

Distributed Hypothesis Generation & Evaluation

Thesis submitted in accordance with the requirements of the
University of Liverpool for the degree of

Doctor in Philosophy

by

Jordan Isaac Robinson

November 2024

To my parents, sister, and Lucy, who changed my life.

Abstract

This thesis addresses the challenges of modelling defeasible reasoning under uncertainty within intelligence analyses, focusing on hypothesis generation and evaluation – an essential set of tasks where analysts must consider distributed, conflicting, uncertain, false, and potentially deceptive information. During analysis, analysts are required to generate the set of mutually exclusive and collectively exhaustive hypotheses to ensure that all the possible hypotheses are evaluated, and crucial hypotheses are not overlooked. With increasing data volumes, resulting from better collection methods and the advent of access to this data via the internet, the complexity of analysis, and the reasoning thereof, will increase also. This thesis is concerned with the development of artificial intelligence tools, techniques, and algorithms that provide analysts with reasoning-support during hypothesis generation and evaluation. The main output from this work is a number of proposals studying whether argumentation theory, computational argumentation, probability and information theory, and deep learning could be combined to offer assistance to analysts in these modelling tasks.

Four techniques are presented and evaluated. The first approach extends a well-known argumentation scheme, deepening its level of specificity, and renders it capable of representing analysts’ reasoning about hypotheses. The second proposal assesses the feasibility of using large language models for hypothesis generation. The third technique can refine an analyst’s prior beliefs about the probability of hypotheses in light of the outputs from semantic evaluation of probabilistic argumentation frameworks. Finally, the fourth method employs abstract argumentation for sensitivity analysis and can identify arguments which, if found to be false or deceptive, would cause the most change to the conclusions drawn from an intelligence analysis. Throughout the thesis, both abstract and real-world examples are used to demonstrate the applicability of the novel techniques and algorithms presented. The proposals are proof of concepts and intended to be used as post-analysis quality checks, ensuring that analysts’ reasoning is rational and includes all the information available to them.

Acknowledgements

I would like to thank my parents, Earl and Sarah Robinson, and my sister, Myah, for their endless support and interest in my work over the past four years. I am also indebted to Lucy Birtles for her encouragement, helpful discussions, and belief in me throughout my time as a student. I would also like to express my gratitude to my extended family, Lucy's family, and my friends who have supported me throughout my studies.

I would also like to thank the Engineering and Physical Sciences Research Council and the Defence Science and Technology Laboratory (Dstl) for funding this research project. Additionally, I am grateful to the Centre for Doctoral Training in Distributed Algorithms and the University of Liverpool for providing me with the necessary training and support to complete the work presented in this thesis. Furthermore, I would like to thank both members of my Independent Progress Assessment Panel, Professor Paul Dunne of the University of Liverpool and Dr Louwe Kuijer of the University of Liverpool, for engaging in interesting discussions, and providing me with invaluable feedback and guidance on how to extend and improve my work. I also wish to thank my thesis examiners, Professor Paul Dunne of the University of Liverpool and Dr Alice Toniolo of the University of St Andrews, for the interesting, insightful, and enjoyable discussions about my research during my viva voce.

I undertook a three month secondment at Dstl as part of my research so I would like to thank Dstl for this. Moreover, I would also like to thank the people in Work Package 4, WP4, of the programme that I was seconded to because they ensured that my time at Dstl was enjoyable and useful. I would also like to thank Alan Knox, who I met during my placement, for the discussions about argumentation theory and intelligence analysis, the work he completed on Operation Fortitude-South (which I describe in more detail in Appendix B), and the interest he showed in the work presented in Chapter 6. One last person that deserves a special mention is Dr Benedict Oakes of the University of Liverpool, whom I lived with while on placement: thank you for the

interesting discussions about your work and mine, and for making our time outside of Dstl fun and memorable.

I would like to thank the Science and Technology Facilities Council's Hartree Centre for the training and use of their high performance computing cluster, called Scafell Pike, which was used to obtain some of the results presented in this thesis.

Finally, I would like to thank my supervisors: Professor Katie Atkinson of the University of Liverpool; Professor Simon Maskell of the University of Liverpool; Professor Chris Reed of the University of Dundee, Scotland; and Mr Todd Robinson of Dstl. Starting with Todd, thank you for your suggestions and feedback which helped to produce the tools and techniques proposed in this thesis. I would also like to thank Chris Reed for his knowledge, suggestions, and engaging discussions throughout the duration of my PhD. Moreover, I would thank Chris for inviting me to join a team of manual argument annotators, led by both Chris and Professor Annette Hautli-Janisz of the University of Passau, Germany, where we worked with an international media organisation and undertook the world's first real-time analysis of argumentative debates over a two-year period. As such, I would also like to thank the annotators, organisers, and the media organisation for that opportunity.

Last but by no means least, there are two people who I cannot thank enough: my internal supervisors, Katie Atkinson and Simon Maskell. I could not have asked for a better set of supervisors, and people, to spend the past four years working with. I have benefitted greatly from your combined knowledge, patience, and friendship. Words cannot begin to describe how grateful I am to you both. Katie, Simon, it has been an honour and a joy to be supervised by you. I will miss our weekly meetings, where knowledge was shared, ideas were debated, stories were told, and many laughs had. You have both contributed more than you realise to making my time as a student a fruitful and rewarding adventure, and you have helped mould me into the researcher I am today – and for this, I am eternally grateful.

Contents

1	Introduction	1
1.1	Research Question	1
1.2	Overview	2
1.3	Contributions	7
1.4	Thesis Structure	8
1.5	Related Publications	10
2	Literature Review	11
2.1	Intelligence Analysis	12
2.1.1	The Intelligence Cycle	13
2.1.2	Types of Intelligence	15
2.1.3	Types of Intelligence Analyst	16
2.1.4	Communicating Probability	17
2.1.5	Structured Analytical Techniques for Intelligence Analysis . .	18
2.2	Argumentation Theory	27
2.2.1	Argumentation Schemes	31
2.3	Computational Argumentation	33
2.3.1	Abstract Argumentation Frameworks	34
2.3.2	Structured Argumentation	41
2.3.3	Argument Mining	46
2.4	Tools for Intelligence Analysis	55
2.4.1	General Approaches	55
2.4.2	Bayesian Methods	56
2.4.3	Argumentation-based Proposals	58
2.5	Summary and Conclusions	61

I	Hypothesis Representation & Generation	64
3	A Scheme for the Representation of Hypotheses	66
3.1	Extension of the Argument from Cause to Effect	66
3.1.1	Multiple Causes	69
3.1.2	The Distinction Between Probability and Confidence	71
3.1.3	Confidence in Causes	73
3.1.4	The Probability of Causes and their Effect	75
3.2	The Argument from Multiple Causes to Effect	76
3.2.1	Extending the Critical Questions	80
3.3	Summary	82
4	On Large Language Models for Hypothesis Generation	84
4.1	Introduction	84
4.2	Methodology	86
4.2.1	Dataset	87
4.2.2	Data Preparation and Annotation	88
4.2.3	Large Language Models	93
4.2.4	Evaluating Model Completions	97
4.3	Results	100
4.3.1	Word-Level Similarity	101
4.3.2	Sentence-Level Similarity	101
4.3.3	Model Hallucinations	102
4.4	Discussion	106
4.5	Summary and Future Work	108
II	Hypothesis Evaluation	110
5	On Human-Machine Bayesian Hypothesis Evaluation	119
5.1	Introduction	119
5.2	Preliminaries	121
5.2.1	Abstract Argumentation Frameworks	121
5.2.2	Probabilistic Argumentation Frameworks	122
5.3	On Human-Machine Bayesian Hypothesis Evaluation	124
5.3.1	Hypotheses and Information	125
5.3.2	Priors and Confidence in the Probabilities of Hypotheses . . .	127
5.3.3	Uncertainty in Items of Information	129

5.3.4	Uncertainty in Attacks	130
5.3.5	The Likelihood	130
5.3.6	Bayes' Theorem	137
5.3.7	Pseudo-code	138
5.4	Experimental Evaluation	143
5.4.1	A Set of Abstract Examples	143
5.4.2	Operation Fortitude-South	153
5.5	Summary and Future Work	161
6	On Diagnostic Arguments in Abstract Argumentation	164
6.1	Introduction	165
6.2	Preliminaries	167
6.3	Diagnostic Argument Identifier	167
6.3.1	Probability Spaces	168
6.3.2	Mutual Information	172
6.3.3	Sensitivity Analysis	174
6.3.4	Pseudo-code	178
6.4	Experimental Evaluation	184
6.4.1	Results from the Running Example	184
6.4.2	A Software Application	187
6.4.3	Operation Fortitude-South	191
6.5	Related Work	195
6.6	Conclusions and Future Work	198
III	Summary of Contributions	200
7	Conclusions and Future Direction	202
7.1	Summary of Contributions	202
7.2	Future Directions	205
	Appendices	231
A	On Large Language Models for Hypothesis Generation	232
A.1	Unique Identifiers	232
A.2	Example of Prompts	233
A.2.1	One-Shot (Unbiased)	233
A.2.2	One-Shot (Biased)	234

A.3	Examples of Model Completions	235
A.3.1	Unbiased Prompting	235
A.3.2	Biased prompting	237
B	Operation Fortitude-South	239
B.1	Dstl's Contribution	239
B.1.1	Step One	240
B.1.2	Step Two	240
B.1.3	Step Three	241
B.1.4	Step Four	242
B.1.5	Step Five	244
B.1.6	Step Six	244
B.1.7	Step Seven	245
B.1.8	Step Eight and Step Nine	245
B.1.9	Concluding Remarks on Dstl's Contribution	245
B.2	List of Information, Evidence, and Assumptions	246
B.3	The Refined ACH Matrix's AIF Argument Graph	247
B.4	Arguments Instantiated Using The ASPIC ⁺ Framework for the Refined ACH Matrix	249
B.5	The Refined ACH Matrix's Argumentation Framework	251
C	On Human-Machine Bayesian Hypothesis Evaluation	253
C.1	Algorithms for Probabilistic Argumentation Frameworks	253
C.2	Distributed Memory Parallel Architecture	254
D	On Diagnostic Arguments in Abstract Argumentation	255
D.1	Hybrid Distributed-Shared Memory Architecture	255

Glossary

- ACH** Analysis of Competing Hypotheses.
- ADU** Argumentative Discourse Units.
- AI** Artificial Intelligence.
- API** Application Programming Interface.
- ASA** All-source analyst.
- CPU** Central Processing Unit.
- DAI** Diagnostic Argument Identifier.
- Dstl** Defence Science and Technology Laboratory.
- EDU** Elementary Discourse Units.
- GPT** Generative Pre-Trained Transformer.
- HUMINT** Human Intelligence.
- IMINT** Imagery Intelligence.
- MASINT** Measurement and Signature Intelligence.
- MBT** Main battle tanks.
- MECE** Mutually exclusive and collectively exhaustive.
- MPI** Message Passing Interface.

NLI Natural Language Inference.

OSINT Open-Source Intelligence.

PrAF Probabilistic Argumentation Framework.

RAM Random Access Memory.

RFI Request for information.

SBERT Sentence Bidirectional Encoder Representations from Transformers.

SD Standard Deviation.

SE Standard Error.

SIGINT Signals Intelligence.

SSA Single-source analyst.

USSR Union of Soviet Socialist Republics.

List of Figures

2.1	The refinement and transformation of data, collected from an operational environment, into items of intelligence (taken from [228]). . . .	13
2.2	The <i>Intelligence Cycle</i>	14
2.3	The Simple Hypotheses technique for multiple hypotheses generation (adapted from [173]).	21
2.4	An example of the Quadrant Hypothesis Generation technique on the time prior to Russia’s invasion of Ukraine.	21
2.5	The Multiple Hypotheses Generator™ (adapted from [173]).	23
2.6	A simple argument containing a single premise supporting a conclusion. . . .	28
2.7	A depiction of Toulmin’s Model of Argument [217].	30
2.8	An example of a Dung-style abstract argumentation framework. . . .	35
2.9	The AIF specification (taken from [11]).	45
2.10	Distinct tasks within the manual argument mining process (adapted from [126]).	47
2.11	The simple structure of inference for (a) convergent, (b) divergent, (c) linked and (d) sequential arguments, where p_i and c_j refers to the i -th and j -th premise and conclusion, respectively (adapted from [126]). . .	50
4.1	An example set of Tweets from the X account of United Kingdom’s Ministry of Defence, published on 25th April 2023.	87
4.2	(a) A combined set of Tweets that contained an intelligence report with two operational hypotheses, denoted Hypo 1 and Hypo 2, and astrategic hypothesis, referred to as Hypo 3, (b) the identification of support for each hypothesis, and (c) the instantiation of the annotation with the dataset.	92

4.3	The two, main hypotheses from the Operation Fortitude-South example instantiated within Walton's argument from cause to effect [236]. The diagram in this figure was adapted from a screenshot of the analysis in <i>OVA+</i> [108].	116
4.4	An instantiation of the reasoning involved in the evidence denoted PORT using Walton's argument from established rule [238] and practical reasoning scheme [241]. The diagram in this figure was adapted from a screenshot of the analysis in <i>OVA+</i> [108].	117
5.1	A flowchart depicting the process for approximating the posterior probability of arguments with embedded hypotheses conditioned on the available information and the defeat of all other hypotheses.	139
5.2	(a) A <i>PrAF</i> with two equally uncertain arguments containing hypotheses, and the resulting posterior estimates computed using the prior relationship in Equation 5.37 and the likelihood output by (b) Algorithm 5.1, (c) Algorithm 5.2, and (d) Algorithm 5.3, where $N = 30,000$. . .	146
5.3	(a) A <i>PrAF</i> with two arguments with embedded hypotheses where $A_H^{(1)}$ was more certain than $A_H^{(2)}$, and the posterior estimates computed using the prior relationship in Equation 5.37 and the likelihood output by (b) Algorithm 5.1, (c) Algorithm 5.2, and (d) Algorithm 5.3, where $N = 30,000$	146
5.4	(a) A <i>PrAF</i> with two certain arguments containing hypotheses and two uncertain information arguments who attack both hypotheses with uncertainty. The posterior estimates for the <i>PrAF</i> computed using the prior relationship in Equation 5.37 and the likelihood output by (b) Algorithm 5.1, (c) Algorithm 5.2, and (d) Algorithm 5.3, where $N = 30,000$	147
5.5	(a) A <i>PrAF</i> with two hypotheses and two information arguments, with one is more certain than the other. The posterior estimates for the <i>PrAF</i> computed using the prior relationship in Equation 5.37 and the likelihood output by (b) Algorithm 5.1, (c) Algorithm 5.2, and (d) Algorithm 5.3, where $N = 30,000$	148
5.6	(a) A <i>PrAF</i> with two hypotheses and two uncertain information arguments, where there is more confidence in the attack from one of the information arguments to $A_H^{(1)}$. Posterior estimates for the <i>PrAF</i> computed using the prior relationship in Equation 5.37 and the likelihood output by (b) Algorithm 5.1, (c) Algorithm 5.2, and (d) Algorithm 5.3, where $N = 30,000$	149

5.7	(a) A <i>PrAF</i> with two hypotheses and two information arguments, where one information argument and its attack to $A_H^{(1)}$ had a greater certainty. The posterior estimates computed using the prior relationship in Equation 5.37 and the likelihood output by (b) Algorithm 5.1, (c) Algorithm 5.2, and (d) Algorithm 5.3, where $N = 30,000$	150
5.8	(a) A <i>PrAF</i> with an asymmetry between $A_H^{(1)}$ and $A_H^{(2)}$, arising from the instantiation of three information arguments, where one of the information arguments had an inducement probability equal to unity. The posterior estimates computed using the prior relationship in Equation 5.37 and the likelihood output by (b) Algorithm 5.1, (c) Algorithm 5.2, and (d) Algorithm 5.3, where $N = 30,000$	151
5.9	Posteriors overtime, where the priors were equal to (a) 0.5 across all time steps – i.e., <i>Fig. 1.2</i> to <i>Fig. 1.8</i> – and (b) 0.5 in the first time step – i.e. <i>Fig. 1.2</i> – and then priors were equal to the previous posterior estimate, across all the remaining time steps – i.e., <i>Fig. 1.3</i> to <i>Fig. 1.8</i>	153
5.10	The posterior estimates for the Operation Fortitude-South argumentation framework. Both arguments containing hypotheses and attacks between them had a probability of inducement of unity. All other arguments were assumed to be information arguments, where information arguments and their attacks had inducement probabilities equal to 0.5. The number of trials was ($N =$) 520×10^6	154
5.11	(a) The execution time and (b) strong scaling speedup for Algorithm 5.5 for an increasing number of processors, where the number of trials was $N = 1 \times 10^6$	158
5.12	(a) The execution time and (b) weak scaling efficiency for Algorithm 5.6 for an increasing number of processors, where the number of trials was $N = \mathcal{P} \times 10^6$	161
6.1	Example of a Dung argumentation framework \mathcal{G}	169
6.2	A sensitive argumentation framework \mathcal{G}^t	177
6.3	A flowchart depicting the order within which the algorithms for the DAI are executed.	179
6.4	The diagnosticity scores for the 16 partitions of labelling vectors for all arguments within the Dung-style argument system \mathcal{G} in Figure 6.1, discovered using preferred semantics.	184

6.5	A violin plot showing the distribution of <i>diagnosticity scores</i> , as well as the median, mean and average absolute change in mutual information, for partitions of argument labels after the removal of each argument from the initial argumentation framework \mathcal{G} , under preferred semantics.	185
6.6	A screenshot of the home screen of the DAI application. The Operation Fortitude-South AIF data was imported, transformed, and instantiated into a Dung argumentation framework. The graph was plotted (right) and evaluated using complete semantics to discover the complete labellings (bottom left).	189
6.7	A screenshot of the DAI application with the Operation Fortitude-South argumentation framework (right), and the corresponding arguments and sub-arguments instantiated using an ASPIC ⁺ argumentation system (left).	189
6.8	A screenshot of the home screen of the DAI, where the argumentation framework \mathcal{G} from Figure 6.1 was instantiated using the application, plotted (right), and evaluated to discover the preferred labellings (bottom left).	190
6.9	A screenshot of a window from the DAI application that displays the results from the diagnosticity vector.	190
6.10	The distribution of change in mutual information, as well as the mean, absolute average, and median diagnosticity scores, for all the arguments in the Operation Fortitude-South argumentation framework, found using preferred semantics.	192
6.11	(a) The execution time and (b) strong scaling speedup for the execution of Algorithms 6.2 and 6.4 for the Operation Fortitude-South argumentation framework, where $\frac{2^{ \mathcal{A} }}{2} = 524,288$ in the initial graph.	196
B.1	The refined ACH matrix for the Operation Fortitude-South example which was annotated using OVA+ [108].	248
B.2	The Dung argumentation framework from the refined Operation Fortitude-South ACH matrix.	252
C.1	Distributed memory architecture employed in the parallel execution of Algorithm 5.3.	254
D.1	An example of the hybrid distributed-shared memory parallel architecture, where each compute node \mathcal{P} had a total of 32 threads \mathcal{T} .	256

List of Tables

2.1	The Probability (or Uncertainty) Yardstick: a method of communicating verbal probabilities and uncertainties for numeric ranges within the unit interval [220].	18
2.2	The complete labellings of the argumentation framework \mathcal{G} (Figure 2.8).	39
4.1	Average BERTScore f1-score ($\pm SD$) on the test dataset, where U and B denote the results from prompts that, respectively, contained unbiased and biased training examples.	103
4.2	Average number of hypotheses in the test dataset closer to model completions than the target hypothesis ($\pm SE$), where U and B denote the results from prompts that, respectively, contained unbiased and biased training examples.	104
4.3	Average S_{NLI} ($\pm SD$) score obtained from the test dataset, where U and B denote the results from prompts that, respectively, contained unbiased and biased training examples. Results highlighted in bold were the highest model type average, respectively.	105
4.4	The refined ACH matrix for Operation Fortitude-South, where “C” and “I” denote row headers that are consistent and inconsistent with the hypotheses (column headers), respectively.	114
5.1	The likelihood conditioned on either $A_H^{(1)}$ or $A_H^{(2)}$ and the ratio between them, i.e. $\frac{A_H^{(1)}}{A_H^{(2)}}$, for the results presented in Figures 5.2 to 5.8.	152
5.2	The likelihoods for the Normandy and Pas de Calais hypotheses and the number of trials employed in the approximation.	154
5.3	The execution time and strong scaling speedup for a fixed problem size.	159

5.4	The execution time and weak scaling speedup for an increasing problem size.	161
6.1	The preferred labellings of the framework displayed in Figure 6.1. . .	169
6.2	The preferred labellings of the \mathcal{G}^l displayed in Figure 6.2.	177
6.3	The degree and eigenvalue centrality for the arguments in the initial AF \mathcal{G}	187
B.1	The initial ACH matrix for Operation Fortitude-South, where “C” and “I” denote row headers that are consistent and inconsistent with the hypotheses (column headers), respectively.	242
B.2	A table containing the argumentation schemes, which were identified by Dstl, corresponding to the row entries in the initial and refined ACH matrix for the Operation Fortitude-South example.	243
B.3	The ORBAT ACH matrix, where “C” and “I” denote row headers pertaining to the Order of Battle that are consistent and inconsistent with the hypotheses, respectively.	243
B.4	The refined ACH matrix for Operation Fortitude-South, where “C” and “I” denote row headers that are consistent and inconsistent with the hypotheses (column headers), respectively.	244

Chapter 1

Introduction

In this chapter, I present my research question, motivate the domain within which the work proposed in this thesis should provide benefit, set out a series of objectives inspired by the problem setting, list the main contributions of this thesis, provide an overview of the structure of this thesis, and list a number of publications that resulted from the work presented in this thesis.

1.1 Research Question

Intelligence analysis is a complex and cognitively challenging task that will become increasingly more so as the intelligence community strives towards incorporating more data in their analyses of novel situations. Intelligence analysts must reason with multiple items of distributed, conflicting, uncertain, false, potentially deceptive, and missing data to arrive at conclusions about what has happened, is happening, and will happen. For intelligence analysis to keep up with the increased demands of the future, the use of artificial intelligence (AI) approaches and algorithms is an obvious choice that could provide decision- and analytical-support to analysts. There are, of course, many different tasks that AI could sensibly be employed for within this fecund research field. However, the proposals made within this thesis focus on the development of AI tools, techniques, and algorithms for all-source intelligence analysts conducting hypothesis generation and evaluation. The overarching problem addressed throughout this thesis is:

By what means may AI be employed for hypothesis generation and evaluation within defence, security, and intelligence settings?

The above question was motivated by a growing need to provide intelligence analysts with reasoning support, due to the complexities of their work. In the quest for answers, a number of different research disciplines were employed, namely: computer science, artificial intelligence, mathematics, philosophy, linguistics, and intelligence analysis.

1.2 Overview

The word *intelligence* can mean a number of different things depending on the context within which it was used. For instance, the noun *intelligence* might refer to a student's ability to acquire and apply knowledge within an education setting or the collection of information of military or political value, as per the *Oxford Dictionary and Thesaurus* definition [163, p. 543]. The work presented in this thesis draws upon the latter definition.

There are two classical works that offer principles that are applicable to intelligence analyses today, namely Sun Tzu's *The Art of War* [219] and Carl von Clausewitz's *On War* [54], where they discuss notions of understanding one's military capabilities, the enemy, strategic planning, and deception. While both works are indeed fundamental to intelligence analysis, further discussion about Sun Tzu's *The Art of War* has been omitted from this thesis because the book is mainly focussed on military strategy. Looking more closely at [228], the Chairman of the Joint Chiefs of Staff for the United States Armed Forces draws their definition of intelligence from Carl von Clausewitz's 1832 book, titled *On War*:

“By ‘intelligence’ we mean every sort of information about the enemy and his country – the basis, in short, of our plans and operations.” [54, p. 117]

The intelligence that Clausewitz refers to is any kind of information about our adversaries that we can use to our advantage. In [54], Clausewitz discusses how a lot of the intelligence reports that commanders receive are contradictory, false and uncertain, and even more so when a country is in conflict with another. He argues that we can resolve the inconsistent data contained within reports by conducting reasoning, guided by the laws of probability and wisdom acquired from the knowledge of people, current affairs, common sense, and experience. He contends that during times of war, there is an increase in the number of intelligence reports, which increases the amount of conflicting information and makes the task of coming to the correct judgement extremely challenging, leading to commanders doubting their own rationale. Although Clausewitz's comments were written in the year 1832, they provide an interesting perspective

on the challenges of intelligence analysis today, where analysts must evaluate several orders of magnitude more conflicting, erroneous, and uncertain data.

Today, in the West, after the initiation of a *Request For Information* (RFI), data is collected from an operational environment by a *single-source analyst*. A RFI is created when a stakeholder, decision maker, or commander requires more information to make a well-informed decision, or when an all-source analyst needs more data to complete an analysis. The required data is then collected by a single-source analyst, using any one of the five disciplines of intelligence collection [139], as part of the *Intelligence Cycle*, which I describe in Section 2.1.1. An all-source analyst, whose job is explained in Section 2.1.3, receives those items of collected information and begins their own analysis. The all-source analyst reasons abductively – using all the collected, already-available, and multi-modal data supplied by single-source analysts – to generate hypotheses, which are then evaluated by the same analyst, using the same set of data.

At the heart of every *good* scientific method is an initial observation that inspires the formation of a hypothesis, which is then tested in a fair experiment where the results are analysed and then communicated in, say, a peer-reviewed article, a report, or even a thesis. We might deem a publication *good* if the results from experiments within that article are repeatable and based on evidence. The generation and evaluation of hypotheses within the context of intelligence settings is no different to hypotheses formed and tested using the scientific method.

Hypothesis generation and evaluation is the cornerstone of all intelligence analyses and there are many analytical techniques that aid analysts with this task, as I describe in Section 2.1.5. Here, the term *hypothesis* is used in a general sense to mean a forecast, explanation, or conclusion based on available information and evidence found, and assumptions made, within the *Intelligence Cycle*. In [173], Pherson and Heuer Jr. define a *good* hypothesis as:

- A statement;
- Testable and falsifiable;
- Based on observations;
- Predicts anticipated results clearly.

It is of the utmost importance that analysts generate the set of plausible, mutually exclusive and collectively exhaustive (MECE) hypotheses because it ensures that they evaluate the set of non-overlapping and possible hypotheses such that no two hypotheses can be observed to be true at the same time, for a given scenario. Here, it is

important to note the distinction between the realisation of a hypothesis and an analyst's prior belief about the probability of a hypothesis. The probability of a hypothesis is based upon the outputs from an intelligence analyst's analysis. There will be a number of different (or collectively exhaustive) hypotheses that the analyst has generated and evaluated in order to produce a set of probabilistic judgements, where the analyst might say that one hypothesis is *more likely* than the other, both hypotheses are equally *likely*, etc. The realisation of a hypothesis happens when one of the set of collectively exhaustive hypotheses actually occurs such that it is an event that has been observed. In this instance, the realised hypothesis is a distinct event such that there is no overlap with any of the other hypotheses that were considered during analysis, due to their mutual exclusivity which necessitates that two or more hypotheses cannot occur at the same time. Another way to think about this is that the set of generated hypotheses will eventually become mutually exclusive, after the observation of one of those hypotheses at some point in time after analysis has concluded. Generating the set of MECE hypotheses ensures that analysts do not miss an important hypothesis by forcing them to evaluate all potential outcomes based on available data. There are a number of different analytical techniques which aid analysts in the generation of MECE hypotheses, which I explore briefly within Section 2.1.5.1. To the best of found knowledge, there have been no proposals for solutions to automated hypothesis generation within intelligence contexts. However, as I state in Section 2.4.1, some have argued that, with the recent advances in the field of natural language processing, large language models could assist intelligence analysts with many tasks [204], where one obvious idea would be to employ them for MECE hypothesis generation.

All-source analysts also employ an assortment of analytical techniques during hypothesis evaluation, as I show in Section 2.1.5.2. Analytical techniques and intelligence analysis are closely intertwined. Analysts within the United States of America are required by law [224] to employ such tools, where appropriate, after the intelligence failings that contributed to the *successful* (from the perpetrators' point of view) terrorist attacks on 11th September 2001 and the arguably *unjustified* invasion of Iraq in 2003 [58]. The Analysis of Competing Hypotheses [101] is an example of a diagnostic, analytical technique that assists analysts by providing them with a structured process during the evaluation of hypotheses. First proposed by Heuer Jr. in the 1970s, the Analysis of Competing Hypotheses enables the analyst to reason with multiple items of information, evidence, and assumptions in order to quantify the probability of hypotheses. Pherson and Heuer Jr., two of the technique's biggest proponents, claim that the procedure reduces confirmation bias [173, p. 164-165] because the relative probability of a hypothesis is computed as a function of how many data were in conflict

with it, and the most likely hypothesis is the one which has the fewest inconsistencies. Techniques like the Analysis of Competing Hypothesis provide analysts with a repeatable and explainable methodology, capturing their reasoning so that other analysts can understand, and even scrutinise, their assessments.

One of the fundamental requirements for reliable and correct intelligence analysis is coherent reasoning about conflicting information, where conclusions can be revised as new data becomes available. Analyses can be, at times, cognitively challenging, and *a fortiori* when the analyst is required to reevaluate the likelihood of hypotheses in light of new information. The use of AI approaches that can automate some of the reasoning tasks that analysts must complete should provide tangible benefit to the intelligence community. Computational models of argument, a sub-field of AI and computer science, can help analysts by conducting some of the logical and non-monotonic reasoning necessary for intelligence analyses. For example, the process by which the probability of a hypothesis is quantified using the Analysis of Competing Hypotheses is somewhat synonymous with how the acceptability of arguments are computed by a semantics within Dung's formative abstract argumentation framework [70]. As the forthcoming definitions in Section 2.1.5.2 demonstrate, the most likely hypothesis, quantified using the Analysis of Competing Hypotheses, is the one that conflicts with the fewest items of information, evidence, and assumptions. Whereas, in Dung's formalism, as presented in Section 2.3.1, a set of arguments is deemed to be *admissible* if that set possesses no internal conflicts and the arguments contained within that set can successfully defend each other from external attacks.

The arguments, and attacks between them, found within intelligence settings are real-world arguments, and hence intelligence analysts do use certain, stereotypical patterns of inference during their evaluation of hypotheses. Argumentation theory, a sub-field of philosophy and linguistics, can help by capturing the reasoning of an analyst in a coherent, defeasible, and explainable manner, which makes argumentation both an effective and suitable technology to employ in the development of tools and techniques for intelligence analysis. Argumentation schemes provide a method of representation for arguments put forth about information, evidence, and assumptions that support hypotheses or their negation, as I show in Chapter 3 and the introduction to Part II. The set of critical questions associated with each scheme can facilitate the identification of attacks between propositions within other arguments, which may enable analysts to identify attacks that they previously did not consider. Moreover, argumentation schemes and their corresponding sets of critical questions have been shown to be useful in the instantiation of Dung-style argument systems [15], enabling automated reasoning about the acceptability of refined arguments (Section 2.3.3.1). Thus, both

argumentation schemes and Dung’s argumentation framework are of great significance to the work presented throughout the remainder of this thesis.

As argued by Clausewitz in [54], the analyst should utilise the laws of probability in order to arrive at a set of reliable, coherent, and correct conclusions. This notion is still indeed valid within the intelligence community to this very day. As I explain in Section 2.1.1, hypotheses are included within an intelligence report (or a *finished intelligence product*) that is disseminated to stakeholders, decision makers, and commanders. The report also contains qualitative probabilistic judgements corresponding to each of the hypotheses contained within it, as I describe in Section 2.1.4, which permits the customers of that product to understand how likely the outcomes are. Intelligence analysis is reasoning that is conducted using probability and under uncertainty – i.e., uncertainty is present within the probabilistic judgements made by an analyst and, during an analysis, it may arise from the source of an item of information, to name a couple of examples. Thus, for AI techniques to offer benefit to the intelligence community, proposed tools should enable the incorporation of both probability and uncertainty. Proposals for hypothesis evaluation tools that incorporate AI, argumentation, probability, and uncertainty are discussed in Section 2.4.

It is important to highlight some of the challenges involved with conducting research into the development of AI tools, techniques and algorithms that support intelligence analysts during analysis. To design, develop and test reliably accurate AI systems, the intelligence community would need to provide researches with a wealth of high quality and correct data. However, much of the data required to build such systems is classified. Even if those data were unclassified for, say, research, intelligence analysts often evaluate rare events for which there is little to no data available to train AI systems on. If the intelligence community were somehow able to train and employ, for example, the capabilities of machine and deep learning within intelligence analyses, model interpretability and transparency is still an issue. Every stage of an intelligence analysis should be auditable and explainable, and not all the current state-of-the-art learning models within the literature are. While there have been many advances in the field of AI, such as in the natural language processing literature with the recent advent and popularity of large language models, such language models in their current state are not proficient at conducting coherent and consistent reasoning. As the results in Chapter 4 demonstrate, such models can also potentially fall victim to hallucinations which would be, at best, unhelpful to the analyst. Tools that can represent and automate some of the analyst’s reasoning whilst also allowing for input from the human should provide the most benefit to analysts, which is why computational argumentation is such an attractive technology. However, as I discuss in Section 2.1.5.2, the develop-

ment of any reasoning- and decision-support tools requires the insights and experience of analysts whose propensity to assist with this task may differ between individuals.

In light of the above, I present the main goals of this thesis below:

1. To provide a theory of representation for hypotheses that are supported by information, evidence, and assumptions that accounts for the defeasible nature of the reasoning, probability, and uncertainty inherent within intelligence analyses.
2. To investigate the feasibility of employing large language models for hypothesis generation within an intelligence domain.
3. To design and develop a formal account capable of refining an intelligence analyst's initial probabilistic judgements by conditioning their prior beliefs on the acceptability of available information. The approach should be tested on a real-world example to assess the potential performance of the proposal in practice.
4. To design and develop explainable algorithms which quantify the sensitivity, robustness, and dependence of the conclusions drawn from an intelligence analysis based on the arguments present within it, and evaluate the proposed approach using a real-world example from the intelligence domain.

1.3 Contributions

The main contributions of this thesis are a series of AI tools, techniques and algorithms that were developed in order to provide decision- and reasoning-support to intelligence analysts during analysis. The contributions are listed below:

- The first contribution – articulation of an argumentation scheme, called the *Argument from Multiple Causes to Effect* – should allow the intelligence analyst to represent their reasoning about items of information, evidence, and assumptions that support hypotheses. Each proposition within the scheme has a probability and confidence interval which corresponds to the proposition's likelihood and uncertainty, respectively. The proposed scheme draws upon Walton's argument from cause to effect [236], extending its level of specificity so that it can be employed within intelligence settings. The scheme is also accompanied by a set of critical questions which could aid the analyst in RFI generation and the identification of conflicts between propositions within abstract formalisms.

- The second contribution of this thesis is a study that assessed the feasibility of employing large language models for hypothesis generation. A number of state-of-the-art language models were used, as well as data which contained open-source intelligence reports from the current conflagration between Russia and Ukraine.
- The third contribution can be divided into two parts, namely: a definition of a hypothesis and a definition of an item of information within the ASPIC⁺ framework [177, 150]; and a formal method that employs those definitions and can refine an analyst’s judgements about the prior probabilities of hypotheses, using the outputs from semantic evaluation of probabilistic argumentation frameworks [133] and Bayes’ theorem [18].
- The fourth contribution employs abstract argumentation for sensitivity analysis. The outputs from Dung’s argumentation frameworks [70] are used to compute the probability of partitions of argument labels in order to compute the change in mutual information [200, 59] before and after the removal of an argument of interest. The technique provides a holistic quantification of the sensitivity, robustness, and dependence of an argumentation framework’s conclusions to change caused by the removal of arguments.
- The final contribution of this thesis is an instantiated argumentation framework, stored in the Argument Interchange Format [11], which contains annotated data from a real-world deception scenario from World War Two, called Operation Fortitude-South. The data stored in the Argument Interchange Format was converted to an ASPIC⁺ argumentation theory [177, 150] and then to a Dung abstract argumentation framework [70], using the procedure described in [23] and Section 2.3.2.3, in order to evaluate the performance of the third and fourth contribution using real-world data.

1.4 Thesis Structure

This thesis is structured as follows:

- **Chapter 1** is this chapter where this thesis’ research question and objectives are defined.
- **Chapter 2** presents an extensive literature survey relevant to the contributions made throughout this thesis. The chapter explores the topics: intelligence anal-

yses, argumentation theory, computational models of argument, and proposed tools and techniques that provide benefit to intelligence analysts in their daily duties.

- **Part I** is comprised of two chapters, Chapters 3 and 4, and makes two proposals with the aim of supporting intelligence analysts in the representation and generation of hypotheses.
 - **Chapter 3** draws upon Walton’s argument from cause to effect [236] and proposes an argumentation scheme for the representation of hypotheses supported by multiple causes, while also enabling the inclusion of uncertainty and probability within each proposition instantiated within the scheme.
 - **Chapter 4** studies whether large language models can be reliably used for hypothesis generation within intelligence contexts. This chapter employs state-of-the-art language models and a dataset from a real-world and current intelligence scenario, containing hypotheses supported by information, to evaluate the feasibility of using such models for hypothesis generation.
- **Part II** examines two approaches, proposed in Chapters 5 and 6, to aid intelligence analysts in hypothesis evaluation tasks. The introduction to Part II presents the method for the manual instantiation of an Argument Interchange Format [11] argument graph, which was transformed into an ASPIC⁺ argumentation theory [177, 150] and then to a Dung argument system [70], and used a scenario pertaining to an Allied deception operation from World War Two, called Operation Fortitude-South. The graph was employed to test the techniques presented in Chapters 5 and 6 on data from a real-world setting.
 - **Chapter 5** proposes an approach for the refinement of intelligence analysts’ prior probabilistic judgements about hypotheses using both probabilistic argumentation frameworks [133] and Bayes’ theorem [18]. The quality of the method is assessed using a series of abstract examples and the argumentation framework introduced at the start of Part II.
 - **Chapter 6** employs Dung’s seminal argumentation framework [70] for sensitivity analysis. Again, the benefit of the technique was assessed using an abstract, running example and the Operation Fortitude-South framework presented in the beginning of Part II. The method was also implemented in a Python [232] application with a graphical user interface in order to show

that results should be relatively easy for intelligence analysts to obtain, whilst also providing useful insights.

- **Part III** is comprised of one chapter, Chapter 7, that concludes the research presented in this thesis and provides some suggestions for possible areas of future work in Sections 7.1 and 7.2, respectively.

1.5 Related Publications

Parts of this thesis appear in the following publications:

- A brief description of intelligence analysis, the *Intelligence Cycle*, an early version of this research's aim and objectives, preliminary results, discussion, and avenues for future work are presented in [186], and the description of the *Intelligence Cycle* in Section 2.1.1 and the diagram in Figure 2.2 were adapted from that document.
- In [188], Dung's abstract argumentation framework [70] was employed for sensitivity analysis. The formal definitions for the probability spaces, mutual information [200, 59] calculations, and sensitivity analysis in Chapter 6 were taken from that publication. The chapter also employs the same running example in Figure 6.1 as the one in the paper. However, the example presented in this thesis uses preferred (instead of complete) semantics to demonstrate how the algorithms in Chapter 6 work.
- A software application that is capable of running the algorithms proposed in Chapter 6 (and [188]) and visualising the results was also presented in [187]. The application presented in Section 6.4.2 is the same as the one in [187], however it is accompanied by a more detailed explanation of how to use it and the outputs.

Chapter 2

Literature Review

In this chapter, I present an overview of the current literature that is pertinent to the topics covered in this thesis. In Section 2.1, I introduce the topic of intelligence analysis. Section 2.2 examines argumentation theory and Douglas Walton’s work on argumentation schemes, as they are foundational to the scheme proposed in Chapter 3 that represents reasoning about hypotheses. Section 2.3 surveys the field of computational argumentation. Here, I introduce Dung’s abstract account of argumentation, the ASPIC⁺ framework and the mapping from the Argument Interchange Format to abstract argumentation frameworks, to enable defeasible reasoning about real-world arguments and the conflicts between them. I end this section with discussion about the process and limitations of both manual and automated argument mining. As I show in the introduction to Part II, the argumentative structure was extracted from an example pertaining to World War Two, using manual argument mining. Those argument data, stored within the Argument Interchange Format, were then transformed into an ASPIC⁺ argumentation theory and then to a Dung abstract argument system in order to enable the testing and evaluation of the approaches proposed in Chapters 5 and 6. Section 2.4 provides an overview of the decision- and reasoning-support tools and techniques for intelligence analysis, studying general, Bayesian, and argumentation-based approaches. Finally, in Section 2.5, I conclude with a summary of the chapter.

2.1 Intelligence Analysis

Following on from Clausewitz's definition of *intelligence*, introduced in Chapter 1, in this section, I examine the topic of intelligence analysis, exploring how it is conducted and the types of analysts involved. I explain why intelligence analysis was transformed into a more scientific pursuit, after some recent intelligence failings, such as the terrorist attacks on 11th September 2001 and the disputed invasion of Iraq in 2003. I examine the analytical techniques frequently employed within the intelligence community, focusing mainly on the process, benefits, and limitations of the Analysis of Competing Hypotheses.

The act of intelligence analysis is similar to that of a metaphorical funnel, where data is collected from an operational environment and condensed into a finished intelligence product, which is then used in the downstream decision-making process (Figure 2.1). For a given operational environment, there will be data that can be collected which might be important. An operational environment contains a number of different operational variables which can make each scenario quite different. Operational variables include things such as the distribution of political responsibility; military capabilities; economic behaviour; social aspects like culture, religious views, and ethnicity within a given area; and the physical environment, to name a few [14]. Through the operation of *Processing and Exploitation*¹, data is collected from the operational environment and transformed into information. Information can be of use to a stakeholder or decision maker, but there is often not much utility in the information when considered on its own. However, when multiple items of information have been analysed and evaluated in conjunction, it results in pieces of intelligence which clearly explain current and future events, along with the inherent uncertainty in that forecast, for a given operational environment. Intelligence analysis, therefore, is the act of reasoning with multiple items of information to come to conclusions about a situation, assessing the likelihood of an adversary's past, present, and future actions based upon available observations or data.

¹I do not consider any *Processing and Exploitation* techniques within this thesis. However, it is important for the reader to be aware of the wider process of intelligence analysis.

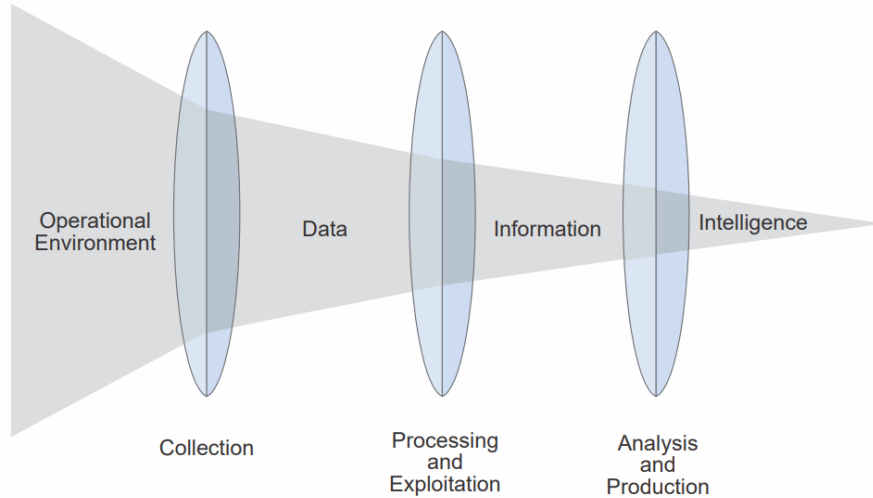


Figure 2.1: The refinement and transformation of data, collected from an operational environment, into items of intelligence (taken from [228]).

2.1.1 The Intelligence Cycle

The *Intelligence Cycle* is displayed in Figure 2.2 and includes five steps, which are: *planning*, *tasking*, *collection*, *analysis*, *production*, and *dissemination* [227, 206]. In the *planning* stage, stakeholders and decision makers discuss and identify what they might need to know about a particular topic within an operational environment – listing all the things they already know, those things that they do not, and what they need to find out – so that they can make a well-informed judgement after the *Intelligence Cycle* has concluded. After *planning*, a RFI is initiated by different types of stakeholders, such as commanders, policymakers, all-source intelligence analysts, amongst others. A RFI can come in different forms, depending on the customer who initiated it [114]. Now a varying number of relevant assets, such as single-source analysts, are then *tasked* with the *collection* of pertinent data from the operational environment, which I discuss further in Sections 2.1.2 and 2.1.3. After *collection*, data is transformed into information through *Processing and Exploitation* techniques, which are not discussed in any tangible detail within this thesis. Intelligence analysts then begin *analysis* of new and already available information, using a variety of diagnostic, contrarian and imaginative thinking techniques, to carry out tasks such as hypothesis generation and evaluation [223]. It is the diagnostic techniques that are the most pertinent to this thesis.

I will, therefore, introduce and discuss in more detail the Analysis of Competing Hypotheses, which is one such technique that is employed in the evaluation of hypotheses using available information, in Section 2.1.5. After *analysis* is complete, intelligence analysts *produce* what is referred to as a *finished intelligence product* or an *intelligence report* that contains the relative probability of hypotheses, along with any items of diagnostic information which would change the likelihood of the lead hypothesis if any one of those items were found to be untrue. The intelligence product is then *disseminated* with customers who make decisions based on the judgements output from an analysis. The process ends when the same customer, who started the *Intelligence Cycle* with their RFI, receives an intelligence report which contains the answer to their original question.



Figure 2.2: The *Intelligence Cycle*.

There are many different approaches to intelligence analyses which differ depending on where you are in the world. In a recent podcast², Mike Baker, a former Central Intelligence Agency agent, describes the difference between the West's approach to the *Intelligence Cycle* and compares it with China's procedures [191]. Baker's so-called Western approach is a targeted approach in which Western intelligence services identify items of information they require, and then *task* the relevant people to collect that data from an operational environment, similar to the process described in Figure 2.2. On the other hand and according to Baker, China collects all the information they can and then employ teams of agents to assess it, extracting any important pieces of information. Baker's comments are in some sense a generalisation because it is highly likely that the West will collect and evaluate more than just targeted information, and China will also have more targeted approaches than the mass collection effort that Baker suggests.

Both approaches have their benefits and issues. For instance, one might spend an unnecessary amount of time sifting through swathes of data to extract a piece of information that would have been quicker to obtain through a targeted collect, when using, what Baker coined, the Chinese approach. Whereas, while using the supposed Western approach, analyses might be impacted by the notion that *you don't know what you don't know*, and therefore both analysts and customers might not always know

²It is important to note that while this citation is from a podcast and it has not undergone the rigour of an academic peer-review, there is currently not any openly available literature that compares and contrasts the different approaches to the *Intelligence Cycle* around the world because this information is often classified.

what they need to collect or include in analyses to make a well-informed evaluation or decision, respectively. This notion is reaffirmed by the late Donald Rumsfeld, the former United States Secretary of Defence, in the following quote taken from a news briefing about the lack of evidence linking the Iraqi government to weapons of mass destruction.

“Reports that say that something hasn’t happened are always interesting to me, because as we know, there are known knowns; there are things we know we know. We also know there are known unknowns; that is to say we know there are some things we do not know. But there are also *unknown unknowns* – the ones we don’t know we don’t know. And if one looks throughout the history of our country and other free countries, it is the latter category that tends to be the difficult ones.” [64]

The *unknown unknowns* problem is exacerbated by the *need to know* policy currently employed within the Western intelligence community [22]. The general idea is that a person is only given the information that they *need to know* to complete their individual task within the *Intelligence Cycle*. While this has the benefit of enforcing security through obfuscation, compartmentalised information can lead to inaccurate analyses due to missing information, and ultimately intelligence failures, such as the terrorist attacks on 11th September 2001 [230]. Proposals have been made to change the *need to know* principal to a *responsibility to provide* paradigm [34, 22]. However, this information-sharing approach has its flaws too, namely sharing too much information increases the likelihood that your adversaries will become aware of your knowledge on a topic, leading to security breaches.

2.1.2 Types of Intelligence

The collection activities conducted by intelligence agencies within the West use a variety of capabilities to obtain the target information required by a stakeholder, policy maker, or decision maker, in answer to their RFI. There are five widely-agreed upon intelligence collection disciplines [139]:

- *Human Intelligence* (HUMINT), which is obtained from human sources and can be split into two categories: overt collection; and covert activities. Overt HUMINT collection is performed openly by military attaches, diplomats, and members of official delegations, amongst others [227]. Covert HUMINT collection is conducted by agents who have agreed to provide information on a foreign nation in a clandestine manner.

- *Signals Intelligence* (SIGINT) is information collected by intercepting signals. The types of signals monitored and intercepted by SIGINT are, for example: *communications intelligence* which monitors things such as telephone traffic, video, fibre optics, or any other means of communication; *electronic intelligence* which employs transmissions from non-communication devices, such as radar, to locate an emitter, its characteristics, and any other systems that are supporting it; and *foreign instrumentation signals intelligence* which collects information on an adversary's weapon systems while they are undergoing testing [226].
- *Imagery Intelligence* (IMINT), which is the act of acquiring representations of objects captured using, for example, photography from hand-held cameras and satellites, or other media sensors, such as radar sensors and lasers, to name a couple.
- *Measurement and Signature Intelligence* (MASINT), which is focused on the collection and analysis of data from scientific and technical sensors, with the goal of identifying the occurrence of events with distinctive features that is often indicative of, for example, an adversary who has conducted experiments with new equipment.
- *Open-Source Intelligence* (OSINT), which involves the collection of data from sources available to the public, such as material on the internet. When open-source information is combined, it is sometimes possible to derive classified information [227]. The future of intelligence analysis and national security will depend on leveraging the ever-increasing capability of AI, machine, and deep learning tools for OSINT [110].

2.1.3 Types of Intelligence Analyst

There are two types of analyst who have different roles within the *Intelligence Cycle*, namely Single-Source Analysts (SSAs) and All-Source Analysts (ASAs). Similar to the process discussed in Section 2.1.1, these analysts are tasked by a customer and answer a RFI. However, the fundamental difference is that SSAs commonly *collect* data and process it turning it into information, while ASAs conduct *analysis* on a variety of different items of information collected and shared by SSAs.

The role of SSAs is to use their expertise with a single source to collect data from an operational environment, using one of the five disciplines of intelligence collection (Section 2.1.2). Then, through a series *Processing and Exploitation* techniques (Section 2.1.1), they refine those data into items of information, which can be employed in

downstream *analysis* and *production*. The information produced by SSAs is sometimes referred to as *raw intelligence*, which is usually in the form of a report which contains the collected data along with a summary of the SSA's findings [53]. However, there is little to no dialogue between an ASA and their single-source counterparts because the assessments generated by the set of SSAs are usually retrieved by the ASA through database queries³.

As the name implies, ASAs make use of all the relevant and available information given to them by their single-source counterparts, to produce *finished intelligence*, in answer to a stakeholder's, decision maker's, or commander's (i.e., the customer's) RFI. The ASA is usually the customer of the SSA's raw intelligence which, when combined with other SSAs' results during an analysis, is used to produce a *finished intelligence product*. The ASA is trained to utilise a variety of diagnostic, contrarian, and imaginative thinking techniques to combine multiple sources of information into a consolidated intelligence product [46]. As previously mentioned in Section 2.1.1, once a stakeholder, decision maker, or commander receives the *finished intelligence product* from the ASA, the *Intelligence Cycle* is complete.

The target audience that will benefit the most from the tools and techniques proposed in this thesis is the ASA as they are the analysts who conduct most of the reasoning with conflicting, uncertain, missing, and potentially false information.

2.1.4 Communicating Probability

Given the uncertainty inherent in intelligence analyses, the evaluated hypotheses and diagnostic information contained within a *finished intelligence product* might not always be correct or, as situations unfold, the true likelihood of a hypothesis could change faster than the *Intelligence Cycle* is able to produce intelligence reports. As such, it is necessary to communicate the probabilities and uncertainties that feature in the output of each analysis.

Describing numerical probability qualitatively is important when assessing the uncertainty in items of information and communicating the likelihood of evaluated hypotheses. Intelligence analysts often employ probability phrases, rather than explicit numbers, because numerical probabilities 'creates illusions of rigour, such that quantifying probability assessments would cause decision makers to see these estimates as being more scientific than they really are' [82, p. 804]. Many Western intelligence agencies have clearly defined standards for communicating probabilities, such as the North Atlantic Treaty Organisation [157], the United States of America [229], Canada

³Personal communication with H. Greenhough on 17th October 2022.

[44], and the United Kingdom [221], to name a few. While the intelligence agencies around the globe employ slightly different probability ranges to describe different verbal probabilities, they share a common goal: to clearly define the uncertainty featuring in the outputs of complex analyses, within an intelligence context. I present the United Kingdom's standard used by Defence Intelligence for probability communication, referred to as the Probability or Uncertainty Yardstick, in Table 2.1. The Probability Yardstick is one such tool that was developed to enable the qualitative communication of probability and uncertainty ranges.

Table 2.1: The Probability (or Uncertainty) Yardstick: a method of communicating verbal probabilities and uncertainties for numeric ranges within the unit interval [220].

Qualitative Probability	Quantitative Probability Range
Remote chance <i>or</i> highly unlikely	Less than 10%
Improbable <i>or</i> unlikely	15% – 20%
Realistic probability	25% – 50%
Probable <i>or</i> likely	55% – 70%
Highly probable <i>or</i> highly likely	75% – 85%
Almost certain	More than 90%

2.1.5 Structured Analytical Techniques for Intelligence Analysis

Structured analytical techniques are the most widely-known and frequently-used tools employed during analyses within the intelligence community. The tools were first developed by two former Central Intelligence Agency analysts, Heuer Jr. and Davis, in the 1970s [50]. After the terrorist attacks on 11th September 2001 and the invasion of Iraq in 2003 on the faulty premise that the country had weapons of mass destruction [58], the United States Congress enshrined in law the use of analytical techniques for all intelligence analyses, where appropriate [224].

Drawing upon some of the notions in the cognitive science literature, the tools' two most notorious proponents, Pherson and Heuer Jr., argue that the methods reduce a number of different cognitive biases by externalising a lot of the internal thought and reasoning processes when analysts are conducting analyses [173]. They argue that the main benefit of using such techniques is in their ability to elicit structured thinking in an intelligence domain, which is so often easily plagued by intuitive and implicit process. All the analytical techniques that I discuss in this thesis are underpinned by accounts of reasoning using dual-process theory, pioneered by Stanovich and West in

[208], which dissects reasoning, and thus analyses, into *System One* and *System Two* processes.

The general notion is that a human has a dichotomy of *minds* in one brain which compete during tasks ranging from simple multiplication to complex analyses. *System One* operates in a fast, automatic, and involuntary manner such that it is efficient and helps reasoners arrive at the correct conclusions for certain, easier situations, such as the answer to $5 \times 5 = X$, which we all intuitively know. When referring to someone's *System One* process, one might refer to their intuition or their ability to recognise and map patterns from previous events to current ones.

System Two, on the other hand, is required to arrive at the correct answer during difficult tasks, such as complex computations or even intelligence analyses. However, *System Two* thought is arduous such that it is only used when a person requires it and, even then, it is still tempting for a reasoner to defer to *System One*. In [116], the late Nobel Prize winner, Daniel Kahneman, explores the difference between *System One* and *System Two* processes through a series of basic examples that the reader must complete using, first, their intuition, which is often found wanting, and then Kahneman provides the reader with the answer, which is generally an answer such that only *System Two* could produce it. In the same spirit, I present one of Kahneman's examples which the reader should try to solve using intuition:

"A bat and ball cost \$1.10.

The bat costs one dollar more than the ball.

How much does the ball cost?" [116, p. 44]

At first glance, the answer is seemingly simple, the cost of the ball is 10¢. Although that answer is intuitive, it is, in fact, wrong. If the ball were to cost 10¢, then the bat would cost \$1.10, making the total \$1.20. The correct answer is 5¢ for the ball and \$1.05 for the bat. It is clear to see from this trivial example that reasoning can easily fall victim to lapses in analytical thought. How much more shall the intelligence reports resulting from analysis contain erroneous judgements from failures in intuition, if analysts are not guided by external processes that elicit *System Two* thinking? Thus, the example above should enable the reader to understand the motivation for structured analytical techniques.

Throughout the remainder of this subsection, I introduce a variety of structured analytical techniques employed within the intelligence community that are relevant to the subsequent chapters in this thesis, focusing mainly on hypothesis generation and evaluation.

2.1.5.1 Hypothesis Generation

I now introduce a number of different methods for multiple hypothesis generation, namely *Simple Hypotheses*, *Quadrant Hypotheses*, and the *Multiple Hypotheses Generator*TM. The choice of technique depends on the complexity of the situation. All the hypotheses generated from these techniques can be employed in downstream evaluation techniques, such as in the Analysis of Competing Hypotheses, which is discussed in Section 2.1.5.2.

Simple Hypotheses Generation. The Simple Hypotheses technique, presented in Figure 2.3, is a group exercise for multiple hypothesis generation that was created by Pherson and Heuer Jr. in [173]. To begin the technique, a group of analysts must come together and review the available information. The analysts should possess diverse expertise for different aspects of the problem, such as differing cultural backgrounds, perspectives, experiences, and styles of thinking. Each person is tasked with generating one to three mutually exclusive hypotheses in the *Idea Generation* and *Brainstorming* stage, writing each hypothesis on separate pieces of paper. Analysts should employ either situational logic (i.e., reasoning about the information and underlying driving forces in the scenario) or comparison with historical examples to produce alternative hypotheses. The alternative hypotheses from all the analysts are then consolidated, ensuring that there are no duplicates, and written on a shared display, such as on a whiteboard. The set of both possible and unique hypotheses are then combined into affinity groups, during the *Idea Grouping* phase. In [173], Pherson and Heuer Jr. then suggest reinstating the problem in different ways to consider opposing outcomes to potentially create new hypotheses. Analysts must then clarify each hypothesis by asking a series of journalistic questions, such as *Who*, *What*, *How*, *When*, *Where* and *Why*? Finally, the most promising hypotheses are appended to the final set of hypotheses, which are then employed in evaluation techniques.

Quadrant Hypothesis Generation. In [173], Pherson and Heuer Jr. proposed Quadrant Hypothesis Generation for instances where a situation has two key dimensions. Using the time prior to Russia's invasion of Ukraine as an example, the four hypotheses that an analyst might have generated using this method are presented in Figure 2.4. The method starts with an analyst, or a team of analysts, constructing a 2×2 matrix and identifying the two dimensions, which were *Russia's objectives* and the *impact on Ukraine* in the case of the example in Figure 2.4. Pherson and Heuer Jr. ask analysts to think of each dimension as continuous scales and to write the extremes of

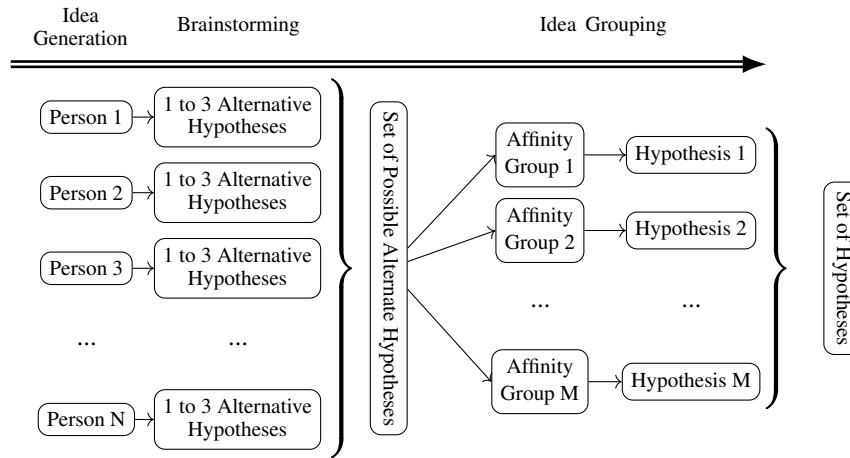


Figure 2.3: The Simple Hypotheses technique for multiple hypotheses generation (adapted from [173]).

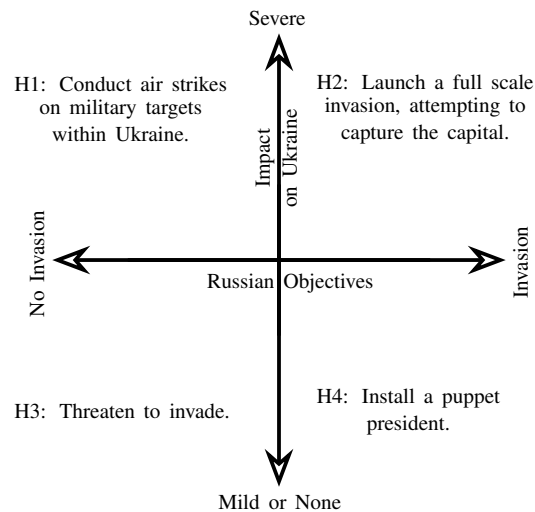


Figure 2.4: An example of the Quadrant Hypothesis Generation technique on the time prior to Russia's invasion of Ukraine.

each driver at the start and end of the horizontal and vertical axes. Analysts then use the dimensions to populate each quadrant with hypotheses. Each quadrant can contain more than one hypothesis and the hypotheses generated are to be viewed as potential end states. Analysts should then identify indicators that monitor whether the situation

is moving towards one of the end states. It is important to note that quadrant hypothesis generation is quicker than the other hypothesis generation methods examined in this thesis. However, this technique does not perform well when there are more than two key dimensions, or in cases where the two driving forces have not been correctly identified.

Multiple Hypothesis Generator™. In [173], Pherson and Heuer Jr. proposed a method for multiple hypothesis generation, called the Multiple Hypotheses Generator™, as shown in Figure 2.5. The procedure begins after the identification of a lead hypothesis. An analyst then enumerates through the components of the main hypothesis, using journalistic questions such as *Who*, *What* and *Why*? This results in a set of permutations which are all the potential hypotheses that one could generate from the lead hypothesis and available information. Analysts should try to ensure that the permutations are mutually exclusive and discard any hypotheses that are not possible. Analysts then score the credibility or plausibility of each permutation, sort the list of permutations from most to least plausible, restate the permutations as hypotheses, and then select the hypotheses from the list that require further analysis.

The three techniques for multiple hypothesis generation provide analysts with a structured approach to generate the set of MECE hypotheses. In [173], Pherson and Heuer Jr. argue that, when used properly, the above techniques reduce: confirmation bias because analysts are forced to generate more than one hypothesis, which might not necessarily be the hypothesis they initially believe to be true; the anchoring effect because analysts cannot assume that the lead hypothesis is always the proper starting point for an analysis; premature closure of analysis when an analyst identifies a hypothesis that is consistent with all the available information and concludes their analysis; and desire for uncertainty reduction and coherence such that analysts are supposedly not fooled into mapping the randomness of events to systematic and coherent stories that explain data. However, whether these benefits apply to analysts' analysis is questionable due to the lack of empirical studies to test these claims.

2.1.5.2 Hypothesis Evaluation

The intelligence community employs a variety of analytical techniques for hypothesis evaluation. As mentioned in Section 2.1.1, these methods can be split into the either *diagnostic*, *contrarian*, or *imaginative thinking* techniques [223]. Diagnostic techniques employ diagnostic reasoning which, in short, is the process of trying refute information and hypotheses instead of trying to confirm them. Examples of diagnostic techniques

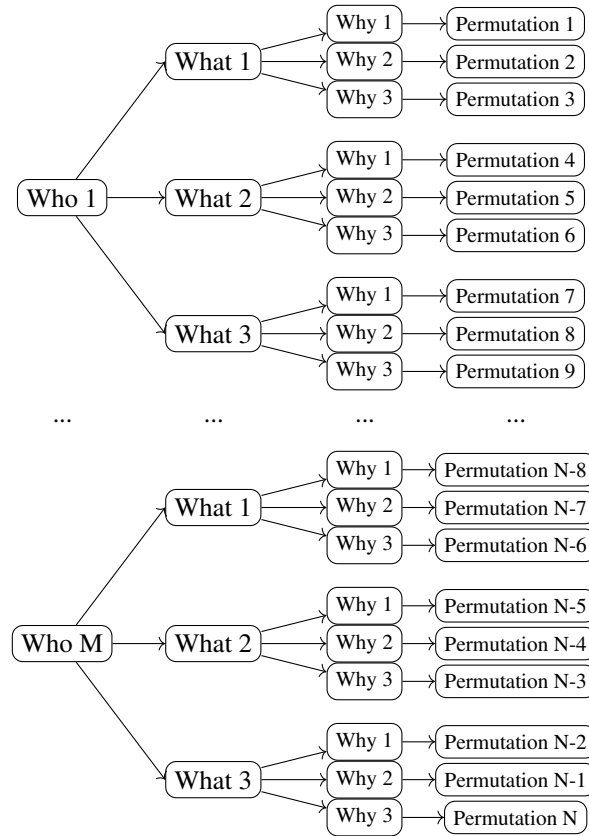


Figure 2.5: The Multiple Hypotheses Generator™ (adapted from [173]).

are the: *Key Assumptions Check* [172], which helps analysts list and evaluate the assumptions underpinning their analysis; *Quality of Information Check* [223], which enables analysts to evaluate the completeness and uncertainty in sources of information; *Indicators of Change* [223], which is where analysts regularly make observations to monitor events, track targets, and identify any unanticipated change; and the *Analysis of Competing Hypotheses* [101], which is a method of evaluating multiple hypotheses and discussed in more detail later on in this subsection.

Contrarian techniques aim to challenge the current thinking and mental models of analysts. Examples of contrarian techniques are: *Devil's Advocacy* [109, 199], which allows analysts to challenge a single, strongly-held view by providing coherent counterexamples; *Team A / Team B* [223], which splits analysts into two (or more) teams who debate about two (or more) competing hypotheses; *High-Impact/Low Probability Analysis* [172], which helps analysts think about rare events, such as a so-called *black*

swan event, and the consequence of such an event; and “*What If?*” analysis [172], which assumes that a future positive or negative event has occurred and works backwards to explain the sequence events that would be required for that event to happen.

Imaginative thinking techniques are crucial for stimulating innovative ideas, solutions, and hypotheses. Examples of imaginative thinking techniques are: *Brainstorming* [162, 172], which is a group task where analysts are given the freedom to generate new ideas and concepts; *Outside-In Thinking* [172], which is employed by analysts to identify driving forces and trends for a given topic so that they can assess a problem in light of outside factors; *Red Team Analysis* [223], which is where a team of analysts debate and try to model how an adversary would think about a scenario; and *Alternative Futures Analysis* [198], which is where analysts systematically explore the different ways that a situation can unfold.

The work proposed throughout the rest of this thesis draws upon one diagnostic technique, namely the Analysis of Competing Hypotheses. As such, the technique is introduced in the remainder of this subsection, along with some discussion pertaining to its benefits and limitations.

The Analysis of Competing Hypotheses. I now examine the Analysis of Competing Hypotheses (ACH) in more detail. The ACH is one of the earliest and, probably, most famous analytical tools, developed by Heuer Jr. in the 1970s [101]. It is a nine step procedure that analysts follow when evaluating a set of MECE hypotheses against a set of information, compiled during the *analysis* phase of the *Intelligence Cycle*, as discussed in Section 2.1.1. Put simply, the ACH structures hypotheses (as column headers) and information, evidence, and assumptions (as row headers) in the form of a matrix, where the consistency of row entries is denoted in each cell. Analysts then count how many row entries were inconsistent with each hypothesis which enables the quantification of the relative probabilities of each hypothesis within an analysis. Pherson and Heuer Jr. state that the technique can be employed by individual analysts or a team [173]. I present the nine steps involved in the ACH below.

1. **Identify hypotheses:** Identify the set of MECE hypotheses.
2. **List information:** List the significant and relevant information, evidence, any assumptions, and missing information that one would expect to be present if a hypothesis were true.
3. **Analyse diagnosticity:** Create a matrix and assess whether each row entry (i.e., the information, evidence, assumptions, and missing information) is consistent,

inconsistent, or not applicable with each hypothesis within the analysis.

4. **Review differences between analysts:** Review where there is disagreement between analysts and decide whether consistency scores should be changed.
5. **Refine hypotheses:** Refine the matrix by either removing or combining superfluous or overlapping hypotheses, respectively.
6. **Draw tentative conclusions:** Make tentatively-held judgements as to the probability of hypotheses by counting the number of inconsistent row entries for each hypothesis. The most likely hypothesis is the one with the fewest inconsistencies.
7. **Analyse sensitivity:** Analyse the sensitivity of the conclusions to identify how dependent the likelihood of each hypothesis is on the items of information, evidence, and assumptions.
8. **Report conclusions:** Report the probabilities of the hypotheses, as well as diagnostic row entries, to stakeholders, decision makers, or commanders.
9. **Identify indicators for future observation:** Identify indicators and milestones for future observations to ascertain whether a situation has evolved or deteriorated, such that the probabilities of hypotheses change.

The ACH has both advantages and disadvantages, with more recent literature highlighting many of the technique's problems. One of the advantages of the ACH is that it is easy to implement, especially in the context of intelligence analysis which is a cognitively demanding task – see Palo Alto Research Center's ACH software [164]. Another is that results from analyses are explainable due to the matrix instantiated in step three. Finally and probably the main benefit of the ACH, according to Pherson and Heuer Jr., is that the technique reduces the confirmation biases present during analyses as analysts evaluate the likelihood of hypotheses based on the number of row entries that are conflict in with them [173, p. 164-165]. However, the claim that the ACH reduces confirmation bias is somewhat weak because Pherson and Heuer Jr. do not provide any accompanying empirical evidence to support that claim. Moreover, a lot of the studies within the literature that tested whether the ACH reduced confirmation bias did not show any statistical significance to that effect [243, 141], apart from in one study where the ACH reduced confirmation bias of participants with no prior experience of analytical techniques or intelligence analyses [129].

Another major issue with the ACH is that it forces analysts to view row entries in isolation which, in turn, reduces the amount of belief chaining in the assessment of hypotheses. This can most likely be attributed to the topology of the matrix instantiated during analyses. However, considering items' consistency or otherwise with a hypothesis in isolation does not make complete sense because there might be cases where, for example, three row entries are deemed not applicable to a hypothesis when considered alone, but are in conflict with that same hypothesis when linked together. This is somewhat similar to a linked argument or collective attack, with a group of premises that either support or conflict with a hypothesis (or conclusion), respectively, as discussed more in Section 2.3.3.1.

Analysts also have to make many judgements about the consistency or otherwise of row entries against hypotheses. For example, an analyst would need to make one hundred judgements for an analysis with five hypotheses and twenty row entries. In the ACH's current state, the judgements made about each cell are ternary (i.e., each cell is deemed either consistent, inconsistent, or not applicable with a hypothesis). However, with some proposals arguing for more expressive diagnosticity scores [115], analysts might be required to assign, say, three probabilities to each cell within the matrix at some point in the future, which would not be practical with the ACH's current setup.

A further issue with ACH is that it models hypotheses using a flat structure, and does not account for the hierarchical nature of hypotheses. For example, a main hypothesis may have a number of sub-hypotheses that must be shown to be true for the overall hypothesis to be true. By flattening the hypotheses within the ACH, some of the nuanced reasoning about hypotheses and their interdependence is lost.

One final point which gives rise to the view that the ACH is not fit for operational use, in its current state, is that the procedure has been altered by the United Kingdom's Department for Defence Intelligence, in their book, titled *Quick Wins for Busy Analysts* [222]. They deviate from the original ACH process by employing both consistency and inconsistency scores in a coarse count to determine the probability of hypotheses. They also ask analysts to assign a credibility score, on a traffic light scale (i.e., red, yellow, and green), to each row entry. However, they do not utilise the uncertainty provided the credibility score, which means probabilities of hypotheses do not incorporate the uncertainty of row entries or their sources. Thus, the only tangible distinction between Heuer Jr.'s ACH and the process within the book, *Quick Wins for Busy Analysts*, is that they compute the relative probability of hypotheses by summing conflicting and confirmatory row entries.

While the orthodoxy of intelligence analysis is based on the scientific method, ac-

According to Janes⁴ analysts orthopraxy is such that they provide assessments of evolving situations, using evaluative techniques such as the ACH. However, these judgements are rarely accompanied by the relevant reasoning required to understand why an analyst made a particular judgement. This makes the work of replicating and automating this reasoning task, using, for example, machine and deep learning, or computational argumentation, extremely difficult. Future tools for intelligence analysis will require both the domain expertise of analysts and academic rigour of researchers to draw out the reasoning underpinning analyses.

2.2 Argumentation Theory

In this section, I introduce argumentation theory, touching upon the study of it, dating from antiquity to contemporary times, where all approaches share a common goal: to formalise patterns of reasoning to better understand how we, as humans, think and communicate through the exchange of arguments.

The study of argumentation dates back to classical times, where the earliest studies of argumentative practice were of the ancient Grecian democracy. Athenians participated in rich debate, in the form of long speeches, surrounding decisions pertaining to the governance of the city, where the importance of a citizen correlated with the persuasiveness of their argumentation [96]. Aristophanes, in his comedies *Wasps* (422 BC) [12] and *Clouds* (423 BC) [104], satirised the use of argumentation to manipulate juries and justify contentious claims, such as reneging on repayment of debts or even justifying the assault of parents by children. Plato, who was a child when these plays were performed, later advanced a more critical stance on Athenian rhetoric, arguing that debate should not merely aim at persuasion but rather seek truth of a particular topic through debate [105]. Plato was interested in *dialectical encounters* where speakers, with different points of view about a subject, would try to arrive at the truth through reasoned argumentation. While Plato was arguably one of the first to examine the mechanics of dialogues between interlocutors, Aristotle's last book, called *Sophistical Refutations*, within the series titled *Topics*, was one of the first attempts within the literature to formalise certain types of dialectical encounters, where agents, specifically a *questioner* and an *answerer*, engaged in the exchange of propositions [13]. The questioner's goal in this dialogue game was such that they would, first, extract a *starting point*, or a thesis, from the answerer and then the questioner would try to refute the answerer's position. The answerer's only objective was to defend their starting position

⁴Personal communication on 18th October 2022.

from attack within a given time frame.

The study of dialectics in ancient times provides some insight into the inspiration for studying argumentation today. Questions – such as *how to successfully defend a thesis from attack?* Or *what is a good starting point?* Or *which proposition should one attack to win the debate?* – are still important today because argumentation is domain-dependent so, for example, a successful attack in one discussion might not apply in another. As such, argumentation was defined by Eemeren *et al.* in [76] as follows:

“Argumentation is a communicative and interactional act complex aimed at resolving a difference of opinion with the addressee by putting forward a constellation of propositions the arguer can be held accountable for to make the standpoint at issue acceptable to a rational judge who judges reasonably.” [76, p. 7]

The irrationality of human agents makes argumentation within debate somewhat messy, in the sense that one might put forth a myriad of propositions in an effort to defend standpoint upon which they firmly believe, especially when that initial opinion is not held by another reasoner.

While argumentation is often employed within contexts where there is a difference of opinion between human agents, it features in all aspects of our lives. An argument, in its simplest form, consists of a premise (or a reason) that supports a conclusion (or a claim). For instance, imagine that a person planned to go for walk and it was raining. That person might sensibly say to themselves that *I should take a coat because it is raining* (Figure 2.6), which is, in fact, an argument put forth within the bounds of their mind. That same person might own more than one coat such that they reason using arguments containing preferences to decide which coat to wear. Finally, ending this simple example, that person might enter a shop whilst on their walk and see two coats that they would like to buy, but one of them was made in an unethical manner such that the notion of values is incorporated into their reasoning about which coat to buy. While there are many more trivial examples that the reader could think of, such as which conserve to spread on their toast before work in the morning or how much milk to put in their coffee, it is easy to see that argumentation plays an important role in the thoughts, actions, and discussions that we, as humans, are frequently involved in.

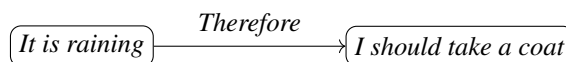


Figure 2.6: A simple argument containing a single premise supporting a conclusion.

Arguments put forth in situations within our daily lives can be far-more complex than a single premise supporting a conclusion. An argument, therefore, can contain a set of premises, ranging from one to many, that leads to a single conclusion. Although the inference between a set of propositions can be different depending on the circumstance, much work within the literature has been completed to map arguments into commonly used inference patterns, which I introduce in more detail in Section 2.2.1.

Argumentation is an extremely useful tool in contexts where the ground truth cannot be used, such as in domains where there is incomplete, uncertain, or conflicting information, like within an intelligence setting. Arguments are tentative proofs for propositions [81, 124] that are capable of aiding defeasible reasoning, where new arguments can be instantiated and reasoned with non-monotonically as new information comes to light. The premises contained within an argument offer a set of presumably-true reasons to believe the claim being made. Usually, the conclusion of an argument deductively holds, if an audience accepts the premises and the inference rule between those premises and the conclusion. However, a listener might reject or attack one or all of the premises, the inference rule, or conclusion. Argumentation, therefore, is a subjective process that varies from person to person and incorporates a reasoner's knowledge and beliefs.

While there are many approaches to represent arguments within the literature, this thesis employs argumentation schemes for presumptive reasoning and abstract accounts of argumentation, which are discussed in more detail later in this section and in Section 2.3, respectively. Arguments are modelled using a set of premises, an inference rule, and a conclusion. The defeasible nature of arguments means that these schemes are not deductively strict, in that although a conclusion is presumed true when the premises hold, that same conclusion might be found to be false for the same set of true premises.

Argumentation has been studied for millennia, as I discussed earlier in this section. A recent example of an argumentation scheme is Toulmin's graphical depiction of an argument, as shown in Figure 2.7, which allows for a more expressive set of premises. Toulmin's description of an argument discretises its constituent propositions into the six following elements:

- a *claim* which is synonymous with the conclusion of an argument;
- the *grounds* which are the premises, in the form of evidence and facts, which support the claim;
- the *warrant* which is an implicit or explicit assumption that links the grounds to the claim;

- a *qualifier* which assigns a strength to a claim;
- a *rebuttal* which is a proposition that is in conflict with the claim; and
- the *backing* which provides additional support for the warrant.

In Toulmin’s model, the fundamental propositions within an argument are the claim, grounds, and warrant. However, the qualifier, rebuttal, and backing can be added to the argument, if necessary. For instance, the assertion that one’s claim might not always be true, in the case of an added qualifier, builds credibility. Alternatively, stating that there are multiple conclusions to be drawn from an argument, in the case of a rebuttal, paints the arguer as an unbiased thinker.

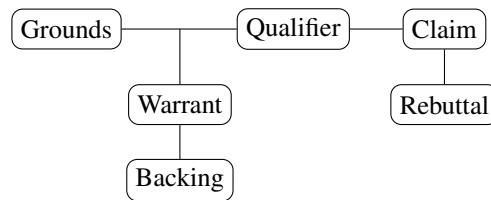


Figure 2.7: A depiction of Toulmin’s Model of Argument [217].

Toulmin’s model is not without its issues. Conflicts within the schema are not well-defined such that the only proposition that can be attacked is the claim, through use of a rebuttal. In real-world argumentation, propositions can conflict with a claim (in the form of a rebutting attack), premise (in the case of an undermining attack), or the inference between a set of premises supporting a conclusion (in the case of an undercutting attack), which are concepts that I discuss in more detail in Section 2.3.2.1 when introducing the ASPIC⁺ framework [177, 150].

Toulmin’s account of argument could be employed for the instantiation of arguments within intelligence settings. It is possible to represent hypotheses as claims supported by observations (or grounds), in accordance with Pherson and Heuer Jr.’s definition of a hypothesis, as stated in Chapter 1. In addition to this, the qualifier in Toulmin’s model of argument could be employed to communicate the strength (i.e., probability or uncertainty) of a hypothesis (or claim) that features in a *finished intelligence product*, as discussed in Section 2.1.4. However, Toulmin’s schema does not allow for the description of uncertainty relating to the grounds supporting a claim which is ever-present in data within intelligence analysis. Although Toulmin’s seminal model of argument has been highly influential, it is not suitable for use within intelligence settings due to the aforementioned reasons.

Since Toulmin's germinal idea, many more representations of arguments have been proposed, such as Walton's argumentation schemes [235], along with their accompanying critical questions which describe the common attacks between propositions and inference rules. In the next subsection, I introduce Walton's argumentation schemes and examine one scheme in more detail – the argument from cause to effect [236] – because of the link between that scheme and the reasoning involved in arguments containing hypotheses.

2.2.1 Argumentation Schemes

Argumentation schemes are powerful tools that can model deductive as well as inductive arguments, and are also able to model defeasible, presumptive, and abductive arguments, which do not possess an inductive or deductive form. Argumentation schemes are an abstract representation of arguments that capture the stereotypical patterns of inference between a set of premises which support a conclusion. The schemes are templates that model propositions and commonly-used inference rules in a variety of domains, such as in legal [248, 24] and intelligence contexts [215], amongst others.

More pertinently and inline with the topics addressed in this thesis, the schemes have been shown to be capable of enabling hypothetical reasoning, which is a type of reasoning that intelligence analysts conduct on a regular basis when, for example, assessing what might happen in the future using available data, or deciding whether they have identified the set of all plausible hypotheses. In [19], Bench-Capon and Prakken model the reasoning between a proponent and opponent in a specific legal case (*California vs. Carney*), and explore whether a warrant would be needed to search a vehicle by asking a series of hypothetical questions pertaining to the facts of the case. This instance of reasoning is somewhat similar to the *What If?* analysis [172], as discussed in Section 2.1.5, where intelligence analysts must envision a future scenario and then work backwards, plotting the course of events required to get to that future point.

Some of the first and most important attempts to formalise these patterns of reasoning were proposed by Hastings in [98], Perelman and Olbrechts-Tyteca in their book titled *The New Rhetoric* [171], and Kienpointner in [121]. Since then, Walton's vast body of work on argumentation schemes [235] has paved the way for the interdisciplinary study of computational models of argument within different domains by enabling researchers to identify and represent arguments. The schemes proposed by Walton are accompanied by a set of critical questions which enables the identification of attacks between arguments; thus, allowing for the evaluation of the acceptability

of those arguments in relation to others within that specific problem setting. Due to complex nature of language and argumentation, especially within intelligence settings, there are a plethora of argumentation schemes that have been identified by this research, such as the argument from cause to effect [236], argument from negative consequences [237], practical reasoning schemes [241], argument from (falsification) of evidence to a hypothesis [239], argument from position to know [240], argument from established rule [238], etc. The aforesaid schemes were employed in the argument annotation of a real-world example and a description of that process is provided in the beginning of Part II. In the subsection to follow, I only consider the argument from cause to effect because it was employed to capture the reasoning underpinning hypotheses in the introduction to Part II and forms the basis of the scheme proposed in Chapter 3.

2.2.1.1 The Argument from Cause to Effect

As stated in Chapter 1, Pherson and Heuer Jr. define a hypothesis as a testable and falsifiable statement that explains available data in a clear and accurate manner. Now, I turn to a more recent, argumentation-based definition of a hypothesis, given by Toniolo *et al.* in [216]:

“A *hypothesis* in intelligence analysis is composed of activities and events that show how the situation has evolved. The argument from cause to effect forms the basis of these hypotheses.” [216, p. 785]

Here, Toniolo reflects Pherson and Heuer Jr.’s definition of a hypothesis, where the reasoning for (or against) a hypothesis is based on the information available to an analyst. The concluding sentence in the quote above highlights that the reasoning between a set of information and a hypothesis can be modelled through the argument from cause to effect [236], which is presented below:

Major Premise: Generally, if *A* occurs, then *B* will (might) occur.

Minor Premise: In this case, *A* occurs (might occur).

Conclusion: Therefore, in this case, *B* will (might) occur.

Walton identifies three critical questions for this scheme:

WCQ1: How strong is the causal generalisation?

WCQ2: Is the evidence cited (if there is any) strong enough to warrant the causal generalisation?

WCQ3: Are there any other causal factors that could interfere with the production of the effect in the given case?

The argument from cause to effect, with its accompanying set of critical questions, provides a good abstraction of the inference between a set of information supporting a hypothesis. Data collected and transformed into information, through *Processing and Exploitation* techniques within the *Intelligence Cycle* (Section 2.1.1), is synonymous with a *cause*, or set of *causes*, where its *effect* is the assertion of a hypothesis which explains those data. The defeasible nature of the scheme is such that as more data is collected, new arguments containing hypotheses can be instantiated and reasoned with. Furthermore, the critical questions provided should enable a reasoner to seriously consider whether there are any other items of information, evidence, and assumptions or arguments that attack either: the inference between those premises and the hypothesis they support (WCQ1); the information, evidence, and assumptions (WCQ2); or another causal factor which supersedes the initial generalisation (WCQ3). The scheme does not provide any critical questions pertaining to potential attacks on the conclusion of the argument. However, in this instance, it is easy to see that, within the context of intelligence analysis, symmetric attacks could sensibly be instantiated between the set of MECE hypotheses because that set contains exhaustive and non-overlapping outcomes such that if one hypothesis were found to be true, then all others would be false. Furthermore, the scheme touches upon a notion of uncertainty in all three propositions (i.e., in the *Major Premise*, *Minor Premise* and *Conclusion*) through the assertion that something *will* or *might* occur. To employ such a scheme within an intelligence setting, the communication of probability and uncertainty in each item of information, evidence, and assumptions, as well as the hypothesis they support, is paramount (Section 2.1.4). In Chapter 3, I provide a more detailed examination of the critical questions and the notion of uncertainty in propositions for this argumentation scheme, resulting in a proposed adaption of the scheme so that it is suitable for use within the intelligence domain.

2.3 Computational Argumentation

The field of computational models of argument is a growing interdisciplinary area of research that spans philosophy, linguistics, computer science, and psychology. The fecundity of ideas within the literature of this youthful field can be divided into a trichotomy that captures a diverse range of models for argumentation, based on the domain and type of argumentation being assessed. In [21], Bentahar *et al.* organise the

three distinct types of model into the following:

- *Monological models*, which study the internal structure of arguments and counterarguments generated from a knowledge base or provided by a third party, where the goal is study the connection between a set of premises and a conclusion, linked by an inference rule. These models examine the different components within an individual argument such that they focus on the micro-structure of arguments, hence the name *monological*, with both Toulmin’s graphical depiction of arguments (Section 2.2) and Walton’s argumentation schemes (Section 2.2.1) falling into this category.
- *Dialogical models* focus on the interchange of attacks between arguments, paying no attention to the internal structure of arguments. Dialogical models are concerned with the macro-structure of arguments and their acceptability such that these models are referred to as defeasible reasoning systems.
- *Rhetorical models* focus on discourses that are *rhetorically* argumentative, paying attention to an audience’s perception of an orator’s argumentation, rather than to achieve general acceptability, as is the case with *monological* and *dialogical* models.

For the remainder of this section, I introduce a particularly influential dialogical model – Dung’s work on abstract argumentation frameworks – and explain how to conduct defeasible reasoning to identify admissible sets of arguments, using both extension- and labelling-based semantics, in Section 2.3.1. Then, in Section 2.3.2, I examine the topic of structure argumentation, which is where both *monological* and *dialogical* models coincide, by introducing the ASPIC⁺ framework, the Argument Interchange Format, and show how to transform annotated argument data into Dung’s formalism. Finally, I study the topic of argument mining, in Section 2.3.3, focusing mainly on manual approaches, but also consider automated methods, with a view to assessing the feasibility and addressing the limitations of argument mining, in its current stage of development within the literature, for intelligence analysis.

2.3.1 Abstract Argumentation Frameworks

Dung’s seminal work on abstract argumentation frameworks made a significant contribution to the field of computational argumentation and non-monotonic reasoning [70]. The underlying notion of his proposal was that arguments and attacks between them can be modelled using a directed graph, where the arguments and attacks are

represented as nodes and edges, respectively. The acceptability of sets of arguments is determined by employing a variety of semantics, where the semantics chosen by a user depends on how sceptical they desire their reasoning to be. This abstraction has been extended to frameworks that can model: support as well as attack [48]; preferences [8] and values [20]; collective attacks from sets of arguments [156]; collective supports and attacks based on evidence [161]; temporal argumentation frameworks, which consider availability intervals for arguments, attacks, and their acceptability [55, 56, 57]; probabilistic argumentation frameworks, such as in the epistemic [212, 213] and constellation [69, 133] approaches, which quantify the uncertainty in arguments and attacks, and thus incorporate these uncertainties into the evaluation of admissible sets of arguments; weighted argument systems [71] that can model the relative strength of attacks through the incorporation of the notion of an inconsistency budget, which represents how much inconsistency is tolerable within an argumentation framework; and abstract dialectical frameworks [35], which provide a richer explanation of the acceptability of arguments through use of an acceptance condition.

Below I provide the formal definitions for Dung's original abstract argumentation framework.

Definition 2.1 A (finite) Dung argumentation framework \mathcal{G} is a tuple $(\mathcal{A}, \mathcal{R})$ which contains a set of arguments \mathcal{A} and binary attack relations $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$ between arguments. For two arguments $a_1, a_2 \in \mathcal{A}$, the argument a_1 attacks a_2 if and only if $(a_1, a_2) \in \mathcal{R}$.

An example of a Dung-style argument graph, comprised of arguments \mathcal{A} (Equation 2.1) and relations \mathcal{R} (Equation 2.2), is presented in Figure 2.8.

$$\mathcal{A} = \{a_1, a_2, a_3, a_4, a_5\} \quad (2.1)$$

$$\mathcal{R} = \{(a_1, a_2), (a_2, a_1), (a_3, a_1), (a_4, a_2), (a_4, a_5), (a_5, a_4)\} \quad (2.2)$$

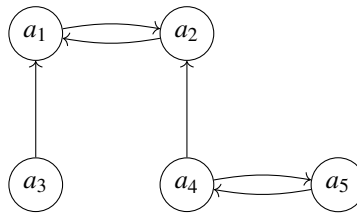


Figure 2.8: An example of a Dung-style abstract argumentation framework.

The evaluation of abstract argumentation frameworks is computed using semantics, which are operations that group sets of *admissible* arguments into either a set of extensions or labellings. *Admissible* sets of arguments adhere to two fundamental properties: *conflict-freeness* and *acceptability*. A set of arguments that possesses the characteristic of *conflict-freeness* is a set which contains arguments that do not attack each other.

Definition 2.2 Let $\mathcal{G} = (\mathcal{A}, \mathcal{R})$ be a Dung argumentation framework and $S \subseteq \mathcal{A}$ be a set of arguments. The set S is *conflict-free* if and only if $\nexists (a_1, a_2) \in \mathcal{R}$ where $a_1, a_2 \in S$.

The notion of *acceptability* considers whether an argument is *defended* against attack by the arguments in its set. An argument within a set of arguments is *defended* if at least one of the arguments within the set attacks an attacker of that argument. An argument is referred to as *acceptable* with respect to its set, if and only if it is defended by an argument within that set.

Definition 2.3 Let $(\mathcal{A}, \mathcal{R})$ be an argumentation framework and $S \subseteq \mathcal{A}$ be a set of arguments. An argument $a_1 \in S$ is *acceptable* with respect to S if and only if $\forall a_2 \in \mathcal{A}$ such that $(a_2, a_1) \in \mathcal{R}$, $\exists a_3 \in S$ such that $(a_3, a_2) \in \mathcal{R}$.

Since the inception of Dung's germinal contribution, many further semantics that uphold the principles of conflict-freeness and acceptability have been proposed [17]. The multitude of semantics in the literature provide a method of automated reasoning which changes depending on how sceptical the user is. Semantics can be broken down into two categories, namely extension- and labelling-based approaches. In short, both methods present equivalent outputs from semantic evaluation, however they are represented in different ways. Extension-based semantics output subsets of admissible arguments, called *extensions*, whilst labelling-based methods assign an argument label to each argument within an argumentation framework, where the set of labelled arguments are stored within sets of *labellings*. Both formalisms, *extensions* and *labellings*, were shown to be analogous to each other [43] such that they contain the same sets of conflict-free and acceptable arguments for an evaluation of a given framework, however it is the way in which these data are represented which differs. Both extensions and labellings were utilised in the work presented throughout this thesis. As such, both approaches are introduced below, starting with extension-based semantics.

Definition 2.4 For an argumentation framework $\mathcal{G} = (\mathcal{A}, \mathcal{R})$ and a set of arguments $S \subseteq \mathcal{A}$, the set S is referred to as a

- *Complete extension* if and only if S is conflict-free and $\forall a_1 \in S, a_1$ is acceptable with respect to S ; thus, a complete extension is an *admissible* one;
- *Grounded extension* if and only if S is a minimal (with respect to set inclusion) complete extension;
- *Preferred extension* if and only if S is a maximal (with respect to set inclusion) complete extension;
- *Stable extension* if and only if S is a preferred extension and $\exists a_1 \in S$ such that $(a_1, a_2) \in \mathcal{R}, \forall a_2 \notin S$.

All the semantics introduced in Definition 2.4 relate to each other and adhere to the two fundamental characteristics, acceptability and conflict-freeness. Sets of admissible arguments form the foundation on which other acceptable sets of arguments can be discovered, such that every stable extension is a preferred extension, every preferred or grounded extension is a complete extension, every complete extension is an admissible set of acceptable and conflict-free arguments. The different types of semantics allows computational agents to carry out sceptical and credulous reasoning, which will be discussed later in this section. For example, sceptical reasoning is conducted through computation of extensions using grounded semantics that returns only one extension which contains the intersection of arguments that feature in all the complete extensions. In the running example presented in Figure 2.8, there were three complete extensions

$$\{a_2, a_3, a_5\}, \{a_3, a_4\} \text{ and } \{a_3\}, \quad (2.3)$$

where the first two sets of arguments are preferred and stable extensions, and the last set, containing the argument a_3 only, is the grounded extension.

Moving onto to labelling-based semantics, which were first proposed within the argumentation literature by Caminada in [42, 41], and subsequently formalised by Caminada and Gabbay in [43]. The general notion of labelling-based semantics is that, while admissibility – computed using complete, grounded, preferred, or stable semantics – was defined in terms of sets of arguments, it is also possible to employ argument labels to describe the admissibility of all arguments within a framework.

Definition 2.5 Let $\mathcal{G} = (\mathcal{A}, \mathcal{R})$ be an argumentation framework and $S \subseteq \mathcal{A}$ be a set of arguments. A *labelling* is a total function $\mathcal{L}(S) : S \mapsto \text{LAB}$ that assigns all arguments $a \in S$ to a label $l \in \text{LAB}$, where $\text{LAB} = \{\text{IN}, \text{OUT}, \text{UND}\}$.

For the purpose of introducing labelling-based semantics, the set of arguments that attack the argument $a_i \in \mathcal{A}$ is defined in Equation 2.4.

$$Att(a_i) = \{a_j \mid (a_j, a_i) \in \mathcal{R}\} \quad (2.4)$$

Similar to extension-based semantics, a labelling can be complete, grounded, preferred, or stable. A complete labelling corresponds to a complete extension if it follows Definition 2.6.

Definition 2.6 For an argumentation framework $\mathcal{G} = (\mathcal{A}, \mathcal{R})$ and a labelling function $\mathcal{L}(\mathcal{A}) : \mathcal{A} \mapsto \text{LAB}$, a *labelling* is referred to as a *complete labelling* if and only if for every $a_i \in \mathcal{A}$, it holds that:

1. if $\mathcal{L}(a_j) = \text{OUT}$, $\forall a_j$ such that $a_j \in Att(a_i)$, then $\mathcal{L}(a_i) = \text{IN}$;
2. if $\exists a_j \in Att(a_i)$ and $\mathcal{L}(a_j) = \text{IN}$, then $\mathcal{L}(a_i) = \text{OUT}$.

Due to the fact that a labelling is a total function, arguments $a_i \in \mathcal{A}$ that are neither labelled IN nor OUT are labelled UND.

Depending on the type of semantics employed, the labelling function can return more than one more labelling. Labelling functions that produce more than one labelling, containing argument labels for all arguments within argument graphs, feature in some of the techniques, tools, algorithms introduced in Chapters 5 and 6 of this thesis. With a slight abuse of notation, a labelling of an argumentation framework which produces more than one labelling is a set of argument labels referred to as

$$\mathcal{L}_N(\mathcal{G}) = \{\mathcal{L}_i(\mathcal{A})\}_{i=1}^N \quad (2.5)$$

where $\mathcal{L}_N(\mathcal{G})$ is a set of N labellings and $\mathcal{L}_i(\mathcal{A})$ is the i -th labelling containing argument labels for all arguments within the framework.

As with extension-based semantics, the definition of a complete labelling can be employed in the definition of other types of labellings.

Definition 2.7 Let $\mathcal{G} = (\mathcal{A}, \mathcal{R})$ be an argumentation framework and $\mathcal{L}_N(\mathcal{G})$ be a set of labellings. A labelling $\mathcal{L}_i(\mathcal{A}) \in \mathcal{L}_N(\mathcal{G})$ is defined as:

- *grounded* if and only if it is a complete labelling with a maximal set of arguments labelled UND;
- *preferred* if and only if it is a complete labelling with a maximal set of arguments labelled IN; and

- *stable* if and only if it is a complete labelling where the set of arguments labelled UND is an empty set.

Continuing the running example for the argumentation framework presented in Figure 2.8 and following the constraints in Definition 2.7, the second labelling $\mathcal{L}_2(\mathcal{A})$ presented in Table 2.2 is a grounded labelling because it possesses a maximal set of arguments labelled UND. Furthermore, the argument a_3 is the only argument within the grounded extension (i.e., $\{a_3\}$ in Equation 2.3) and it is also the only argument labelled IN within the grounded labelling. The first and third labellings, $\mathcal{L}_1(\mathcal{A})$ and $\mathcal{L}_3(\mathcal{A})$, are both preferred and stable labellings because they contain a maximal set of arguments labelled IN and the set of arguments labelled UND is an empty set, respectively. Looking back to the sets of preferred and stable extensions (i.e., $\{a_2, a_3, a_5\}$ and $\{a_3, a_4\}$ in Equation 2.3), the arguments that feature in both extensions are also labelled IN within the first and third labelling. Thus, it is clear to see that complete extensions and labellings form the foundations for all other extension- and labelling-based semantics, respectively. Furthermore, the set of complete, grounded, preferred, and stable labellings are equivalent to their extension-based counterparts.

Table 2.2: The complete labellings of the argumentation framework \mathcal{G} (Figure 2.8).

Labellings, $\mathcal{L}(\mathcal{G})$	a_1	a_2	a_3	a_4	a_5
$\mathcal{L}_1(\mathcal{A})$	OUT	IN	IN	OUT	IN
$\mathcal{L}_2(\mathcal{A})$	OUT	UND	IN	UND	UND
$\mathcal{L}_3(\mathcal{A})$	OUT	OUT	IN	IN	OUT

Although the computation of a semantics enables the evaluation of the acceptability of arguments within an argumentation framework, it does not wholly capture the justification status of arguments across the sets of extensions or labellings. For instance, in the case of extension-based semantics, an argument contained in all the sets of extensions might be viewed as a *stronger* argument because it is both acceptable and conflict-free across all extensions. For labelling-based semantics, an argument that possesses the same argument label across the set of all labellings might cause a human agent to question the acceptance status of such an argument. On the contrary, an argument that only features in one extension out of a set of extensions or possesses an argument label that only features in one labelling across a set of labellings might be deemed *weaker* by human agent due to there being less reasons to believe it. The notion of sceptical and credulous acceptance mathematically defines the two aforementioned

concepts with regards to abstract argumentation, where an argument's justification status depends on its membership to sets within the set of extensions, or an argument's labels across a set of labellings.

Definition 2.8 (Sceptical / Credulous Acceptance in Extension-Based Semantics).

Let $\mathcal{G} = (\mathcal{A}, \mathcal{R})$ be an argumentation framework and \mathcal{E} be a set of extensions discovered through computation of some semantics \mathcal{S} . An argument $a_i \in \mathcal{A}$ is:

- *sceptically accepted* if and only if $a_i \in e_j, \forall e_j \in \mathcal{E}$;
- *credulously accepted* if and only if $\exists e_j \in \mathcal{E}$ such that $a_i \in e_j$.

Definition 2.9 (Sceptical / Credulous Acceptance in Labelling-Based Semantics).

Let $\mathcal{G} = (\mathcal{A}, \mathcal{R})$ be an argumentation framework, $\mathcal{L}_N(\mathcal{G}) = \{\mathcal{L}_i(\mathcal{A})\}_{i=1}^N$ be a set of labellings discovered through computation of some semantics \mathcal{S} , and $l \in \text{LAB}$ be the same argument label l out of the set of possible labels LAB, where $\mathcal{L}_i(a_j)$ denotes the label of the j -th argument in the i -th labelling. The argument $a_j \in \mathcal{A}$ is:

- *sceptically labelled* if and only if $\mathcal{L}_i(a_j) = l, \forall \mathcal{L}_i(\mathcal{A}) \in \mathcal{L}_N(\mathcal{G})$;
- *credulously labelled* if and only if $\exists \mathcal{L}_i(\mathcal{A}) \in \mathcal{L}_N(\mathcal{G})$ such that $\mathcal{L}_i(a_j) = l$.

Arguments that are labelled IN across all labellings are referred to *sceptically-IN*, whereas arguments that are labelled IN for one labelling are referred to as *credulously-IN*. This naming convention also extends to arguments labelled OUT and UND.

An interesting point to note is that semantics or argument graph topologies which produce a single extension or labelling are where sceptical and credulous acceptance coincide.

Dung's abstract argumentation frameworks are an excellent choice of model for use within intelligence scenarios. The frameworks enable automated and defeasible reasoning with conflicting arguments, where those arguments might contain propositions that instantiate items of information, evidence, assumptions, hypotheses, and even the inference between them. The defeasibility of abstract argumentation frameworks is a consequence of those frameworks being directed graphs where arguments can be added or removed and the semantics computed again, ensuring that new sets of conclusions are based on the most recent data, as new information comes to light. Furthermore, techniques employed within the intelligence community, such as the ACH (as discussed in Section 2.1.5), try to reduce confirmation bias by assessing the likelihood of hypotheses as a function of how inconsistent each hypothesis was with each row entry in the matrix. Dung's abstract formalism mirrors this approach by assessing the acceptability of arguments using attacks between arguments only.

2.3.2 Structured Argumentation

While Dung's seminal account of abstract argumentation allows for the identification of sets of admissible arguments, its level of abstraction means it pays no attention to the internal structure and acceptability of those arguments. Although this characteristic enables the evaluation of Dung-style argument systems and the acceptability arguments thereof, it precludes the formalism alone from being employed in real-world applications and situations. Thus, *structured argumentation* was developed to formalise arguments' internal structure and acceptability.

In *structured argumentation*, arguments and attacks are instantiated using a formal language. The arguments within structured argumentation frameworks contain premises and conclusions of arguments which are made explicit and contained within the formal language, where the inferences and attacks between these propositions are formally defined. There are many approaches to structured argumentation, such as assumption-based argumentation frameworks [30], defeasible logic programming [202], the Carneades model [88] of argument structure and evaluation, and the ASPIC⁺ framework [177, 150]. However, only the latter is introduced as it was the only approach employed in the testing and evaluation of tools and algorithms that utilised abstract argumentation applied to real-world problems in the subsequent chapters of this thesis.

2.3.2.1 The ASPIC⁺ Framework

One such structured framework is the ASPIC⁺ framework, which adopts an intermediate level of abstraction to provide an abstract account on the acceptability of structured arguments [177, 150]. The ASPIC⁺ framework can model structured argumentation problems using strict and defeasible inference rules, with preferences between defeasible rules, as well as a knowledge base which contains sets of necessary axioms (or facts), ordinary premises, and assumptions. However, I restrict the ASPIC⁺ framework so that it is only comprised of a knowledge base containing the set of ordinary premises and defeasible inference rules without preferences, as they are the only conditions that pertain to the work presented throughout this thesis.

Definition 2.10 ([177]) An *argumentation system* is tuple $AS = (\mathcal{L}, ^-, \mathcal{R})$ where

- \mathcal{L} is a logical language;
- $-$ is a contrariness function $- : \mathcal{L} \mapsto 2^{\mathcal{L}}$; and
- \mathcal{R} is a set of defeasible rules.

Definition 2.11 ([177]) Let \mathcal{L} be a logical language and $\bar{\cdot}$ be a contrariness function, where ϕ and ψ are statements within that language, such that $\phi, \psi \in \mathcal{L}$, and $\bar{\phi}$ and $\bar{\psi}$ are sets containing statements that conflict with ϕ and ψ , respectively. It follows that

- ϕ is called a *contrary* of ψ if and only if $\phi \in \bar{\psi}$;
- ϕ and ψ are *contradictory* if and only if $\psi \in \bar{\phi}$ and $\phi \in \bar{\psi}$, denoted by $\phi = -\psi$.

The arguments formed using the ASPIC⁺ framework are defined as inference trees which are created by applying defeasible inference rules on objects within the logical language. I refer to $p \Rightarrow q$ as a defeasible rule $r \in \mathcal{R}$, where p is the antecedent and q is the consequent.

Definition 2.12 ([177]) For the restricted version of an ASPIC⁺ argumentation system $(\mathcal{L}, \bar{\cdot}, \mathcal{R})$ presented in this thesis, a *knowledge base* \mathcal{K} is a set of *ordinary* premises \mathcal{K}_p , such that $\mathcal{K} \subseteq \mathcal{L}$ and $\mathcal{K} = \mathcal{K}_p$.

Definition 2.13 ([177]) An *argumentation theory* is a pair $AT = (\mathcal{K}, AS)$ where \mathcal{K} is a knowledge base containing the set of ordinary premises \mathcal{K}_p only, such that $\mathcal{K} = \mathcal{K}_p$, and AS is an argumentation system.

Arguments are derived from the knowledge base, where each argument A is obtained from the set of ordinary premises within the knowledge base \mathcal{K}_p of an argumentation theory AT ; $Prem(A)$ returns all the ordinary premises within \mathcal{K}_p which support A , $Conc(A)$ returns A 's conclusion, and $Sub(A)$ returns all of A 's sub-arguments.

Definition 2.14 ([177]) Let \mathcal{K} be a knowledge base in an argumentation system $(\mathcal{L}, \bar{\cdot}, \mathcal{R})$. An argument A is defined as

- $A = \{\phi\}$ if and only if $\phi \in \mathcal{K}$ where $Prem(A) = \{\phi\}$, $Conc(A) = \{\phi\}$, and $Sub(A) = \{\phi\}$.
- $A = \{A_1, \dots, A_n \Rightarrow \psi\}$ if and only if A_1, \dots, A_n are arguments and there exists a defeasible rule in the argumentation system AS such that $Conc(A_1), \dots, Conc(A_n) \Rightarrow \psi \in \mathcal{R}$; $Prem(A) = Prem(A_1) \cup \dots \cup Prem(A_n)$; $Conc(A) = \psi$; and $Sub(A) = Sub(A_1) \cup \dots \cup Sub(A_n) \cup \{A\}$.

Attacks from one argument to another are represented through the contrariness function, and successful attacks are defined as defeats. The ASPIC⁺ framework can model *rebutting*, *undermining*, and *undercutting* attacks. A *rebutting* attack is one where an argument attacks the conclusion of another. An *undermining* attack is one

where the conclusion of an argument is contrary to the premise of another. An *undercutting* attack is one where an argument's defeasible inference from a set of premises to a conclusion is attacked by another argument. While both rebutting and undercutting attacks can only feature in structured argumentation that allows for defeasible inference rules, undercutting attacks are not included in the new work presented in this thesis. Furthermore, as the ASPIC⁺ formalism has been restricted to the set of defeasible rules without preferences, all conflicts that feature in the sets output by the contrariness function are deemed to be defeats, as per in Definition 2.15.

Definition 2.15 ([177]) For the arguments $A, B \in \mathcal{K}$, the argument A *defeats* B , when:

- A *rebutts* B (on B') if and only if $\text{Conc}(A) \in \overline{\psi}$ for some $B' \in \text{Sub}(A)$ where $B' = \{B''_1, \dots, B''_n \Rightarrow \psi\}$;
- A *undermines* B (on ψ) if and only if $\text{Conc}(A) \in \overline{\psi}$ for some $B' = \psi$, $\psi \in \text{Prem}(B)$.

As in [177], structured argumentation theories along with the set of defeats can be employed in the instantiation of Dung abstract argumentation graphs. Remembering that a Dung-style argument system is a tuple $(\mathcal{A}, \mathcal{R})$ with a set of arguments \mathcal{A} and defeats $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$, as defined in Section 2.3.1.

Definition 2.16 ([177]) A Dung argumentation framework \mathcal{G}_{AT} corresponding to an argumentation theory AT is a pair $(\mathcal{A}, \mathcal{R})$ with a set of arguments \mathcal{A} and relations \mathcal{R} instantiated using the respective arguments (Definition 2.14) and defeats (Definition 2.15) within the theory.

Once the ASPIC⁺ framework has been translated to a Dungian graph, all the well-known semantics can be used to compute the admissible sets of arguments, using either extension- or labelling-based approaches. The mapping from an ASPIC⁺ argumentation theory to Dung's argumentation frameworks has been applied to real-world scenarios, such as for reasoning in the mathematics [167] and intelligence [215] domains, to name a couple.

2.3.2.2 The Argument Interchange Format

The Argument Interchange Format (AIF) is a community-led attempt to gather a variety of types of works within the computational argumentation literature by providing a shared ontology to facilitate future research and development of argumentation-based

tools and techniques [52]. The ontology acts as an abstract medium that allows researchers to employ any logical language they so choose in order to create argument systems, whilst also providing them with the added benefit of going between languages and formalisms. For instance, if a user were to conduct argument analysis on a debate using an annotation scheme, such as the Inference Anchoring Theory [37], and save their data in the AIF, then they would also be able to semantically evaluate the acceptability of arguments in their analysis by mapping it from the AIF to an ASPIC⁺ argumentation theory and then to a Dung argumentation framework, which can be evaluated using all the well-known semantics. As such, the AIF provides a solid foundation on which real-world applications for argumentation can be based. In Section 2.3.3.1, I briefly introduce a number of proposals within the literature that employed the AIF to store structured argument data that was transformed from unstructured natural language corpora through argument analysis and annotation.

The specification for the AIF ontology is presented in Figure 2.9. The AIF ontology has two parts, namely the *Upper Ontology* and the *Forms Ontology* [181, 180]. The *Upper Ontology* is comprised of *information nodes* (*I-nodes*) and *scheme nodes* (*S-nodes*), allowing users to build the nodes and edges found within argument systems. Depending on the context, *information nodes* store the data points within an argument analysis, such as locutions and propositions, whilst *scheme nodes* capture general patterns of reasoning, such as the inference between a set of premises supporting a conclusion or a conflict between *I-nodes*. As such, scheme nodes are the instantiation of: rule-application nodes (*RA-nodes*), indicating an inference from at least one *I-node* to another; conflict-application nodes (*CA-nodes*), indicating a conflict between two *I-nodes*; or preference-application nodes (*PA-nodes*), which annotate preferences between *I-nodes*.

The *Forms Ontology* employs the nodes and edges in the Upper Ontology and allows users to refine the simple patterns of reasoning captured, instantiating different theoretical argumentative forms, such as the argumentation schemes discussed in Section 2.2.1, thus allowing users to attain a better understanding of argumentation by modelling the intricacies found within it. The refinement of stereotypical patterns of reasoning into argumentation schemes is important when conducting argumentation mining, which I discuss more in Section 2.3.3.

The AIF ontology's main representational language is a directed graph. Graphs provide a structured and systematic way of describing argumentation without the constraints of a logic [52], while also aligning with many of the accounts of argumentation proposed within the literature. An AIF argument graph \mathcal{G}_{AIF} , not to be confused with the AIF ontology specification presented in Figure 2.9, is defined in Definition 2.17.

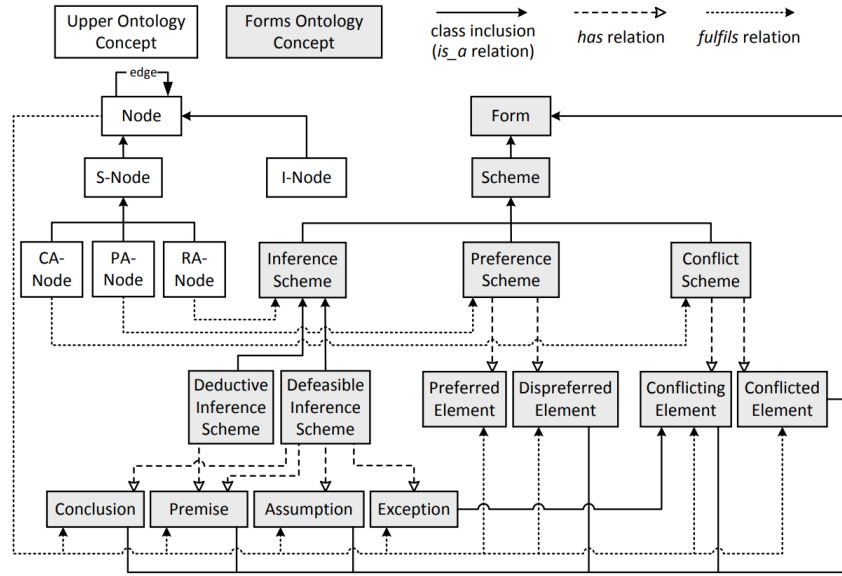


Figure 2.9: The AIF specification (taken from [11]).

Definition 2.17 Let $\mathcal{G}_{AIF} = (V, E)$ be an AIF argument directed graph which is a pair (V, E) where

1. $V = I \cup RA \cup CA$ is the set of vertices in \mathcal{G}_{AIF} , where I are the I -nodes, RA are the RA -nodes, and CA are the CA -nodes;
2. $E \subseteq V \times V \setminus I \times I$ is the set of edges in \mathcal{G}_{AIF} ;
3. if and only if $v \in V \setminus I$, then v has at least one direct predecessor and successor;
4. if and only if $v \in RA$, then v has at least one predecessor and successor in the form of a *premise* and *conclusion*, respectively;
5. if and only if $v \in PA$, then v has exactly one predecessor v_i and one direct successor v_j that instantiates the form *preferred* and *dispreferred element*, respectively, where $v_i \neq v_j$; and
6. if and only if $v \in CA$, then v has exactly one predecessor and successor, respectively termed *conflicting* and *conflicted elements*.

2.3.2.3 Translating from the AIF to ASPIC⁺

After introducing a restricted case of the ASPIC⁺ framework (Section 2.3.2.1), which can be employed in the instantiation of a Dung argumentation framework (Definition 2.16), presenting the AIF (Section 2.3.2.2), and defining how to instantiate AIF argument graphs (Definition 2.17), I now present a method capable of mapping argument data stored in the AIF to an argumentation theory in the ASPIC⁺ framework, which enables the semantic evaluation of arguments from real-world analyses [23].

Definition 2.18 For an AIF directed argument graph $\mathcal{G}_{AIF} = (V, E)$ with a set of I -nodes, forms \mathcal{F} and a set of relations that link the nodes in \mathcal{G} to nodes in \mathcal{F} , an ASPIC⁺ argumentation theory AT based on \mathcal{G}_{AIF} is defined as follows:

1. $\mathcal{L} = I \cup RA$;
2. $\mathcal{K}_p = \{v \in I \mid v \text{ is an initial node}\}$ where $v \in \mathcal{K}_p$;
3. \mathcal{R} is the smallest set of inference rules $r_k : v_1, \dots, v_n \Rightarrow v$ where there exists a node $v_i \in RA$ such that:
 - (a) v_i fulfils a *defeasible* scheme in the Forms Ontology \mathcal{F} ; and
 - (b) v_i 's direct predecessors are *premises* of the form v_1, \dots, v_n and v_i 's direct successor is a conclusion of the form v .
4. $v_h \in \bar{v}_j$ if and only if $\exists v_i \in CA$ such that v_i has a direct predecessor v_h and direct successor v_j .

The translation from the AIF to an ASPIC⁺ argumentation theory to a Dung argumentation framework enables automated reasoning about real-world arguments and the conflicts between them. The importance of this mapping will become apparent in the work presented in the introduction to Part II: it was employed to convert the outputs from an argument analysis of a deception operation during World War Two from the AIF to a Dung-style argument system that was used to evaluate the techniques proposed in Chapters 5 and 6.

2.3.3 Argument Mining

In this subsection, I examine the topic of argument(ation) mining, introducing manual and automated approaches in Sections 2.3.3.1 and 2.3.3.2, respectively, and highlight some of the problems associated with both methods in Section 2.3.3.3. Generally speaking, argument mining is the act of extracting propositions from unstructured

natural language corpora and identifying inference links between them, instantiating arguments that capture the reasoning involved. Both manual and automated argument mining employ a pipeline to achieve the aforementioned goal, however they differ in the tools they use to do so which, therefore, impacts things such as the speed and accuracy of argument analysis. While I examine the manual approach to argument mining in more detail within this subsection because it features in the work presented throughout Part II, I also include some discussion pertaining to automated methods as they are an obvious avenue for future work from this thesis.

2.3.3.1 Manual Argument Mining

Understanding manual argument mining provides a good starting point from which to grasp automated approaches. There are a wide variety of manual argument annotation software available on the internet, such as *Araucaria* [183], *Rationale* [85], *Carneades* [88] and the *Online Visualisation of Argument (OVA+)* [108]. Most of these applications require the annotator to, first, manually extract argumentative propositions and then identify the inference structure between them, forming arguments which contain a simple structure of inference between a set of premises leading to a conclusion. After the identification of the simple structure and depending on the requirement of how detailed an analysis should be, an annotator may choose to refine the simple structure, instantiating argumentation schemes, such as Walton's set of schemes [235] discussed in Section 2.2.1, which better capture the inference within arguments. Moreover, the instantiation of argumentation schemes, with their accompanying set of critical questions, enables analysts to identify attacks between the propositions within the arguments in their analysis. The general pipeline for manual argument mining is presented in Figure 2.10.

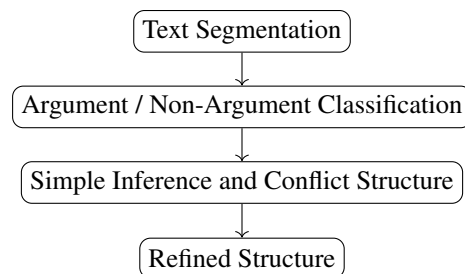


Figure 2.10: Distinct tasks within the manual argument mining process (adapted from [126]).

Text Segmentation. Partitioning of unstructured natural language corpora is the first step within manual argument analysis. The literature on text segmentation is divided into two categories, namely the identification of *elementary discourse units* or *argumentative discourse units*. *Elementary discourse units* (EDUs) are defined as a set of non-overlapping text segments extracted from natural language. There a number of different definitions for an EDU such that they can be either be individual sentences [174], clauses [89, 86], or prosodic units [102], to name a few. The main thing to note here is that the identification and extraction of EDUs should remain consistent throughout analysis so, for example, if an annotator chooses to split text using sentences, then sentence extraction should be the method of segmentation throughout their study.

Moving onto *argumentative discourse units* (ADUs), first introduced by Peldszus and Stede in [168] where they define them as the minimal atoms of arguments, or more commonly known as propositions. In some cases, ADUs might be much longer than EDUs as this approach to segmentation splits text up into parts which contribute to argumentation. Segmentation into ADUs is more challenging when compared to its EDU counterpart. One analyst might believe that a certain excerpt of a text contains argumentative power while another might not. In addition to this, the complexities of language makes dissecting corpora into its set of ADUs difficult. In [193], Saint-Dizier discusses the topic of *dislocation* in text as it pertains to ADU extraction. A *dislocation* in a sentence is a syntactic change where one ADU is embedded within another such that there is propositional content to either side (i.e., to the left and right) of another proposition's boundaries. A further challenge when segmenting using ADUs comes from the use of *enthymemes* within natural language. An *enthymeme* is an implied premise of an argument [103]. One of the main reasons for enthymemes within corpora is because we, as humans, want to be as efficient as possible when communicating. So, human agents often leave things unsaid because they are seemingly obvious. However, this presents problems for the argument analyst whose job it is to make reasoning explicit. The two aforementioned challenges make text segmentation an extremely challenging task for both human and computational agents.

Argument / Non-Argument Classification. In short, this part of the manual process involves deciding which of the excerpts form the propositions of arguments within a given text. Often, in the case of manual argument mining, the decision as to whether an EDU or ADU is argumentative or not occurs during the segmentation phase, but automated approaches must conduct this classification task separately [126].

Simple Inference and Conflict Structure. Once an argument analyst has segmented the text and classified which segments are argumentative, they begin the task of studying the set of segments and identifying the inference and attacks between them, instantiating a simple structure. Analysts should pay close attention to the pattern of inference employed because the structure of the inference might not be as simple as a single premise supporting a conclusion.

In [90], Groarke *et al.* note the variety of ways within which inference can present itself in the simple structure, as displayed in Figure 2.11. *Convergent arguments*, as shown in Figure 2.11a, contain more than one independent premise which supports a conclusion, such that the removal of one premise does not weaken the inference of another premise supporting the same conclusion. *Divergent arguments* contain the same premise supporting multiple conclusions, as presented in Figure 2.11b, however these arguments are not so common in real-world argumentation. *Linked arguments* contain multiple premises that are connected and support a single conclusion, as per Figure 2.11c, such that each premise requires the others for the inference to the conclusion to hold. If one of the premises was to be attacked or omitted from a person’s argumentation, then the argument would be significantly weakened. *Sequential arguments*, as displayed in Figure 2.11d, contain a series of premises leading to a conclusion where, for example, the first premise supports the second, the second premise supports the third, and so on, until the final premise leads to a conclusion. It is important to note that sequential arguments could be seen as a series of arguments that sequentially support each other such that the conclusion of one argument is the premise of the next. In reality, argumentation is not as simple as the types of arguments discussed above because we assert a *constellation of propositions* to make our position *acceptable to a rational judge who judges reasonably*, as per Eemeren *et al.*’s definition of argumentation [76] that was quoted in Section 2.2. Real-world argumentation contains combinations of the aforementioned simple structures which makes manual argument analysis a challenging and time-consuming task, even for expert argument annotators.

A notion of attack should be considered during the instantiation of the simple structure to better understand which propositions are in conflict with others. Although I have touched upon the different types of attack when talking about Toulmin’s model of argument in Section 2.2, Dung’s abstract argumentation frameworks in Section 2.3.1, and the ASPIC⁺ framework in Section 2.3.2, it is important to note that the simple structure can provide a graphical representation of conflict. The applications mentioned earlier in this section – i.e., *Araucaria*, *Rationale*, *Carneades* and *OVA+* – make it easier for analysts to diagrammatically describe rebutting, undercutting, and undermining attacks at the propositional level.

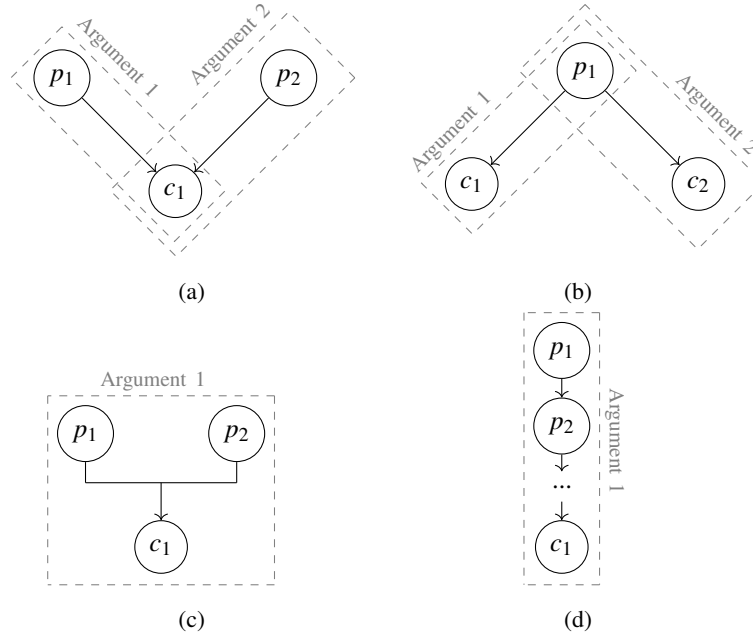


Figure 2.11: The simple structure of inference for (a) convergent, (b) divergent, (c) linked and (d) sequential arguments, where p_i and c_j refers to the i -th and j -th premise and conclusion, respectively (adapted from [126]).

Refined Structure. After the instantiation of a simple structure, the analyst is able to identify stereotypical patterns of inference, thus, adapting the simple structure into a refined one through use of argument annotation schemata, such as Walton's set of argumentation schemes, as discussed in Section 2.2.1. The critical questions within these schemes provide analysts with the most commonly used set of attacks between propositions. The critical questions enable the argument analyst to identify attacks which they might not have considered otherwise, during the instantiation of the simple structure phase. Furthermore, argumentation schemes aim to make reasoning explicit and, therefore, enthymemes, where a proposition is either not said or left implied, must be identified and incorporated into refined arguments within this stage of analysis. The combination of modelling arguments from the simple structure using argumentation schemes, plus their critical questions to identify conflicts between propositions, and explicitly incorporating enthymemes, makes instantiating the refined structure the most

cognitively challenging and time-consuming task within manual argument analysis. Although the refinement of the inference is hard, it enables analysts to fully understand the argumentation within their problem setting. Again, the argument annotation software – i.e., *Araucaria*, *Rationale*, *Carneades* and *OVA+* – introduced earlier in this section provides analysts with the necessary tools to instantiate the refined structure.

Software for Manual Argumentation Mining. As mentioned earlier in this subsection, there have been many proposals within the literature to provide software which aids in the visualisation and analysis of arguments, such as *Araucaria*, *Rationale*, *Carneades*, and *OVA+*. Looking to Janier *et al.*'s *OVA+* as an example, the software is a web application that enables users to annotate unstructured natural language corpora, converting it into diagrammatic representations of arguments which allows for further analysis. The software has been used in a number of research tasks – such as in the analysis of BBC Radio Four's Moral Maze [127], the 2016 United States presidential debates [234], and more recently, BBC One's Question Time [99], the simulation of mathematical reasoning [167], and intelligence analysis [253] – showing how the inference between propositions can be formally represented as either simple inference rules or through the instantiation of argumentation schemes. The application also enables the representation of attack between propositions.

While *OVA+* is a useful tool for instantiation and visualisation of inference structures, it does not provide users with automated reasoning about the acceptability of arguments and conflicts between them. *The Online Argument Structures Tool* [203], commonly known as *TOAST*, is one such implementation that utilises the AIF, ASPIC⁺ framework, and the mapping from AIF to an ASPIC⁺ argumentation theory to a Dung argumentation framework, as discussed in Section 2.3.2, in order to provide users with the ability to evaluate the acceptability of real-world arguments.

2.3.3.2 Automated Argument Mining

The aim of automated argument mining is to employ techniques from the natural language processing and computational linguistic literature to extract structured arguments from unstructured natural language corpora. The realisation of reliable and accurate automated approaches should enable the instantiation of abstract argumentation frameworks, which can then be evaluated using many of the well-known semantics; thus, enabling automated reasoning of real-world problems at speed and scale. Generally, the tasks involved in the automation of the argument mining process can be split into two stages, namely *argumentative component detection* and *relation prediction*.

Looking first to *argumentative component detection*. This task is usually split into two distinct sub-tasks which are the classification of propositions with argumentative power and detection of their textual boundaries. As with the manual approach, much of the literature pertaining to automated methods for argument mining obtains ADUs through text segmentation. While there are some studies which assume that the textual boundaries of propositions have been detected through other means – such as in [207, 73] – so that they can solely focus on classifying claims and premises, many of the other works within the literature do not make this assumption. The remainder of the literature employs a series of grammatical and syntactic features (i.e., punctuation, verb tenses, part-of-speech tags, n -grams, and discourse markers) to train machine and deep learning models to extract ADUs. The machine learning models which have been trained include: Support Vector Machines [149, 166, 207, 73, 72]; Logistic Regression [132, 185, 154]; Naïve Bayes classifiers [151, 149, 27, 73]; Maximum Entropy classifiers [151, 149]; Parsing algorithms, which use a context-free grammar [84, 169, 77]; and Decision Trees and Random Forests [80, 73]. In [94], Habernal and Gurevych formulated the argumentative component detection task as an entity recognition problem, where each word in a corpus is tagged as either the beginning, inside or outside of an ADU. Since then, approaches have adopted the aforementioned tagging formulation and proposed various deep learning models, such as use of recurrent neural networks [112, 5, 143] and transformer architectures [145, 5, 143], for ADU extraction.

Relation prediction is the second and most challenging stage within the automated argument mining process. This stage is focused on the *prediction*, rather than the *detection* or *classification*, of both support and conflict relations between propositions to form either arguments or attacks between them, respectively. The prediction of supports and attacks is all that one needs to instantiate an argumentation framework and can be thought of as somewhat similar to the instantiation of the simple structure when conducting manual analysis. A number of machine and deep learning models have been proposed to predict relations, such as, again, Support Vector Machines [207, 155, 144], Parsing algorithms [84, 169, 77], Logistic Regression [154], Textual Entailment [40, 113], Recurrent Neural Networks [155, 77, 143], and Transformer architectures [192, 143].

Fully automated pipelines. While the difficulty significantly increases with each stage of the automated argument mining pipeline, this has not stopped some from proposing fully automated end-to-end approaches, taking unstructured natural language corpora and extracting the structure of inference and conflicts between propositions [77, 130, 83, 45]. To achieve the aforementioned pipelines, all these approaches

assume that a relation (i.e., an inference or conflict) occurs between a single source and target only. Although this is a good first step, the proposals were not capable of modelling the reasoning about arguments that contain more than one premise supporting a conclusion. Thus, as a result of this constraint, the proposed models do not predict and instantiate Walton’s argumentation schemes, which is arguably the most challenging task but also offers the most potential benefit in terms of understanding argumentation at scale.

Walton’s argumentation schemes in automated approaches. The automated relation prediction stage can also include the prediction of Walton’s argumentation schemes, which is similar to the instantiation of the refined structure in manual analysis and allows for more detailed representation of inference between propositions. In [78], Feng and Hirst propose one of the first approaches that predicts Walton’s schemes, where they trained and tested on Reed and Rowe’s *Araucaria* database [183] and achieved accuracies of 63–91% and 80–94% in one-against-others classification and pairwise classification, respectively. Since then, there have not been many attempts to automatically predict Walton’s schemes which can be attributed to the difficulty of the task. However, there have been a few proposals within the literature that have attempted to mine, for example, bespoke schemes to capture reasoning involved in consumer product reviews [247] (omitting evaluation results using *recall* and *precision* due to lack of standardised testing datasets for argument mining) and a number of Walton’s original schemes, using: a combination of Support Vector Machines, Naïve Bayes classifiers, and Decision Trees in [125] (with F-scores between 0.78 and 0.91); Tree Kernels in [134] (with F-scores between 0.65 and 0.92); Bayesian Networks in [123] (where the overall average accuracies for the test set were 27.5%); and a combination of Transformer architectures, Textual Entailment, and probabilistic soft logic in [113] (with F-scores between 0.42 and 0.79 across two distinct training and test datasets).

2.3.3.3 Challenges and Limitations

Both manual and automated approaches to argument mining have advantages and disadvantages. The advantage of manual analysis is that it is thorough, it enables an annotator to grasp the reasoning within data, and once the simple structure has been instantiated, analysts can refine it, instantiating Walton’s argumentation schemes. The disadvantage of a manual approach is that it is cognitively demanding, requires a lot of training before an annotator is proficient, and manual analyses can take a considerable amount of time. The main benefit of automated approaches is that the process of

mining arguments is much quicker. However, this speed comes at a cost. The accuracy of automated methods for argument mining can be quite low because of a cascade of errors resulting from the combination of errors from each model, for each stage of the pipeline. Thus, the technology, in its current state, employed to conduct automated argument mining does not outperform manual annotation. Moreover, many of the automated pipelines are trained on a specific domain and perform poorly when those same models are used in different scenarios. On the contrary and once proficient, a manual annotator can go between domains at ease.

Argument data. The lack of consistently annotated argument data poses a real problem for the development of automated approaches to argument mining. There are five reasons for this. The first is due to the amount of time it takes analysts to conduct manual argument mining, which means there are not many openly-available datasets for the development of automated techniques. Creating structured argument data from unstructured text is a complex and time-consuming task that requires teams of annotators, where there might not be a high inter-annotator agreement due to the subjectivity and complexity of analysis, even between the most skilled annotators. Second, different researchers have employed a variety of annotation schemes and methodologies in previous works which means that if, for example, we were to provide two research groups with the same set of corpora, then the resulting sets of instantiated arguments and attacks may well be very different. Third, in years gone by, different research groups have stored their argument data in different formats. While this is not necessarily a problem, it makes implementing and comparing automated approaches more difficult than it needs to be. Argument data saved in the AIF (Section 2.3.2.2) and stored on shared databases, such as AIFdb [128], aims to solve this particular problem as it provides a common platform for researchers to develop models using a consistent datatype. Fourth, there is simply not enough annotated argument data to train, for example, an encoder-decoder recurrent neural network or a transformer to produce a context-independent pipeline of models. Finally, the nuance of argumentation makes it such that many of the machine and deep learning models are trained on data from specific domains – such as within legal, scientific, and political settings, to name a few – where they perform poorly when they are employed within settings that they were not constructed for.

Enthymemes and Walton’s argumentation schemes. Enthymemes pose another challenge when trying to automate the argument mining process, specifically when trying to predict Walton’s argumentation schemes. Often in natural language, proposi-

tions are left implied. While this is not an issue for the manual annotator, this makes predicting argumentation schemes a real challenge for automated techniques. Currently in the literature, the automated approaches make the assumption that a single premise supports a single conclusion. However, many argumentation schemes contain more than one premise. Given the fact that some of these premises will be left unsaid within corpora, training machine and deep learning models to reliably predict Walton's schemes, whilst also trying to identify enthymemes, is a difficult research question that is currently unanswered.

For a comprehensive overview of the process, challenges, and limitations involved in automated argument mining, refer to [135] and [126].

2.4 Tools for Intelligence Analysis

In this section, I examine the current literature on proposed techniques and tools for the *analysis* phase of the *Intelligence Cycle*, with a particular focus on evaluative techniques, such as the ACH. I begin by discussing general approaches, then examine a number of Bayesian methods, and finally consider tools that employ argumentation for this task.

2.4.1 General Approaches

Software tools have been developed to support the workflows of intelligence analysts. A lot of the applications pertinent to this thesis have extended the methodology of the ACH (Section 2.1.5) so that it functions on a computer, making it accessible to analysts and more efficient than analysis conducted by hand [26, 164]. In [26], Billman *et al.* show how software applications can provide the intelligence community with the added benefit of monitoring the judgements, performance, and bias of analysts. Tools which provide an audit trail of the reasoning within an analysis are important for instances where a mistake has occurred. Such tools would have been useful in the assessment of the failings of the intelligence services before the terrorist attacks on 11th September 2001 and before the invasion of Iraq in 2003 on a faulty premise.

While some tools have mapped the ACH process to an application, other works have taken the methodology of the ACH and incorporated more rigorous reasoning techniques. For example, in [175], Pope and Jøsang propose the use of subjective logic within the ACH, which is the first example within the literature to combine a logic with the procedure. The proposed approach enhances the ACH by allowing analysts to apply abductive and deductive reasoning within their analysis, while also minimis-

ing reasoning errors. Another proposal [60] combines granular computing with the ACH, amalgamating similar items of information into chunks, referred to as granules, which are then grouped into rough sets, where inference is conducted using three-way decisions; thus, aiding analysts when reasoning with items of similar information instantiated in the diagnosticity matrix of the ACH.

With the advent and popularity of large language models, such as OpenAI's ChatGPT⁵, it is obvious that some researchers might think that those models are capable of performing some of the analytical tasks that analysts do on a daily basis. For instance, Snidaro states that

“Large language models have the potential to revolutionise intelligence operations by providing intelligence officers with powerful tools for information processing and analysis.” [204, p. 449]

More pertinent to this thesis, whether these models alone are capable of, first, generating the set of plausible, MECE hypotheses (Chapter 1) and second, reasoning with a set of incomplete, conflicting, and uncertain information, is a research question that is currently unanswered. Large language models are good at generating text on previously seen examples, which they have been trained on. However, intelligence analysis often includes information for which there is no training data. Therefore, such models might not perform so well when tasked to generate and evaluate hypotheses on these types of novel situations. I discuss this point in more detail in Chapter 4, where I test whether large language models are capable of generating the set of MECE hypotheses for a recent, real-world intelligence scenario.

2.4.2 Bayesian Methods

Bayes' theorem [18] is a mathematical formula, first introduced by the British mathematician and Reverend Thomas Bayes in 1764, that provides a method of refining an initial probability judgement given some information. The formula enables the computation of a *posterior* probability which describes the probability of an event using an initial belief, referred to as a *prior*, and the *likelihood* of the observation of a given set of data in light of that event being true.

Bayes' law has been shown to be capable of revising probability estimates pertaining to hypotheses within intelligence settings, specifically in problems of strategic warning [251, 74]. In [252], Zlotnick examples how Bayes' theorem could have complemented analysis on events leading to the 1941 Pearl Harbour attacks and the invasion

⁵<https://chatgpt.com/auth/login> (Accessed on: 12.07.2024)

of South Korea in 1950. In [79], Fisk provides another pertinent and early comparison of the use of Bayes' formula with other conventional methods for intelligence analysis at the time, using a border dispute between China and the Union of Soviet Socialist Republics (USSR), as an example. In both works, Zlotnick and Fisk show how the refined probabilities, computed using a Bayesian approach, are better estimates of the likelihood of hypotheses. A rendition of Bayes' law, as it pertains to the aforementioned intelligence examples, is presented below

$$P(h_1|I) = \frac{P(h_1)P(I|h_1)}{P(I)} \quad (2.6)$$

where $P(h_1|I)$ is the *posterior* probability of a hypothesis h_1 given the available information I , $P(h_1)$ is an intelligence analyst's *prior* belief about the probability of a hypothesis, $P(I|h_1)$ is the *likelihood* which is the probability of the information given the hypothesis is true, and $P(I)$ is the probability of the information.

Depending on the number of hypotheses in the assessment, the probability of the information can be computed in different ways. The probability of the information when there are two hypotheses, h_1 and h_2 , within an analysis is

$$P(I) = P(I|h_1)P(h_1) + P(I|h_2)P(h_2). \quad (2.7)$$

When there is more than two hypotheses within an analysis, the probability of the information for N hypotheses can be found through Equation 2.8.

$$P(I) = \sum_{i=1}^N P(I|h_i)P(h_i) \quad (2.8)$$

According to Zlotnick [251], Bayesian methods should not become an alternative to intelligence analysis, they should act as a reliability check for use after analysis, after making an initial set of probabilistic judgements, helping analysts understand whether their set of conclusions are plausible. Although Zlotnick argued this in 1967, his argument stood the test of time, with a recent study showing that the post-analytic use of Bayes' theorem yielded large improvements in analysts' accuracy, when compared with the vanilla ACH [117].

A number of other Bayesian techniques have been shown to be capable of hypothesis evaluation within the intelligence domain. One such technique is the Bayesian network [111], which is an extension of Bayes' theorem. These networks employ directed acyclic graphs to model the probabilities of hypotheses and evidence (as the nodes in the graph) and their likelihoods (as the edges). There have been some proposals within the literature that show how the use of Bayesian networks could aid an intelligence

analysis, specifically analysis conducted using the ACH [231, 118, 195, 196]. Further statistical models have been proposed, such as a multinomial-Dirichlet model for multiple hypothesis evaluation [68]. All these networks, however, require the user to input the network's structure and probabilities, which could make this task a tedious one if there is a lot of data. As the intelligence community moves towards reasoning with more information, through the incorporation of additional data collected through OSINT (Section 2.1.2), instantiating a graph and assigning probabilities manually will not be practical.

In [39], Burns advocates against employing more sophisticated statistical techniques that require probabilistic experts because intelligence analysts are often not trained in formal statistical and probability theory, and they usually reason with only a few hypotheses and items of information. So, many of the proposed techniques, such as the aforementioned Bayesian networks and multinomial-Dirichlet model, are foreign to the analyst such that their use adds unneeded complexity to an already complex task. Burns suggests a simpler approach in [38], where one elicits the set of MECE hypotheses, the information (which is referred to as evidence by Burns), the likelihood of observing the information given the hypotheses, and the prior probabilities of hypotheses, from intelligence analysts, which ultimately enables the computation of posteriors, through Bayes' theorem. The approach taken by Burns reaffirms the soundness of the earlier proposals undertaken by Fisk and Zlotnick and, again, suggests that Bayesian approaches should only be employed after analysis has concluded in order to enable an analyst to assess the soundness of their initial probabilistic judgements.

2.4.3 Argumentation-based Proposals

The use of argumentation for intelligence analysis is a relatively niche and new research field, with only a handful of ideas, tools, and techniques for hypothesis evaluation proposed within the literature. However, the specific task of hypothesis generation has not been studied from an argumentation perspective in any published research.

Computational argumentation is a seemingly good fit for tools within the *analysis* stage of the *Intelligence Cycle* (Section 2.1.1) for several reasons. First, the use of argumentation schemes, as discussed in Section 2.2.1, should enable intelligence analysts to make the reasoning underpinning their analysis explicit, which is beneficial when the reasoning within an analysis is often left implied (Section 2.1.5). Second, the critical questions in schemes should enable analysts to repeatably identify conflicts between propositions, where they might recognise attacks between data which they otherwise might not have. Third, reasoning about the acceptability of arguments containing hy-

potheses, information, evidence, and assumptions can be done in a logical manner because of the well-known argumentation semantics (Section 2.3.1). Finally, computational argumentation enables defeasible and non-monotonic reasoning such that as new information comes to light, additional propositions can be incorporated into analyses, thus instantiating new arguments and attacks, which can be reasoned with to come to an updated set of conclusions.

In [210], Tecuci *et al.* propose a Wigmorean probabilistic inference network⁶ that draws upon Toulmin's representation of argument, as discussed in Section 2.2, to reason about hypotheses and information within the ACH. While the authors were one of the first to identify the relationship between the ACH and argumentation, the use of Toulmin's account, with the constraint that conflict can only occur as a rebuttal, means that the proposed model does not allow for the variety of attacks which might be present within an intelligence analysis.

Walton's argumentation schemes, as discussed in Section 2.2.1, have been employed for hypothesis evaluation. In [152], Murukannaiah *et al.* combine the ACH with argumentation schemata and conduct a small study, which included 5 undergraduate and 15 graduate students, to compare their proposal with the vanilla ACH, as defined in Section 2.1.5. They show that the incorporation of argumentation schemes into the ACH increased: the completeness of analysis due to increased coverage of belief search; the length of belief chaining; the explicitness of assumptions; and the repeatability of the conclusions drawn from analysis, for all participants within the study. They also argue that the inclusion of argumentation schemes in the ACH increased the ease of use of the technique. However, the subjects within the test had no former experience of analytical techniques within an intelligence domain, so the argument for ease of use as it pertains to trained intelligence analysts needs to be explored. Moreover, whether all the benefits mentioned translate to skilled analysts still needs to be tested, but this initial study shows the positive impact that argumentation, specifically the use of argumentation schemes, could have on techniques for hypothesis evaluation.

The link between the ACH and argumentation schemes has also been explored through two very different examples on the Defence Science and Technology Laboratory's *eleatics* website [63]. The first example utilised data pertaining to Operation

⁶Wigmore [244] charts provide a method of diagrammatically representing arguments to reason with evidence in the legal domain. The diagrams model support between statements (or evidence) within a legal case. Wigmore proposed more than thirty different types of support that all have their individual diagrammatic representations. Each support is assigned a *force*, which denotes how strong a support is and ranges from having no effect to either a very strong positive or negative one.

Fortitude-South⁷, which was an Allied deception effort during World War Two to make the Axis command believe that the Allies would invade through the shortest and most obvious path, from Dover to Pas de Calais, when in fact the Allies went via a longer route, from Portsmouth to Normandy. The second example studies the 1989 Jesuits Massacre⁸ in El Salvador, where the truth about who committed the atrocity only came to light years later. Here, in both examples, the ACH and argumentation schemes were combined to show, firstly, how argumentation schemes could be used with the ACH and second, to demonstrate how both techniques can model the changes in the evaluation of hypotheses over time as new information comes to light. Furthermore, the Operation Fortitude-South scenario is of particular importance to this thesis as the example provided by the Defence Science and Technology Laboratory was manually annotated for arguments and conflicts, as described in the introduction to Part II, and subsequently employed to test the methods proposed in Chapters 5 and 6.

There exists a mapping from data stored within the AIF to the populated diagnosticity matrix found with the ACH. In [253], Zografistou *et al.* conduct an argument analysis on an intelligence scenario, selecting the arguments which contain hypotheses, and then translate the analysis stored within the AIF to a diagnosticity matrix, with a view to making argument analysis look more familiar to intelligence analysts. The relationship between argumentation frameworks and the diagnosticity matrix has not been formally defined within the literature. However, as I show in more detail in the introduction to Part II and Appendix B, data within a diagnosticity matrix can be mapped to the AIF, and then to abstract argumentation frameworks. Therefore, the mapping from the AIF to the ACH, and vice versa, is potentially a form of morphism, which should be investigated further to understand whether it is indeed a morphism, its type, and the necessary constraints within which this mapping can occur.

In [189], Robinson employed a utility function to consider the value of adding arguments and attacks into probabilistic argumentation frameworks [133] to aid in the generation of RFIs during the *collection* stage of the *Intelligence Cycle* (Section 2.1.1). This approach was then extended in [190] to a simple, real-world example to show that approach can provide benefit to analysts.

The last piece of notable work discussed in this section was proposed by Toniolo *et al.* in [215], where they produced probably the most comprehensive implementation and evaluation of an argumentation-based tool for collaborative intelligence analysis. The software presented enables a number of analysts to: communicate amongst

⁷<https://dstl.github.io/electrics/argumentation/fortitude/> (Accessed on: 10.02.2023)

⁸<https://dstl.github.io/electrics/argumentation/muc3/jesuits> (Accessed on: 10.02.2023)

themselves; forage for information; instantiate arguments using Walton's argumentation schemes (Section 2.2.1); identify conflicts between items of information through the instantiation of said schemes; evaluate the resulting argumentation framework, using the well-known argumentation semantics (Section 2.3.1) and conduct defeasible reasoning after the incorporation of new arguments and attacks as more information becomes available; and finally, analysts can ask questions which are answered, using a technique called crowdsourcing. The authors used intelligence analysts, the subject matter experts for this domain, to evaluate the performance of their tool, who deemed the tool to be generally useful in improving their daily activities, as well as having the potential to benefit complex analysis and training novice analysts.

2.5 Summary and Conclusions

In this chapter, I have presented a survey of the literature related to the work I present in the upcoming chapters. The main focus of this research is the development of techniques and tools which provide benefit to intelligence analysts, specifically ASAs. In Section 2.1, I discussed the *Intelligence Cycle*, the types of intelligence and analysts, how analysts communicate probability and uncertainty, and a number of analytical techniques employed during hypothesis generation and evaluation. In Section 2.2, I introduced argumentation theory as a method for the representation of arguments, focusing mainly on Walton's argument from cause to effect [236]. Section 2.3 presents computational approaches to modelling defeasible reasoning about conflicting arguments, as well as some discussion about both manual and automated argument mining. Finally, in Section 2.4, a number of general, Bayesian, and argumentation-based proposals for decision- and reasoning-support tools for intelligence analysts were discussed.

The literature presented in this chapter provides a number of important points that will be extended by the work presented in the rest of this thesis, which I summarise below.

Intelligence Analysis. The aim of this thesis is to provide decision-support tools and algorithms for intelligence analysts. As such, it is important to understand the *Intelligence Cycle*, the types of intelligence and analysts, the definition of a hypothesis (Chapter 1), how uncertainty is communicated within intelligence reports, and the analytical techniques employed in the generation and evaluation of hypotheses. It is crucial for intelligence analysts to generate the set of MECE hypotheses to ensure that

they are evaluating all the potential and plausible hypotheses within a given operational environment. In Chapter 4, I evaluate the feasibility of using large language models for MECE hypotheses generation. Finally, intelligence analysis is inherently argumentative and defeasible, in that analysts are tasked with reasoning with conflicting information, where they are required to revise their judgements as new information becomes available. One such tool that aids the intelligence analyst in their evaluation of such scenarios is the ACH. An example intelligence analysis using the ACH has been mapped to an argumentation framework in the introduction to Part II and Appendix B in order to enable the assessment of the decision-support techniques proposed in Chapters 5 and 6.

Argumentation Theory. Argumentation theory plays a pivotal role in the representation of arguments and attacks between propositions within those arguments. Walton's argumentation schemes [235] provide a method that is capable of modelling the stereotypical patterns of reasoning involved in intelligence analyses. In addition to this, the set of critical questions identified by Walton should enable analysts to repeatably identify attacks between hypotheses, information, evidence, and assumptions. These schemes will be examined in more detail in the introduction to Part II when I show how the diagnosticity matrix within the ACH can be mapped to an argumentation framework containing instantiated argumentation schemes. One of Walton's schemata, the argument from cause to effect [236], can represent the reasoning underpinning arguments which contain a hypothesis. In Chapter 3, the argument from cause to effect and its set of critical questions is examined in more detail, and then extended to incorporate propositions that contain items of information, evidence, assumptions, and hypotheses, as well as qualitative probability and uncertainty ranges

Computational Argumentation. Techniques from the fecund field of computational models of argument provide a number of mechanisms which are useful when applying argumentation theory to real applications. There have been a number of fundamental contributions from this research area. First, Dung's seminal work on abstract argumentation frameworks and Prakken's ASPIC⁺ framework which, when combined, enables the evaluation of the acceptance status of structured arguments. In Chapter 5, the ASPIC⁺ framework's definition of arguments is extended to produce a definition of a hypothesis and a definition of information to aid in the refinement of prior probabilities of hypotheses. The proposed, structured argumentation-based definition of a hypothesis in Chapter 5 is in accordance with Pherson and Heuer Jr.'s definition of a hypothesis in Chapter 1. In Chapter 6, Dung's framework is combined with probability and infor-

mation theory in order to employ abstract argumentation for sensitivity analysis within the ACH, discussed in Section 2.1.5.2. Second, the AIF enables researchers to store argument data in a consistent data type, which can then be mapped to Dung's framework for semantic evaluation. Finally, the formalised approaches to manual and automated argument mining facilitate the transformation of unstructured natural language corpora into structured argument data. All three aforementioned contributions have enabled the assessment of the argumentation-based decision-support tools and techniques proposed in Chapters 5 and 6.

Tools for Intelligence Analysis. There have been proposals within the literature which employ general, Bayesian, and argumentation-based approaches in the development of tools to provide decision- and analytical-support for intelligence analysts. In Chapter 4, experiments were conducted to assess the feasibility of leveraging large language models for MECE hypothesis generation. Chapter 5 demonstrates how both argumentation and Bayes' theorem can be employed in a formal account to refine an analyst's initial prior beliefs using a semantically-evaluated set of arguments containing information. All three of the techniques proposed in Chapters 4, 5, and 6 should be utilised as support tools only, acting as a reliability check for analysts after conducting their own analyses, in line with Zlotnick's suggestion in [251] (Section 2.4.2).

Part I

Hypothesis Representation & Generation

In this part, I introduce two techniques that study hypothesis representation and generation. In Chapter 3, I consider hypothesis representation and examine Walton's argument from cause to effect [236], demonstrating how the scheme can be extended to increase its specificity so that it is fit for operational use within intelligence settings. Similar to the notion put forth by Toniolo *et al.* in [216] and discussed in Section 2.2.1.1, the scheme includes items of information, evidence, and assumptions – as premises – that support a hypothesis – the conclusion of the argument. All propositions have an associated qualitative description of their probability and uncertainty, in accordance with the Probability (or Uncertainty) Yardstick [221] (Section 2.1.4). The hypotheses incorporated within the proposed scheme are statements that follow Pheron and Heuer Jr.'s definition of a *good* hypothesis [173], as previously discussed in Chapter 1. Then, in Chapter 4, I present a feasibility study which assessed whether large language models can be employed for MECE hypothesis generation. Large language models were an obvious choice of technology due to their ability to conduct conditional text generation. However, as the results presented in Chapter 4 show, whether those models are able to generate *good* hypotheses for novel, intelligence scenarios is questionable. In conformity with Zlotnick's recommendation in [251] (Section 2.4.2), the techniques proposed in this part, and the next, are to be employed after an ASA has completed their own analysis and should provide the analyst with the assurance that their judgements are reasonable and logical, in light of the available data. Moreover, the argumentation scheme proposed in Chapter 3 should increase the explainability of analyses by fully capturing analysts' reasoning about data and assumptions that support hypotheses.

Chapter 3

A Scheme for the Representation of Hypotheses

In this chapter, I present an argumentation scheme to capture the reasoning about hypotheses supported by information, evidence, and assumptions. The argumentation scheme is an extension of Walton's argument from cause to effect [236], which was introduced in Chapter 2. Not only does the extended scheme enable analysts to model support for a hypothesis using multiple premises, it allows the analyst to describe the probabilities and uncertainties associated with each proposition instantiated within the scheme.

3.1 Extension of the Argument from Cause to Effect

In Chapter 1, I introduced Pherson and Heuer Jr.'s definition of a hypothesis within intelligence settings. A hypothesis within this context is a falsifiable statement that explains available data in an accurate fashion [173]. In Section 2.2.1.1, I also discussed a more recent, argumentation-based definition of a hypothesis proposed by Toniolo *et al.* in [216], which employs Walton's argument from cause to effect [236] and agrees with Pherson and Heuer Jr.'s definition, arguing that a hypothesis is comprised of a series of information about activities and events which describe how a situation has developed. Here, I return to the argument from cause to effect and discuss some of the issues associated with it, making it unfit for operational use within intelligence settings, in its current state. I then take these concerns and propose a series of extensions which render the scheme capable of representing hypotheses that are supported by multiple

items of information, evidence, and assumptions. The proposed scheme also enables the inclusion of uncertainty and probability for past, present, and future situations to fully represent an analyst's reasoning.

Recall Walton's argument from cause to effect [236] with its accompanying set of critical questions from Chapter 2:

Major Premise: Generally, if A occurs, then B will (might) occur.

Minor Premise: In this case, A occurs (might occur).

Conclusion: Therefore, in this case, B will (might) occur.

Walton identifies three critical questions for this scheme:

WCQ1: How strong is the causal generalisation?

WCQ2: Is the evidence cited (if there is any) strong enough to warrant the causal generalisation?

WCQ3: Are there any other causal factors that could interfere with the production of the effect in the given case?

Consider the following example from a fictitious scenario based upon the *Intelligence Cycle*, as discussed in Section 2.1.1. Relations are deteriorating between two bordering countries, Country X and Country Y. After some discussion and *planning*, stakeholders from Country X want to understand whether Country Y is preparing to invade, so they initiate a RFI which tasks an ASA with analysis of the situation. Unfortunately, the ASA does not possess any information on the evolving scenario, but they hold an assumption about the leader of Country Y, namely that they are a dictator with a thirst for power and resources. To understand more about the situation, the ASA tasks a SSA, through the initiation of another RFI, to employ IMINT, one of the five disciplines of intelligence collection [139], to collect data from the operational environment pertaining to Country Y's movements of personnel and military equipment close to Country X's border.

Suppose that during their collection using satellite imagery, the SSA captured a photograph which *likely* contained a large and unknown number of people close to the border of Country X and in response to the RFI raised by the ASA, the SSA sends a report asserting that observation, its likelihood, and the confidence in that estimate. While the SSA has employed their expertise to analyse the image, what is shown in the picture is not fully conclusive, which is why the SSA's observation was accompanied by the qualitative probability range associated with the descriptor *likely*. Now, as per the *Intelligence Cycle*, the ASA generates a set of MECE hypotheses using the received,

raw intelligence and evaluates the likelihood of those hypotheses using a technique, such as the ACH, in line with the *analysis* phase of the *Cycle*. The ASA may want to represent their reasoning about those hypotheses using an argumentation scheme. Using the raw intelligence provided by the SSA, the ASA instantiates Walton's argument from cause to effect [236], which might look something like the following:

Major Premise: Generally, if Country Y has *likely* deployed a large and unknown number of people close to their border with Country X, then it is *highly likely* that Country Y will invade Country X.

Minor Premise: The IMINT analyst asserts that "Country Y has *likely* deployed a large and unknown number of people to their border with Country X".

Conclusion: Therefore, in this case, it is *highly likely* that Country Y will invade Country X.

Here, the ASA utilises the causal generalisation in the major premise to conclude that an invasion of Country X by Country Y is highly likely because of the deployment of a large and unknown number of people close Country X's border.

Looking to the critical questions, one could argue that the causal generalisation is weak (WCQ1) because the information is not strong enough to warrant the *invasion* conclusion (WCQ2). Although there is a large and unknown number of people close to Country X's border, the formation may or may not be associated with Country Y's military. Thus, the argument, in its current state, possesses a weak inference from cause to effect. One could argue that a proposition was left implied by the ASA, namely their assumption that Country Y's leader is totalitarian and aggressive. For repeatable and explainable analysis, any assumptions or prior domain knowledge which features in the ASA's reasoning – in the form of enthymemes – should be included within an analysis. The last critical question (WCQ3) pertains to the presence of other causal generalisations which could be drawn from the same set of information. It should be easy for the reader to think of other hypotheses (or claims) which could sensibly be made using the same item of available information like, for example, Country Y is attempting to bolster their defences along a part of their border due to deteriorating relations with Country X. However, the production of such an effect requires more than one cause to produce a stronger causal generalisation and thus, a stronger argument that contains a concrete representation of the reasoning.

3.1.1 Multiple Causes

Now the above argumentation scheme is able to model the reasoning in scenarios where an ASA is required to instantiate an argument with a single cause leading to a single effect. However, in reality, analysis is never that simple.

Recall step two of the ACH in Section 2.1.5.2. The ASA must list the significant and relevant information, evidence, assumptions, and anything else that they believe is missing from their knowledge base. Here, an important point to note is that the intelligence community may refer to the rows of an ACH matrix as *evidence*, but row entries in Heuer Jr.'s original ACH can either be information, evidence, assumptions, or any missing information that the analyst believes should be included within an analysis. In the same spirit as Heuer Jr.'s original contribution, the row entries are assumed to contain either information, evidence, or assumptions, with the caveat that any missing information should be included as information.

In step three of the ACH, the analyst sequentially goes through each row entry in the ACH matrix and assesses whether the row entries support or conflict with each hypothesis by marking whether each item was consistent or inconsistent, respectively, and anything unrelated is denoted *not applicable*. In line with the current procedures for hypothesis evaluation within the intelligence community, there are usually multiple data points which support (or attack) a hypothesis. In this chapter, these supporting data are assumed to be synonymous with causes that should form part of the ASA's argument for a hypothesis, if and only if they are relevant to the conclusion being made. As such, there will almost certainly be arguments where more than one datum produces a linked support for a hypothesis. Due to the process detailed in the ACH, the causes supporting a hypothesis can be grouped into three different categories, namely *information*, *evidence*, and *assumptions*, which I explore in more detail below.

Information. The first case of multiple causes is one where an analyst receives more than one item of raw intelligence from a set of SSAs and wants to instantiate more than piece of information as a premise in an argument where a hypothesis is the conclusion. Turning back to the running example, after initiating a RFI, the ASA receives another item of raw intelligence from a different single-source counterpart, in addition to the satellite imagery from the IMINT analyst. This information might be such that when considered in isolation, there is a weak inference from it to a conclusion due to a weak causal generalisation within the argument from cause to effect. As previously mentioned, the raw intelligence from the IMINT analyst might not provide enough information to conclude that an invasion is imminent as that item alone bares little to no

relevance to the proposed effect. There are a number of opposing hypotheses which the ASA could generate using that single piece of available information, such as the formation of people at the border was a protest about the deteriorating relations between both countries. However, imagine that a covert HUMINT source stationed within Country Y and located geographically close to the border of Country X sent a report to the ASA asserting that the crowd of people were wearing military uniform. Not only does this corroborate the IMINT analyst's report on the observations from the satellite imagery, it would suggest that the formation of people were military personnel. It would be remiss of the ASA to exclude both items of information (i.e., the satellite imagery and the report from a HUMINT source) in an argument. In this toy example, the combination of the two causes would enable a stronger inference to the effect which, in this case, is the *invasion* hypothesis. As such, any scheme employed in the representation of hypotheses supported by information should allow for more than one item of raw intelligence, if necessary, which the argument from cause to effect does not allow for in its current form.

Evidence. There may be specific data points within the ASA's analysis known to be true such that those items of raw intelligence are treated as evidence. Evidence in this context can be thought of as certain information such that the probability of that observed event is equal to unity. The ASA may, however, doubt the veracity of the source who reported that raw intelligence, if any of the SSA's prior reports were found to be, for example, incorrect, biased, or deceptive. The distinction between probability and uncertainty is discussed in Section 3.1.2.

Ensuring that evidence features within the proposed scheme is important because evidence is a premise that cannot be defeated. Evidence can be formally represented in terms of structured argumentation through, for example, the set of axiomatic premises¹ \mathcal{K}_n within the ASPIC⁺ framework [177, 150]. The intelligence community reasons using real-world arguments. As defined in Section 2.3.2.3, real-world data stored within the AIF can be mapped to an ASPIC⁺ argumentation theory and then to a Dung-style argument system in order to allow a computational agent to reason about the acceptability of those arguments. The premises containing evidence will always be sceptically accepted such that they will feature in every extension or be labelled IN for, respectively, both extension- or labelling-based semantics. While this thesis is focused on developing tools for intelligence settings, such investigative and inquiry-based reasoning is carried out in a variety of different domains, such as within investigations by

¹Although this thesis does not consider axiomatic premises when dealing with the ASPIC⁺ framework, it is important to note that the formalism is capable of representing premises that cannot be defeated.

law enforcement or by companies who want to make the best decision based upon the information, evidence, and assumptions available to them. Thus, in such settings, ensuring that argumentation schemes can differentiate between evidence and information should make the representation of the reasoning about hypotheses more accurate and robust.

Assumptions. As alluded to at the start of this section, analysts may hold some assumptions which they draw upon while conducting reasoning. Moreover, defining and checking assumptions is common practice within intelligence analysis and there are a number of different tools that analysts employ to do this, such as the *Key Assumptions Check* [172], which I briefly introduced in Section 2.1.5.2. Such assumptions are made explicit within step two of the ACH and should also be clearly instantiated within the proposed models of an analyst's argumentation pertaining to hypotheses.

Recall from the running example that the ASA assumes that Country Y's leader is an authoritarian dictator with a thirst for power and resources. This assumption featured in the analyst's reasoning such that they concluded that an invasion was *high likely*. Without this assumption, the causal generalisation does not seem strong at all such that the argument can be attacked using all three critical questions. For instance, when evaluating the reasoning involved with the information from the IMINT analyst only, one could argue that the causal generalisation is weak (CQ1) because the information (i.e., a grouping of people) is not necessarily indicative of an invasion (CQ2) and other hypotheses (CQ3), such as a protest or Country Y bolstering up their defences, could be supported by the same item of raw intelligence, thus explaining why the formation of people were at the border. The argument within this example would be much stronger if it also included the analyst's assumption about the nature of Country Y's leader.

3.1.2 The Distinction Between Probability and Confidence

In a general sense, the probability of a cause or its effect, respectively, pertains to the likelihood of a SSA observing an event or an ASA's judgement about probability in light of the available data. Whereas, confidence captures an analyst's belief about uncertainty in their probabilistic judgements (i.e., how confident the analyst is in the likelihoods output from their analysis or the information received from their single-source counterparts). As discussed in Section 2.1.4, the intelligence community determines and communicates probabilities and uncertainties qualitatively by employing tools, such as the Probability (or Uncertainty) Yardstick, but how levels of confidence are communicated in practice is not well understood because analysts often only in-

clude the probabilities of hypotheses within *finished intelligence products* [67]. Both the ASA and SSA make probabilistic judgements. However, there is also a distinction in the confidence of those judgements for both the set of SSAs and the ASA because of, first, the uncertainty in a SSA's measurement and second, how much faith the ASA puts in the SSA's report.

Consider first the quantification of probability and confidence from the SSA's perspective. Think back to the IMINT analyst's collection of satellite imagery from the operational environment which they shared with the ASA as an item of raw intelligence. Suppose that a different SSA conducted another collection and discovered that there was *likely* thirty main battle tanks (MBTs) close to Country X's border. Based upon the domain expertise acquired by the SSA during previous collections, the IMINT analyst *likely* recognised the shape of thirty MBTs in the captured image. The confidence in that probabilistic judgement depends on a number of factors, such as the time of day that these images were captured, any cloud cover obstructing the image, the experience of the IMINT analyst, etc. For instance, if the images were taken in the middle of the daytime and with good visibility, then the IMINT analyst should have higher confidence in the conclusions they draw from their data. However, imagine that the image was acquired at dusk and under poor visibility, the analyst might not be so sure in this scenario.

Now think about the estimation of probability and confidence from the ASA's viewpoint, which is the most pertinent to the tools, techniques, and algorithms proposed within this thesis. The ASA receives a report from the IMINT analyst which includes the assertion that "*there were likely thirty MBTs close to Country X's border*", and the confidence in this judgement according to the SSA is *high* due to good visibility and the time of day when the image was taken. The ASA is now required to incorporate this raw intelligence into their analysis. While the ASA could blindly instantiate the aforementioned report as a premise within an argument for the *invasion* hypothesis, as well as including the probability (i.e., *likely*) and confidence (i.e., *high*) of such a premise, the ASA should evaluate the confidence they have in that source and the reported propositional content before including it within an argument. Arguments for a hypothesis that contain the information received from a SSA possess the argument structure of *argumentum ad alia* which is where, in this case, the ASA makes an argument that is based upon what another person has previously asserted [36]. Certain annotation schemes, such as Inference Anchoring Theory [37], enable the annotation of reported speech by partitioning natural language corpora into sets of locutions, reported propositions, and the actual propositional content contained within those assertions. Drawing on speech acts theory [75], one of the most prominent argumentation-based accounts of inter-

locutor communication, the reported speech and the propositional content are linked through an *illocutionary force* which combines utterances with propositions that form part of the argumentation being put forth. The ASA must make a judgement about how much confidence they hold in the propositional content reported by the SSA. Consider the case where it was known that the particular SSA who sent the report usually drew inaccurate conclusions from satellite imagery. In this instance, the ASA might assign a lower confidence to the assertion and the probability of the event contained within it (i.e., *likely*) such that the cause provides less support for the hypothesis within the argument in question.

Confidence also features in the assumptions made by the ASA during their analysis. For example, think back to the assumption at the start of this chapter: “Country Y’s leader is *likely* a dictator with a thirst for power and resources”. The ASA would assign a high confidence to this assumption if the *modus operandi* of Country Y’s leader remained unchanged. However, consider the case where Country Y’s leader recently decided to hold a fair and democratic election. Here, the ASA would indeed be required to lower the confidence of their *dictator* assumption due to increased uncertainty caused by a change in the leader’s most recent actions.

In the subsequent subsections, I discuss notions of confidence in causes and the probability of information, evidence, and assumptions, as well as hypotheses from the perspective of the ASA.

3.1.3 Confidence in Causes

As I described in the previous subsection, when the ASA receives items of reported information or evidence, or when they make assumptions, there will always be an associated confidence due to the context within which intelligence analyses are conducted. Confidence has been handled by an adapted version of the ACH process within the handbook, *Quick Wins for Busy Analysts* [222], where each row within the ACH matrix is assigned a colour (i.e., green, yellow, red) which is synonymous with a datum’s uncertainty. However, this approach does not incorporate these confidence levels into any of the downstream judgements about the probabilities of hypotheses, as I described in Chapter 2.

ASAs reason with information, evidence, and assumptions that are conflicting, incomplete, potentially deceptive, and false. This makes reasoning in such a space difficult due to the inherent uncertainty in every data point within an analysis. Looking back to the running example and the information shared with the ASA by the covert HUMINT analyst, the ASA must think critically about the information that their single-

source counterpart has shared. For instance, the ASA could conduct a coarse count, counting the number of times the HUMINT agent was right or wrong previously, to assign a confidence to the reported information. Alternatively, as the information reported by the HUMINT agent is a witness account, the ASA might draw upon arguments about witness testimony and the critical questions within, say, Walton's argument from witness testimony [242, 178], which are listed below:

WTCQ1: Is what the witness said internally consistent?

WTCQ2: Is what the witness said consistent with the known facts of the case (based on evidence apart from what the witness testified to)?

WTCQ3: Is what the witness said consistent with what other witnesses have (independently) testified to?

WTCQ4: Is there some kind of bias that can be attributed to the account given by the witness?

WTCQ5: How plausible is the statement asserted by the witness?

By sequentially working through each of the above critical questions, the ASA is able to reason about the confidence in the reported information and its source, the HUMINT agent. The first question (WTCQ1) pertains to the consistency of the HUMINT's assertion, which in this case is found to have no internal conflicts. The reasoning involved with the second (WTCQ2) and third (WTCQ3) critical question is somewhat related because the satellite imagery observed a large number of people in Country Y and close to the border Country X. In this case, the images of the grouping of people would be classified as evidence, unless the photograph was a fake or taken prior to the time frame required for analysis (i.e., before the relations between both countries had deteriorated). Given the photograph was taken after the ASA raised a RFI, the uncertainty within the image is not about its contents, but who the people in the image are associated with, and what their plans and goals are. As the HUMINT agent reported that the formation of people were wearing military uniform, the HUMINT agent's account is consistent with the known facts (WTCQ2) and what others have independently testified to (WTCQ3). The fourth critical question (WTCQ4) forces the ASA to think about any biases which might feature in the report from the HUMINT source. For example, imagine that the ASA knew that the HUMINT agent owned a stock investment portfolio that contained the vendors of military equipment for both Country X and Y. One could assume that the covert information source might be influenced by profit and motivated to report that the grouping of people were wearing military uniform. Thus,

in this case, the HUMINT agent's assertion could potentially be biased and the confidence in the propositional content within their statement should be lowered as a result. Finally, the ASA must consider the plausibility of the witness' statement (WTCQ5), which is indeed believable given the deteriorating relationship between both countries and the satellite imagery from the IMINT source.

Although the above example focuses on a HUMINT source, the ASA should reason about the confidence in reports of both information and evidence, and also for any assumptions made by the ASA that feature within the analysis. The prescribed method of choice for uncertainty quantification should be left up to the analyst. However, one can easily see that incorporating the confidence of items of information, evidence, and assumptions within arguments becomes increasingly important as the amount of sources who report conflicting data increases. Expressing such uncertainty within arguments about hypotheses should enable a better representation of the ASA's reasoning, as well as enabling them to produce better probabilistic estimates.

3.1.4 The Probability of Causes and their Effect

The set of premises supporting a conclusion in the form of a set of causes leading to an effect, respectively, are closely related. The probability of items of information, evidence, and assumptions and how they affect the probability of a hypothesis should be no different. For instance, imagine that a hypothesis was supported by two causes, both of which had a probability of *highly unlikely*. In this instance, one would expect the probability of the hypothesis supported by those premises to be lower also. ASAs employ analytical techniques, such as the ACH in Section 2.1.5.2, to calculate the probability of a hypothesis by conducting a coarse count of the number of times that row entries are inconsistent with the hypothesis in question, which provides a robust evaluation of a hypothesis' probability. The probabilistic judgements made by the analyst often concern novel events, such that those probabilities are epistemic and incorporate an analyst's subjective belief. It may seem that the use of conditional probability is a perfect fit for the representation and modelling of the dependence of a set of premises on the hypotheses they support. However, it is incredibly challenging to reliably compute the probability of observing, say, an item of information conditioned on a hypothesis, especially for circumstances that haven't happened before, which is often the case within intelligence settings. To compute the likelihood of a set of row entries conditioned on a hypothesis, the analyst would be required to calculate the joint probability of a hypothesis and the information, evidence, and assumptions, as well as make a probabilistic judgement about that hypothesis' prior. As many of the situa-

tions being assessed concern non-repeatable events, it is highly unlikely that those joint probabilities and priors will be correct or reliable. As I will show in Chapter 5, an analyst's set of prior probabilities for hypotheses can be refined through an approach that combines Bayes' theorem and probabilistic argumentation frameworks. However, for the purposes of this chapter, the proposed argumentation scheme, the *Argument from Multiple Causes to Effect*, only includes marginal probabilities.

3.2 The Argument from Multiple Causes to Effect

The ASA is required evaluate the set of MECE hypotheses using the information and evidence shared by SSAs and their own assumptions to produce estimates of the probability of all hypotheses that feature within their analysis, as per the ACH as discussed in Section 2.1.5.2. As in Section 2.1.1, after analysis, the ASA produces a *finished intelligence product*, containing the lead hypothesis, its probability of occurrence, and any items of diagnostic information, which is then disseminated with stakeholders for downstream decision-making. In Section 3.1, I discussed how Walton's argument from cause to effect is capable of modelling the inference from a single item of information, evidence, or assumption to a hypothesis. However, as I explored in the previous section, the reasoning underpinning a hypothesis is often never as simple as a single cause supporting an effect. For any proposed scheme to provide tangible benefit to the intelligence community, it should allow analysts to include as many supporting items of information, evidence, and assumptions as they believe are necessary, along with their associated probability and confidence levels, in order to fully describe the outputs from their analysis. Moreover, during analysis, analysts may formulate narratives associated with data that supports a given hypothesis to aid in the evaluation of those data and the judgement of the probability of hypotheses. In [25], Bex et al. propose a comprehensive framework for evaluating evidence in criminal cases within the legal domain, using causal stories and argumentation. This hybrid theory is relevant to intelligence analysis, where analysts often deal with complex, incomplete, conflicting, deceptive, and missing data. In such contexts, constructing plausible narratives could help the analyst understand possible scenarios, whilst also ensuring that those stories are grounded in logical reasoning, through use of argumentation. The scheme proposed in this chapter also draws upon argumentation, namely an adaptation of an argumentation scheme, to enable analysts to produce sound representations of the reasoning underpinning hypotheses.

In this subsection, I take Walton's general scheme, the argument from cause to

effect, and deepen its level of specificity to enable it to be applied specifically in intelligence contexts where probabilistic reasoning is used. The proposed scheme is capable of representing an argument for a hypothesis that is supported by multiple causes with probabilities from a set of SSAs, and the ASA's confidence in those reports, as well as the analyst's judgement about the probability of a hypothesis. I now present the adapted scheme below:

An Argument from Multiple Causes to their Effect:

Major Premise: Generally, if the set of information I_1, \dots, I_N reported by the sources S_{I_1}, \dots, S_{I_N} and evidence E_1, \dots, E_M reported by the sources S_{E_1}, \dots, S_{E_M} have been observed, and based upon the assumptions A_1, \dots, A_K , with some probability and in some confidence interval, then the hypothesis H has occurred/is occurring/will occur with some probability $P(H)$.

Minor Information Premise 1: The source S_{I_1} asserts the information I_1 was observed with some probability $P(I_1)$, and the confidence in that assertion and the source's estimate is $\pm\sigma_{I_1}$.

...

Minor Information Premise N: The source S_{I_N} asserts the information I_N was observed with some probability $P(I_N)$, and the confidence in that assertion and the source's estimate is $\pm\sigma_{I_N}$.

Minor Evidence Premise 1: The source S_{E_1} asserts evidence E_1 was observed with certainty such that its probability $P(E_1)$ equals unity, and the confidence in that assertion and the source's estimate is $-\sigma_{E_1}$.

...

Minor Evidence Premise M: The source S_{E_M} asserts evidence E_M was observed with certainty such that its probability $P(E_M)$ equals unity, and the confidence in that assertion and the source's estimate is $-\sigma_{E_M}$.

Minor Assumption Premise 1: The assumption A_1 employed during analysis has a probability $P(A_1)$ and a confidence interval equal to $\pm\sigma_{A_1}$.

...

Minor Assumption Premise K: The assumption A_K employed during analysis has a probability $P(A_K)$ and a confidence interval equal to $\pm\sigma_{A_K}$.

Conclusion: Therefore, the hypothesis H has occurred/is occurring/will occur with some probability $P(H)$.

The above argumentation scheme can model the reasoning involved with data points which support hypotheses within the ACH. The scheme enables ASAs to instantiate all relevant and necessary items of information, evidence, and assumptions, as well as their associated probabilities and confidence intervals. The scheme can also model the reasoning about past, present, and future hypotheses, which makes it fit for use in not only

intelligence settings, but also in other investigative and inquiry-based settings, where a person is required to reason under uncertainty, such as in law enforcement, business intelligence, legal cases, etc. The distinction between the probability of an item of information and evidence, and the confidence interval assigned by the ASA, is made clear which enables them to allocate a strength (or weight) to the set of minor premises. The probability in the conclusion represents the analyst's judgement about the likelihood of a hypothesis, and it is a possible representation such that it is not checked for consistency against the set of probabilities and confidence intervals contained within each premise. As I will show in Chapter 5, the probability of a hypothesis (i.e. the conclusion of the argument) can be verified using the set of premises. However, the addition of those refined probabilities into the above scheme, the argument from multiple causes to effect, is a task left for future work. An important point to highlight here is that the confidence interval assigned to a premise that contains evidence is always negative (i.e., $-\sigma_{E_i}$). The reason for this is that the probability of a *Minor Evidence Premise* will always equal unity – i.e., $P(E_i) = 1$ – because, as I described in Section 3.1.1, evidence is an item of factual information. However, the analyst might, for example, doubt the honesty of the source who reported that fact. Finally, the scheme enables enthymemes to be explicitly instantiated which should enable other analysts to understand the entirety of an ASA's thinking during analysis.

Returning to the running example for the penultimate time. Recall that:

- The ASA assumed that “Country Y's leader is *likely* (i.e., $0.55 < P(A_1) < 0.7$) a dictator with a thirst for power and resources” with a small uncertainty interval (i.e., $\pm\sigma_{A_1} = 0.05$) initially. However, Country Y's leader recently decided to hold a fair and democratic election such that the uncertainty of the *dictatorship* premise increased (i.e., $\pm\sigma_{A_1} = 0.2$).
- The IMINT source S_{I_1} reported that “Country Y has *likely* deployed a large and unknown number of people close to their border with Country X” and the ASA's confidence in that assertion is high such that the uncertainty interval is small (i.e., $\pm\sigma_{I_1} = 0.01$).
- A different IMINT source S_{I_2} reports “there was *likely* (i.e., $0.55 < P(I_2) < 0.7$) thirty MBTs close to Country X's border” but there is a larger uncertainty interval (i.e., $\pm\sigma_{I_2} = 0.5$) associated with this report because the image was collected at dusk, when there was a lot of cloud cover, and the source frequently makes wrong observations.
- A covert HUMINT source S_{I_3} asserts that “the crowd of people in Country Y and

at Country X's border was *almost certainly* (i.e., $P(I_3) > 0.9$) wearing military uniform". However, the ASA deems the uncertainty in this source's estimate to be high (i.e., $\pm\sigma_{I_3} = 0.5$) because the source owns stock investments of Country X and Y's vendors of military equipment.

Using the details above, the argument from multiple causes to effect can be instantiated as follows:

Major Premise: Generally, if "Country Y has *likely* (i.e., $0.55 < P(I_2) < 0.7$) deployed a large and unknown number of people close to their border with Country X", "there was *likely* (i.e., $0.55 < P(I_2) < 0.7$) thirty MBTs close to Country X's border", "the crowd of people in Country Y and at Country X's border was *almost certainly* (i.e., $P(I_3) > 0.9$) wearing military uniform", and based upon the assumption that "Country Y's leader is *likely* (i.e., $0.55 < P(A_1) < 0.7$) a dictator with a thirst for power and resources", and according to the assigned confidence intervals, then "it is *highly likely* (i.e., $0.75 < P(H_1) < 0.85$) that Country Y will invade Country X".

Minor Information Premise 1: The source S_{I_1} asserts that "Country Y has *likely* (i.e., $0.55 < P(I_2) < 0.7$) deployed a large and unknown number of people close to their border with Country X", and the confidence in that assertion and the source's estimate is equal to $\sigma_{I_1} = 0.01$.

Minor Information Premise 2: The source S_{I_2} asserts "there was *likely* (i.e., $0.55 < P(I_2) < 0.7$) thirty MBTs close to Country X's border", and the confidence in that assertion and the source's estimate is equal to $\pm\sigma_{I_2} = 0.5$.

Minor Information Premise 3: The source S_{I_3} asserts that "the crowd of people in Country Y and at Country X's border was *almost certainly* (i.e., $P(I_3) > 0.9$) wearing military uniform", and the confidence in that assertion and the source's estimate is equal to $\pm\sigma_{I_3} = 0.5$.

Minor Assumption Premise 1: The ASA assumed that "Country Y's leader is *likely* (i.e., $0.55 < P(A_1) < 0.7$) a dictator with a thirst for power and resources" with a confidence interval equal to $\pm\sigma_{A_1} = 0.2$.

Conclusion: Therefore, the ASA asserts that "it is *highly likely* (i.e., $0.75 < P(H_1) < 0.85$) that Country Y will invade Country X".

Here, each minor premise is assigned a probability based on either the SSA's or ASA's judgement and depending on whether the premise is item of information, evidence, or assumption. The ASA then employs the confidence interval to specify how much they believe a source's assertion, which impacts how much support each premise provides for a hypothesis. The proposed scheme should enable analysts to fully explain their reasoning about hypotheses, probabilities, and uncertainty, as well as reducing the number of enthymemes present within analysis.

3.2.1 Extending the Critical Questions

As with any other argumentation scheme, the extended scheme should be accompanied by a set of critical questions which enables the analyst to critically assess the confidence of each premise, its source, and the probabilistic estimates made within the argument. The critical questions for the proposed scheme were adapted from the questions found within Walton's original argument from cause to effect. The adapted set of critical questions are presented below:

- CQ1: How strong is the causal generalisation?
- CQ2: Are there any potential items of information or evidence that have not been collected which should be in order to strengthen the causal generalisation?
- CQ3: Are the probabilities of the information, evidence, and assumptions reasonable?
- CQ4: Are the confidence intervals of the information, evidence, and assumptions reasonable?
- CQ5: Are there any other reports of information and evidence, or assumptions, which disagree with the propositional content within each minor premise?
- CQ6: Are there any items of information, evidence, or assumptions that would increase the uncertainty in a minor premise's confidence interval?
- CQ7: If one (or more) of the minor premises were found to be false, would the strength of the inference change?
- CQ8: Are the set of information, evidence, and assumptions (if there is any) strong enough to warrant the causal generalisation?
- CQ9: Are there any other causal factors that interfere with the production of the hypothesis in the given case?
- CQ10: Are there any other hypotheses which can be inferred from the minor premises available?
- CQ11: Are there any other arguments containing other hypotheses with a different set of minor premises that are in conflict with this argument?

The eleven critical questions above were systematically derived from Walton's set of questions in the original argument from cause to effect, with some slight alterations and additions to ensure the questions were appropriate for use within intelligence settings. The extended questions should enable the analyst to critically evaluate the argument from multiple causes to effect by focusing their attention on: the soundness of the argument; any weak points which require the collection of more information through RFI generation; the quality of arguments; and whether the probabilities and confidence levels are reasonable.

Coming back to the running example for the last time, CQ1 is the same as WCQ1 in Section 3.1. The ASA uses CQ1 to assess how strong the argument for the invasion hypothesis is which, for this illustration, they find to be very strong because they have listed all the relevant information and assumptions within the generalisation.

Moving onto to CQ2. Although the ASA has instantiated an argument using the scheme, this question enables them to think about whether they require more information to make their argument and if they do, then they should raise another RFI to obtain the missing raw intelligence. Perhaps, after reading *Minor Information Premise 2*, the ASA would like to know the type of MBT at Country X's border and therefore, task another covert HUMINT source to collect this raw intelligence.

CQ3 enables the ASA to consider whether the probabilistic estimates are reasonable and if they are not, then the ASA should alter those probabilities or change the confidence interval. For example, the source S_{I_3} in the *Minor Information Premise 3* asserts that it is almost certain that the formation of people at Country X's border were wearing military uniform. However, the ASA might question how close the HUMINT source was to the observation, how good their eyesight is, the visibility on that day, or whether the source was in fact biased, which was the case for this example. The fact that the source is biased would indicate that the ASA should either reduce the likelihood of the observation or the increase the uncertainty interval.

CQ4 is similar to CQ3. However, CQ4 deals with instances where the probabilities of information, evidence, and assumptions are reasonable but the confidence interval is not. Perhaps the ASA's confidence was too large (small) for the information in a minor premise and should be decreased (increased).

ASAs try to reduce confirmation bias when conducting analyses, as I briefly discussed when introducing the ACH in Section 2.1.5.2. Utilising conflicting information, evidence, and assumptions in the assessment of premises for arguments containing hypotheses is akin to that practice (CQ5). When a data point disagrees with one (or more of the) minor premise(s) within an argument, the ASA might reduce either the probability, confidence interval, or both for the conflicted minor premise(s).

CQ6 forces the ASA to think about whether there are any other items of raw intelligence or assumptions that would increase the uncertainty of a minor information or evidence premise. For example, if the ASA received a report stating there was a show for MBT enthusiasts close to the border of Country X at the same time as the observation made by the source S_{I_2} , who asserted *Minor Information Premise 2*, then this would increase the uncertainty in the propositional content reported by the source S_{I_2} , and ultimately decrease the probability of the *invasion* hypothesis.

Considering how the strength of the causal generalisation changes as a result of false propositional content within any of the minor premises is important (CQ7). Such a question should enable the ASA to consider how dependent the hypothesis is on one (or more) of the premise(s) which support it. Moreover, this question is similar to step seven of the ACH where the analyst is required to sequentially remove each row entry from the ACH matrix in order to assess whether the probability of hypotheses changes.

CQ8 is similar to WCQ2 in Section 3.1 but has been extended to include the items of information, evidence, and assumptions found with the proposed scheme.

CQ9 is the same as WCQ3.

CQ10 draws upon WCQ3 but for the case of hypotheses. CQ10 allows the ASA to think about the production of alternative hypotheses using the same set of minor premises instantiated within the scheme.

Finally, CQ11 enables the ASA to think about the argument as a whole. The ASA must assess whether there are any other arguments containing alternative hypotheses and a different set of minor premises that might be in conflict with the one instantiated using the scheme.

3.3 Summary

In this chapter, I have discussed Walton's argument from cause to effect and accompanying critical questions with the context of intelligence settings. I explored the different cases where the specifics of the original scheme is not fit for operational use within such settings. I then proposed an adapted version of the scheme so that it is capable of representing the inference from a set of premises – namely information, evidence, and assumptions – to a hypothesis, which is the conclusion of the argument. The more detailed scheme also allows the ASA to include probabilistic judgements about the causes and hypothesis under scrutiny, as well as a description of the uncertainty present within the propositional content of each minor premise. I also proposed an extension to Walton's set of critical questions to enable the ASA to critically evaluate

their argumentation. The extended scheme should enable ASAs, as well as other type of analyst reasoning within investigative and inquiry-based settings, to fully capture their rationale, but this claim should be tested empirically in future work to ascertain whether the scheme works in practice.

This chapter has demonstrated that argumentation schemes can be employed to represent intelligence analysts' reasoning about the support for a hypothesis that is provided by items of information, evidence, and assumptions. In the next chapter, I assess the feasibility of using AI, specifically large language models, for MECE hypothesis generation for a real-world intelligence scenario.

Chapter 4

On Large Language Models for Hypothesis Generation

While the previous chapter focussed on hypothesis representation, this chapter concerns hypothesis generation. In this chapter, I present the results from experiments on a number of state-of-the-art large language models employed for hypothesis generation within an intelligence context. Models were tasked with generating a number of hypotheses using information from a real-world intelligence scenario. The outputs from models were assessed on three axes, namely: the word-level semantic similarity between target hypotheses and model completions; the sentence-level semantic similarity between target hypotheses and generations; and the probability that a set of model completions contained hallucinations. I then discuss those results and address whether large language models, in their current state, are an effective type of model for hypothesis generation within an intelligence context. I conclude the chapter with a summary and identify future avenues to extend the work.

4.1 Introduction

Hypothesis generation is a fundamental process within the *Intelligence Cycle*. It occurs at the start of the *analysis* stage, where analysts must generate the set of MECE hypotheses to ensure that they evaluate the set of all possible and plausible outcomes for a given scenario. As I discussed in Section 2.1.5.1, there are a number of different analytical techniques for hypothesis generation, such as *Simple Hypotheses Generation*, *Quadrant Hypothesis Generation*, and the *Multiple Hypotheses Generator*TM

[173]. The ASA employs the aforementioned tools depending on the situation being evaluated. For instance, the ASA might use *Simple Hypothesis Generation* when trying to gain a better understanding of a time critical and unfolding scenario, *Quadrant Hypothesis Generation* when there are two clear forces driving a situation, or the *Multiple Hypotheses Generator*TM when trying to explore the space of all potential hypotheses. While these techniques help analysts generate *good* hypotheses, the use of AI, specifically large language models, could aid the process and allow analysts to identify previous unconsidered hypotheses.

Of late, large language models have taken the world by storm, and their applications are numerous and seemingly ever-increasing. For example, large language models have been employed in: general-purpose AI chat bots, which can retrieve and present information in the form of text [201, 214, 87, 158, 122], and even implement algorithms in specific programming languages [51]; creative work, such as short-form writing, for idea generation [93], poetry [49], and works of fiction in multiple languages [182], as well as long-form writing, for co-writing screenplays and theatre scripts [148]; knowledge representation and reasoning [209, 246, 140, 28, 137, 61]; law, for case prediction [95] and question answering [249, 29, 120, 194, 107]; and social sciences, in order to both analyse [147, 170] and simulate human behaviour and relationships [165], to name a few pertinent examples.

Many of the recent, exciting advances within the field of natural language processing can be attributed to the capabilities of large language models. As I discussed in Chapter 2, some researchers, such as Snidaro, have made the claim that such models could potentially “*revolutionise*” [204, p. 449] the *analysis* tasks within *Intelligence Cycle*. One obvious question that arises from such a statement is whether large language models are, in fact, capable of generating previously unconsidered hypotheses for intelligence analysts. As I describe in Section 4.2.3, large language models are capable of conducting conditional text generation, based upon a provided context. As a task, hypothesis generation requires an analyst to reason abductively, in that they must observe, understand, and synthesise raw intelligence and employ that data to infer hypotheses. In [136], Liu, Neubig, and Andreas show that large language models perform well at some tasks involving deductive and inductive reasoning. However, their reasoning abilities dwindle when asked to conduct abductive reasoning.

In the context of the current capabilities of large language models, this chapter makes one specific contribution, a feasibility study was conducted which considered whether large language models could be leveraged for hypothesis generation within intelligence settings. A number of different, state-of-the-art language models were tested to assess how similar model completions were to target hypotheses, at the sentence-

and word-level, and the probability of model hallucinations was quantified. The results presented in this chapter are a first step towards automated hypothesis generation for intelligence analysts, and should help guide future research on the topic.

The chapter is structured as follows. The methodology employed in the study of large language models for hypothesis generation is described in Section 4.2. The dataset used in the testing is introduced in Section 4.2.1, and the preparation and annotation of it is reported on in Section 4.2.2. The large language models employed in the study and how they were prompted to generate hypotheses is explained in Sections 4.2.3 and 4.2.3.1, respectively. The way in which the outputs from those models were evaluated is made clear in Section 4.2.3. Results from a number of experiments are presented in Section 4.3 and discussed in Section 4.4. The chapter concludes with a summary and avenues for future work in Section 4.5.

4.2 Methodology

In this section, I describe the methods employed in the creation and use of a dataset, which contained hypotheses supported by sets of information. The dataset was employed to prompt a series of large language models to ascertain whether such models could be used to generate the set of MECE hypotheses. Due to the nature of intelligence analysis, many *finished intelligence products* are classified. There are a number of reasons for this, such as the reports generated by ASAs may contain sensitive information about people and collection methods, to a name a couple. The only openly available data, which includes such unclassified intelligence products, can be found on the X, formerly Twitter, account of the United Kingdom's Ministry of Defence¹, where they have been providing updates on the ongoing invasion (or 'special military operation', as the Russian government refer to it) between Russia and Ukraine. The Ministry of Defence started providing updates on developments at the start of the conflict to counter the amount of misinformation on the internet.

An important point to note here is that the scheme proposed in Chapter 3, the argument from multiple causes to effect, was not employed in the prompting of large language models in this chapter. The reason for this is that there was no way to verify whether a datum that supported a hypothesis was an item of information, evidence or assumption. Thus, it was assumed that the Tweets in the dataset from the United Kingdom's Ministry of Defence only contained information and hypotheses.

¹<https://x.com/DefenceHQ> (Accessed on: 10.02.2024)

4.2.1 Dataset

The United Kingdom’s Ministry of Defence provided this research with a total of 6,536 Tweets from their X account, for the purposes of assessing the feasibility of employing large language models for hypothesis generation. The Tweets provided date from 1st February 2022 to 23rd October 2023. A subset of the shared Tweets contained intelligence reports that included at least one hypothesis. An example, comprised of three Tweets², is presented in Figure 4.1. The Tweets contained either information or a hypothesis, where each statement was accompanied by a qualitative description of probability (i.e, *high likely* and *likely*), in line with the Probability (or Uncertainty) Yardstick, as described in Chapter 2.



Figure 4.1: An example set of Tweets from the X account of United Kingdom’s Ministry of Defence, published on 25th April 2023.

The first and second Tweet can be grouped into items of information which support the hypothesis asserted in the third Tweet. The first Tweet contains some raw intelligence shared by the Ukrainian General Staff and states that the rate of Russian casualties *highly likely* fell by 30%. An analyst from the United Kingdom’s Ministry

²<https://x.com/DefenceHQ/status/1650739827552186370> (Accessed on: 10.02.2024)

of Defence then supplements this information with a corroborating sentence about the ‘*exceptionally heavy Russian casualties over [the] January-March 2023*’ time period. This validates the Ukrainian General Staff’s assessment and implies that Russia’s daily casualty rate was always likely to decrease because it was exceedingly high in the months prior to the report in April 2023.

The second Tweet appeals to data communicated by the Ukrainian General Staff which indicated that the average number of daily Russian casualties decreased from 776 to 568. The second Tweet concludes with a statement made by the United Kingdom’s Ministry of Defence containing their belief about the uncertainty in the raw intelligence shared by the Ukrainian General Staff, where they deemed it ‘*likely accurate*’. The presentation and use of raw intelligence, probability, and uncertainty within the analysis for this particular example is similar to the proposal made in this thesis for an argumentation scheme to represent hypotheses, as presented in Chapter 3. For instance, one can think about the information and probabilities shared by the Ukrainian General Staff as a report from a SSA and the inclusion of such a datum in an analysis is accompanied by a verbal uncertainty quantification made by an ASA, which in this case is an analyst from the United Kingdom’s Ministry of Defence.

As previously mentioned, the third Tweet contains a hypothesis, which was in accordance with Pherson and Heuer Jr.’s definition, as per Chapter 1. The hypothesis is a statement in the most literal sense of the word because it is a single sentence. Many of the other hypotheses within the data shared by the United Kingdom’s Ministry of Defence were sentences. However, there were instances where the same items of information supported more than one hypothesis, and the annotation of such cases is described in more detail in Section 4.2.2.1. Similar to the items of raw intelligence, each hypothesis contained a qualitative probabilistic judgement pertaining to the likelihood of that hypothesis.

In the next subsection, I describe the preparation and annotation of the dataset so that it could be employed for the fine-tuning and prompting of a number of large language models.

4.2.2 Data Preparation and Annotation

The 6,536 Tweets shared with this research by the United Kingdom’s Ministry of Defence contained all the Tweets published on their account over a one and half year period. A subset of the Tweets contained intelligence reports which were similar to the example presented in Figure 4.1. Tweets that potentially included intelligence products were identified and extracted using a set of unique identifiers, which are presented

in Appendix A.1. The dataset was partially pre-processed using Python [232], and a combination of Pandas [211] and Regular Expressions [4], to extract potential intelligence reports from the dataset using the unique identifiers. This reduced the initial set of 6,536 Tweets to 3,197 Tweets.

The strings contained within each set of Tweets were split into sentences, a type of EDU (Section 2.3.3.1) that is widely employed throughout the natural language processing literature. Hypotheses were identified manually within the reduced dataset according to Pherson and Heuer Jr.’s definition of a hypothesis in [173]. Drawing upon the most literal use of the word *statement*, hypotheses were assumed to be one sentence in length. In reality, a hypothesis might be longer than one sentence. However, this assumption was employed to ensure that the annotation of the data was consistent.

4.2.2.1 Strategic, Operational, and Tactical Hypotheses

In cases where there was more than one hypothesis that was supported by the same set of information, I tried to ascertain whether the raw intelligence supported either a tactical, operational, or strategic hypothesis. In a broad sense, the difference between those three types of hypothesis is the time horizon upon which they are observed, their scope, and their level of detail. The difference between the aforementioned types of hypotheses is made apparent when looking to the different definitions of strategic, operational, and tactical intelligence. For example, strategic intelligence is defined by the United States’ Department of Defence as the

“intelligence required for the formation of policy and military plans at national and international levels.” [225, p. 522]

Strategic hypotheses focus on overall objectives which could take a great deal of time to come to fruition. For instance, imagine the hypothesis: *Country A, our neighbour and adversary, will highly likely invade.* The outworking of such a hypothesis may take a considerable amount of time because of all the planning and logistics involved in such an operation. The realisation of that hypothesis depends on the level of readiness of that hostile nation, as well as the success of a number of operational objectives, which must be achieved before the strategic hypothesis occurs.

The United States’ Department of Defence also defines operational intelligence which can, again, be employed to gain an understanding of operational hypotheses, and strategic ones too. Operational intelligence is defined as

“intelligence that is required for planning and conducting campaigns and

major operations to accomplish strategic objectives within theatres or operational areas.” [225, p. 399]

Here, the link between strategic and operational intelligence is made explicit. Operational intelligence informs strategic objectives and thus, strategic hypotheses. The raw intelligence received, and hypotheses generated and evaluated, at an operational level shed light on the probability of strategic hypotheses. It would be remiss of the analyst to omit such operational, raw intelligence from their reasoning about strategic hypotheses.

Tactical hypotheses are the final type of hypothesis. Such hypotheses pertain to situations and events that are evolving now. For instance, a commander might raise a RFI, in accordance with the *Intelligence Cycle*, as discussed in Chapter 2, concerning the movement of a foreign country’s military personnel, who just so happen to be situated on the same battlefield as them. The ASA, who received collected information pertaining to that evolving, tactical situation may well use such data in combination with other items of operational and tactical intelligence in order to understand what the adversary intends to do operationally and strategically. In this instance, the tactical hypothesis, which has been evaluated using tactical intelligence, should aid in the evaluation of operational hypotheses, and operational hypotheses and intelligence should be employed in the evaluation of strategic ones too. Tactical hypotheses were not observed within the dataset provided by the United Kingdom’s Ministry of Defence, which was most likely due to their level of classification and the fact that the United Kingdom is not currently at war. However, understanding the rich interplay between such hypotheses should provide a good understanding of how, once evaluated, each type of hypothesis can be employed in the evaluation of others. The relationship between strategic, operational, and tactical hypotheses seemingly forms something akin to a tree structure, where the evaluations of operational and tactical hypotheses could aid in the assessment of strategic hypotheses. However, formally defining such a relationship is a task left for future work.

Upon studying the dataset, it was noted that for Tweets which contained more than one hypothesis, there were a series of operational hypotheses supported by operational intelligence, which usually culminated in support for a strategic hypothesis. The annotation process for those Tweets was the same and goes as follows. First, the task was to split the Tweets into sentences, identify the strings that contained hypotheses and decide whether they were either strategic or operational hypotheses. Second, using the chronology of sentences within the Tweets and the type of hypothesis (i.e., operational or strategic), the first information and operational hypothesis strings were appended to

a dataset as a feature and label, respectively. Third, the prior hypothesis string was added to the information string, if it was relevant to the next hypothesis, as well as any subsequent intelligence, creating a new information string which was, again, appended to the dataset as a feature and accompanied by its label, the next hypothesis. This procedure continued for all the hypotheses within a set of Tweets until there were no more hypotheses to continue the process.

An example annotation of a set of Tweets that contained strategic and operational hypotheses is presented in Figure 4.2. The set of Tweets pertains to the Wagner group’s attempted coup d’état on 24th June 2023. The first two hypotheses (i.e. Hypo 1 and Hypo 2) were operational ones that describe specific, short-term actions that were potentially taken by the Russian state media and government to respond to the mutiny. Both hypotheses were supported by items of information that contained data collected from the operational environment, with the goal of providing support for both hypotheses. The first hypothesis (i.e. Hypo 1) was supported by the assertion that Russian media attempted to respond to the insurrection in three phases (i.e. Info 1). The second hypothesis (i.e. Hypo 2) was supported by the first item of information (i.e. Info 1), the first hypothesis (i.e. Hypo 1), and some discussion about how Vladimir Putin was actively involved in the mutiny’s resolution which, as a result, was handled without any killing. A further interesting point about the sentence containing the second hypothesis (i.e., Hypo 2) is that it contains both a hypothesis and an item of information, which the EDU segmentation approach was not able to capture. For instance, if the strings were split into reconstructed propositions using ADU segmentation, then Hypo 2 could be refined into the following:

- **Info 2.1:** Wagner Telegram channels have largely gone silent.
- **Hypo 2:** The Russian government has almost certainly intervened and taken control of the Wagner group’s Telegram channels.

The ADU segmentation method above is capable of dealing with dislocations (Section 2.10) within sentences. Moreover, reconstruction of assertions into propositional form should reduce the number of enthymemes that language models would need to reason with. However, investigating whether ADU segmentation would provide any tangible increase in model performance is a task left for future work.

Finally, consider the third hypothesis (i.e. Hypo 3) in Figure 4.2 which, in this particular case, is an example of a strategic hypothesis. Here, the projection strength is strategic because maintaining an image of strength will most likely lead to desired outcomes, such as domestic stability and legitimacy, deterrence against internal and

external threats, and preservation of power and long-term control of Russia and the Eastern world.

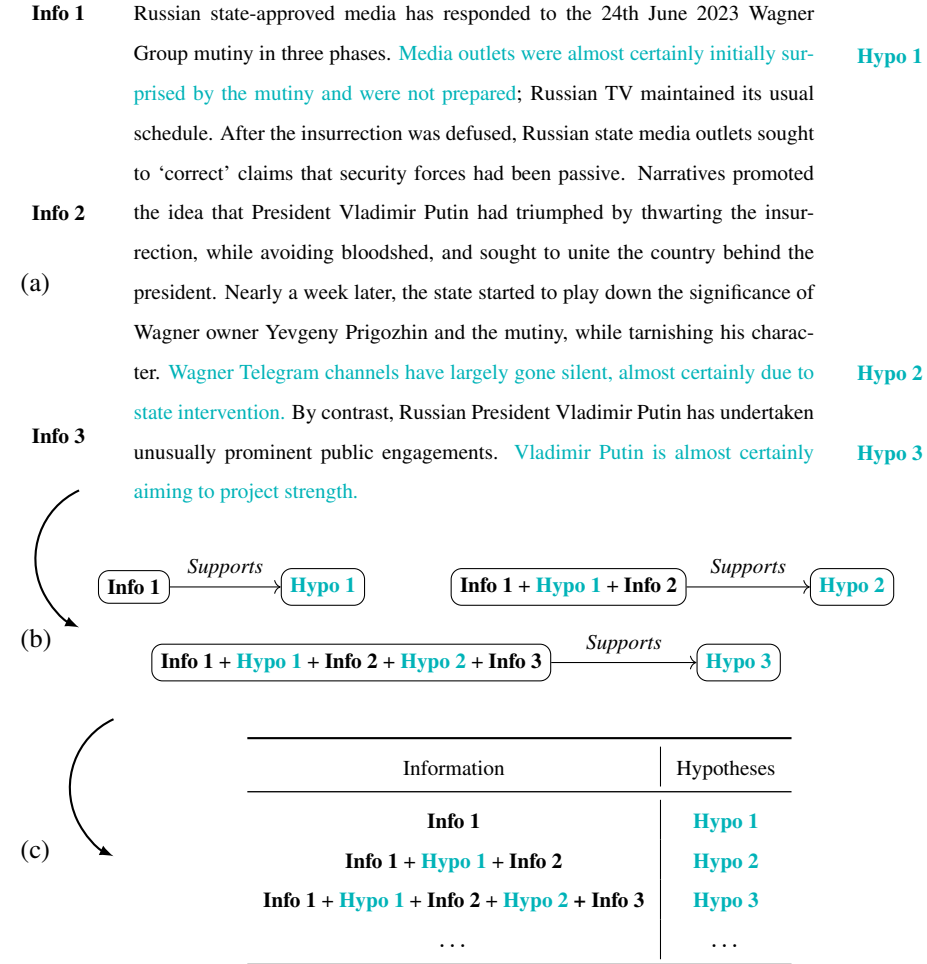


Figure 4.2: (a) A combined set of Tweets that contained an intelligence report with two operational hypotheses, denoted Hypo 1 and Hypo 2, and astrategic hypothesis, referred to as Hypo 3, (b) the identification of support for each hypothesis, and (c) the instantiation of the annotation with the dataset.

4.2.2.2 Training, Validation, and Test Datasets

After completing the manual annotation process described in the previous subsection, a total of 1,027 datapoints were identified, extracted, and instantiated within a dataset

that contained information (features) that supported hypotheses (labels). The dataset was partitioned for training (75%), validation (15%), and testing (10%). The data was randomly shuffled using Numpy [97], a widely-used Python [232] library for data science, using the same random seed³, so that fine-tuned models were trained and validated, and standard and fine-tuned models were tested, using the same data.

In the subsection to follow, I introduce the standard and fine-tuned large language models that were employed in this feasibility study and describe how those models were prompted to generate hypotheses using the aforesaid dataset.

4.2.3 Large Language Models

Large language models are trained on vast swathes of data which renders them capable of understanding natural language corpora and generating human-like text. They do this through a combination of transformer architectures and a self-attention mechanism, first introduced by Vaswani *et al.* in [233]. One of the most well-known language models is OpenAI’s ChatGPT [158], which employs their Generative Pre-Trained Transformer (GPT) models behind the user interface. GPT models were trained by OpenAI on data up to a specified date⁴. OpenAI provide an Application Programming Interface (API) to enable researchers, industry specialists, and enthusiasts to use their models. This API was employed to fine-tune the GPT 3.5 model for one, two, and three epochs, as well as prompt the standard and fine-tuned models listed below:

- gpt-3.5-turbo-1106 (GPT 3.5), trained on data up to September 2021;
- gpt-3.5-turbo-1106, fine-tuned for 1 epoch (GPT 3.5, 1 epoch);
- gpt-3.5-turbo-1106, fine-tuned for 2 epochs (GPT 3.5, 2 epochs);
- gpt-3.5-turbo-1106, fine-tuned for 3 epochs (GPT 3.5, 3 epochs);
- gpt-4-0125-preview (GPT 4), trained on data up to December 2023;
- gpt-4o (GPT 4o), trained on data up to October 2023.

The temperature of the above models can be altered to change the creativity of model outputs. The temperature ranges between zero and unity, and forces a given model’s output to, respectively, be either deterministic or random. Throughout the remainder of this chapter, model temperatures were equal to either 0.1, 0.4, 0.7 or 1

³<https://numpy.org/doc/stable/reference/random/generated/numpy.random.seed.html> (Accessed on: 20.09.2024)

⁴<https://platform.openai.com/docs/models> (Accessed on : 17.10.2024)

because this study assessed the change in performance when the creativity of model generations was altered.

In the next subsection, I describe the process by which prompts were generated and input into the models, using a call to OpenAI's API.

4.2.3.1 Prompt Engineering

The models introduced in the previous subsection were prompted to generate textual outputs, which are referred to as completions or generations from now on. One of the advantages of using large language models over conventional machine learning resides in their ability to perform zero-shot and few-shot learning. Zero-shot learning allows large language models to generate text without any prior examples, whereas with few-shot learning, models are given a handful of examples to learn from before generating a response [245, 119, 153]. Throughout the remainder of this chapter, both zero-shot and few-shot learning are referred to as k -shot learning, where k is the number of examples given to a model within a prompt.

Prompt Structure. The structure of each prompt was the same. Every prompt contained an introduction and a number k of training examples before the model was tasked with generating a given number of completions. The introduction provided each model with: a description of its role (i.e., an AI assistant); the task at hand (i.e., generating a hypothesis using the available information); a description of a hypothesis in accordance with Pherson and Heuer Jr.'s definition, as presented in Chapter 1; and a request for the model to generate a hypothesis. The introduction employed in each prompt is displayed below:

'You are an Artificial Intelligence assistant for an intelligence analyst who is analysing the conflict between Russia and Ukraine. Your job is to generate a hypothesis using a set of information that you have received. In this context, a hypothesis is a sentence, it is testable and falsifiable, it is based on the information you received, and it predicts anticipated results clearly. Please generate one hypothesis based on the inputted information.'

After the introduction, a number k of training examples were added to the prompt. Two different types of example were added to prompts, namely: unbiased examples; and biased examples. Starting with unbiased examples, each model was provided with the same set of k randomly chosen items of information that supported k hypotheses. The process by which those examples were sampled is made clear in Algorithm 4.1,

and an example of an unbiased one-shot learning prompt produced by that algorithm is presented in Appendix A.2.1.

Algorithm 4.1: Unbiased k -shot learning.

Input : A dataset D , of length M , containing hypotheses H (labels) supported by information I (features);
A number of examples k to train the model with;
A temperature T ;
An API to OpenAI's GPT Model.

Output: An array of $M \times N$ models completions.

```

1  $Seed \leftarrow 0$            // Specify the random seed in order to ensure the same training
   examples are used for all target hypotheses
2  $Completions \leftarrow \text{array}(Completions, M \times N)$  // Create an empty array to store the  $M \times N$ 
   model completions
3 for  $i = 1$  to  $M$  do
4    $Prompt \leftarrow \text{Introduction}$            // Append the introduction to the prompt
5    $\tilde{D} = \text{remove}(D, i)$  // Remove the  $i$ -th hypothesis  $H_i$  and its supporting item of
   information  $I_i$  from the training dataset
6   if  $k > 0$  then
7     for  $j = 1$  to  $k$  do
8        $(\tilde{I}, \tilde{H}) \sim \tilde{D}$  // Sample the information  $\tilde{I}$  supporting a hypothesis  $\tilde{H}$ 
       without replacement from the training dataset
9        $Prompt \leftarrow \text{Information} : \tilde{I}$ 
10       $Prompt \leftarrow \text{Hypothesis} : \tilde{H}$ 
11      /* Append the sampled information  $\tilde{I}$  and hypothesis  $\tilde{H}$  to the
       prompt */
12    $Prompt \leftarrow \text{Information} : I_i \in D$  // Append the target information to the prompt
13    $Prompt \leftarrow \text{Hypothesis} :$  // Leave intentionally blank for the model to
   complete
14    $Completions[i] \leftarrow \text{Model}(Prompt, N, T)$  // Task the model to generate  $N$  hypotheses
   using the prompt and temperature  $T$ , and append the result
15 return  $Completions$ 
```

The second method of training example selection employed a biased approach to reflect the way we, as humans, learn: it is far easier to learn how to do something when provided with a similar example. This notion should be no different for large language models that are required to learn how to generate plausible hypotheses within the context of this study, and chapter. To identify the best training examples, each information string was encoded in a fixed-size, sentence-level semantic vector embedding – using Sentence Bidirectional Encoder Representations from Transformers (SBERT) [184], which is discussed in more detail in Section 4.2.4.2 – and the cosine similarity between

the target information, which supported the target hypothesis, and all other information strings was computed. The k most semantically similar information strings, at the sentence-level, were then included in the prompt as training examples. Algorithm 4.2 formally presents the procedure for biased training example selection. The outputs from the algorithm, for the same target information and hypothesis as the unbiased approach, where k also equalled unity, is displayed in Appendix A.2.2.

Algorithm 4.2: Biased k -shot learning using SBERT [184].

Input : A dataset D , of length M , containing hypotheses H (labels) supported by information I (features);
A number of examples k to train the model with;
A temperature T ;
An API to OpenAI's GPT Model.

Output: An array of $M \times N$ models completions.

```

1  $E = \text{embed}(I)$  // Calculate SBERT embeddings of the information
2  $\text{Completions} \leftarrow \text{array}(\text{Completions}, M \times N)$  // Create an empty array to store the  $M \times N$ 
   model completions
3 for  $i = 1$  to  $M$  do
4    $\text{Prompt} \leftarrow \text{Introduction}$  // Append the introduction to the prompt
5    $S \leftarrow \text{array}(S, |I| - 1)$  // Create an empty array, of length  $|I| - 1$ , to store
   similarity scores and indices
6   if  $k > 0$  then
7     for  $j = 1$  to  $M$  do
8       if  $i \neq j$  then
9          $S[j] \leftarrow [\cos\_sim(E_i, E_j), j]$  // Compute the cosine similarity between
           the  $i$ -th and  $j$ -th information embedding, if and only if the
            $j$ -th embedding is not the target information, and append
           the result and the index  $j$  to the similarity array  $S$ 
10       $\tilde{S} \leftarrow \text{top\_sim\_indices}(S, k)$  // Save the indices of the  $k$  most similar
           embeddings
11      forall  $j \in \tilde{S}$  do
12         $\text{Prompt} \leftarrow \text{Information} : I_j$ 
13         $\text{Prompt} \leftarrow \text{Hypothesis} : H_j$ 
14        /* Append the  $j$ -th most similar item of information  $I_j$ 
           supporting the hypothesis  $H_j$  to the prompt */
15       $\text{Prompt} \leftarrow \text{Information} : I_i$  // Append the target information to the prompt
16       $\text{Prompt} \leftarrow \text{Hypothesis} :$  // Leave intentionally blank for the model to
           complete
17       $\text{Completions}[i] \leftarrow \text{Model}(\text{Prompt}, N, T)$  // Generate  $N$  hypotheses from the model
           using the prompt and temperature  $T$ , and append the result
18 return  $\text{Completions}$ 

```

After adding k training examples, the target information was added to the prompt, as shown in Lines 12 and 15, and the model was tasked with generating ten hypotheses,

in Lines 13 and 16, of Algorithms 4.1 and 4.2, respectively.

4.2.4 Evaluating Model Completions

In the following subsections, I describe the metrics employed in the evaluation of completions made by each model. The bespoke application for the large language models proposed in this chapter meant there was not a wealth of data and techniques to evaluate the completions output by each model. Generations from the models were assessed on three criteria, specifically: the mean word-level semantic similarity between target hypotheses and model completions; the average sentence-level semantic similarity between target hypotheses and generations; and the mean probability of hallucination in a model's outputs.

4.2.4.1 Word-Level Semantic Similarity

The word-level similarity between model completions and the target hypothesis was computed using BERTScore [250], which is a task agnostic language generation metric for a set of reference and prediction strings. The process by which BERTScore computed the word-level semantic similarity goes as follows:

1. The words in each reference and prediction string are embedded using pre-trained contextual word embeddings from transformers, such as Bidirectional Encoder Representations from Transformers (BERT) [65] or the Robustly Optimised BERT Pre-training Approach, more formally known as RoBERTa [138], to name a couple. In this study, BERTScore employed RoBERTa with the hash code: `roberta-large_L17_no-idf_version=0.3.12(hug.trans=4.37.2)-rescaled`.
2. The pairwise cosine similarity between all the words within the prediction and reference embeddings are computed.
3. Each predicted word embedding is then matched to the most similar reference embedding, and vice versa, in order to compute precision, recall, and f1 scores.
4. The importance of rare words can then be incorporated into the aforementioned scores using Inverse Document Frequency [205]. However, this task is optional and was omitted from the BERTScores presented in this chapter.
5. Finally, the precision, recall, and f1 scores are linearly rescaled.

For the purpose of obtaining results for this chapter, the predictions were the N model completions and the reference was the target hypothesis in the dataset.

4.2.4.2 Sentence-Level Semantic Similarity

While BERTScore is capable of assessing the average similarity of word tokens across a set of prediction and reference strings, it does not allow for semantic comparison at the sentence-level. By combining BERT [65] and the Siamese network [197], SBERT [184] was employed to quantify the semantic sentence similarity between the set of N model completions and the target hypothesis. The procedure for computing sentence embeddings using SBERT goes as follows. Reference and prediction strings are input into BERT or RoBERTa, the outputs from either of those models are put through a pooling operation to create fixed-sized sentence embeddings, and the Siamese network is used to update weights so that the sentence embeddings are semantically meaningful such that they can be compared using the cosine similarity. This study employed a pre-trained sentence transformer, which used the aforementioned process, with the hash code: `all-MiniLM-L6-v2`⁵. The resulting cosine similarity scores between all target hypotheses within the dataset and the N model completions were employed in order to conduct a count that measured the number of other hypotheses in the dataset that were closer to the model completions than the target hypothesis. The process by which this count was computed is described in Algorithm 4.3.

4.2.4.3 Hallucination Detection

Determining whether large language models are capable of producing a diverse set of plausible, MECE hypotheses is important. To generate such hypotheses, the temperatures of models was increased, as discussed at the start of Section 4.2.3, so that model completions were creative. However, increasing the diversity of model generations can also cause hallucinations. Thus, understanding whether models were hallucinating was vital, especially if such models are to be employed within intelligence settings. For instance, it would be quite unhelpful and misleading to an analyst if a model were to generate an implausible hypothesis because it contained a hallucination, and the intelligence community may not want to employ such models in practice as a result.

To assess whether model completions contained hallucinations, I turned to Self-CheckGPT with Natural Language Inference (NLI) [142], which combined another large language model (DeBERTa-v3-large [100]) fine-tuned for textual entailment to

⁵The same hash code was used for identifying and including the k most similar, biased training examples within a given prompt in Section 4.2.3.1.

Algorithm 4.3: Compute the average number of hypotheses that were closer, at the sentence-level, to the N model completions than the target hypothesis.

Input : An indexed list H containing M hypotheses from the dataset D ;
 An indexed list of hypotheses H_N , of size N , generated by the model.

Output: A list of counts \mathbb{C} representing the number of hypotheses that were closer to the N model completions than the target hypothesis, for all hypotheses in D .

```

1  $\tilde{H} \leftarrow [H \ H_N]$  // Concatenate the list of hypotheses  $H$  from  $D$  and the list
   hypotheses  $H_N$  generated by the model
2  $E \leftarrow \text{embed}(\tilde{H})$  // Compute the sentence-level embeddings using SBERT [184]
3  $\mathbb{C} \leftarrow \text{zeros}(\mathbb{C}, M)$  // Create an array of zeros to store the closeness counts for
   each of the target hypotheses in  $D$ 
4 for  $t = 1$  to  $M \in H$  do
5    $\text{Closeness} \leftarrow \text{array}(\text{Closeness}, M - 1)$  // Create an array of zeros to store the
   cosine similarity scores of all the hypotheses within  $D$ , except the
   target hypothesis
6    $\text{Target\_Sim} = 0$  // Set the average cosine similarity between the target
   hypothesis and the  $N$  model completions to zero
7   for  $i = 1$  to  $|E| - N$  do
8     /* Iterate through the indices of embeddings for hypotheses in the
       list  $H$  */
9     if  $i \neq t$  then
10       $\text{Other} \leftarrow \text{zeros}(\text{Other}, N)$  // Create an array of zeros to store the
       cosine similarity scores between  $N$  model completions and the
        $i$ -th hypothesis embedding  $E_i$ .
11      for  $j = |E| - N$  to  $|E|$  do
12        /* Iterate through the indices of embeddings for hypotheses
          output by the model */
13         $\text{Other}[j] \leftarrow \text{cos\_sim}(E_i, E_j)$  // Compute the cosine similarity between
          the  $j$ -th model generation and the  $i$ -th hypothesis in  $D$ ,
          and append the result
14       $\text{Closeness}[i] \leftarrow \text{mean}(\text{Other})$  // Compute the average cosine similarity
          and append the result to the closeness array
15      else if  $i = t$  then
16         $\text{Target} \leftarrow \text{array}(\text{Target}, N)$ 
17        for  $j = |E| - N$  to  $|E|$  do
18           $\text{Target} \leftarrow \text{cos\_sim}(E_i, E_j)$  // Compute the cosine similarity between
          the target and the  $N$  model completions
19         $\text{Target\_Sim} \leftarrow \text{mean}(\text{Target})$  // Compute the average and save the result
20       $\mathbb{C}_i = 0$  // Set the count of the number of other hypotheses that were
       closer to the  $N$  model completions than the target to zero
21      for  $\text{mean}(\text{Other}) \in \text{Closeness}$  do
22        if  $\text{mean}(\text{Other}) > \text{Target\_Sim}$  then
23           $\mathbb{C}_i = \mathbb{C}_i + 1$  // If the mean cosine similarity between the model
          completions and another hypothesis in  $D$  is greater than the
          target's cosine similarity, add unity to the count
24       $\mathbb{C}[t] \leftarrow \mathbb{C}_i$  // Append the counts for the  $i$ -th hypothesis
25 return  $\mathbb{C}$ 

```

predict the probability of a contradiction between a reference string and a set of predictions. SelfCheckGPT is a robust, zero-resource, and black-box approach to hallucination detection. The procedure normally identifies contradictions between a completion, generated using a more-deterministic temperature, and a set of N generations that used a higher and more creative temperature. However, for the purposes of this study, the process was adjusted and the set of target hypotheses were employed as the *deterministic* samples. The adapted method for hallucination detection using SelfCheckGPT for one target hypothesis went as follows:

1. Identify and store the target hypothesis t_i ;
2. Prompt the model and sample N hypotheses S^n , where $n \in \{1, \dots, N\}$, using a temperature of either 0.1, 0.4, 0.7, or 1;
3. For each of the hypotheses S^n sampled from the model, compute the probability of a contradiction between the target hypothesis t_i and S^n , using the logits associated with entailment and contradiction output by the large language model, DeBERTa-v3-large, such that

$$P(\text{contradict}|t_i, S^n) = \frac{\exp(z_c)}{\exp(z_c) + \exp(z_e)} \quad (4.1)$$

where z_e and z_c are the logits of *entailment* and *contradiction*, respectively. Probabilities closer to unity implied that a contradiction was more likely, and vice versa for entailment.

4. The hallucination score was computed as the average of the probability of a contradiction between the target hypothesis and N model completions, where

$$S_{NLI}(t_i) = \frac{1}{N} \sum_{n=1}^N P(\text{contradict}|t_i, S^n). \quad (4.2)$$

4.3 Results

In this section, results from experiments which employed the models listed in Section 4.2.3 and the test dataset in Section 4.2.2.2 are presented. The test dataset contained a total of 103 samples. Each model was tasked with the generation of ten hypotheses, where $N = 10$, using the same prompts. The reason for this is simple. Imagine that two ASAs were provided with the same items of information and both were asked to generate three hypotheses. While those analysts might generate hypotheses that were semantically similar at the sentence-level, it is unlikely that they would generate

exactly the same hypotheses. Thus, tasking a large language model with the generation one hypothesis and expecting it to be the same as the target hypothesis is unreasonable. So, each large language model was tasked with generating ten hypotheses under the assumption that as the models made more completions, there was a higher chance that at least one of the generations would be relatively close to the target.

4.3.1 Word-Level Similarity

The mean BERTScore f1-scores, plus or minus the Standard Deviation (*SD*), are presented in Table 4.1. Looking to both the model and prompt-type averages, generally speaking, as the temperature was increased, the f1-score decreased; thus, increased model temperatures came at the expense of word-level similarity. The highest model and prompt-type average was observed for the GPT 3.5, 2 epochs model and biased ten-shot learning prompts, respectively, where both had a temperature of 0.1. The highest average f1-score was produced by the GPT 3.5, 3 epochs model which used a temperature of, again, 0.1 and five-shot biased learning. Moreover, as models were provided with more training examples, where k was greater than or equal to one, the f1-score increased, which shows that models were able to learn terminology, at the word-level, that was similar to what an analyst might say. Finally, prompts that contained biased training examples obtained higher average f1-scores when compared to their unbiased counterparts, which confirmed that biasing training examples improved the language models' ability to generate words that were similar to the intended output.

4.3.2 Sentence-Level Similarity

The average number of hypotheses, plus or minus the Standard Error (*SE*), in the test dataset which were semantically closer, at the sentence-level, to the ten model completions than the target hypotheses are presented in Table 4.2. Zero-shot learning considerably outperformed all other prompt-types across all models and temperatures, where the GPT 3.5, 2 epochs model, prompted using zero-shot learning and a temperature of 0.4, had the lowest average. The fine-tuned models were expected to, and did, outperform all other models with regards to the closeness of their generations to target hypotheses, as per Table 4.2. While this result is interesting, it was anticipated because all fine-tuned models were trained and validated on 75% and 15%, respectively, of the dataset that contained information and hypotheses. However, the standard GPT 3.5 model produced completions that were closer to target hypotheses than GPT 4 and 4o's outputs, which was unexpected. Both GPT 4 and 4o were trained by Ope-

nAI using data which may have contained information about the ongoing conflagration between Russia and Ukraine, whereas the standard GPT-3.5 model was not. Furthermore, biasing the k -shot training examples, using Algorithm 4.2, did seem to increase the closeness of model completions to target hypotheses, on average. However, this result is not so significant because the closeness results produced by zero-shot learning prompts were much better than all other prompt-types.

4.3.3 Model Hallucinations

The mean probability of a contradiction between a target hypothesis t_i and ten model completions S^n , plus or minus the SD , for all the hypotheses in the test dataset are presented in Table 4.3. The lowest prompt-type and model means were seen in zero-shot learning prompts and the GPT-4 model, respectively. However, the completions from all models probably possessed hallucinations because the average probability of a hallucination was greater than 50% for all models, prompt-types, and temperatures. The prompt-type and model averages with the lowest probability of a hallucination were noted in models which employed a temperature of unity. This result is interesting because one would expect completions generated using a lower temperature to contain fewer hallucinations. However, this was not the case. Upon observation of the completions from models that employed a temperature of unity and 0.1, the reason for this becomes slightly clearer. Completions generated using a temperature of 0.1 tended to be almost the same, if not identical. Whereas, all the generations that used a temperature of unity were different. This meant that if a model with a lower temperature generated a hypothesis that contained hallucinations, then all the other completions highly likely contained the same hallucination also. On the contrary, when a model with a higher temperature hallucinated, that model would almost never generate the same hypothesis in the following completions; thus, providing those models with the opportunity to produce novel hypotheses which did not contain hallucinations. An example of this is provided in Appendix A.3 for completions that were generated using a temperature of 0.1 and unity. The examples used the unbiased and biased one-shot learning prompts with the same target hypothesis, respectively, displayed in Appendices A.2.1 and A.2.2, which I described earlier in Section 4.2.3.1.

Table 4.1: Average BERTScore f1-score ($\pm SD$) on the test dataset, where U and B denote the results from prompts that, respectively, contained unbiased and biased training examples.

Models	Temperature	Zero-Shot	One-Shot		Five-Shot		Ten-Shot		Model Average
			U	B	U	B	U	B	
GPT 3.5	0.1	86.6 \pm 1.3	86.7 \pm 1.3	86.8 \pm 1.3	86.9 \pm 1.3	86.9 \pm 1.3	86.9 \pm 1.4	87.0\pm1.6	86.8 \pm 1.4
	0.4	86.5 \pm 1.3	86.8 \pm 1.4	86.7 \pm 1.3	86.9 \pm 1.3	86.9 \pm 1.4	86.9 \pm 1.4	86.9 \pm 1.4	86.8 \pm 1.4
	0.7	86.5 \pm 1.3	86.7 \pm 1.4	86.7 \pm 1.4	86.8 \pm 1.3	86.9 \pm 1.4	86.8 \pm 1.4	86.9 \pm 1.4	86.8 \pm 1.4
	1	86.4 \pm 1.3	86.7 \pm 1.4	86.6 \pm 1.3	86.8 \pm 1.3	86.8 \pm 1.4	86.8 \pm 1.4	86.9 \pm 1.5	86.7 \pm 1.4
GPT 3.5, 1 epoch	0.1	87.6 \pm 1.9	87.6 \pm 1.7	88.6 \pm 3.1	88.3 \pm 2.1	88.7 \pm 2.5	88.7 \pm 2.1	89.3\pm2.1	88.4 \pm 2.3
	0.4	87.5 \pm 1.9	87.7 \pm 1.8	88.2 \pm 2.8	88.0 \pm 2.0	88.4 \pm 2.5	88.2 \pm 2.0	88.8 \pm 2.3	88.1 \pm 2.3
	0.7	87.2 \pm 1.8	87.2 \pm 1.7	87.7 \pm 1.7	87.4 \pm 1.7	87.8 \pm 2.4	87.7 \pm 1.9	88.1 \pm 2.4	87.6 \pm 2.1
	1	86.4 \pm 1.7	86.1 \pm 1.5	86.6 \pm 2.3	86.3 \pm 1.7	86.7 \pm 2.3	86.5 \pm 1.7	87.0 \pm 2.5	86.5 \pm 2.0
GPT 3.5, 2 epochs	0.1	87.4 \pm 1.8	88.4 \pm 2.2	89.4\pm3.5	89.2 \pm 2.1	89.4\pm2.7	88.9 \pm 2.0	89.3 \pm 2.5	88.9\pm2.6
	0.4	87.2 \pm 1.7	88.1 \pm 2.1	88.8 \pm 3.0	88.7 \pm 2.0	89.0 \pm 2.5	88.5 \pm 2.1	88.9 \pm 2.4	88.5 \pm 2.4
	0.7	87.0 \pm 1.6	87.5 \pm 1.9	88.2 \pm 2.7	87.9 \pm 1.9	88.3 \pm 2.4	87.8 \pm 1.9	88.2 \pm 2.4	87.9 \pm 2.2
	1	86.2 \pm 1.4	86.5 \pm 1.8	86.9 \pm 2.2	86.8 \pm 1.8	87.1 \pm 2.2	86.8 \pm 1.7	87.0 \pm 2.3	86.8 \pm 1.9
GPT 3.5, 3 epochs	0.1	87.1 \pm 1.8	87.9 \pm 2.2	88.4 \pm 2.7	89.2 \pm 2.0	89.5\pm2.6	89.1 \pm 2.2	89.3 \pm 2.5	88.7 \pm 2.4
	0.4	87.1 \pm 1.7	87.6 \pm 2.1	88.1 \pm 2.6	88.9 \pm 2.0	89.0 \pm 2.3	88.9 \pm 2.0	89.0 \pm 2.3	88.4 \pm 2.3
	0.7	86.7 \pm 1.6	87.2 \pm 1.9	87.6 \pm 2.3	88.1 \pm 2.1	88.3 \pm 2.3	88.1 \pm 2.0	88.4 \pm 2.3	87.8 \pm 2.2
	1	86.2 \pm 1.5	86.4 \pm 1.8	86.7 \pm 2.2	86.9 \pm 2.0	87.0 \pm 2.1	86.9 \pm 1.9	87.2 \pm 2.2	86.8 \pm 2.0
GPT 4	0.1	86.4 \pm 1.5	86.9 \pm 1.4	87.0 \pm 1.5	87.0 \pm 1.7	87.1 \pm 1.7	87.0 \pm 1.6	87.3\pm1.8	86.9 \pm 1.6
	0.4	86.4 \pm 1.5	86.9 \pm 1.5	87.0 \pm 1.5	87.0 \pm 1.6	87.3\pm1.7	87.1 \pm 1.7	87.1 \pm 1.6	87.0 \pm 1.6
	0.7	86.4 \pm 1.5	86.9 \pm 1.5	86.9 \pm 1.6	86.9 \pm 1.5	87.1 \pm 1.6	87.0 \pm 1.6	87.1 \pm 1.6	86.9 \pm 1.6
	1	86.4 \pm 1.4	86.9 \pm 1.6	86.9 \pm 1.5	86.8 \pm 1.6	87.0 \pm 1.4	86.9 \pm 1.5	87.2 \pm 1.7	86.9 \pm 1.6
GPT 4o	0.1	86.2 \pm 1.3	86.8 \pm 1.5	86.6 \pm 1.5	88.7 \pm 1.5	88.5 \pm 1.6	88.9 \pm 1.7	89.0\pm1.6	87.8 \pm 1.9
	0.4	86.1 \pm 1.3	86.9 \pm 1.5	86.8 \pm 1.6	88.7 \pm 1.5	88.5 \pm 1.7	88.9 \pm 1.7	88.8 \pm 1.7	87.8 \pm 1.9
	0.7	86.2 \pm 1.3	87.5 \pm 1.8	87.6 \pm 1.7	88.7 \pm 1.7	88.6 \pm 1.8	88.9 \pm 1.6	88.7 \pm 1.7	88.0 \pm 1.9
	1	86.1 \pm 1.3	87.3 \pm 1.6	87.3 \pm 1.7	88.5 \pm 1.6	88.4 \pm 1.8	88.7 \pm 1.7	88.5 \pm 1.7	87.8 \pm 1.9
Prompt-type Average	0.1	86.9 \pm 1.7	87.4 \pm 1.9	87.8 \pm 2.7	88.2 \pm 2.0	88.4 \pm 2.4	88.2 \pm 2.1	88.5\pm2.3	
	0.4	86.8 \pm 1.7	87.3 \pm 1.8	87.6 \pm 2.4	88.0 \pm 1.9	88.2 \pm 2.2	88.1 \pm 2.0	88.3 \pm 2.2	
	0.7	86.7 \pm 1.6	87.2 \pm 1.7	87.4 \pm 2.2	87.7 \pm 1.8	87.8 \pm 2.1	87.7 \pm 1.9	87.9 \pm 2.1	
	1	86.3 \pm 1.4	86.7 \pm 1.7	86.8 \pm 1.9	87.0 \pm 1.8	87.2 \pm 2.0	87.1 \pm 1.8	87.3 \pm 2.1	

Table 4.2: Average number of hypotheses in the test dataset closer to model completions than the target hypothesis ($\pm SE$), where U and B denote the results from prompts that, respectively, contained unbiased and biased training examples.

Models	Temperature	Zero-Shot	One-Shot		Five-Shot		Ten-Shot		Model Average
			U	B	U	B	U	B	
GPT 3.5	0.1	4.3 ± 0.7	20.8 ± 2.6	18.7 ± 2.5	18.3 ± 2.5	17.1 ± 2.4	17.2 ± 2.5	16.9 ± 2.4	16.2 ± 9.1
	0.4	3.7 ± 0.7	20.0 ± 2.6	17.4 ± 2.5	19.1 ± 2.6	16.9 ± 2.4	17.6 ± 2.5	16.5 ± 2.4	15.9 ± 9.1
	0.7	4.4 ± 0.7	19.4 ± 2.6	17.6 ± 2.6	19.2 ± 2.6	16.3 ± 2.5	18.0 ± 2.5	16.1 ± 2.4	15.9 ± 9.2
	1	4.7 ± 0.7	19.1 ± 2.6	17.7 ± 2.5	18.9 ± 2.6	16.0 ± 2.4	17.5 ± 2.5	16.2 ± 2.5	15.7 ± 9.2
GPT 3.5, 1 epoch	0.1	5.5 ± 0.7	17.9 ± 2.6	19.7 ± 2.8	19.5 ± 2.7	19.9 ± 2.8	19.1 ± 2.7	21.1 ± 2.7	17.5 ± 9.9
	0.4	4.3 ± 0.7	16.8 ± 2.6	17.7 ± 2.7	16.9 ± 2.6	18.1 ± 2.7	17.2 ± 2.7	17.4 ± 2.6	15.5 ± 9.6
	0.7	3.4 ± 0.6	17.7 ± 2.7	17.3 ± 2.7	16.1 ± 2.6	17.7 ± 2.7	17.2 ± 2.6	18.1 ± 2.6	15.3 ± 9.6
	1	3.8 ± 0.6	19.5 ± 2.8	17.3 ± 2.6	18.6 ± 2.6	18.1 ± 2.6	20.6 ± 2.8	19.1 ± 2.7	16.7 ± 9.8
GPT 3.5, 2 epochs	0.1	3.4 ± 0.6	19.1 ± 2.7	19.0 ± 2.7	19.9 ± 2.9	20.1 ± 2.8	21.0 ± 2.9	18.9 ± 2.6	17.3 ± 10.1
	0.4	3.3 ± 0.6	17.2 ± 2.6	16.6 ± 2.6	17.5 ± 2.6	16.7 ± 2.6	17.5 ± 2.7	17.5 ± 2.6	15.2 ± 9.6
	0.7	3.4 ± 0.6	16.6 ± 2.6	15.9 ± 2.6	16.5 ± 2.7	17.3 ± 2.6	16.8 ± 2.6	17.0 ± 2.6	14.8 ± 9.5
	1	4.0 ± 0.6	16.6 ± 2.5	16.0 ± 2.7	17.7 ± 2.7	16.0 ± 2.5	19.3 ± 2.7	18.1 ± 2.7	15.4 ± 9.5
GPT 3.5, 3 epochs	0.1	3.9 ± 0.7	18.3 ± 2.6	19.2 ± 2.6	19.1 ± 2.7	19.0 ± 2.5	22.7 ± 2.8	22.6 ± 2.8	17.8 ± 9.8
	0.4	3.8 ± 0.7	15.5 ± 2.5	17.1 ± 2.6	17.7 ± 2.7	17.7 ± 2.5	16.9 ± 2.5	19.1 ± 2.5	15.4 ± 9.3
	0.7	4.1 ± 0.7	13.9 ± 2.3	15.3 ± 2.3	16.0 ± 2.5	18.7 ± 2.7	16.0 ± 2.4	19.4 ± 2.6	14.8 ± 9.1
	1	3.9 ± 0.6	16.0 ± 2.4	15.7 ± 2.5	15.5 ± 2.4	15.5 ± 2.4	18.7 ± 2.6	17.9 ± 2.5	14.8 ± 9.0
GPT 4	0.1	4.0 ± 0.6	17.6 ± 2.5	16.4 ± 2.4	17.4 ± 2.4	16.7 ± 2.4	18.1 ± 2.5	15.9 ± 2.4	15.1 ± 8.9
	0.4	4.5 ± 0.7	17.1 ± 2.5	15.7 ± 2.3	17.2 ± 2.5	15.3 ± 2.4	16.7 ± 2.5	15.8 ± 2.4	14.6 ± 8.9
	0.7	5.6 ± 0.9	16.9 ± 2.5	16.5 ± 2.4	16.4 ± 2.5	16.1 ± 2.5	16.6 ± 2.5	16.1 ± 2.5	14.9 ± 9.0
	1	5.4 ± 0.8	16.6 ± 2.5	16.1 ± 2.3	17.6 ± 2.5	16.5 ± 2.6	16.8 ± 2.5	16.4 ± 2.5	15.1 ± 9.0
GPT 4o	0.1	5.0 ± 0.8	18.2 ± 2.4	17.9 ± 2.5	19.3 ± 2.6	16.5 ± 2.4	18.8 ± 2.5	15.7 ± 2.3	15.9 ± 8.9
	0.4	5.1 ± 0.8	17.3 ± 2.4	17.4 ± 2.3	18.2 ± 2.5	16.1 ± 2.3	17.1 ± 2.4	16.0 ± 2.3	15.3 ± 8.7
	0.7	5.1 ± 0.8	16.7 ± 2.4	15.8 ± 2.3	17.8 ± 2.5	16.2 ± 2.4	17.2 ± 2.5	16.1 ± 2.4	15.0 ± 8.8
	1	5.1 ± 0.8	16.6 ± 2.5	17.0 ± 2.4	17.2 ± 2.5	15.8 ± 2.3	16.9 ± 2.5	16.7 ± 2.4	15.0 ± 8.8
Prompt-type Average	0.1	4.4 ± 2.9	18.6 ± 10.7	18.5 ± 10.7	18.9 ± 11.0	18.2 ± 10.6	19.5 ± 11.0	18.5 ± 10.6	
	0.4	4.1 ± 2.8	17.3 ± 10.5	17.0 ± 10.4	17.8 ± 10.7	16.8 ± 10.3	17.2 ± 10.6	17.1 ± 10.3	
	0.7	4.3 ± 3.0	16.9 ± 10.4	16.4 ± 10.3	17.0 ± 10.7	17.0 ± 10.6	17.0 ± 10.4	17.1 ± 10.4	
	1	4.5 ± 2.9	17.4 ± 10.5	16.7 ± 10.4	17.6 ± 10.5	16.3 ± 10.3	18.3 ± 10.9	17.4 ± 10.6	

Table 4.3: Average S_{NLI} ($\pm SD$) score obtained from the test dataset, where U and B denote the results from prompts that, respectively, contained unbiased and biased training examples. Results highlighted in bold were the highest model type average, respectively.

Models	Temperature	Zero-Shot	One-Shot		Five-Shot		Ten-Shot		Model Average
			U	B	U	B	U	B	
GPT 3.5	0.1	0.64 ± 0.34	0.65 ± 0.34	0.64 ± 0.35	0.64 ± 0.35	0.61 ± 0.35	0.61 ± 0.33	0.58 ± 0.35	0.62 ± 0.34
	0.4	0.63 ± 0.33	0.64 ± 0.33	0.65 ± 0.33	0.63 ± 0.33	0.61 ± 0.34	0.60 ± 0.33	0.59 ± 0.33	0.62 ± 0.33
	0.7	0.61 ± 0.33	0.65 ± 0.32	0.65 ± 0.32	0.63 ± 0.32	0.63 ± 0.32	0.61 ± 0.32	0.60 ± 0.31	0.63 ± 0.32
	1	0.59 ± 0.31	0.59 ± 0.31	0.59 ± 0.31	0.58 ± 0.30	0.56 ± 0.31	0.56 ± 0.30	0.57 ± 0.31	0.58 ± 0.31
GPT 3.5, 1 epoch	0.1	0.61 ± 0.30	0.66 ± 0.28	0.64 ± 0.29	0.64 ± 0.29	0.58 ± 0.32	0.60 ± 0.31	0.61 ± 0.31	0.62 ± 0.30
	0.4	0.62 ± 0.28	0.63 ± 0.26	0.63 ± 0.27	0.65 ± 0.26	0.60 ± 0.29	0.59 ± 0.28	0.59 ± 0.28	0.61 ± 0.27
	0.7	0.61 ± 0.25	0.62 ± 0.26	0.64 ± 0.26	0.63 ± 0.24	0.61 ± 0.25	0.59 ± 0.26	0.62 ± 0.27	0.62 ± 0.25
	1	0.57 ± 0.22	0.58 ± 0.21	0.59 ± 0.22	0.61 ± 0.22	0.58 ± 0.22	0.59 ± 0.22	0.60 ± 0.22	0.59 ± 0.22
GPT 3.5, 2 epochs	0.1	0.60 ± 0.30	0.64 ± 0.29	0.62 ± 0.30	0.63 ± 0.28	0.59 ± 0.32	0.62 ± 0.31	0.60 ± 0.31	0.61 ± 0.30
	0.4	0.60 ± 0.26	0.62 ± 0.27	0.62 ± 0.27	0.63 ± 0.26	0.59 ± 0.27	0.60 ± 0.27	0.60 ± 0.27	0.61 ± 0.27
	0.7	0.62 ± 0.25	0.63 ± 0.24	0.65 ± 0.24	0.63 ± 0.24	0.62 ± 0.25	0.61 ± 0.26	0.61 ± 0.25	0.62 ± 0.25
	1	0.54 ± 0.22	0.58 ± 0.23	0.60 ± 0.24	0.61 ± 0.23	0.59 ± 0.22	0.59 ± 0.23	0.58 ± 0.22	0.58 ± 0.23
GPT 3.5, 3 epochs	0.1	0.63 ± 0.28	0.64 ± 0.28	0.65 ± 0.28	0.64 ± 0.29	0.61 ± 0.30	0.62 ± 0.29	0.62 ± 0.29	0.63 ± 0.29
	0.4	0.63 ± 0.26	0.63 ± 0.25	0.63 ± 0.25	0.63 ± 0.25	0.62 ± 0.26	0.62 ± 0.25	0.62 ± 0.25	0.63 ± 0.25
	0.7	0.64 ± 0.25	0.64 ± 0.24	0.65 ± 0.24	0.64 ± 0.24	0.64 ± 0.24	0.62 ± 0.25	0.62 ± 0.24	0.63 ± 0.24
	1	0.54 ± 0.22	0.59 ± 0.22	0.66 ± 0.22	0.66 ± 0.22	0.60 ± 0.22	0.64 ± 0.21	0.63 ± 0.21	0.62 ± 0.22
GPT 4	0.1	0.58 ± 0.35	0.60 ± 0.35	0.62 ± 0.35	0.60 ± 0.34	0.59 ± 0.35	0.60 ± 0.34	0.61 ± 0.33	0.60 ± 0.35
	0.4	0.59 ± 0.34	0.59 ± 0.34	0.61 ± 0.33	0.60 ± 0.33	0.58 ± 0.34	0.59 ± 0.33	0.59 ± 0.32	0.59 ± 0.33
	0.7	0.59 ± 0.33	0.61 ± 0.33	0.61 ± 0.33	0.59 ± 0.33	0.59 ± 0.33	0.59 ± 0.32	0.58 ± 0.32	0.59 ± 0.33
	1	0.57 ± 0.32	0.58 ± 0.33	0.57 ± 0.32	0.54 ± 0.31	0.60 ± 0.31	0.56 ± 0.32	0.56 ± 0.30	0.57 ± 0.31
GPT 4o	0.1	0.62 ± 0.34	0.61 ± 0.34	0.59 ± 0.35	0.58 ± 0.34	0.60 ± 0.34	0.59 ± 0.32	0.54 ± 0.34	0.59 ± 0.34
	0.4	0.62 ± 0.33	0.61 ± 0.32	0.60 ± 0.33	0.57 ± 0.32	0.60 ± 0.33	0.56 ± 0.33	0.56 ± 0.33	0.59 ± 0.33
	0.7	0.62 ± 0.32	0.61 ± 0.30	0.59 ± 0.32	0.57 ± 0.30	0.58 ± 0.31	0.54 ± 0.33	0.56 ± 0.32	0.58 ± 0.31
	1	0.55 ± 0.32	0.61 ± 0.31	0.69 ± 0.29	0.68 ± 0.29	0.62 ± 0.30	0.65 ± 0.31	0.67 ± 0.28	0.64 ± 0.30
Prompt-type Average	0.1	0.61 ± 0.32	0.63 ± 0.32	0.63 ± 0.32	0.62 ± 0.32	0.60 ± 0.33	0.60 ± 0.32	0.59 ± 0.32	0.60 ± 0.32
	0.4	0.62 ± 0.30	0.62 ± 0.30	0.62 ± 0.30	0.62 ± 0.29	0.60 ± 0.31	0.59 ± 0.30	0.59 ± 0.30	0.60 ± 0.30
	0.7	0.62 ± 0.29	0.63 ± 0.28	0.63 ± 0.28	0.61 ± 0.28	0.61 ± 0.28	0.59 ± 0.29	0.60 ± 0.29	0.60 ± 0.29
	1	0.56 ± 0.27	0.59 ± 0.27	0.62 ± 0.26	0.61 ± 0.26	0.59 ± 0.26	0.60 ± 0.26	0.60 ± 0.26	0.60 ± 0.26

4.4 Discussion

The results from experiments that tested the feasibility of using a number of state-of-the-art large language models for hypothesis generation within intelligence settings were inconclusive. While large language models were able to generate hypotheses using a set of information and k -shot learning examples, the findings do not confirm whether they have the potential to, as Snidaro put it in [204], “*revolutionise*” the task of hypothesis generation, for a number of reasons. The examined models were able to learn the relevant terminology employed by intelligence analysts, and increasing the temperature of model outputs increased the range of hypotheses observed. Closeness of model completions was not increased by increasing the number of k -shot learning examples. The closeness results for completions from zero-shot learning performed better than all other prompt-types, which affirms the work of Liu, Neubig, and Andreas in [136], where they posit that language models perform better abductive reasoning when they are provided with inference instructions, instead of few-shot learning examples. Increasing model temperature had no tangible impact on the number of potential model hallucinations as the average probability that a set of model completions contained a hallucination was greater than 50%, for all models, temperatures, and k -shot learning approaches. When model temperatures were at their lowest (i.e., 0.1), model generations were not diverse and many of the strings output by models were the same, which would be, at best, unhelpful to the analyst. The models that were fine-tuned for one, two, and three epochs produced completions closest to target hypotheses at the sentence-level, using zero-shot learning prompts. This suggests that fine-tuned models are better than the most recent, standard, and state-of-the-art models for hypothesis generation. Whether the completions from the GPT 4 and 4o models, fine-tuned for one, two, and three epochs, would be closer to target hypotheses than the outputs from the fine-tuned GPT 3.5 models is an unanswered question left for future work.

The experiments in this chapter demonstrate that large language models can be employed for hypothesis generation within the context of the conflict between Russia and Ukraine. However, the models should be tested in different settings to understand whether they can produce a diverse set of *good* hypotheses for a variety of novel scenarios. Model generations were not examined to see whether they were *good* hypotheses, in accordance with Pherson and Heuer Jr.’s definition [173]. Moreover, this study did not assess whether completions contained the qualitative probabilistic judgements, such as those found in the Probability (or Uncertainty) Yardstick, introduced in Section 2.1.4, which are required to be included within *finished intelligence products*.

As the intelligence community moves towards utilising AI in the future, there is

a growing need to understand whether large language models could be employed for different tasks within the *analysis* stage of the *Intelligence Cycle*. In this chapter, I have shown that hypothesis generation, a fundamental task within the *Cycle*, can be completed by large language models. However, the extent to which these tools produce *good* hypotheses is questionable. There are number of issues with state-of-the-art large language models which must be addressed before they are used operationally for intelligence analyses. Starting with the first and most fundamental problem, large language models are trained on an exorbitant amount of data and they perform well at completing tasks that they previously observed during training. However, hypothesis generation, within intelligence contexts, is juxtaposed to that notion of training. Analysts are required to conduct abductive reasoning, a task which large language models are not proficient at [136], using the available information and evidence, as well as their assumptions, where the amount of data could vary from a lot to a little. The scenario being assessed by the analyst may be novel such that there are no previous examples to help guide an analyst's forecasts. While the standard GPT 3.5 model, which was not trained on any data pertaining to the current conflict between Russia and Ukraine, performed relatively well across all experiments, this does not confirm whether other language models will generalise to other, more nuanced and novel situations that they have not been trained on.

The second issue pertains to attention dilution. The results show that as the models were provided with more training examples, the models' completions and target hypotheses became increasingly less semantically similar, at the sentence-level. For large language models to provide real benefit to the intelligence community in the future, the analyst should be able to provide a given model with as much context as they believe is necessary to generate hypotheses, without impacting model performance.

Third, producing models that can robustly generate diverse hypotheses without hallucinating is a challenge, but a necessary task for intelligence analysis in the future. Although the model temperature did not, roughly speaking, impact the probability of a hallucination, models of the future should focus on decreasing said probability across all model generations to ensure that large language models can be reliably used for MECE hypothesis generation.

Fourth, models were tasked with generating both operational and strategic hypotheses, which may have decreased model performance. Whether these models can differentiate between those two types of hypotheses and generate them, if asked to, should be studied and understood in the future.

Fifth, the effects of the granularity of segmentation on model performance was not assessed in this chapter. In Section 2.3.3.1, the distinction between an EDU and

ADU was made clear. The method for segmentation in this chapter employed an EDU approach. However, the example in Figure 4.2 and brief discussion in Section 4.2.2.1 demonstrates how dislocations can be present within the data such that a hypothesis is either embedded within or split between the propositional content in a clause containing information. As I discussed in Section 4.2.2.1, segmenting the data into ADUs and reconstructing any missing or implicit material, such as resolving anaphoric references and dislocations, might increase language model performance. However, this is a task that is left for future work.

Finally, given the nature of intelligence analysis, much of the information available to an analyst is classified and not openly accessible. Fine-tuning the models employed in this chapter, using both classified and unclassified data, might increase their performance. However, for obvious reasons, this is a task that the intelligence community would need to undertake as future work.

4.5 Summary and Future Work

In this chapter, a study was conducted which measured the feasibility of leveraging large language models for hypothesis generation with intelligence settings. Tweets from the X, formerly Twitter, account of the United Kingdom’s Ministry of Defence, were annotated and instantiated within a dataset that contained hypotheses supported by information. The Tweets contained unclassified intelligence reports. A number of state-of-the-art large language models were fine-tuned and prompted using those data. The results from the feasibility study were inconclusive. While large language models can generate hypotheses using a set of information as context within a prompt, whether or not the hypotheses generated were *good*, in terms of Pherson and Heuer Jr.’s definition [173], is questionable. Prompts that employed a zero-shot learning approach outperformed all others, producing completions that were the closest to the intended target hypothesis. All the results presented contained hallucinations so, again, there was no guarantee that the hypotheses generated by models were plausible. Moreover, the problem of attention dilution was noted when more training examples were included in prompts. Although large language models are indeed very popular and there is much ongoing research to study their potential use-cases, this study shows that the technology, in its current state, cannot be employed for reliable hypothesis generation within intelligence settings. Hypothesis generation requires the analyst to reason abductively, a task language models struggle with.

The application of large language models proposed in this chapter is novel and

quite niche. As such, there are many ideas for future work. First, the curated dataset employed in the fine-tuning and prompting of language models could be partitioned into two datasets, containing information supporting either operational or strategic hypotheses, which would enable us to understand whether model performance increases when the definition of a hypothesis is stricter. Second, as I described in Section 4.2.2.1, hypotheses generated and evaluated at the tactical level should aid in the evaluation of operational hypotheses, and the set of evaluated operational hypotheses should inform the probability of strategic hypotheses too. The relationship between evaluated tactical, operational, and strategic hypotheses could form a tree structure. Understanding and formalising this relationship should help intelligence analysts evaluate hypotheses in the future. Third, the quality of model completions should be assessed to understand whether: the hypotheses generated are *good*, according to Pherson and Heuer Jr.’s definition in Chapter 1; and model generations contain qualitative probability judgements, as per the Probability (or Uncertainty) Yardstick in Section 2.1. Moreover, the precision of those probabilistic judgements should be quantified to understand whether they are in agreement with an analyst’s judgement. Fourth, as language models are not proficient in abductive reasoning, using a set of inference instructions, instead of few-shot learning examples, similar to the proposal discussed by Liu, Neubig, and Andreas in [136], should be considered. Finally, future research could investigate whether large language models can conduct hypothesis generation using analytical techniques, such as *Quadrant Hypothesis Generation* that was described in Section 2.1.5.1, where the goal would be to prompt a given model to identify the two, key driving factors pertaining to a situation, to name one example.

Up until now, this thesis has considered the use of argumentation schemes and large language models to represent and generate hypotheses, respectively. In Part II, two techniques are proposed that focus on providing decision- and reasoning-support for ASAs when conducting hypothesis evaluation tasks.

Part II

Hypothesis Evaluation

In this part, I present two techniques for aiding hypothesis evaluation within the *analysis* stage of the *Intelligence Cycle*. In Chapter 5, I present the first approach which combines probabilistic argumentation frameworks [133] with Bayes' theorem [18] in order to condition an intelligence analyst's probabilistic judgements – i.e., their assessments about the prior probabilities of hypotheses, information, evidence, and assumptions – on a semantically evaluated set of arguments which feature within their analysis. In Chapter 6, I propose a second method which combines Dung's abstract argumentation frameworks [70] with probability and information theory to emulate the task of sensitivity analysis, which is usually employed in step seven of the ACH, as I discussed in Section 2.1.5.2. Recall that the methods in Part I were to be employed after the ASA had concluded their own analysis. The techniques proposed within this part are also to be used after the conclusion of an analysis, in agreement with Zlotnick's suggestion in [251], which was previously discussed in Section 2.4.2. Both the approaches in Chapters 5 and 6 should provide benefit to analysts by, respectively, computationally refining analysts' prior probabilistic judgements in light of a set of semantically evaluated arguments and quantifying diagnostic arguments, acting to ensure that analysts' conclusions are reasonable, rational, and based upon all the available data.

A real-world example, called Operation Fortitude-South, was employed to test quality and benefits of both the algorithms proposed in this part of the thesis. Operation Fortitude was an Allied deception operation in World War Two conducted prior, during, and after the Normandy landings and invasion of Europe, to counter the German offensive. As Donovan puts it

'Fortitude consisted of two primary operations. Fortitude-North was designed to convince the Germans that the Allied invasion in 1944 would come through Norway and Sweden vice France. Fortitude-South was developed to convince Hitler and his staff that the primary invasion site in France would be Pas de Calais and other potential invasion sites, such as Normandy, were only diversions.' [66, p. 5]

The overall objective of such an operation was to force the Axis' attention away from Normandy to enable a successful Allied invasion, while also delaying German reinforcements to the area. In [66, p. 13-18], Donovan provides a comprehensive description of the plans, goals, and events that occurred as result of Operation Fortitude-South.

I now briefly introduce the scenario, as well as the manual argument mining process undertaken to annotate the data, within this introduction to Part II. The annotated data from the Operation Fortitude-South scenario was captured within the AIF [11] representation and then mapped to an ASPIC⁺ argumentation theory [177, 150] and then to

a Dung argument system [70], as discussed in Section 2.3.2.3, in order to enable the evaluation of the proposed arguments.

As discussed in Section 2.4.3, Alan Knox, at the United Kingdom's Dstl, produced the first proposal which studied the Operation Fortitude-South example using both the ACH and argumentation schemes. The work on this is presented on Dstl's *eleatics* website⁶ [63]. All the annotated data pertaining to that scenario was taken from their website and employed in experiments throughout the rest of this part of the thesis. While applying argumentation schemes to the ACH is not a new concept and in [152], Murukannaiah *et al.*'s proposed combination of the techniques has been shown to provide benefit to analysts, as discussed in Section 2.4.3, the combination of the techniques had only been applied to fictitious examples prior to Knox's work on Operation Fortitude-South. Knox again demonstrates that the ACH can be combined with argumentation schemes for real-world scenarios.

The manual annotation of argument data from the Operation Fortitude-South scenario, stored in an AIF argument graph and translated to an ASPIC⁺ argumentation theory and then to a Dung argumentation framework, is one contribution of this thesis. The methodology for the manual argumentation mining process and consequent framework instantiation is documented below. The research presented in this introduction to Part II is an extension of Dstl's work on the same example. A full description of Dstl's contribution and how it relates to the work presented in this thesis has been included in Appendix B.1.

As previously discussed in Section 2.4.3, Zografistou *et al.* show in [253] that argument data stored within the AIF can be converted to an ACH matrix, as long as the annotator chooses which propositions contain hypotheses. The work presented in this part of the thesis demonstrates that the diagnosticity matrix, created during the ACH, can also be mapped to the AIF to enable computational reasoners to evaluate the acceptability of arguments present within an intelligence analysis. The methodology of the ACH process is such that analysts are required to list all the plausible and MECE hypotheses, as well as all the information, evidence, and assumptions. Therefore, all the data necessary to instantiate an argumentation framework is already present within the diagnosticity matrix. Moreover, one could argue that this matrix is inherently argumentative because analysts must decide which row entries provide support for or attack hypotheses, and it is this reasoning that I have elicited from the ACH matrices proposed by Knox on Dstl's *eleatics* website.

The remainder of this introduction to Part II describes the extension to Knox's

⁶<https://Dstl.github.io/eleatics/argumentation/fortitude/> (Accessed on: 10.02.2023)

work, which resulted in an argumentation framework to enable the testing and evaluation of the two techniques proposed within Chapters 5 and 6 of this thesis.

Dstl’s Contribution to the Operation Fortitude-South Example. On Dstl’s *eleatics* website, Knox demonstrates how the ACH can be combined with argumentation schemes using Operation Fortitude-South as an example. An important point to note is that the analysis is conducted from the perspective of the German forces and what the Allied commanders believed the Axis command would have concluded as the most likely invasion site. The Allied nations would have conducted their own analysis about what the Germans may or may not have believed at the time. The point of this analysis was to understand whether the ACH could in fact be employed in an analysis about what others should believe, given the available information, using a deception operation as an example. Knox loosely follows the ACH process, which I described in Section 2.1.5.2, and starts by identifying the set of MECE hypotheses, which were reduced to a set of plausible ones that were found to be:

- *Pas de Calais*: The main invasion will land in Pas de Calais.
- *Normandy*: The main invasion will land in Normandy.

An interesting point to note here is that the above hypotheses are strategic ones. Recall from Section 4.2.2.1, the distinction between strategic, operational, and tactical hypotheses. Now, imagine that the Allies had two different ways to arrive at Normandy. Both of the ways to get to Normandy should sensibly be hypotheses. However, the distinct two routes are operational hypotheses as they operationally describe how the Allies might go about achieving their strategic goal, which was to invade mainland Europe through Normandy.

Returning back to the work completed by Knox, he then lists the significant and relevant information, evidence, and assumptions, which are presented in Appendix B.2. Next, he prepare an initial diagnosticity matrix, presented in Table B.1, containing the two hypotheses and the information, evidence, and assumptions, and populate each cell with consistency scores based upon whether the row entries were inconsistent or otherwise with each hypothesis. An important point to note here is that Knox assumed that each cell within an ACH matrix was an application of one of Walton’s argumentation schemes⁷. Then, he condensed the initial matrix and removed superfluous row entries, resulting in a refined ACH matrix, which is presented in Table 4.4 and B.4. In both

⁷<https://dstl.github.io/eleatics/argumentation/fortitude/schemes.xhtml> (Accessed on: 10.02.2023)

the initial and refined diagnosticity matrices, the most likely hypothesis from the Axis' perspective was deemed to be the *Pas de Calais* hypothesis because it was the least inconsistent with the available row entries. Knox also conducted sensitivity analysis and concluded that ORBAT (see Appendix B.2 for a definition of this item) was the most diagnostic item of information. ORBAT is a row entry which accrued other items and whether ORBAT provided support for or was in conflict with either of the hypotheses was determined by conducting another ACH, which I describe in more detail in Section B.1.3. A more detailed description of Dstl's contribution, how the work presented in this thesis extends the work by Knox at Dstl, and the analysis pertaining to ORBAT is provided in Appendix B.1.

Table 4.4: The refined ACH matrix for Operation Fortitude-South, where “C” and “I” denote row headers that are consistent and inconsistent with the hypotheses (column headers), respectively.

Information, Evidence, and Assumptions	Hypotheses	
	Normandy	Pas de Calais
PORT	I	C
EXPANSION	C	I
STRONGPOINT	C	I
SPIES	I	C
SIGINT	I	C
SHORTEST	I	C
AIRPLAN	I	C
ORBAT	I	C

The Manual Annotation Process. While there is much documented on the Operation Fortitude-South scenario, the only data employed within the analysis presented in this thesis was data collected from Dstl's *eleatics* website. The manual argument mining process, which I discussed in Section 2.3.3.1, was employed in the annotation of the Operation Fortitude-South dataset. All the propositions discovered within the dataset were taken from the *eleatics* website and put through the entirety of the manual argument mining process (i.e., text segmentation, argument / non-argument classification, simple inference and conflict structure, and refined structure instantiation), culminating in a set of arguments, instantiated using Walton's argumentation schemes, for each hypothesis and item of information, evidence, and assumption found within the

dataset. While Knox assumed that each cell within an ACH matrix was an instantiation of one of Walton’s argumentation schemes, this thesis extends the work on Dstl’s *eleatics* website by relaxing this constraint, in order to allow cells within a matrix to be the application of a varying number of sequentially linked arguments that provide support for a hypothesis or its negation. I now discuss how the two, main hypotheses and the row entries in the refined ACH matrix were annotated using *OVA+* [108], which allowed the argument data to be stored in the AIF.

Hypotheses. Similar to the notion first proposed by Toniolo *et al.* in [216] and the scheme proposed within this thesis for the representation of hypotheses in Chapter 3, hypotheses within the analysis of Operation Fortitude-South were represented using Walton’s original argument from cause to effect [236], instead of using the scheme proposed in Chapter 3. However, an interesting avenue for future work might be to employ the scheme in Chapter 3 as an argumentation-based summary of the reasoning presented in the diagnosticity matrix after the conclusion of an ACH in order to demonstrate the scheme’s usefulness. The hypotheses are presented in Figure 4.3. To make the reasoning about hypotheses explicit, the propositions instantiated within arguments about hypotheses were: the hypothesis; the negation of the other hypothesis; and a causal generalisation – for both the *Normandy* and *Pas de Calais* hypotheses. The mutual exclusivity of hypotheses was modelled by the causal generalisation which meant that if one of the hypotheses were found to be true (false), then the other one would be false (true). For example, if the *Normandy* hypothesis was true, then the *Pas de Calais* would be false. Attacks between hypotheses were instantiated as follows. First, recall from Section 2.2.1.1 that WCQ3 in Walton’s argument from cause to effect [236] asks whether there are any other causal generalisations that interfere with the production of an effect. Thus, symmetric attacks were instantiated between the causal generalisations for arguments containing opposing hypotheses with alternate generalisations. Both the causal propositions were instantiated as ordinary premises within the ASPIC+ framework [177, 150], which allows those mutual attacks to be represented also within an abstract argumentation framework, after translating from the AIF to ASPIC+ then to a Dung-style argument system. Symmetric attacks were also identified between each of the hypotheses due to the conflicting outcomes (i.e., the Allied forces were planning to invade either *Normandy* or *Pas de Calais*). Finally, symmetric attacks were also instantiated between hypotheses and their complements because a hypothesis and its negation should not be included in the same extension or labelling found after semantic evaluation.

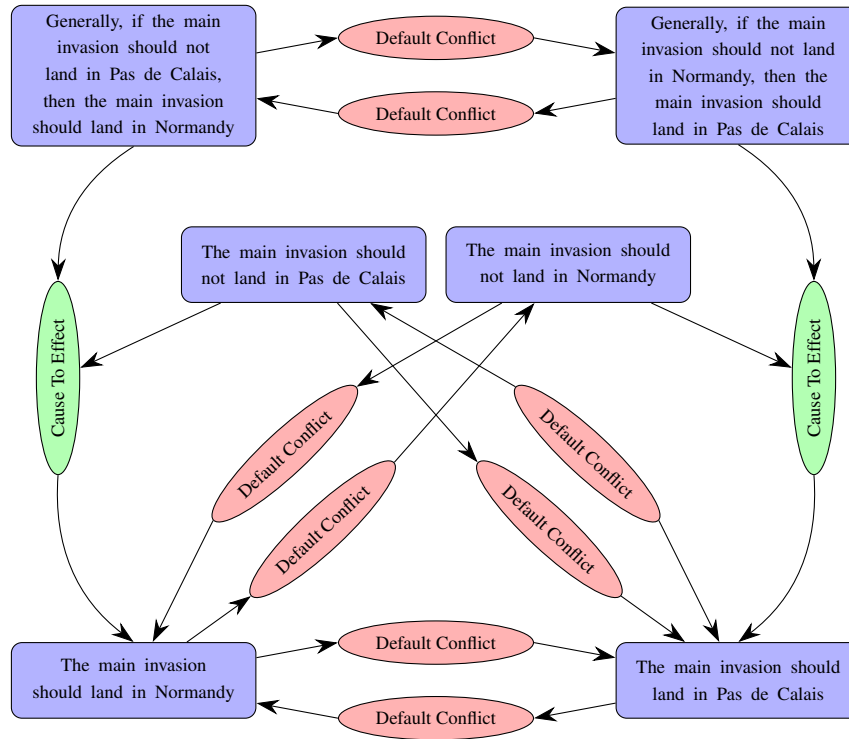


Figure 4.3: The two, main hypotheses from the Operation Fortitude-South example instantiated within Walton's argument from cause to effect [236]. The diagram in this figure was adapted from a screenshot of the analysis in *OVA+* [108].

Information, Evidence, and Assumptions. Each cell within the ACH matrix was assumed to be comprised of a varying number of sequentially linked arguments containing information, evidence, or assumptions that either provided support for a hypothesis or its negation. Row entries that were consistent with a hypothesis in the ACH matrix were instantiated as premises within arguments that culminated in direct support for a hypothesis (i.e., the conclusion of argument for a hypothesis), whereas row entries which were inconsistent with a hypothesis provided support for the hypothesis' negation; thus, indirectly attacking the hypothesis due to the method of symmetric attack instantiation discussed earlier. For example, consider the item of evidence denoted as PORT, in Table B.1, which is inconsistent (i.e., I) and consistent (i.e., C) with the *Normandy* and *Pas de Calais* hypotheses, respectively, because Normandy did not have a deep water port in 1944 but Pas de Calais did. Now look to Figure 4.4. The reasoning underpinning the use of ports and military practice pertaining to invasions

must first be made explicit through the instantiation of Walton’s argument from established rule [238]. The conclusion of the argument from established rule (i.e., *the allied military invasion must use a major port*) was employed as a premise, along with domain knowledge about whether Normandy or Pas de Calais had deep water ports at the time, within Walton’s practical reasoning scheme [241], to conclude that the invasion should land in Pas de Calais, not Normandy. Relaxing the assumption that each cell was the application of one of Walton’s argumentation schemes enabled a stronger and more logical inference from the set of premises to the conclusion for the *Pas de Calais* hypothesis in the example shown in Figure 4.4.

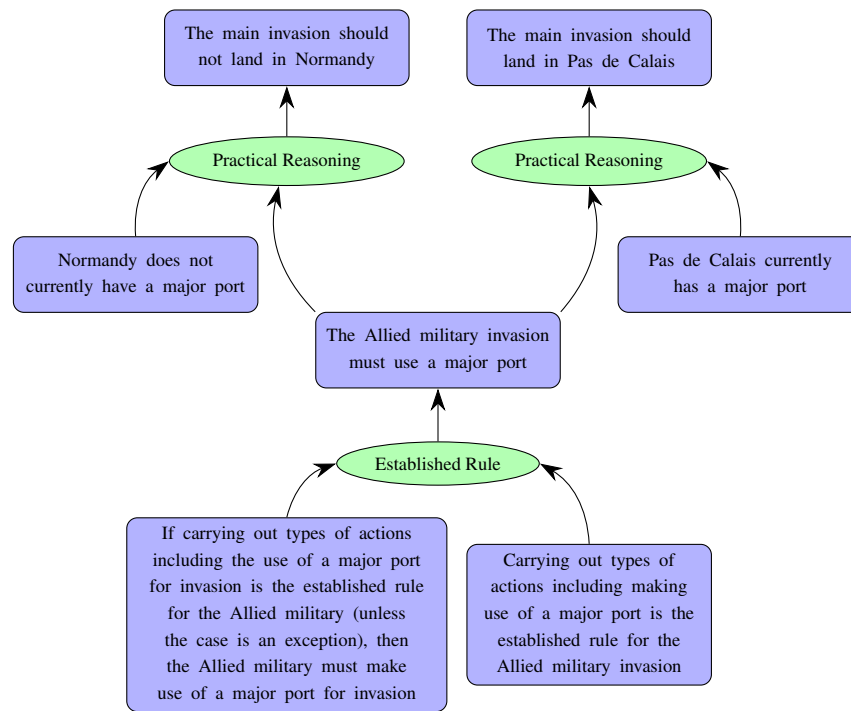


Figure 4.4: An instantiation of the reasoning involved in the evidence denoted PORT using Walton’s argument from established rule [238] and practical reasoning scheme [241]. The diagram in this figure was adapted from a screenshot of the analysis in OVA+ [108].

From the ACH to AIF. While there were two ACH matrices presented by Knox on Dstl’s *eleatics* website, namely the initial⁸ and refined⁹ ACH matrices, only the latter was employed in the testing of the approaches proposed in Chapters 5 and 6. The refined ACH matrix is presented in Table B.4 in Appendix B.1.5. The refined ACH matrix was converted to structured argument data using the same methodology as described earlier for each row entry within the matrix, under the assumption that the reasoning about each cell could be modelled using a varying number of sequential instantiations of Walton’s schemes. The AIF argument graph, annotated using *OVA+*, for the refined ACH matrix is presented in Figure B.1 in Appendix B.3. The AIF graph was then translated to an ASPIC⁺ argumentation theory and then to Dung’s seminal framework for evaluation. The arguments and sub-arguments instantiated using the ASPIC⁺ framework are presented in Appendix B.4. The Dung argumentation graph instantiated, using the translation procedure described in Section 2.3.2.3, is presented in Figure B.2 in Appendix B.5.

⁸<https://dstl.github.io/eleatics/argumentation/fortitude/matrix-at-3.xhtml> (Accessed on: 10.02.2023)

⁹<https://dstl.github.io/eleatics/argumentation/fortitude/matrix-at-4.xhtml> (Accessed on: 10.02.2023)

Chapter 5

On Human-Machine Bayesian Hypothesis Evaluation

In this chapter, I propose a definition for a hypothesis and a definition of information in terms of the ASPIC⁺ framework, which I introduced in Section 2.3.2.1. I then combine those definitions with probabilistic argumentation frameworks [133], Bayes' theorem, and an analyst's prior beliefs about the probability of hypotheses and information – in the form of a human-machine team – in order to refine a human agent's probabilistic judgements using a set of semantically evaluated arguments. I provide a series of abstract examples that draw upon, what is assumed to be, a typical intelligence scenario to show how the proposed algorithms work in practice. I then test the approach using the Operation Fortitude-South argumentation framework, which was mapped from the ACH, to AIF, to an ASPIC⁺ argumentation theory, and then to a Dung argument system, as described in the introduction to Part II. The chapter concludes with a summary of the contribution and some discussion pertaining to avenues for future work.

5.1 Introduction

Intelligence analysis includes many different tasks, such as identifying, processing, and making sense of past and present data in order to provide forecasts on future developments for stakeholders and decision makers. Analysts are tasked with generating and evaluating hypotheses using the information available to them during the *analysis* phase of the *Intelligence Cycle*, as described in Section 2.1.1. The output from their analysis is an intelligence report (or *finished intelligence product*) which usually

contains a set of probabilities pertaining to the likelihood of each hypothesis, as well as any diagnostic information, which is then shared with stakeholders in the *dissemination* stage of the *Cycle*. Although analysts employ a variety of techniques during analysis, evaluative tools, such as the ACH, are not without their deficiencies. For example, as described in Section 2.4.3, in [152], Murukannaiah *et al.* show how the use of the ACH alone does not necessarily mean that analysts are reasoning with multiple items of information to evaluate hypotheses. In fact, the ACH forces analysts to consider the consistency or otherwise of individual items of information due to the matrix which analysts are required to instantiate during evaluation. As I discussed in Section 2.1.5.2, an analyst may decide that certain row entries within the ACH are not applicable to a hypothesis when considered in isolation. However, those same items could conflict with, or provide support for, a hypothesis when they are combined. As such, analysts should possess tools which can reason with all the information, evidence, and assumptions that feature in their analysis.

While the ACH was intended to minimise confirmation bias, the probabilities of the hypotheses reported to stakeholders and decision makers will inevitably contain a human agent's prior beliefs about a particular domain, judgement errors, and cognitive biases. The number of errors and biases included in those probability judgements could vary from analyst to analyst because they come from, for example, different backgrounds and hold different societal views. This means that two analysts evaluating the same set of hypotheses using, say, identical row entries from the ACH may arrive at different probabilistic assessments. The benefits of utilising a computational agent that can assess the acceptability of data points, and condition the probabilities of hypotheses on the acceptability of arguments within their analysis, are two-fold. First, employing a computational agent to conduct the reasoning about hypotheses in the light available data will ensure that analyses are repeatable, consistent, and coherent amongst different analysts. Second, utilising the computational power of an automated reasoner means that the probabilistic judgements made by human agents about hypotheses can include all the information, thus providing a more holistic evaluation.

As I discussed in Section 2.4.2, previous work by Zlotnick in [251] and Edwards *et al.* in [74] argued that Bayes' theorem [18] is capable of revising probabilistic estimates and that the technique should be employed within intelligence analyses, after the *analysis* phase of the *Intelligence Cycle* has concluded. Further support for this argument was provided by both Zlotnick in [252] and Fisk in [79], where Bayes' theorem was employed for the revision of estimates in real-world examples, namely the events leading to the 1941 Pearl Harbour attacks and the border dispute between China and the USSR, respectively.

To this end, this chapter proposes a technique which is able to refine the probabilistic judgements made by analysts using the acceptability of arguments found within the ACH and instantiated within probabilistic argumentation frameworks. The proposed algorithms are not intended to replace an analyst's domain expertise and skill during analyses. The technique presented is to be employed after the analyst has concluded their analysis (step seven of the ACH) and before the dissemination of their findings (step eight of the ACH), acting similar to a quality assurance check conducted in light of the results from semantic evaluation of arguments within the evaluated ACH matrix; thus, providing benefit to the intelligence community.

The contribution made in this chapter can be divided into two parts:

- Pherson and Heuer Jr.'s definition of a hypothesis in [173] is employed to define, for the first time, a hypothesis in terms of structured argumentation, specifically the ASPIC⁺ framework.
- A novel set of algorithms are proposed, which build upon the first contribution, in order to condition a human agent's tentatively-held probabilistic judgements about hypotheses, using the output from semantic evaluation of probabilistic argumentation frameworks [133] and Bayes' theorem [18], in a similar fashion to the proposals made by Zlotnick, Fisk, and Edwards *et al.*, which were mentioned earlier in this section and in Section 2.4.2. The method presented describes how to combine the knowledge of a human analyst with the rationality of a computational agent, in a human-machine team, to evaluate hypotheses. The algorithm should benefit intelligence analysts by providing more holistic estimates of the likelihoods of hypotheses that are based upon the admissibility of the information.

5.2 Preliminaries

In this section, I introduce the relevant prerequisites necessary for understanding the proposed algorithms presented in Section 5.3.7.

5.2.1 Abstract Argumentation Frameworks

Recall the definition of abstract argumentation frameworks from Section 2.3.1. A Dung argument graph is a pair $(\mathcal{A}, \mathcal{R})$ that contains a set of arguments \mathcal{A} and a binary attack relation $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$. We say that a_1 attacks a_2 if and only if $(a_1, a_2) \in \mathcal{R}$ and $a_1, a_2 \in \mathcal{A}$. The semantics of such frameworks are operations which group *admissible*

sets of arguments that possess two characteristics: *conflict-freeness*; and *acceptability*, as respectively stated in Definitions 2.2 and 2.3 in Section 2.3.1.

While the literature characterises the semantics into either extension- or labelling-based semantics, both approaches employ different methods to realise the same end. The algorithms proposed in this chapter employ extension-based methods. Complete semantics were chosen to evaluate the outputs from the proposed technique. Remember that, as per Definition 2.4 in Section 2.3.1, a set of arguments S is a *complete extension* if and only if S is conflict-free and $\forall a_1 \in S, a_1$ is acceptable with respect to S ; thus, making S an admissible set.

In the following subsection, I introduce Li, Oren, and Norman’s probabilistic argumentation framework [133], a new and previously undefined framework within the context of this thesis, because Algorithms 5.1, 5.2, and 5.3, in Section 5.3.7, employed the formalism to compute likelihoods.

5.2.2 Probabilistic Argumentation Frameworks

Probabilistic argumentation frameworks [133] extend Dung’s account of abstract argumentation by assigning each argument and relation a probability of inducement.

Definition 5.1 (*Probabilistic Argumentation Frameworks*). A probabilistic argumentation framework (PrAF) is a tuple

$$PrAF = (A, P_A, R, P_R) \quad (5.1)$$

where (A, R) is a Dung argument system, and P_A and P_R are probability functions defined over, respectively, the set of arguments, such that $\forall a_i \in A, P_A : a_i \rightarrow (0, 1]$, and the set of attack relations, such that $\forall (a_i, a_j) \in R, P_R : (a_i, a_j) \rightarrow (0, 1]$.

Probabilities are computed through random inducement of Dung argumentation frameworks, referred to as *induced* frameworks or graphs from now on. The user-chosen uncertainties of arguments and attacks within the PrAF (i.e., P_A and P_R) describe the probabilities associated with the existence of those arguments and relations, and the instantiation of an *induced* framework is a realised Dung argument system sampled from the aforementioned probability functions.

Definition 5.2 (*Induced Argumentation Frameworks*). For a PrAF (A, P_A, R, P_R) , an *induced* graph $\mathcal{G}^I = (\mathcal{A}^I, \mathcal{R}^I)$ is observed if and only if the following holds:

- $\mathcal{A}^I \subseteq A$;

- $\mathcal{R}^I \subseteq R \cap (\mathcal{A}^I \times \mathcal{A}^I)$;
- $\mathcal{A}^I = \{a_i \in A : P_A(a_i) \geq r_{a_i}\}$;
- $\mathcal{R}^I = \{(a_i \in \mathcal{A}^I, a_j \in \mathcal{A}^I) : P_R((a_i, a_j)) \geq r_{(a_i, a_j)}\}$

where r_{a_i} and $r_{(a_i, a_j)}$ are elements in the unit interval such that $r_{a_i}, r_{(a_i, a_j)} \in (0, 1] \subset \mathbb{R}$.

The pseudo-code for a function which can instantiate an induced framework is presented in Algorithm C.1 in Appendix C.1.

The set of all induced graphs from a probabilistic framework is denoted $I(\text{PrAF})$.

The probability of inducing a Dung-style argument system from a PrAF is computed using the product of the probabilities of the arguments and defeat relations which did and did not feature in an induced graph. Thus, for a set of induced arguments \mathcal{A}^I , we need to identify the set of relations that *might* appear within an induced framework, which is defined as

$$\tilde{R} = R \cap (\mathcal{A}^I \times \mathcal{A}^I). \quad (5.2)$$

It follows that we can calculate the probability of instantiating some induced graph, under some semantics \mathcal{S} , as shown in Equation 5.3, where a and r are, respectively, arguments and relations that were and were not sampled from the PrAF. Both the arguments and relations are assumed to be sampled independently such that the joint probability of the realisation of an induced framework is computed through a series of products. This assumption states that the existence of one argument is independent from the existence of other arguments within a given induced graph. Moreover, the sampling of relations from the PrAF is also assumed to be independent from the existence of other relations in an induced graph. However, the relations are not sampled entirely independently because they depend on the existence of certain induced arguments, as shown in Equation 5.2.

$$P_{\text{PrAF}}^{\mathcal{S}}(\mathcal{G}^I) = \prod_{a \in \mathcal{A}^I} P_A(a) \prod_{a \in A \setminus \mathcal{A}^I} [1 - P_A(a)] \prod_{r \in \mathcal{R}^I} P_R(r) \prod_{r \in \tilde{R} \setminus \mathcal{R}^I} [1 - P_R(r)] \quad (5.3)$$

The semantic evaluation function $\zeta^{\mathcal{S}}(\mathcal{G}^I, \mathcal{X})$, as stated in Definition 5.3, quantifies whether a set of arguments is contained within an extension of an induced argumentation framework. The function returns *true* if and only if the set of arguments \mathcal{X} is contained within an extension, found through computation of the semantics.

Definition 5.3 (Semantic Evaluation Function). Let $\mathcal{G}^I = (\mathcal{A}^I, \mathcal{R}^I)$ be an induced argument graph, \mathcal{S} be a semantics, \mathcal{E} be a set of extensions of \mathcal{G}^I with respect to

the semantics \mathcal{S} , \mathcal{X} be a user-chosen set of arguments, and $\zeta^{\mathcal{S}} : (\mathcal{A}^I, \mathcal{R}^I) \times 2^{\mathcal{A}^I} \mapsto \{0, 1\}$ be a semantic evaluation function. The semantic evaluation function returns a value of unity, i.e. $\zeta^{\mathcal{S}}(\mathcal{G}^I, \mathcal{X}) = 1$, if and only if the set of arguments \mathcal{X} features within an extension, such that $\mathcal{X} \in \mathcal{E}$. The semantic evaluation function returns zero, i.e. $\zeta^{\mathcal{S}}(\mathcal{G}^I, \mathcal{X}) = 0$, if and only if the set of arguments do not feature in any of the extensions, such that $\mathcal{X} \notin \mathcal{E}$.

The pseudo-code for the semantic evaluation function is included in Algorithm C.2 in Appendix C.1, for the reader's benefit.

The probability of observing a set of arguments \mathcal{X} is computed by counting the number of times those arguments were discovered to be acceptable, with respect to the chosen semantics \mathcal{S} , within the extensions of the set of all induced argument graphs $I(PrAF)$, such that

$$P_{PrAF}^{\mathcal{S}}(\mathcal{X}) = \sum_{\mathcal{G}^I \in I(PrAF)} P_{PrAF}^{\mathcal{S}}(\mathcal{G}^I) \text{ where } \zeta^{\mathcal{S}}(\mathcal{G}^I, \mathcal{X}) = true. \quad (5.4)$$

In this subsection, I have introduced PrAFs, a formalism that provides a method of quantifying the probability of observing a set of acceptable arguments. In the following section, I present the two contributions of this chapter: a definition of arguments with embedded hypotheses and a definition of information arguments, using the ASPIC⁺ framework; and a novel technique, comprised of a series of algorithms, that is capable of refining an intelligence analyst's prior beliefs using PrAFs and Bayes' theorem.

5.3 On Human-Machine Bayesian Hypothesis Evaluation

While Dung's seminal account of abstract argumentation allows for the identification of sets of admissible arguments, its level of abstraction means it pays no attention to the internal structure of those arguments. The use of structured argumentation is beneficial within the context of intelligence analyses because it should provide analysts with a finer-grained representation and understanding of their reasoning. The ASPIC⁺ framework was chosen for this work because it adopts an intermediate level of abstraction, whilst also offering an abstract account of the acceptability of structured arguments [177, 150]. Recall from Section 2.3.2.1 that the ASPIC⁺ argumentation theory was restricted to the set of ordinary premises, i.e. $\mathcal{K} = \mathcal{K}_p$, and defeasible rules, i.e. $\mathcal{R} = \mathcal{R}_d$, without preferences.

The remainder of this section proceeds as follows. In Section 5.3.1, I employ Pheron and Heuer Jr.'s definition of a hypothesis, as discussed in Chapter 1, and reformulate it in terms of the ASPIC⁺ framework; thus, rendering it capable of representing hypotheses, while also defining how arguments containing information are instantiated using the same formalism. I then define how: an analyst can assign prior probabilities to the set of hypotheses in Section 5.3.2; to employ the probability of inducement in order to describe the uncertainty in sources of information and attacks in Sections 5.3.3 and 5.3.4, respectively; to compute the likelihood of the information and all other hypotheses being, respectively, justified and defeated, given the hypothesis of interest was found to be acceptable, in Section 5.3.5; to employ Bayes' theorem to obtain the probability of a hypothesis given a semantically evaluated information set in Section 5.3.6; and to conclude the section, in Section 5.3.7, I present the pseudo-code for the proposed algorithms which, when combined, are capable of refining an analyst's initial probabilistic beliefs based upon the acceptability of the arguments present within their analysis.

5.3.1 Hypotheses and Information

Recall from Chapter 1, a hypothesis is defined as a statement which employs falsifiable observations to forecast anticipated results clearly. Remember also the ASPIC⁺ framework and the logical language \mathcal{L} from an *argumentation system*, as per Definition 2.10 in Section 2.3.2.1. In this chapter, it is assumed that there is a set of n hypotheses $H_{\mathcal{L}}$ which are statements within the logical language, as described in Definition 5.4.

Definition 5.4 For a given logical language \mathcal{L} , the set $H_{\mathcal{L}} \subset \mathcal{L}$ contains n hypotheses, where $H_{\mathcal{L}} = \{h_{\mathcal{L}}^{(1)}, \dots, h_{\mathcal{L}}^{(n)}\}$, $h_{\mathcal{L}}^{(i)}$ refers to the i -th hypothesis and each hypothesis is a statement within the logical language, such that $\forall h_{\mathcal{L}}^{(i)} \in H_{\mathcal{L}}, h_{\mathcal{L}}^{(i)} \in \mathcal{L}$.

With regards to structured argumentation, I assume that a hypothesis is a statement that is embedded within the conclusion of an argument so that it is supported by a set of reasons to believe it. This assumption is in agreement with the argumentation scheme presented in Chapter 3, where hypotheses form the conclusion of both the original and adapted versions of Walton's argument from cause to effect [236]. This assumption is also aligned with Toniolo *et al.*'s definition of a hypothesis in [216], where it was assumed that an extension of an argumentation framework was a hypothesis. The extensions in Toniolo *et al.*'s proposal contain an argument with an embedded hypothesis and the corresponding, acceptable sub-arguments from an ASPIC⁺ argumentation theory that support that hypothesis. The definition of an argument containing a hypothesis

within the ASPIC⁺ formalism is presented in Definition 5.5.

Definition 5.5 For an argumentation theory AT , a hypothesis $h_{\mathcal{L}}^{(i)} \in H_{\mathcal{L}}$ is the conclusion of an argument $A_H^{(i)}$, such that $\text{Conc}(A_H^{(i)}) = h_{\mathcal{L}}^{(i)}$. The hypothesis $h_{\mathcal{L}}^{(i)}$ is supported by a set of m premises, such that $\text{Prem}(A_H^{(i)}) = \{p_1, \dots, p_m\}$. Therefore, using the definition of an argument in the ASPIC⁺ framework (Definition 2.14 in Section 2.3.2.1), an argument $A_H^{(i)}$ containing a hypothesis $h_{\mathcal{L}}^{(i)}$ is assumed to have the following defeasible inference

$$A_H^{(i)} = \{p_1, \dots, p_m \Rightarrow h_{\mathcal{L}}^{(i)}\}.$$

Following Definition 5.5, a human agent must select a set of n arguments $H_A = \{A_H^{(1)}, \dots, A_H^{(n)}\}$ containing hypotheses, where $A_H^{(i)}$ is the i -th argument with the i -th hypothesis $h_{\mathcal{L}}^{(i)}$ as its conclusion.

After choosing the arguments with embedded hypotheses, the analyst is left with a set of statements within the logical language that are either information, evidence, or assumptions. For the purposes of this chapter, I restrict the set of remaining statements and assume that they only include items of information, as stated in Definition 5.6. However, I note that one could represent information arguments as evidence by assigning a value of unity to that argument's inducement probability, as I briefly discuss in Section 5.3.3.

Definition 5.6 For a logical language \mathcal{L} and hypotheses $H_{\mathcal{L}} \subset \mathcal{L}$, the information statements are a set $I_{\mathcal{L}} \subset \mathcal{L}$ that contains \tilde{n} items of information which are the remaining statements within the logical language, such that $I_{\mathcal{L}} = \mathcal{L} \setminus H_{\mathcal{L}}$, where $I_{\mathcal{L}} = \{i^{(1)}, \dots, i^{(\tilde{n})}\}$ and $\forall i^{(j)} \in I_{\mathcal{L}}, i^{(j)} \in \mathcal{L}$.

Similar to arguments which include hypotheses as their conclusion, the statements of information can also be instantiated within arguments in the ASPIC⁺ formalism, as asserted in Definition 5.7. The difference between an argument containing a hypothesis and one that contains information is that the argument containing information does not possess a hypothesis as its conclusion.

Definition 5.7 For an argumentation theory (\mathcal{K}_p, AS) , a set of information statements $I_{\mathcal{L}} = \{i^{(1)}, \dots, i^{(\tilde{n})}\}$ (Definition 5.6), and a set of arguments containing information only $I_A = \{A_I^{(1)}, \dots, A_I^{(\tilde{n})}\}$, the i -th information argument $A_I^{(i)} \in I_A$ is one where its conclusion $\text{Conc}(A_I^{(i)}) = i^{(i)}$ is in the logical language, i.e. $i^{(i)} \in \mathcal{L}$, and it is not a hypothesis, i.e. $i^{(i)} \notin H_{\mathcal{L}}$. As per Definition 2.14 in Section 2.3.2.1, the argument $A_I^{(i)}$ can have the following defeasible inferences:

- $A_I^{(i)} = \{i^{(i)}\}$; or

$$\bullet A_I^{(i)} = \{i^{(1)}, \dots, i^{(\tilde{n}-1)} \Rightarrow i^{(j)}\}.$$

The human agent can choose as many or as few information statements as they desire in order to instantiate arguments containing information only. However, as I mentioned earlier, it is assumed that after the analyst has chosen the set of arguments with embedded hypotheses, the remaining arguments are comprised of information alone. The set of arguments with their embedded hypotheses H_A and the set of arguments containing information I_A are employed in the evaluation of those hypotheses, and those sets should be kept separate to allow for a fair assessment, as stated in Definition 5.8. This is a common procedure within the intelligence community as individual hypotheses can only be attacked by data in the row entries within, for example, the ACH.

Definition 5.8 For an argumentation framework corresponding to an ASPIC⁺ argumentation theory $\mathcal{G}_{AT} = (\mathcal{A}, \mathcal{R})$, the set of arguments containing information $I_A \in \mathcal{A}$ is assumed to be the difference between the set of arguments within \mathcal{G}_{AT} and the set of arguments containing hypotheses $H_A \in \mathcal{A}$, chosen by the user, such that $I_A = \mathcal{A} \setminus H_A$ where H_A and I_A are closed under union, such that $H_A \cup I_A = \mathcal{A}$, and intersection, such that $H_A \cap I_A = \emptyset$.

In this subsection, I have defined arguments containing hypotheses and arguments comprised of information with respect to the ASPIC⁺ framework. The arguments instantiated using this formalism can be converted to a Dung argument system, using Definition 2.16 in Section 2.3.2.1, and also to a PrAF, as long as the human agent assigns a probability of inducement to each argument and relation within the framework. In the following subsection, I show how the analyst can assign prior probabilities to hypotheses in the logical language, and how the confidence in their probabilistic judgements can be represented by inducement probabilities from Li, Oren, and Norman's PrAF.

5.3.2 Priors and Confidence in the Probabilities of Hypotheses

Based on the information available to them and their analysis, an intelligence analyst must make a judgement about the prior probability of each hypothesis. And the analyst can employ a number of different techniques to that effect. For example, using the ACH for evaluation, the analyst is able to compute a prior for each hypothesis within their analysis. Under the assumption that the probability of a hypothesis is equivalent to the probability of that hypothesis as a statement within the logical language, Equation

5.5 describes the bounds of those probabilistic judgements, such that

$$0 \leq P(h_{\mathcal{L}}^{(i)}) \leq 1, \quad \forall h_{\mathcal{L}}^{(i)} \in H_{\mathcal{L}}. \quad (5.5)$$

The set of hypotheses chosen by the analyst must be mutually exclusive

$$P(h_{\mathcal{L}}^{(i)}, h_{\mathcal{L}}^{(j)}) = 0, \quad \forall h_{\mathcal{L}}^{(i)}, h_{\mathcal{L}}^{(j)} \in H_{\mathcal{L}} \text{ where } i \neq j, \quad (5.6)$$

and collectively exhaustive

$$\sum_{i=1}^n P(h_{\mathcal{L}}^{(i)}) = 1, \quad \forall h_{\mathcal{L}}^{(i)} \in H_{\mathcal{L}}, \quad (5.7)$$

where n refers to the number of hypotheses chosen by the user.

The reason that it was assumed that the priors of hypotheses were MECE is two-fold. First, looking to the mutual exclusivity of hypotheses, as shown in Equation 5.6, this assumption ensures that if one hypothesis were found to be true, then the other hypotheses within an analysis should be false. Second, hypotheses should be collectively exhaustive, as stated in Equation 5.7, to ensure that the analyst is assessing all the plausible hypotheses pertaining to their analysis.

As mentioned earlier in this subsection, it was assumed that the prior probability of a hypothesis was equivalent to the probability of a statement within the logical language. It was also assumed that the probability of a hypothesis, a statement within the logical language, was equivalent to the probability of an argument containing a hypothesis, such that

$$P(h_{\mathcal{L}}^{(i)}) \iff P(A_H^{(i)}), \quad \forall h_{\mathcal{L}}^{(i)} \in H_{\mathcal{L}} \text{ and } \forall A_H^{(i)} \in H_A. \quad (5.8)$$

The probability of a hypothesis as a statement or within an argument – i.e. $P(h_{\mathcal{L}}^{(i)})$ or $P(A_H^{(i)})$, respectively – is not to be confused with the probability of inducing an argument containing a hypothesis, i.e. $P_A(A_H^{(i)})$, where

$$P(A_H^{(i)}) \not\iff P_A(A_H^{(i)}). \quad (5.9)$$

The key distinction between the two probabilities in Equation 5.9 is that the probability of the hypothesis $P(A_H^{(i)})$ is intended to be an intelligence analyst's prior belief, resulting from their analysis, such as from the outputs from an ACH, whereas the probability $P_A(A_H^{(i)})$ is the inducement probability of an argument with an embedded hypothesis, and is assumed to be the representation of the analyst's confidence in their prior beliefs. The importance of this difference becomes apparent when I introduce Bayes' theorem, in Section 5.3.6.

5.3.3 Uncertainty in Items of Information

As alluded to in Equation 5.9, the inducement probabilities provide a method of describing the confidence in an analyst's set of priors for hypotheses. The uncertainty in an item of information can also be described using the same method. Uncertainty in items of information may arise from, for example, the truth of an assertion within the information or the confidence in its source. The intelligence community already possesses tools which help analysts describe the inherent uncertainty in row entries of the ACH. One example is the Probability (or Uncertainty) Yardstick [221], which I introduced in Section 2.1.4. Moreover, as I discussed in Section 2.1.5.2, in the United Kingdom's Department for Defence Intelligence's *Quick Wins for Busy Analysts* [222], analysts employ a *traffic light* system – i.e. *green*, *yellow*, and *red* – to describe the uncertainty of row entries within the ACH. However, even in this more-expressive version of the ACH, analysts are not required, or even shown how, to use the traffic light system in their reasoning about hypotheses. While this uncertainty quantification is not used in the evaluation of hypotheses, such metrics could be employed within the algorithms proposed in this chapter, through assignment of inducement probabilities, in order to describe an analyst's confidence in arguments formed from the row entries of the ACH.

An analyst may decide that data contained within an information argument is certain, and PrAFs are capable of representing this notion. In this instance, the analyst should assign a value of unity to the information argument's inducement probability such that it will always feature in the set of all induced argument graphs. Such an argument could be thought of as evidence because there is no uncertainty in its existence.

One interesting point to note here is that arguments containing information alone are arguments that include either one or more statements from the logical language, as per Definition 5.7, where each statement may have its own uncertainty score. The probability of inducement for an information argument is the combination of uncertainty scores for each item of information within the argument. Therefore, the analyst should reason with the information statements within that argument in order to arrive at its probability of inducement. While methods for combining uncertainties in this context are outside of the remit of this chapter and thesis, it is important to note the distinction between local and global uncertainty scores that, respectively, arise from the uncertainty of individual information statements and the argument-level uncertainty, which is a combination of uncertain information statements. For the remainder of this chapter, it is assumed that the analyst has assigned a global, argument-level uncertainty score, in the form of an inducement probability, for all information arguments.

5.3.4 Uncertainty in Attacks

So far, I have described the uncertainty in arguments. However, an analyst might identify uncertainty in an attack between information arguments, or from an information argument to an argument with an embedded hypothesis. For instance, as the ACH should ideally be conducted as a group exercise, an analyst, called Analyst A, might say to some Analyst B: *I am not so sure that this row entry attacks that hypothesis, I do not think that this attack is so strong, or I disagree that there is an attack between those information arguments.* In this particular instance, a prudent analyst may still want to include that attack within their analysis, but Pherson and Heuer Jr.'s current version of the ACH does not allow for this. Inducement probabilities of attack relations provide an avenue to represent such disagreement and uncertainty.

In this thesis, I omit discussion about attacks from arguments containing hypotheses to information arguments because such attacks cannot exist within the ACH formalism. Furthermore, attacks between arguments within embedded hypotheses are assumed to always be certain due to the mutual exclusivity of hypotheses.

5.3.5 The Likelihood

The likelihood of observing the information and not the other hypotheses, given the existence and acceptability of the hypothesis of interest, can be computed using PrAFs. Computing this probability is important because it should allow the intelligence analyst to understand how probable the information and defeat of all other hypotheses is, given that the hypothesis they are interested in is assumed to be true. This conditional probability would ordinarily be quite difficult for an analyst to compute, but it is a task where computational argumentation could provide tangible benefit to the analyst. There are three different approaches for the computation of the likelihood, which are introduced throughout this subsection. The first method assumes that the acceptability of the information and the hypothesis of interest are dependent. The second assumes that the acceptability of information and the lead hypothesis is completely independent. The third method employs the independence assumption from the second to approximate the likelihood using the probability of all other hypotheses either not existing nor being accepted, reducing the number of multiplications, in order to use the proposed technique on large PrAFs. In all three approaches, the existence and acceptability of all other hypotheses was assumed to be independent.

5.3.5.1 Assuming Dependence Between Hypotheses and Information

At a glance, assuming dependence between evaluated information arguments and a hypothesis is intuitive because the acceptability of a hypothesis should depend on the amount of information which either supports or conflicts with it. For this approach, one must first quantify three things, namely: the probability of acceptance of the i -th argument with the i -th hypothesis embedded within its conclusion; the joint probability of acceptance of the set of information arguments and the i -th argument that contains the i -th hypothesis; and, finally, the joint probability of all other arguments with embedded hypotheses not existing nor being acceptable, except for the i -th argument containing the i -th hypothesis because that is the argument for which the posterior is approximated.

The probability of existence and acceptance of the i -th hypothesis. The first, and by far the simplest, quantity that we are interested in calculating is the probability of observing the argument $A_H^{(i)}$ in the extensions across the set of all induced graphs. Substituting the argument $A_H^{(i)}$ into Equation 5.4, we are now able to quantify its probability of existence and acceptance, as shown in Equation 5.10.

$$P_{PrAF}^S(\{A_H^{(i)}\}) = \sum_{\mathcal{G}^I \in I(PrAF)} P_{PrAF}^I(\mathcal{G}^I) \quad \text{where } \zeta^S(\mathcal{G}^I, \{A_H^{(i)}\}) = true \quad (5.10)$$

The existence of the argument $A_H^{(i)}$ depends on the inducement probability assigned to it by the analyst, i.e. $P_A(A_H^{(i)})$. The inducement probability of an argument containing a hypothesis was assumed to be a representation of the analyst's confidence in their prior probabilistic judgements, and it is not to be confused with the prior itself, as previously shown in Equation 5.9. The probability that the argument $A_H^{(i)}$ will be accepted, i.e. $P_{PrAF}^S(\{A_H^{(i)}\})$, depends on the topology of the PrAF, and whether it was acceptable, according to the chosen semantics \mathcal{S} , in a subset of induced frameworks.

For the sake of brevity, $P_{PrAF}^S(\{A\})$ and $P_{PrAF}^S(\{A_1, A_2, \dots, A_\infty\})$ is, respectively, denoted $P_{PrAF}^S(A)$ and $P_{PrAF}^S(A_1, A_2, \dots, A_\infty)$ throughout the remainder of this chapter. With a slight abuse of notation, joint probabilities which are the union of sets of arguments, e.g. the set of information arguments I_A and an argument containing a hypothesis $A_H^{(i)}$, such as $P_{PrAF}^S(I_A \cup \{A_H^{(i)}\})$, may also be referred to as either $P_{PrAF}^S(I_A \cup A_H^{(i)})$, $P_{PrAF}^S(I_A; A_H^{(i)})$, or $P_{PrAF}^S(A_I^{(1)}, \dots, A_I^{(\tilde{n})}; A_H^{(i)})$.

The probability of existence and acceptance of all information arguments and the i -th hypothesis. We now wish to compute the joint probability of observing the set of information arguments I_A and the argument that contains the hypothesis of interest,

such that

$$P_{PrAF}^S(I_A, A_H^{(i)}) = P_{PrAF}^S(I_A \cup A_H^{(i)}) \quad (5.11)$$

$$= P_{PrAF}^S(A_I^{(1)}, \dots, A_I^{(n)}; A_H^{(i)}), \quad (5.12)$$

where the probability in Equation 5.12 can be computed, again, using PrAFs, in accordance with Equation 5.4.

The probability of all other hypotheses not existing nor being acceptable. The final probability required to compute the likelihood is the probability of all other arguments with embedded hypotheses not existing nor being found to be acceptable. The reason for this is that if one argument containing a hypothesis is found to be true, then all others must be false, due to the mutual exclusivity of hypotheses. In order to compute this probability, we must identify the set of arguments with embedded hypotheses that are assumed to be untrue when the argument $A_H^{(i)}$ exists and is acceptable. It follows that this set is just the complement of the set of hypotheses H_A where the argument of interest $A_H^{(i)}$ has been removed, such that

$$H_A^{c,(i)} = H_A \setminus \{A_H^{(i)}\} \quad (5.13)$$

is the set of arguments with embedded hypotheses without the argument containing the hypothesis of interest.

Using the set of remaining hypotheses $H_A^{c,(i)}$, shown in Equation 5.13, the joint probability of non-existence and defeat of all other arguments with embedded hypotheses is presented in Equation 5.14. Assuming that the presence and acceptance of those arguments is independent, the joint probability in Equation 5.14 can be reduced to a series of products of marginal probabilities, as shown in Equation 5.15.

$$P_{PrAF}^S(\neg H_A^{c,(i)}) = P_{PrAF}^S(\neg A_H^{(1)}, \dots, \neg A_H^{(n-1)}) \quad (5.14)$$

$$= P_{PrAF}^S(\neg A_H^{(1)}) \dots P_{PrAF}^S(\neg A_H^{(n-1)}) \quad (5.15)$$

While one could count the number of times that an argument with an embedded hypothesis was not present within the extensions of the set of all induced argument graphs, this computation is superfluous because the number of times those arguments were acceptable is already counted, as I show later in the pseudo-codes for all three approaches presented in Section 5.3.7. Using the axioms of probability, the probability that an argument did not exist and was not acceptable is just the probability that it was accepted subtracted from unity, such that

$$P_{PrAF}^S(\neg A_H^{(j)}) = 1 - P_{PrAF}^S(A_H^{(j)}). \quad (5.16)$$

By substituting the notion presented in Equation 5.16 into 5.15 for all arguments within the set $H_A^{c,(i)}$, it follows that

$$P_{PrAF}^S(\neg H_A^{c,(i)}) = \prod_{A_H^{(j)} \in H_A^{c,(i)}} [1 - P_{PrAF}^S(A_H^{(j)})] \quad (5.17)$$

where $P_{PrAF}^S(A_H^{(j)})$ is computed in a similar fashion to Equation 5.10 but for the j -th argument containing the j -th hypothesis.

The likelihood. Now we have all that we need to compute the likelihood, which is just the probability that the information arguments and the lead hypothesis were acceptable, and all the remaining arguments containing hypotheses did not exist nor were acceptable, conditioned on the acceptability of the argument containing the hypothesis of interest. More formally,

$$P_{PrAF}^S(I_A, \neg H_A^{c,(i)} | A_H^{(i)}) = \frac{P_{PrAF}^S(I_A \cup A_H^{(i)} \cup \neg H_A^{c,(i)})}{P_{PrAF}^S(A_H^{(i)})} \quad (5.18)$$

where the numerator is the joint probability of: acceptance of the information; the argument containing the hypothesis of interest $A_H^{(i)}$ being true; and the probability of non-acceptance for all other hypotheses. The denominator is the probability of the lead hypothesis.

Under the assumption that the probability of the argument with the embedded hypothesis $A_H^{(i)}$ and the set of information arguments I_A are independent from all other unacceptable hypotheses, the joint probability on the numerator of Equation 5.18 can be split into the product of two probabilities, as shown in Equation 5.19.

$$P_{PrAF}^S(I_A, \neg H_A^{c,(i)} | A_H^{(i)}) = \frac{P_{PrAF}^S(I_A, A_H^{(i)}) P_{PrAF}^S(\neg H_A^{c,(i)})}{P_{PrAF}^S(A_H^{(i)})} \quad (5.19)$$

By substituting Equations 5.12 and 5.17 into Equation 5.19, the likelihood of acceptance of the argument containing the hypothesis of interest and the set of information arguments, and defeat of all other hypotheses, conditioned on the acceptability of the argument $A_H^{(i)}$, across N induced graphs, can be approximated using Equation 5.20.

$$P_{PrAF}^S(I_A, \neg H_A^{c,(i)} | A_H^{(i)}) = \frac{P_{PrAF}^S(A_I^{(1)}, \dots, A_I^{(\tilde{n})}, A_H^{(i)}) \prod_{A_H^{(j)} \in H_A^{c,(i)}} [1 - P_{PrAF}^S(A_H^{(j)})]}{P_{PrAF}^S(A_H^{(i)})} \quad (5.20)$$

As shown in Equation 5.20, the likelihood of the information and the other, un-accepted arguments containing hypotheses, given the argument with the embedded hypothesis is accepted, can be computed using a dependence assumption. However, computing the likelihood using this method is not without its issues because the dependence assumption, i.e. $P_{PrAF}^S(A_I^{(1)}, \dots, A_I^{(\tilde{n})}, A_H^{(i)})$, necessitates that all information arguments and an argument containing a hypothesis are in an extension of an induced framework. While this is not an issue for small PrAFs, computing the likelihood will become more challenging when employing real-world data with this approach. For instance, for small PrAFs, the likelihood can almost always be computed even for information arguments that attack each other and the argument with embedded hypothesis, if the number of induced argumentation frameworks N is large enough. However, for large, real-world PrAFs, it is highly unlikely that all information arguments and an argument containing a hypothesis will feature in the same extension, especially when a subset of information arguments attack the hypothesis of interest, because of the potential variance of argumentation frameworks that can be induced. Thus, for a sufficiently large enough PrAF, with information arguments that possess a more definite probability of inducement, the joint probability will become increasingly small, if not zero, because there will be fewer induced graphs that contain those arguments as nodes with no attacks between them.

5.3.5.2 Assuming Independence Between Evaluated Arguments

One solution to the problem pertaining to Equation 5.20 is to relax the dependence assumption. In this subsection, I will reformulate the likelihood using the assumption that the probability of existence and acceptability of the set of information arguments and the argument $A_H^{(i)}$ are independent.

To begin, Equation 5.12 can be rewritten so that the joint probability of the set of information arguments and the argument $A_H^{(i)}$ is the product of marginals, as shown in Equation 5.21.

$$P_{PrAF}^S(I_A, A_H^{(i)}) = P_{PrAF}^S(A_I^{(1)}) \dots P_{PrAF}^S(A_I^{(\tilde{n})}) P_{PrAF}^S(A_H^{(i)}) \quad (5.21)$$

Substituting Equation 5.21 into Equation 5.19 results in a likelihood comprised of independent probabilities, such that

$$\frac{P_{PrAF}^S(I_A, \neg H_A^{c(i)} | A_H^{(i)})}{P_{PrAF}^S(A_I^{(1)}) \dots P_{PrAF}^S(A_I^{(\tilde{n})}) P_{PrAF}^S(\neg A_H^{(1)}) \dots P_{PrAF}^S(\neg A_H^{(n-1)}) P_{PrAF}^S(A_H^{(i)})} = \quad (5.22)$$

where the probability of the argument with an embedded hypothesis $A_H^{(i)}$ cancels out, such that

$$P_{PrAF}^S(I_A, \neg H_A^{c,(i)} | A_H^{(i)}) = P_{PrAF}^S(A_I^{(1)}) \dots P_{PrAF}^S(A_I^{(\tilde{n})}) P_{PrAF}^S(\neg A_H^{(1)}) \dots P_{PrAF}^S(\neg A_H^{(n-1)}) \quad (5.23)$$

and reduces to

$$P_{PrAF}^S(I_A, \neg H_A^{c,(i)} | A_H^{(i)}) = \prod_{A_I^{(j)} \in I_A} P_{PrAF}^S(A_I^{(j)}) \prod_{A_H^{(j)} \in H_A^{c,(i)}} [1 - P_{PrAF}^S(A_H^{(j)})]. \quad (5.24)$$

With regards to Equation 5.24, an interesting point to note is that the probability only depends on two terms: the probability of the information; and the probability that the other arguments containing other hypotheses did not exist and were not acceptable. In a sense, this result is argumentative because the likelihood depends on the information available and the probability that all other hypotheses were defeated. While the argument of interest $A_H^{(i)}$ is not employed in the quantification of the likelihood, its acceptability across the set of all induced argument graphs is implied by the probability that all other hypotheses were unacceptable, due to the mutual exclusivity of hypotheses. Moreover, the likelihood is somewhat analogous to the ACH's attempt to reduce confirmation bias, as I discussed in Section 2.1.5.2, because the most likely hypothesis within the ACH is the one that conflicts with the fewest number of row entries, and the likelihood in Equation 5.24 depends on the defeat of other hypotheses, not the acceptability of the hypothesis that we are interested in.

5.3.5.3 Towards Real-World Data: Solving the Floating Point Underflow Problem

While the likelihood in Equation 5.24 can be approximated using PrAFs, as the number of information arguments increases, the joint probability of the semantically evaluated information set will tend towards zero, such that

$$\lim_{|I_A| \rightarrow \infty} \prod_{A_I^{(j)} \in I_A} P_{PrAF}^S(A_I^{(j)}) \approx 0. \quad (5.25)$$

Although this is not particularly an issue for small, toy problems, it is indeed a significant problem when working with real-world data with a sufficiently large number of arguments, creating a floating point underflow condition such that a computer cannot represent the joint probability of information in memory. To resolve this, there is one

obvious, slight adaptation to the calculation of the likelihood which is described in more detail below.

Using Equation 5.24, which contains the assumption about the independence of the probability of the information, the ratio of the likelihoods conditioned on the arguments $A_H^{(i)}$ and $A_H^{(k)}$ is presented in Equation 5.26.

$$\frac{P_{PrAF}^S(I_A, \neg H_A^{c,(i)} | A_H^{(i)})}{P_{PrAF}^S(I_A, \neg H_A^{c,(k)} | A_H^{(k)})} = \frac{\prod_{A_I^{(j)} \in I_A} P_{PrAF}^S(A_I^{(j)}) \prod_{A_H^{(j)} \in H_A^{c,(i)}} [1 - P_{PrAF}^S(A_H^{(j)})]}{\prod_{A_I^{(m)} \in I_A} P_{PrAF}^S(A_I^{(m)}) \prod_{A_H^{(m)} \in H_A^{c,(k)}} [1 - P_{PrAF}^S(A_H^{(m)})]} \quad (5.26)$$

Due to the assumption about independence between the information arguments and hypotheses, the joint probability of information arguments in both the numerator and denominator of Equation 5.26 are equal, such that

$$\frac{\prod_{A_I^{(j)} \in I_A} P_{PrAF}^S(A_I^{(j)})}{\prod_{A_I^{(m)} \in I_A} P_{PrAF}^S(A_I^{(m)})} = 1, \quad \forall A_I^{(j)}, A_I^{(m)} \in I_A. \quad (5.27)$$

Equation 5.27 enables the simplification of Equation 5.26 through the removal of the joint probability of information for both likelihoods, resulting in Equation 5.28.

$$\frac{P_{PrAF}^S(I_A, \neg H_A^{c,(i)} | A_H^{(i)})}{P_{PrAF}^S(I_A, \neg H_A^{c,(k)} | A_H^{(k)})} = \frac{\prod_{A_H^{(j)} \in H_A^{c,(i)}} [1 - P_{PrAF}^S(A_H^{(j)})]}{\prod_{A_H^{(m)} \in H_A^{c,(k)}} [1 - P_{PrAF}^S(A_H^{(m)})]} \quad (5.28)$$

The likelihood, therefore, is proportional to the probability of defeat of all other arguments with embedded hypotheses, such that

$$P_{PrAF}^S(I_A, \neg H_A^{c,(i)} | A_H^{(i)}) = k \cdot \prod_{A_H^{(j)} \in H_A^{c,(i)}} [1 - P_{PrAF}^S(A_H^{(j)})] \quad (5.29)$$

where k is a constant equal to the joint probability of acceptable information arguments.

Using the independence assumption, the joint probability of the information is the same for all arguments containing hypotheses. Thus, the proportion of the likelihood that affects it is the probability of all other hypotheses being defeated, as shown in Equation 5.30, which can be employed as an analogue for the approximation of the likelihood.

$$P_{PrAF}^S(I_A, \neg H_A^{c,(i)} | A_H^{(i)}) \propto \prod_{A_H^{(j)} \in H_A^{c,(i)}} [1 - P_{PrAF}^S(A_H^{(j)})], \quad \forall A_H^{(i)} \in H_A. \quad (5.30)$$

In this subsection, I have defined how to compute the likelihood of the information and defeat of all other hypotheses given the argument containing the hypothesis of interest was acceptable. Three different approaches that assumed dependence and independence between information and hypotheses were presented in Equations 5.20, 5.24, and 5.30. In the next subsection, I show how the likelihood and an intelligence analyst's set of priors are combined with Bayes' theorem to compute the posterior probability of a hypothesis.

5.3.6 Bayes' Theorem

Turning to Bayes' theorem, we can compute the probability of the hypothesis of interest, conditioned on the probability of the information arguments and the probability of all other hypotheses not existing nor being acceptable. The calculation of Bayes' theorem is based upon the assumption that the likelihood that was approximated using PrAFs is equivalent to the likelihood of the information arguments and defeat of all other hypotheses given the hypothesis of interest is acceptable, such that

$$P_{PrAF}^S(I_A, \neg H_A^{c,(i)} | A_H^{(i)}) \iff P(I_A, \neg H_A^{c,(i)} | A_H^{(i)}) \quad (5.31)$$

Following this assumption, the probability of an argument containing a hypothesis $A_H^{(i)}$ conditioned on the set of acceptable information I_A and the defeat of all other arguments with embedded hypotheses $H_A^{c,(i)}$ is found to be

$$P(A_H^{(i)} | I_A, \neg H_A^{c,(i)}) = \frac{P(I_A, \neg H_A^{c,(i)} | A_H^{(i)}) P(A_H^{(i)})}{\sum_{j \in H_A} P(I_A, \neg H_A^{c,(j)} | A_H^{(j)}) P(A_H^{(j)})} \quad (5.32)$$

where $P(I_A, \neg H_A^{c,(i)} | A_H^{(i)})$ and $P(I_A, \neg H_A^{c,(j)} | A_H^{(j)})$ are the likelihoods, which can be approximated using either Equations 5.20, 5.24 or 5.30, and $P(A_H^{(i)})$ and $P(A_H^{(j)})$ is the human agent's prior beliefs, as defined in Equation 5.8, for the i -th and j -th argument containing a hypothesis, respectively.

The distinction between an analyst's prior beliefs about a hypothesis and the inducement probabilities of arguments containing hypotheses is made apparent in Equation 5.32. Here, priors are represented as either $P(A_H^{(i)})$ or $P(A_H^{(j)})$, respectively, for the i -th and j -th hypothesis. Those probabilities are the output from an intelligence analyst's analysis and follow the axioms presented in Equations 5.5, 5.6, 5.7, and 5.8. On the contrary, the likelihoods – i.e., $P(I_A, \neg H_A^{c,(i)} | A_H^{(i)})$ and $P(I_A, \neg H_A^{c,(j)} | A_H^{(j)})$ – incorporate the analyst's confidence in their prior probabilistic judgements by employing the inducement probability of information arguments and hypotheses to alter the number

of times those arguments (and attacks between them) were instantiated and acceptable across N induced graphs.

5.3.7 Pseudo-code

In this subsection, I present the algorithms for human-machine Bayesian hypothesis evaluation to make the approach discussed in Sections 5.3.1 to 5.3.6 concrete. There are four, distinct algorithms. Algorithms 5.1, 5.2, and 5.3 approximate the likelihood of the information and the defeat of all other hypotheses, given the argument $A_H^{(i)}$ is acceptable. Algorithm 5.1 computes the likelihood using the assumption of dependence between the set of information arguments and arguments with embedded hypotheses, using Equation 5.20. Algorithm 5.2 assumes independence between information arguments and hypotheses, and computes the likelihood using Equation 5.24. Algorithm 5.3 employs the same independence assumption and solves the floating point underflow condition by using only the defeat of all other hypotheses to approximate the likelihood, in line with Equation 5.30. Both Algorithms 5.1 and 5.2 take as input: a PrAF; a set of arguments containing hypotheses H_A ; a set of information arguments I_A ; a semantics \mathcal{S} ; and the number of induced graphs N . Algorithm 5.3 takes the same input, however it does not require the set of information arguments for the algorithm. The output from Algorithms 5.1, 5.2, and 5.3 is a list of n likelihoods for all arguments with embedded hypotheses chosen by the human agent.

Algorithm 5.4 uses Bayes' theorem, as defined in Equation 5.32, to combine a human agent's probabilistic judgements about the prior probabilities of hypotheses and the outputs from Algorithms 5.1, 5.2, and 5.3.

A depiction of how the four algorithms were combined is provided in Figure 5.1.

5.3.7.1 Bayes' Theorem

Algorithm 5.4 employs Bayes' theorem to produce a posterior estimate of the probability of the i -th argument containing the i -th hypothesis, given the set of evaluated information arguments and defeat of all other hypotheses. The algorithm takes as input: a set of arguments containing hypotheses H_A chosen by the analyst; the analyst's belief about the prior probability of each hypothesis; and the list of likelihoods, which were approximated using either Algorithms 5.1, 5.2, or 5.3. The output from the algorithm is a refined probabilistic estimate of the probability of the argument $A_H^{(i)}$ which incorporates the analyst's prior beliefs, uncertainty pertaining to the information and hypotheses, and the acceptability of arguments found within the human agent's analysis.

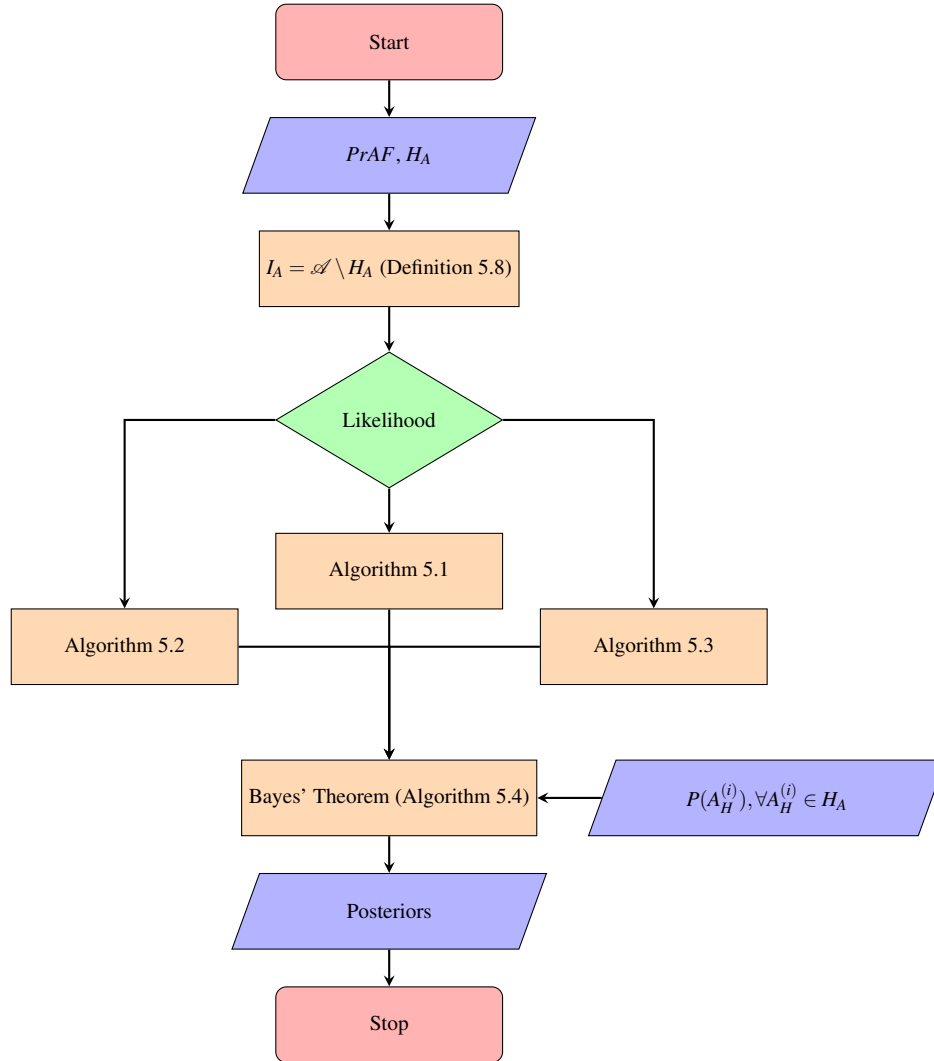


Figure 5.1: A flowchart depicting the process for approximating the posterior probability of arguments with embedded hypotheses conditioned on the available information and the defeat of all other hypotheses.

Algorithm 5.1: Compute the likelihood using Equation 5.20.

Input : $A \text{ PrAF} = (A, P_A, R, P_R)$;
 A set of arguments with embedded hypotheses H_A ;
 A set of arguments containing information I_A ;
 A semantics S ;
 A number of trials N

Output: A list of $|H_A|$ likelihoods $P_{PrAF}^S(I_A, \neg H_A | A_H)$ for each argument with an embedded hypothesis $A_H^{(i)}$.

```

1  $A_H = [A_H^{(1)}, \dots, A_H^{(n)}] \leftarrow H_A$  // Convert the set  $H_A$  to an indexed array  $A_H$ 
2  $A_{HandI} = [\ ]^1, \dots, [\ ]^n$  // Create an array containing  $n = |H_A|$  empty arrays
3 for  $i = 1$  to  $n$  do
4    $A_{HandI}^{(i)} \leftarrow [A_H^{(i)}, A_I^{(1)}, \dots, A_I^{(n)}]$  // Set  $A_{HandI}^{(i)}$  equal to the  $i$ -th argument with an
   embedded hypothesis and all the information arguments
5  $C_H \leftarrow \text{zeros}(C_H, |A_H|)$  // Create an array of zeros, of size  $|A_H|$ , to count the
   number of times an argument containing a hypothesis was acceptable
6  $C_{HandI} \leftarrow \text{zeros}(C_{HandI}, |A_H|)$  // Create an array of zeros, of size  $|A_H|$ , to count
   the number of times an argument containing a hypothesis and all the
   information was acceptable
7 for  $1$  to  $N$  do
8    $\mathcal{G}^I \leftarrow \text{INDUCE AF}(PrAF)$  // Algorithm C.1 in Appendix C.1
9   forall  $i \in A_H$  do
10     $C_H^{(i)} = \text{SEMANTIC EVALUATION FUNCTION}(\mathcal{G}^I, A_H^{(i)}, S, C_H^{(i)})$  // Algorithm C.2 in
    Appendix C.1
11     $C_{HandI}^{(i)} = \text{SEMANTIC EVALUATION FUNCTION}(\mathcal{G}^I, A_{HandI}^{(i)}, S, C_{HandI}^{(i)})$  // Algorithm C.2 in
    Appendix C.1
12  $P_{PrAF}^S(I_A, \neg H_A | A_H) \leftarrow \text{zeros}[P_{PrAF}^S(I_A, \neg H_A | A_H), |A_H|]$  // Create an array of zeros, of
    size  $A_H$ , to store the likelihoods
13 for  $i \in A_H$  do
14    $P_{PrAF}^S(A_H^{(i)}) = 0; P_{PrAF}^S(I_A, A_H^{(i)}) = 0; P_{PrAF}^S(\neg H_A^{c(i)}) = 0$  // Set to zero the probability of
    the  $i$ -th argument containing a hypothesis, the joint probability of
    the information and the  $i$ -th argument with an embedded hypothesis,
    and the probability of defeat of all other hypotheses
15   forall  $j \in A_H$  do
16     if  $A_H^{(i)} = A_H^{(j)}$  then
17        $P_{PrAF}^S(A_H^{(i)}) = \frac{C_H^{(j)}}{N}$  // Compute  $A_H^{(i)}$ 's probability
18        $P_{PrAF}^S(I_A, A_H^{(j)}) = \frac{C_{HandI}^{(j)}}{N}$  // Compute the joint probability of  $A_H^{(i)}$  and
       the information arguments
19     else
20       if  $P_{PrAF}^S(\neg H_A^{c(i)}) = 0$  then
21          $P_{PrAF}^S(\neg H_A^{c(i)}) = 1 - \frac{C_H^{(j)}}{N}$  // Compute the probability of defeat for
          $A_H^{(j)}$ , if the probability of defeat is equal to zero
22       else
23          $P_{PrAF}^S(\neg H_A^{c(i)}) = P_{PrAF}^S(\neg H_A^{c(i)}) \times \left(1 - \frac{C_H^{(j)}}{N}\right)$  // Compute the probability
         of defeat for  $A_H^{(j)}$  and multiply by  $P_{PrAF}^S(\neg H_A^{c(i)})$ , if
          $P_{PrAF}^S(\neg H_A^{c(i)})$  is non-zero
24    $P_{PrAF}^S(I_A, \neg H_A^{c(i)} | A_H^{(i)}) = \frac{P_{PrAF}^S(I_A, A_H^{(i)}) P_{PrAF}^S(\neg H_A^{c(i)})}{P_{PrAF}^S(A_H^{(i)})}$  // Compute the likelihood
    conditioned on  $A_H^{(i)}$  (Equation 5.20)
25    $P_{PrAF}^S(I_A, \neg H_A | A_H) \leftarrow P_{PrAF}^S(I_A, \neg H_A^{c(i)} | A_H^{(i)})$  // Append the result
26
27 return  $P_{PrAF}^S(I_A, \neg H_A | A_H)$ 

```

Algorithm 5.2: Compute the likelihood using Equation 5.24.

Input : $A \text{ PrAF} = (A, P_A, R, P_R)$;
 A set of arguments with embedded hypotheses H_A ;
 A set of arguments containing information I_A ;
 A semantics S ;
 A number of trials N

Output: A list of $|H_A|$ likelihoods $P_{PrAF}^S(I_A, \neg H_A | A_H)$ for each argument with an embedded hypothesis $A_H^{(i)}$.

```

1  $A_H = [A_H^{(1)}, \dots, A_H^{(n)}] \leftarrow H_A$  // Convert the set  $H_A$  to an indexed array  $A_H$ 
2  $A_I = [A_I^{(1)}, \dots, A_I^{(n)}] \leftarrow I_A$  // Convert the set  $I_A$  to an indexed array  $A_I$ 
3  $\mathcal{C}_H \leftarrow \text{zeros}(\mathcal{C}_H, |A_H|)$  // Create an array of zeros, of size  $|A_H|$ , to count the
   number of times an argument containing a hypothesis was acceptable
4  $\mathcal{C}_I \leftarrow \text{zeros}(\mathcal{C}_I, |A_I|)$  // Create an array of zeros, of size  $|A_I|$ , to count the
   number of times each information argument was acceptable
5 for 1 to  $N$  do
6    $\mathcal{G}^I \leftarrow \text{INDUCE AF}(PrAF)$  // Algorithm C.1 in Appendix C.1
7   forall  $i \in A_H$  do
8      $\mathcal{C}_H^{(i)} = \text{SEMANTIC EVALUATION FUNCTION}(\mathcal{G}^I, A_H^{(i)}, S, \mathcal{C}_H^{(i)})$  // Algorithm C.2 in
       Appendix C.1
9   forall  $i \in A_I$  do
10     $\mathcal{C}_I^{(i)} = \text{SEMANTIC EVALUATION FUNCTION}(\mathcal{G}^I, A_I^{(i)}, S, \mathcal{C}_I^{(i)})$  // Algorithm C.2 in
      Appendix C.1
11  $P_{PrAF}^S(I_A) = 0$  // Set the probability of evaluated information arguments to
   zero
12 forall  $i \in A_I$  do
13   if  $P_{PrAF}^S(I_A) = 0$  then
14      $P_{PrAF}^S(I_A) = \frac{\mathcal{C}_I^{(i)}}{N}$  // Compute the probability of  $A_I^{(i)}$ , if  $P_{PrAF}^S(I_A)$  is equal
       to zero
15   else
16      $P_{PrAF}^S(I_A) = P_{PrAF}^S(I_A) \times \left(\frac{\mathcal{C}_I^{(i)}}{N}\right)$  // Compute the probability of  $A_I^{(i)}$  and
       multiply by  $P_{PrAF}^S(I_A)$ , if  $P_{PrAF}^S(I_A)$  is non-zero
17  $P_{PrAF}^S(I_A, \neg H_A | A_H) \leftarrow \text{zeros}[P_{PrAF}^S(I_A, \neg H_A | A_H), |A_H|]$  // Create an array of zeros, of
   size  $A_H$ , to store the likelihoods
18 for  $i \in A_H$  do
19    $P_{PrAF}^S(\neg H_A^{c(i)}) = 0$  // Set the probability of defeat of all other hypotheses
   to zero
20   forall  $j \in A_H$  do
21     if  $A_H^{(i)} \neq A_H^{(j)}$  then
22       if  $P_{PrAF}^S(\neg H_A^{c(i)}) = 0$  then
23          $P_{PrAF}^S(\neg H_A^{c(i)}) = 1 - \frac{\mathcal{C}_H^{(j)}}{N}$  // Compute the probability of defeat for
            $A_H^{(j)}$ , if the probability of defeat is equal to zero
24       else
25          $P_{PrAF}^S(\neg H_A^{c(i)}) = P_{PrAF}^S(\neg H_A^{c(i)}) \times \left(1 - \frac{\mathcal{C}_H^{(j)}}{N}\right)$  // Compute the probability
           of defeat for  $A_H^{(j)}$  and multiply by  $P_{PrAF}^S(\neg H_A^{c(i)})$ , if
            $P_{PrAF}^S(\neg H_A^{c(i)})$  is non-zero
26    $P_{PrAF}^S(I_A, \neg H_A^{c(i)} | A_H) = P_{PrAF}^S(I_A) P_{PrAF}^S(\neg H_A^{c(i)})$  // Compute the likelihood conditioned
   on  $A_H^{(i)}$  (Equation 5.24)
27    $P_{PrAF}^S(I_A, \neg H_A | A_H) \leftarrow P_{PrAF}^S(I_A, \neg H_A^{c(i)} | A_H)$  // Append the result
28 return  $P_{PrAF}^S(I_A, \neg H_A | A_H)$ 

```

Algorithm 5.3: Compute the likelihood using Equation 5.30.

Input : A $PrAF = (A, P_A, R, P_R)$;
 A set of arguments with embedded hypotheses H_A ;
 A semantics S ;
 A number of trials N

Output: A list of $|H_A|$ likelihoods $P_{PrAF}^S(I_A, \neg H_A | A_H)$ for each argument with an embedded hypothesis $A_H^{(i)}$.

```

1  $A_H = [A_H^{(1)}, \dots, A_H^{(n)}] \leftarrow H_A$  // Convert the set  $H_A$  to an indexed array  $A_H$ 
2  $\mathcal{C}_H \leftarrow \text{zeros}(\mathcal{C}_H, |A_H|)$  // Create an array of zeros, of size  $|A_H|$ , to count the
   number of times an argument containing a hypothesis was acceptable
3 for 1 to  $N$  do
4    $\mathcal{G}^I \leftarrow \text{INDUCE AF}(PrAF)$  // Algorithm C.1 in Appendix C.1
5   forall  $i \in A_H$  do
6      $\mathcal{C}_H^{(i)} = \text{SEMANTIC EVALUATION FUNCTION}(\mathcal{G}^I, A_H^{(i)}, S, \mathcal{C}_H^{(i)})$  // Algorithm C.2 in
       Appendix C.1
7  $P_{PrAF}^S(I_A, \neg H_A | A_H) \leftarrow \text{zeros}[P_{PrAF}^S(I_A, \neg H_A | A_H), |A_H|]$  // Create an array of zeros, of
   size  $A_H$ , to store the likelihoods
8 for  $i \in A_H$  do
9    $P_{PrAF}^S(\neg H_A^{c(i)}) = 0$  // Set the probability of defeat of all other hypotheses
   to zero
10  forall  $j \in A_H$  do
11    if  $A_H^{(i)} \neq A_H^{(j)}$  then
12      if  $P_{PrAF}^S(\neg H_A^{c(i)}) = 0$  then
13         $P_{PrAF}^S(\neg H_A^{c(i)}) = 1 - \frac{\mathcal{C}_H^{(j)}}{N}$  // Compute the probability of defeat for
           $A_H^{(j)}$ , if the probability of defeat is equal to zero
14      else
15         $P_{PrAF}^S(\neg H_A^{c(i)}) = P_{PrAF}^S(\neg H_A^{c(i)}) \times \left(1 - \frac{\mathcal{C}_H^{(j)}}{N}\right)$  // Compute the probability
          of defeat for  $A_H^{(j)}$  and multiply by  $P_{PrAF}^S(\neg H_A^{c(i)})$ , if
           $P_{PrAF}^S(\neg H_A^{c(i)})$  is non-zero
16   $P_{PrAF}^S(I_A, \neg H_A^{c(i)} | A_H^{(i)}) = P_{PrAF}^S(\neg H_A^{c(i)})$  // Compute the likelihood conditioned on  $A_H^{(i)}$ 
   (Equation 5.30)
17   $P_{PrAF}^S(I_A, \neg H_A | A_H) \leftarrow P_{PrAF}^S(I_A, \neg H_A^{c(i)} | A_H^{(i)})$  // Append the result
18 return  $P_{PrAF}^S(I_A, \neg H_A | A_H)$ 

```

Algorithm 5.4: An algorithm to compute Bayes' theorem using Equation 5.32.

Input : A set of arguments with embedded hypotheses H_A ;
 Prior probabilities of arguments containing a hypothesis $\forall A_H^{(i)} \in H_A, P(A_H^{(i)})$;
 A list of likelihoods conditioned on each hypothesis $P_{PrAF}^S(I_A, \neg H_A | A_H)$

Output: A posterior probability $P(A_H^{(i)} | I_A, \neg H_A^{c(i)})$ for the argument $A_H^{(i)}$ conditioned on the acceptability of information arguments and the defeat of all other hypotheses.

```

1  $A_H = [A_H^{(1)}, \dots, A_H^{(n)}] \leftarrow H_A$  // Convert the set  $H_A$  to an indexed array  $A_H$ 
2  $P(I_A, \neg H_A) = 0$  // Set the total probability equal to zero
3 forall  $j \in A_H$  do
4    $P(I_A, \neg H_A) = P(I_A, \neg H_A) + P_{PrAF}^S(I_A, \neg H_A^{c(j)} | A_H^{(j)}) P(A_H^{(j)})$  // Multiply the likelihood
   // conditioned on  $A_H^{(j)}$  by the human agent's  $j$ -th prior  $P(A_H^{(j)})$ 
5  $P(A_H^{(i)} | I_A, \neg H_A^{c(i)}) = \frac{P_{PrAF}^S(I_A, \neg H_A^{c(i)} | A_H^{(i)}) P(A_H^{(i)})}{P(I_A, \neg H_A)}$  // Compute Bayes' theorem (Equation 5.32)
6 return  $P(A_H^{(i)} | I_A, \neg H_A^{c(i)})$ 

```

5.4 Experimental Evaluation

To demonstrate how the proposed approach was able to refine probabilistic judgements about hypotheses using semantically evaluated information, Algorithms 5.1, 5.2, 5.3, and 5.4 were implemented in Python [232] and evaluated using a set of simple, abstract examples that contained two hypotheses, with some discussion pertaining to the outputs, in Section 5.4.1. Then, in Section 5.4.2, I present a result obtained, using Algorithm 5.3, from the Operation Fortitude-South argumentation framework, introduced in the beginning of Part II, in order to assess whether the proposed algorithms could be employed on real-world scenarios. The result from the Operation Fortitude-South scenario was computed using a high performance computing cluster¹, so I present some strong scaling speedup and weak scaling efficiency results for two parallel implementation of Algorithm 5.3 in Section 5.4.2.1.

While the results in presented Sections 5.4.1 and 5.4.2 focused on two arguments with embedded hypotheses only, Algorithms 5.1, 5.2, 5.3, and 5.4 can be run with more than two hypotheses. However, results for scenarios with more than two hypotheses were omitted from this thesis for the sake of brevity.

5.4.1 A Set of Abstract Examples

In this subsection, I present the results from running the algorithms in Section 5.3.7 on a set of abstract examples, which were assumed to be analogous to the types of

¹The high performance computing cluster employed was called Scafell Pike: <https://www.hartree.stfc.ac.uk/technologies/high-performance-computing/> (Accessed on: 02.10.2024).

frameworks that one might instantiate within an evolving intelligence scenario. I altered the number of information arguments that were in conflict with arguments that contained hypotheses, as well as changed the set of inducement probabilities, in order to understand how they affected the revised, posterior estimates.

Experimental Setup. All the results presented in this subsection were conducted on a Windows machine with an AMD Ryzen 9 5900X 12-Core 3.7GHz Central Processing Unit (CPU) and 32Gb of Random Access Memory (RAM). Each experiment employed complete extensions which were computed using a disjunctive logic programming system [131]. The number of trials was equal to ($N =$) 30,000 to ensure that posterior estimates could be compared, and the average time to complete execution was 870 ± 30 seconds.

I now describe the methodology pertaining to the instantiation of arguments which contained either hypotheses or information only, and the attacks emanating from them.

The instantiation of arguments containing hypotheses and their attacks. While intelligence analysts are required to make judgements about the prior probability of hypotheses, they may not be so confident in their judgements. Therefore, the probability of inducement for arguments with embedded hypotheses was assumed to be between zero and unity, such that

$$0 \leq P_A(A_H^{(i)}) \leq 1, \forall A_H^{(i)} \in H_A. \quad (5.33)$$

Due to the mutual exclusivity of hypotheses, symmetric attacks were instantiated between arguments containing hypotheses with an inducement probability of unity. This ensured that if more than one argument containing a hypothesis featured within an induced argument graph, then those arguments would attack each other. More formally,

$$\forall A_H^{(i)}, A_H^{(j)} \in H_A, P_R((A_H^{(i)}, A_H^{(j)})) = 1, \quad (5.34)$$

where $i \neq j$.

Finally, it was assumed that arguments containing hypotheses could not attack information arguments. This approach is in accordance with the ACH, as discussed in Section 2.1.5.2, where hypotheses cannot attack row entries.

The instantiation of information arguments and their attacks. The probability of inducement for information arguments was assumed to be the same as arguments with embedded hypotheses, such that

$$0 \leq P_A(A_I^{(i)}) \leq 1, \forall A_I^{(i)} \in I_A. \quad (5.35)$$

However, the method of attack instantiation for information arguments differed slightly from its hypothesis counterpart. It was assumed that information arguments could attack any other argument within the PrAF with a probability of inducement within the unit interval, such that

$$\forall A_I^{(i)} \in I_A, \text{ and } \forall a_j \in A, 0 \leq P_R\left((A_I^{(i)}, a_j)\right) \leq 1. \quad (5.36)$$

I begin this experimental evaluation by considering scenarios that possessed two arguments containing hypotheses, with a topological symmetry between them, in order to understand how both altering the analyst's choice of prior and a PrAF affected posterior estimates. To ensure that Bayes' theorem could be successfully computed for two hypotheses, priors were assumed to have the following relationship:

$$P(A_H^{(1)}) = 1 - P(A_H^{(2)}). \quad (5.37)$$

5.4.1.1 Hypotheses and No Information

I begin with the simplest experiments, which were conducted on a probabilistic framework with two arguments containing hypotheses and no information, as shown in Figure 5.2a. The inducement probabilities for both arguments was set to 0.5 to represent a lack of confidence in the analyst's prior probabilistic judgements about both hypotheses. The posterior estimates presented in Figures 5.2b, 5.2c, and 5.2d remain approximately equal to the analyst's set of priors because both hypotheses were found to be justified roughly the same number of times across N induced graphs.

Now suppose the human agent was less confident in the prior probability of, say, the second argument $A_H^{(2)}$ containing a hypothesis, as shown in Figure 5.3a. One would expect the posterior estimates for the argument $A_H^{(1)}$ to be greater than $A_H^{(2)}$ for all equivalent priors, and the results presented in Figures 5.3b, 5.3c, and 5.3d confirmed this.

5.4.1.2 Hypotheses and Information

While Figures 5.2 and 5.3 show that the proposed technique is promising for scenarios with only two arguments with embedded hypotheses and no information arguments, in reality, intelligence analysts must reason with all the available information in order to evaluate the probability of hypotheses. As such, the remainder of results presented within this subsection employed PrAFs which contained two or more information arguments. The inducement probabilities of information arguments and their attacks were altered to understand the impact that the acceptability of the information had on the

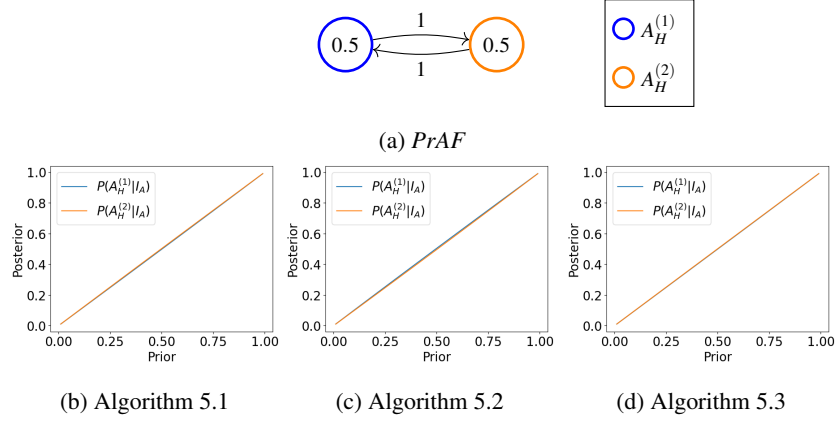


Figure 5.2: (a) A *PrAF* with two equally uncertain arguments containing hypotheses, and the resulting posterior estimates computed using the prior relationship in Equation 5.37 and the likelihood output by (b) Algorithm 5.1, (c) Algorithm 5.2, and (d) Algorithm 5.3, where $N = 30,000$.

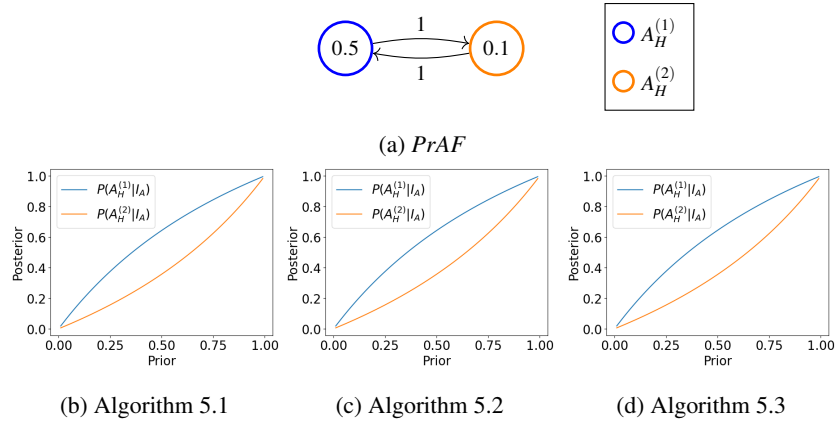


Figure 5.3: (a) A *PrAF* with two arguments with embedded hypotheses where $A_H^{(1)}$ was more certain than $A_H^{(2)}$, and the posterior estimates computed using the prior relationship in Equation 5.37 and the likelihood output by (b) Algorithm 5.1, (c) Algorithm 5.2, and (d) Algorithm 5.3, where $N = 30,000$.

posterior estimates of the hypotheses in question. Finally, all arguments with embedded hypotheses were assigned an inducement probability of unity so that they were always instantiated in the set of N induced argument frameworks.

To start, the first experiment with information featured two information arguments with, again, a topological symmetry between them, as displayed in Figure 5.4a. The posterior estimates for both hypotheses were approximately equal to all the human agent's priors, as demonstrated in Figures 5.4b, 5.4c, and 5.4d. This result is intuitive due to the symmetry in the graph, which arises from the two uncertain information arguments that attack both hypotheses with an equal uncertainty, and the symmetric conflicts between arguments containing hypotheses.

The second experiment containing information considered whether altering the uncertainty of information arguments alone affected posterior estimates. The inducement probability of the information argument that attacked the first hypothesis was increased from 0.5 to 0.9, as shown in Figure 5.5a, which, in turn, meant that the posterior estimates for the argument $A_H^{(2)}$ were greater than $A_H^{(1)}$ for all equivalent priors, as shown in Figures 5.5b, 5.5c, and 5.5d. An interesting point to note here is the difference between the graph plotted in Figure 5.5b and the ones in Figures 5.5c and 5.5d, which are the same. The joint probability of the information is a constant which is cancelled out

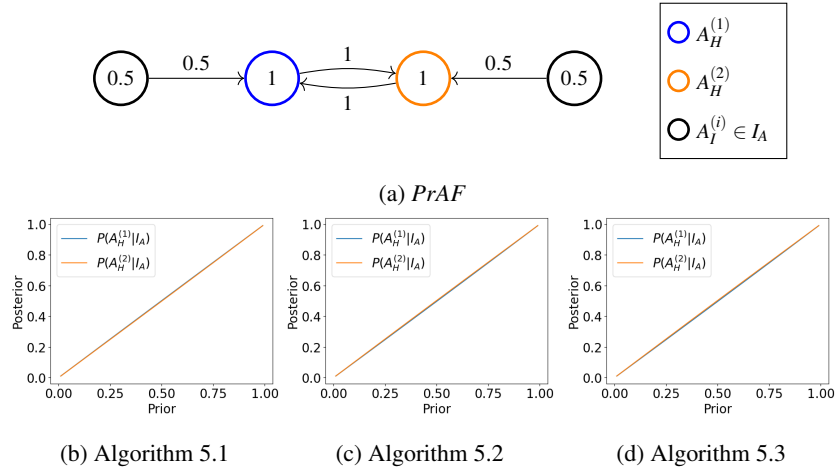


Figure 5.4: (a) A $PrAF$ with two certain arguments containing hypotheses and two uncertain information arguments who attack both hypotheses with uncertainty. The posterior estimates for the $PrAF$ computed using the prior relationship in Equation 5.37 and the likelihood output by (b) Algorithm 5.1, (c) Algorithm 5.2, and (d) Algorithm 5.3, where $N = 30,000$.

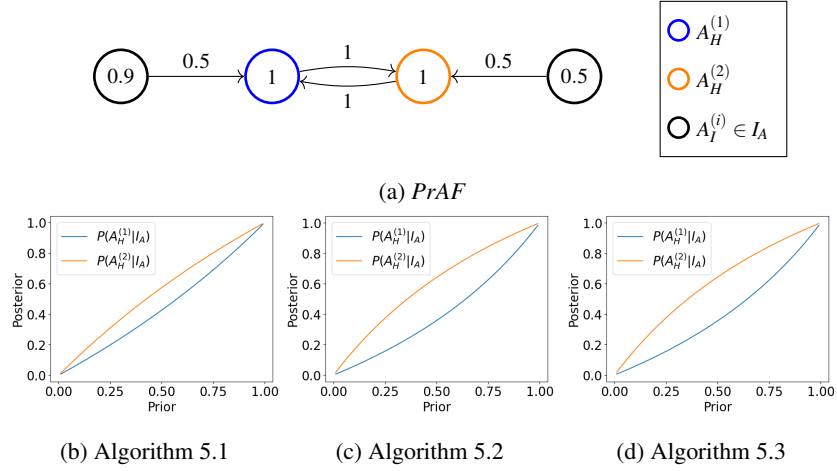


Figure 5.5: (a) A *PrAF* with two hypotheses and two information arguments, with one is more certain than the other. The posterior estimates for the *PrAF* computed using the prior relationship in Equation 5.37 and the likelihood output by (b) Algorithm 5.1, (c) Algorithm 5.2, and (d) Algorithm 5.3, where $N = 30,000$.

when computing Bayes' theorem in the case of Algorithm 5.2 and that same probability does not feature in any of the outputs from Algorithm 5.3. The results from Algorithm 5.1 do, however, incorporate the acceptability of information and a hypothesis within the joint probability in the numerator of Equation 5.20, which is why there is a smaller difference between posterior estimates for both hypotheses in Figure 5.5b.

The third experiment on PrAFs with hypotheses and information assessed the impact of changing the uncertainty of attacks from information arguments to arguments containing hypotheses. To conduct this test, all information arguments were assigned a probability of inducement of 0.5. The inducement probability of the information argument which attacked the first hypothesis was increased from 0.5 to 0.9, as per Figure 5.6a. Looking to Figures 5.6b, 5.6c, and 5.6d, the revised probability estimate for the argument containing the second hypothesis $A_H^{(2)}$ was greater than $A_H^{(1)}$ for all equivalent priors, which, again, is intuitive. Although both information arguments feature, on average, in the same number of induced argument graphs, the first $A_H^{(1)}$ and second $A_H^{(2)}$ arguments containing hypotheses were attacked by information arguments in 45% and 25% of the induced graphs, respectively.

Increasing the inducement probability of attacks to a hypothesis produces a more significant change in posterior estimates when compared to increasing the inducement probability of information arguments which attack hypotheses, in the case of Algorithm

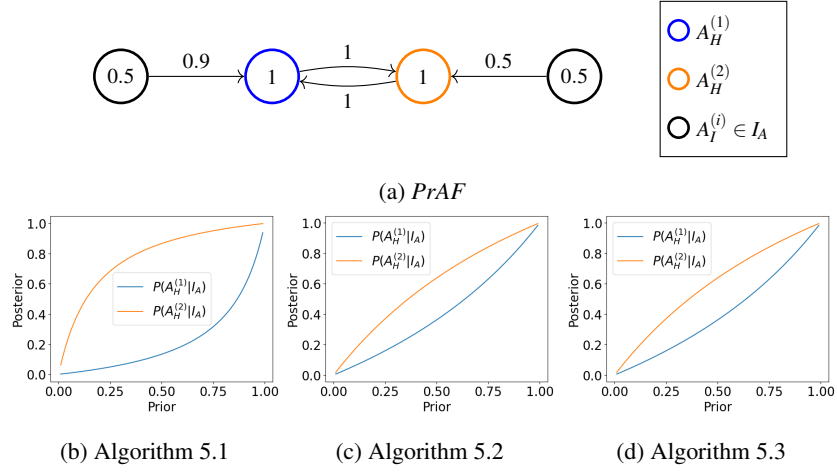


Figure 5.6: (a) A *PrAF* with two hypotheses and two uncertain information arguments, where there is more confidence in the attack from one of the information arguments to $A_H^{(1)}$. Posterior estimates for the *PrAF* computed using the prior relationship in Equation 5.37 and the likelihood output by (b) Algorithm 5.1, (c) Algorithm 5.2, and (d) Algorithm 5.3, where $N = 30,000$.

5.1. Compare the graphs in Figures 5.5a and 5.6a. While the posterior probability of the second argument containing a hypothesis is greater than the first in both charts, the change in the revised probabilities is much greater in Figure 5.6a, even though the average number of times that the hypothesis was attacked was approximately the same. This result can most likely be attributed to the assumption about dependence, in Equation 5.20, between information and a hypothesis. This result is slightly less instinctive but it still agrees with intuition because, in Figure 5.5a, there will be more extensions of induced frameworks that include the first argument containing the hypothesis $A_H^{(1)}$ and the information arguments that attacks it. Whereas, in Figure 5.6a, the same information arguments will be in fewer induced argument frameworks but, when it is included, it will attack the first hypothesis 90% of the time; thus, both arguments will feature in a smaller number of extensions over the set of induced graphs, and the joint probability $P_{PrAF}^S(I_A, A_H^{(1)})$ will be smaller in Equation 5.20 as a result.

The fourth experiment containing hypotheses and information studied increasing the inducement probability of both an information argument and its attack to the first argument with an embedded hypothesis, as shown in Figure 5.7a. As expected, the posterior estimates for the second argument containing a hypothesis were greater than the first for all equivalent priors, as demonstrated in Figures 5.7b, 5.7c, and 5.7d.

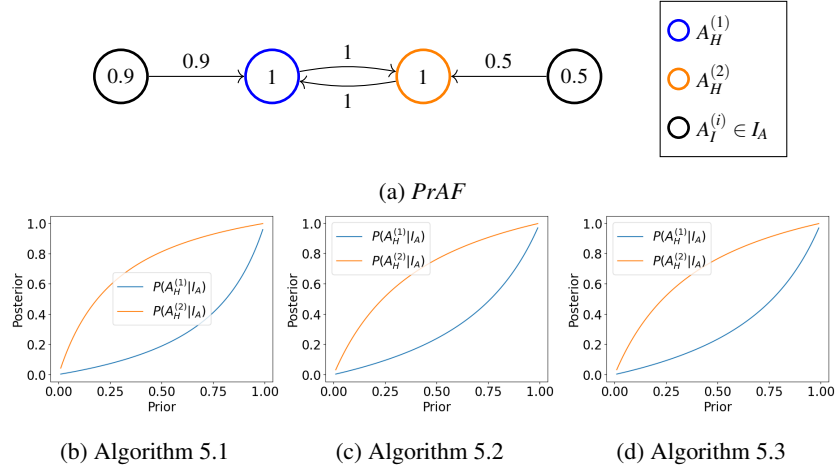


Figure 5.7: (a) A *PrAF* with two hypotheses and two information arguments, where one information argument and its attack to $A_H^{(1)}$ had a greater certainty. The posterior estimates computed using the prior relationship in Equation 5.37 and the likelihood output by (b) Algorithm 5.1, (c) Algorithm 5.2, and (d) Algorithm 5.3, where $N = 30,000$.

Asymmetrical topology between arguments containing hypotheses. Thus far, I have considered *PrAFs* that possessed a topological symmetry between arguments containing hypotheses. However, many of the argumentation graphs instantiated using real-world data will be asymmetrical. To understand whether the proposed approach could produce correct, reliable, and intuitive results in such settings, I present a framework that contained three information arguments, as per Figure 5.8a. In this example, the first argument containing a hypothesis $A_H^{(1)}$ is defended by two information arguments: the first defender has an inducement probability equal to 0.5 and attacks the second hypothesis $A_H^{(2)}$; and the second has a probability of inducement equal to unity and is in symmetric conflict with another information argument that attacks $A_H^{(1)}$. Recall from Section 5.3.3 that information arguments with a probability of inducement equal to unity can be thought of as evidence because there is no uncertainty associated with its existence. For the example displayed in Figure 5.8a, one would expect the posteriors of the argument $A_H^{(1)}$ to be greater than the second $A_H^{(2)}$ for all equivalent priors because an item of evidence comes to the first hypothesis' defence, and the results in Figures 5.8b, 5.8c, and 5.8d affirmed this notion.

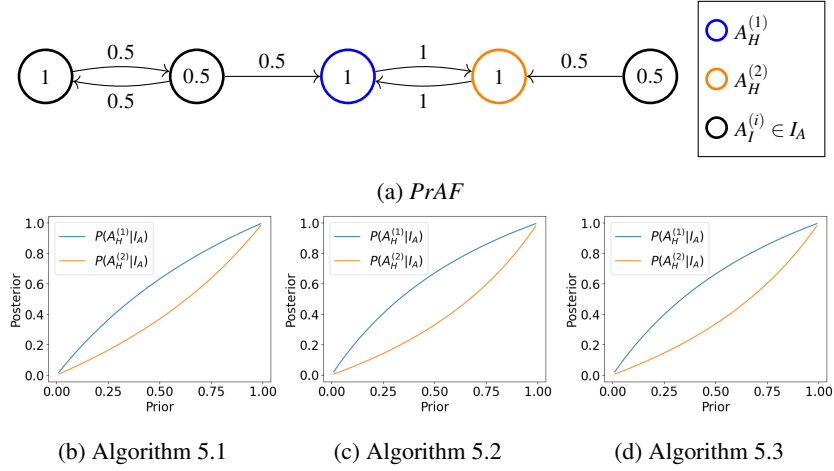


Figure 5.8: (a) A *PrAF* with an asymmetry between $A_H^{(1)}$ and $A_H^{(2)}$, arising from the instantiation of three information arguments, where one of the information arguments had an inducement probability equal to unity. The posterior estimates computed using the prior relationship in Equation 5.37 and the likelihood output by (b) Algorithm 5.1, (c) Algorithm 5.2, and (d) Algorithm 5.3, where $N = 30,000$.

The ratio of the likelihoods and the floating point underflow problem. The likelihoods conditioned on either the first $A_H^{(1)}$ or second $A_H^{(2)}$ argument containing a hypothesis, as well as the ratio between those likelihoods, i.e. $\frac{A_H^{(1)}}{A_H^{(2)}}$, for Figures 5.2 to 5.8 are presented in Table 5.1. Consider first the ratio of the likelihoods for Algorithms 5.1, 5.2, and 5.3. If the ratio was above (below) unity, then the likelihood conditioned on the argument $A_H^{(1)}$ ($A_H^{(2)}$) was greater than $A_H^{(2)}$ ($A_H^{(1)}$). When the ratio approximately equalled unity, both likelihoods conditioned on the arguments $A_H^{(1)}$ and $A_H^{(2)}$ were approximately equal. The ratios for all three algorithms were, roughly speaking, either above, below, or equal to unity for the same figures in Table 5.1, which confirms that the independence assumptions made in Algorithms 5.2 and 5.3 produced sensible outputs in line with the results from Algorithm 5.1. While the ratios of likelihoods were approximately equal across all experiments for Algorithms 5.2 and 5.3, they were only roughly equal in Figures 5.2, 5.3, and 5.4 for all three algorithms. All sets of ratios from Algorithms 5.2 and 5.3 were greater than Algorithm 5.1 for the experiments in Figures 5.5 to 5.8 because of the incorporation of the information and a hypothesis in the joint probability within the numerator of Equation 5.20, implemented in Algorithm 5.1. However, the dependence between evaluated information and hypotheses in Al-

gorithm 5.1 meant that the likelihoods tended towards zero with the inclusion of more information arguments in PrAFs.

Now consider the floating point underflow condition and the solution to it provided by Algorithm 5.3 and Equation 5.30, as discussed in Section 5.3.5.3. In Table 5.1, the likelihoods become increasingly small for the outputs from Algorithms 5.1 and 5.2 in Figures 5.5 to 5.8, when compared to the results from Algorithm 5.3. In fact, after the addition of a third information argument, in Figure 5.8, the likelihoods from Algorithms 5.1 and 5.2 are the smallest. Algorithm 5.3, however, is able to compute an approximately equal ratio of the likelihood, when compared with Algorithm 5.2, whilst also ensuring that the likelihoods do not tend towards zero.

Thus, Table 5.1 confirms that the results from Algorithms 5.2 and 5.3, the algorithms which assume independence, provide a good approximation of the set of likelihoods and produce results in agreement with Algorithm 5.1. Moreover, Algorithm 5.3 should be capable of computing likelihoods for real-world scenarios, with much larger PrAFs, because its estimates do not include the probability of information arguments which could cause a floating point underflow condition.

Table 5.1: The likelihood conditioned on either $A_H^{(1)}$ or $A_H^{(2)}$ and the ratio between them, i.e. $\frac{A_H^{(1)}}{A_H^{(2)}}$, for the results presented in Figures 5.2 to 5.8.

Figure	Likelihood Conditioned Using								
	Algorithm 5.1			Algorithm 5.2			Algorithm 5.3		
	$A_H^{(1)}$	$A_H^{(2)}$	$\frac{A_H^{(1)}}{A_H^{(2)}}$	$A_H^{(1)}$	$A_H^{(2)}$	$\frac{A_H^{(1)}}{A_H^{(2)}}$	$A_H^{(1)}$	$A_H^{(2)}$	$\frac{A_H^{(1)}}{A_H^{(2)}}$
5.2	0.4963	0.5011	0.9904	0.5038	0.4955	1.0166	0.4971	0.4992	0.9957
5.3	0.9010	0.5003	1.8009	0.8993	0.4996	1.7999	0.9020	0.4976	1.8128
5.4	0.0422	0.04191	1.007	0.0615	0.0622	0.9882	0.2482	0.2516	0.9865
5.5	0.1007	0.1357	0.7428	0.1124	0.2013	0.5583	0.2498	0.4516	0.5530
5.6	0.0113	0.0737	0.1539	0.0638	0.1123	0.5683	0.2535	0.4468	0.5673
5.7	0.0565	0.2437	0.2318	0.1148	0.3671	0.3126	0.2479	0.8101	0.3060
5.8	0.0091	0.0053	1.7109	0.0420	0.0207	2.0349	0.2507	0.1268	1.9766

Posterior Estimates of Hypotheses Over Time. Although producing a refined probabilistic estimate about the probability of hypotheses is important, intelligence analysts may be required to understand how such estimates have evolved over time. To show that the proposed algorithms are capable of producing posteriors throughout time, Fig-

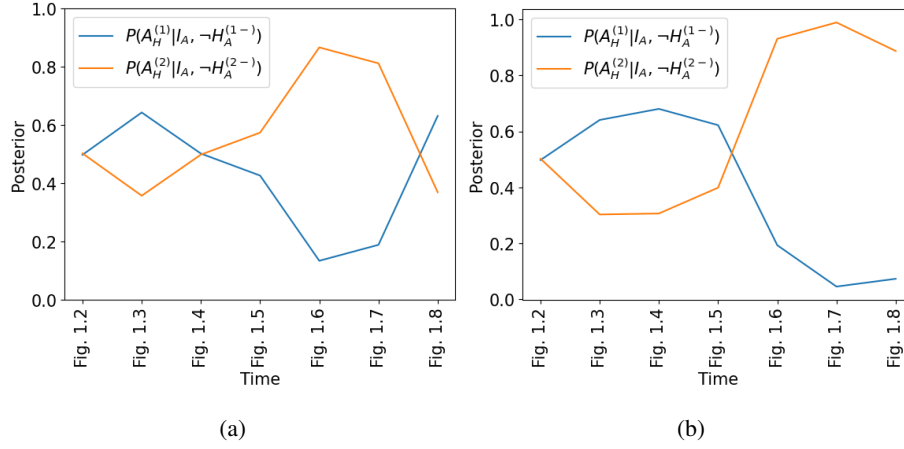


Figure 5.9: Posteriors overtime, where the priors were equal to (a) 0.5 across all time steps – i.e., *Fig. 1.2* to *Fig. 1.8* – and (b) 0.5 in the first time step – i.e. *Fig. 1.2* – and then priors were equal to the previous posterior estimate, across all the remaining time steps – i.e., *Fig. 1.3* to *Fig. 1.8*.

ure 5.9 displays posterior estimates computed using two different priors and the likelihoods from the results in Figures 5.2 to 5.8, which was assumed to capture how likelihoods might change over time.

5.4.2 Operation Fortitude-South

A high performance computing cluster² and two parallel implementations of Algorithm 5.3 were required to obtain results for the Operation Fortitude-South argumentation framework. The method of parallelisation, strong scaling speedup, and weak scaling efficiency is discussed in more detail in Section 5.4.2.1.

To ensure that results could be computed from the Operation Fortitude-South argumentation framework, the arguments containing hypotheses and the inducement probability of each argument and relation needed to be defined. Recall from the introduction to Part II that the two hypotheses were:

- $Conc(A_H^{(Nor)})$: The main invasion should land in Normandy;
- $Conc(A_H^{(PdC)})$: The main invasion should land in Pas de Calais.

²The high performance computing cluster employed was called Scafell Pike: <https://www.hartree.stfc.ac.uk/technologies/high-performance-computing/> (Accessed on: 02.10.2024).

Arguments containing hypotheses and attacks between them were assigned an inducement probability of unity so that both hypotheses featured in all induced graphs. All other arguments were assumed to be information arguments, as per Definition 5.8. The probability of inducement was set equal to 0.5 for all information arguments and attacks emanating from them. The semantics of choice was complete semantics due to the speed at which complete extensions can be computed on larger frameworks. The number of trials was equal to $(N =) 520 \times 10^6$. Here, the reader should note that the reason for the aforementioned number of trials N was because the counts output from the speedup and efficiency tests were combined to provide a better approximation. The likelihoods for the two arguments with embedded hypotheses for the Operation Fortitude-South argumentation framework are presented in Table 5.2. The posterior estimates are presented in Figure 5.10 and were computed using the same prior relationship defined in Equation 5.37 in Section 5.4.1.

Table 5.2: The likelihoods for the Normandy and Pas de Calais hypotheses and the number of trials employed in the approximation.

	Likelihood	Number of Trials, N
$P(I_A, H_A^{(Nor-)} A_H^{(Nor)})$	0.2499	520×10^6
$P(I_A, H_A^{(PdC-)} A_H^{(PdC)})$	0.2500	

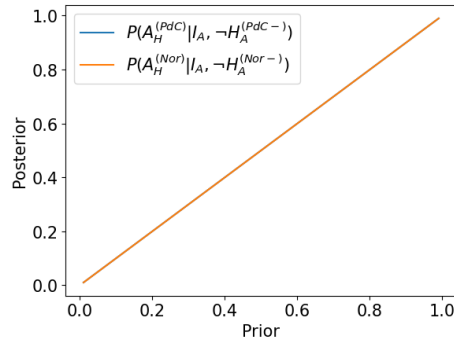


Figure 5.10: The posterior estimates for the Operation Fortitude-South argumentation framework. Both arguments containing hypotheses and attacks between them had a probability of inducement of unity. All other arguments were assumed to be information arguments, where information arguments and their attacks had inducement probabilities equal to 0.5. The number of trials was $(N =) 520 \times 10^6$.

Looking to the graph of posterior estimates in Figure 5.10. The posterior probabilities were approximately equal for all equivalent priors because there was a negligible difference between both the likelihoods presented in Table 5.2. As I describe in Appendix B, Operation Fortitude-South was so successful that Hitler believed the Normandy landings were a diversionary attack for an entire month after D-Day occurred [66, p. 15]. Although the Axis' command had presumably conducted their own analysis, using a trusted set of sources and analysts, and deemed Pas de Calais to be the most likely invasion site, as we now know, this assessment was in fact wrong. Had the Axis' command possessed a tool that was able to refine their prior probabilistic judgements on all the available information and the defeat of all other hypotheses, they may have arrived at the correct conclusion, potentially changing the outcome of World War Two.

One further, interesting point to note about the results in Table 5.2 and Figure 5.10 is that they do not encompass all the likelihoods that could be approximated and used in the computation of the posterior probabilities of the two hypotheses. The reason for this becomes apparent when one studies the conclusions of the arguments, namely Arg 34 to Arg 52, in Appendix B.4. The arguments and sub-arguments used to test the proposal made in this chapter were instantiated using the ASPIC⁺ framework which, when translated from the AIF to ASPIC⁺ to a Dung argumentation framework, can result in a number of arguments with the same conclusion. Although those arguments possess the same hypothesis embedded as a conclusion, they are supported by different sub-arguments and therefore, they are different arguments. Future work should employ the algorithms proposed in this chapter and consider how to combine the likelihoods for arguments with the same hypothesis embedded within its conclusion to produce a global posterior estimate for a hypothesis when using real-world data. However, it should be noted that combining such estimates should be done with care because of the juxtaposition between the rationale underpinning the field of probability and the extensions or labellings output by a semantics within the computational argumentation literature. For instance, in argumentation, one might say that a conclusion is stronger if one is able to instantiate more arguments with that proposition and, as a result, more of those arguments (with that same conclusion) might be found to be acceptable. On the contrary, if one were using probability theory to simply multiply likelihoods or posteriors, then stronger arguments (with more reasons to believe them) would be penalised for being many in number. One sensible approach might be to calculate likelihoods and posteriors for all arguments with embedded hypotheses and provide all those results to the intelligence analyst to avoid information loss through aggregation.

5.4.2.1 Parallel Implementations of Algorithm 5.3, Strong Scaling Speedup, and Weak Scaling Efficiency

Algorithm 5.3 is, what is referred to as, *embarrassingly parallel* because PrAFs employ a Monte Carlo approximation approach in order to compute the probability of sets of arguments. As such, the instantiation of N induced argument graphs was independent. This meant that the `for` loop, in Lines 3 to 6 of Algorithm 5.3, could be spread across a given number of compute nodes, using distributed memory parallelism, and the resulting counts gathered and summed up afterwards.

Algorithms 5.5 and 5.6 are adaptations of Algorithm 5.3 which employed a divide-and-conquer approach to parallelisation to study strong and weak scaling, respectively, as well as to obtain results for the Operation Fortitude-South example. To understand whether Algorithms 5.5 and 5.6 were in fact faster than Algorithm 5.3, its sequential counterpart, a series of experiments that measured strong scaling speedup and weak scaling efficiency were conducted.

Experimental Setup. The results presented in this subsection were obtained on a cluster of 846 2×128 Intel Xeon Gold E5-6142³ CPUs, each with 16 cores, a clock speed of 2.5GHz (up to 3.7GHz), and 192Gb of RAM. Algorithms 5.5 and 5.6 were implemented in Python [232] and parallelised using *mpi4py* [62], a widely employed Python package for parallelisation of codes using distributed memory architecture. An example of the distributed memory architecture employed in the execution of Algorithms 5.5 and 5.6 is presented in Figure C.1 in Appendix C.2. The experiments which measured strong scaling speedup and weak scaling efficiency employed a power-of-two number – i.e. 1, 2, 4, ... 256 – of CPUs from the cluster. An interesting point to note is that while high performance computing clusters are capable of executing hybrid distributed-shared memory parallelism, this approach was not implemented and is left for future work.

Strong scaling speedup. In [6], Amdahl argued that the speedup of an algorithm is limited by the fraction of the code which cannot be parallelised and must be run in serial. Strong scaling speedup is governed by Amdahl’s law such that it measures how the speedup changes with a varying number of processors for a fixed problem size which, in the context of this chapter, is the number of trials N . Algorithm 5.5 employs distributed memory parallelism to compute the likelihood in Equation 5.30

³<https://ark.intel.com/content/www/us/en/ark/products/120487/intel-xeon-gold-6142-processor-22m-cache-2-60-ghz.html> (Accessed on: 02.10.2024)

for a constant number of trials.

Algorithm 5.5: A parallel implementation of Algorithm 5.3 that employs a distributed memory architecture and a fixed problem size N .

Input : A $PrAF = (A, P_A, R, P_R)$;
 A set of arguments with embedded hypotheses H_A ;
 A semantics S ;
 A number of processors \mathcal{P} ;
 A number of trials N .

Output: A list of $|H_A|$ likelihoods $P_{PrAF}^S(I_A, \neg H_A | A_H)$ for each argument with an embedded hypothesis $A_H^{(i)}$.

```

1 do in parallel
2    $A_H = [A_H^{(1)}, \dots, A_H^{(n)}] \leftarrow H_A$            // Convert the set  $H_A$  to an indexed array  $A_H$ 
3    $\mathcal{C}_H \leftarrow \text{zeros}(\mathcal{C}_H, |A_H|)$            // Create an array of zeros, of size  $|A_H|$ , to count
   the number of times an argument containing a hypothesis was
   acceptable
4   for 1 to  $\frac{N}{\mathcal{P}}$  do
5      $\mathcal{G}^I \leftarrow \text{INDUCE AF}(PrAF)$            // Algorithm C.1 in Appendix C.1
6     forall  $i \in A_H$  do
7        $\mathcal{C}_H^{(i)} = \text{SEMANTIC EVALUATION FUNCTION}(\mathcal{G}^I, A_H^{(i)}, S, \mathcal{C}_H^{(i)})$  // Algorithm C.2 in
       Appendix C.1
8   Send  $\mathcal{C}_H$  to main processor  $\mathcal{P}_1$ 
9 if Processor is  $\mathcal{P}_1$  then
10   Receive  $\mathcal{C}_H$  and sum counts belonging to the same argument containing a hypothesis
11   Continue with Lines 7 to 18 in Algorithm 5.3 to compute the likelihood conditioned on each argument
   containing a hypothesis

```

Strong scaling speedup is calculated as

$$\text{Strong scaling speedup} = \frac{t(1)}{t(\mathcal{P})} \quad (5.38)$$

where $t(1)$ and $t(\mathcal{P})$ are the execution times for, respectively, one and \mathcal{P} processors, for a fixed problem size of N trials.

In a perfect world, the speedup and execution time should, respectively, be proportional and inversely proportional to the number of processors employed in the parallelised algorithm, such that

$$\text{Ideal speedup} = \mathcal{P} \quad \text{and} \quad \text{ideal execution time} = \frac{t(1)}{\mathcal{P}}, \quad (5.39)$$

where \mathcal{P} is the number of processors.

However, in real applications, such ideal speedups and execution times are rarely possible because of the overhead of parallelisation which arises from constraints, such as the cost of communication between nodes across a network.

The execution time and strong scaling speedup results for experiments on Algorithm 5.5 with a fixed problem size are presented in Figure 5.11 and Table 5.3. The execution time decreased and speedup increased with an increasing number of processors. The results confirmed good scaling when $\mathcal{P} = 1, 2, 4, 8$, and 16, acceptable scaling when $\mathcal{P} = 32$ and 64, and poor scaling when $\mathcal{P} = 128$ and 256. This was most likely caused by the decreased number of trials that each processor was required to complete – i.e. $\frac{N}{128}$ or $\frac{N}{256}$ per processor – which, in turn, decreased each processor’s workload and execution time, whilst also increasing the communication cost across the network, causing a bottleneck. It’s worth mentioning that the number of trials – i.e., $N = 1 \times 10^6$ – was quite small and increasing the fixed problem size by a factor of, for example, ten, twenty, etc, would increase the strong scalability of the experiments with 128 and 256 processors.

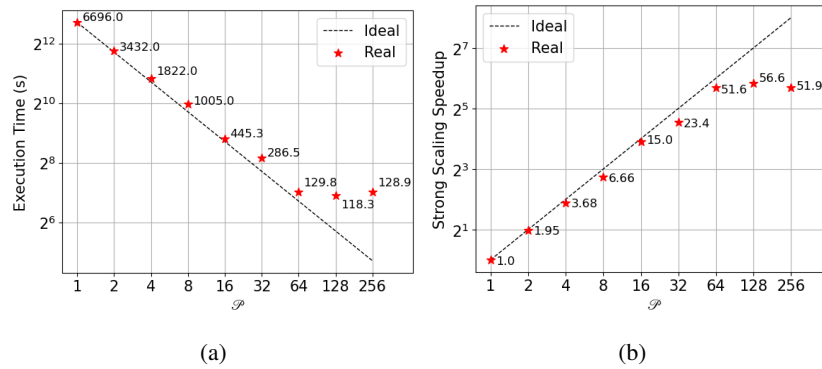


Figure 5.11: (a) The execution time and (b) strong scaling speedup for Algorithm 5.5 for an increasing number of processors, where the number of trials was $N = 1 \times 10^6$.

Weak scaling efficiency. While Amdahl’s law provides the upper bound of speedup for a fixed problem size, the benefit of parallelising and executing algorithms on high performance computing clusters is that the size of a problem can be increased. As Gustafson noted in [92], the size of a problem scales with the amount of available resources. Weak scaling measures parallel performance when the problem size increases. Both the number of processors and the problem size are increased, whilst also ensuring that each processor has a constant workload. Algorithm 5.6, again, employs distributed memory parallelism to compute the likelihood in Equation 5.30 for an increasing problem size, where the number of trials is proportional to the number of processors.

Weak scaling efficiency is computed in a similar manner to the strong scaling

Table 5.3: The execution time and strong scaling speedup for a fixed problem size.

Processors \mathcal{P}	Execution Time (s)	Speedup	Number of Trials N
1	6,696	1.00	1×10^6
2	3,432	1.95	
4	1,822	3.68	
8	1,005	6.66	
16	445.3	15.0	
32	286.5	23.4	
64	129.8	51.6	
128	118.3	56.6	
256	128.9	51.9	

speedup, such that

$$\text{Weak scaling efficiency} = \frac{t(1)}{t(\mathcal{P})}, \quad (5.40)$$

where the efficiency is a real number within the unit interval.

As the workload is held constant across \mathcal{P} processors, the ideal efficiency and execution time should, respectively, be equal to unity and the time taken for one processor to complete N trials, as shown in Equation 5.41.

$$\text{Ideal efficiency} = 1 \quad \text{and} \quad \text{ideal execution time} = t(1). \quad (5.41)$$

The weak scaling execution time and efficiency for experiments on Algorithm 5.6 with an increasing problem size are presented in Figure 5.12 and Table 5.4. The execution time increased and efficiency decreased with an increasing number of processors and trials. The results displayed good efficiency when $\mathcal{P} = 1, 2, 4, 8, 16$ and 32, acceptable efficiency when $\mathcal{P} = 64$, and poor efficiency when $\mathcal{P} = 128$ and 256. The reason for these results is two-fold. First, as previously mentioned, increasing the number of processors, increased the communication costs across the network, even for the almost completely parallel implementation presented in Algorithm 5.6. Second, increasing the number of trials N as a function of the number of processors \mathcal{P} , increased the chance of instantiating a more complex induced argument graph which was computationally harder to evaluate using a semantics. In Algorithm 5.6, work was distributed evenly across the set of processors. Each processor was assigned the same workload which meant that if a processor instantiated a number of more challenging induced frame-

works, then the main processor \mathcal{P}_1 would be required to wait until that processor completed its set of tasks. One solution to this problem would be to implement dynamic task allocation, where nodes are only assigned more work once they have completed tasks. However, this would, again, only increase the amount of communication during runtime and increase execution times. The benefit of employing either static or dynamic work distribution in Algorithms 5.5 and 5.6 is a task left for future work.

Algorithm 5.6: A parallel implementation of Algorithm 5.3 that employs a distributed memory architecture and an increasing problem size $N \times \mathcal{P}$.

Input : A $PrAF = (A, P_A, R, P_R)$;
 A set of arguments with embedded hypotheses H_A ;
 A semantics S ;
 A number of processors \mathcal{P} ;
 A number of trials N .

Output: A list of $|H_A|$ likelihoods $P_{PrAF}^S(I_A, \neg H_A | A_H)$ for each argument with an embedded hypothesis $A_H^{(i)}$.

```

1 do in parallel
2    $A_H = [A_H^{(1)}, \dots, A_H^{(n)}] \leftarrow H_A$            // Convert the set  $H_A$  to an indexed array  $A_H$ 
3    $\mathcal{C}_H \leftarrow \text{zeros}(\mathcal{C}_H, |A_H|)$            // Create an array of zeros, of size  $|A_H|$ , to count
   the number of times an argument containing a hypothesis was
   acceptable
4   for 1 to  $N$  do
5      $\mathcal{G}^I \leftarrow \text{INDUCE AF}(PrAF)$            // Algorithm C.1 in Appendix C.1
6     forall  $i \in A_H$  do
7        $\mathcal{C}_H^{(i)} = \text{SEMANTIC EVALUATION FUNCTION}(\mathcal{G}^I, A_H^{(i)}, S, \mathcal{C}_H^{(i)})$  // Algorithm C.2 in
       Appendix C.1
8   Send  $\mathcal{C}_H$  to main processor  $\mathcal{P}_1$ 
9 if Processor is  $\mathcal{P}_1$  then
10    $N = N \times \mathcal{P}$            // Calculate the total number of trials
11   Receive  $\mathcal{C}_H$  and sum counts belonging to the same argument containing a hypothesis
12   Continue with Lines 7 to 18 in Algorithm 5.3 to compute the likelihood conditioned on each argument
   containing a hypothesis

```

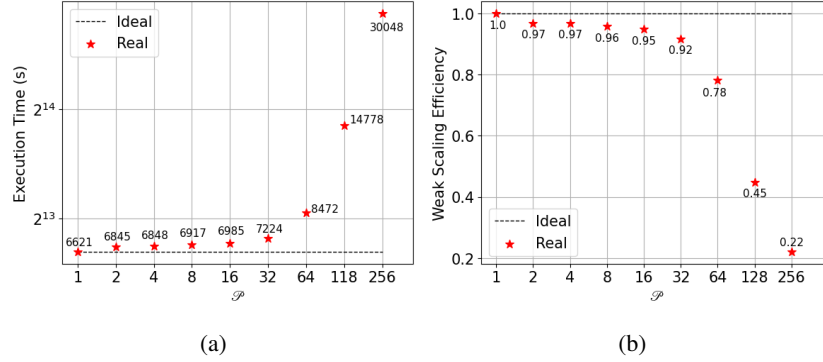


Figure 5.12: (a) The execution time and (b) weak scaling efficiency for Algorithm 5.6 for an increasing number of processors, where the number of trials was $N = \mathcal{P} \times 10^6$.

Table 5.4: The execution time and weak scaling speedup for an increasing problem size.

Processors \mathcal{P}	Execution Time (s)	Efficiency	Number of Trials N
1	6,621	1.00	1×10^6
2	6,845	0.97	2×10^6
4	6,848	0.97	4×10^6
8	6,917	0.96	8×10^6
16	6,985	0.95	16×10^6
32	7,224	0.92	32×10^6
64	8,472	0.78	64×10^6
128	14,778	0.45	128×10^6
256	30,048	0.22	256×10^6

5.5 Summary and Future Work

In this chapter, I have taken the ASPIC⁺ framework, previously outlined in Section 2.3.2.1, and Pherson and Heuer Jr.’s definition of a hypothesis, introduced in Chapter 1, and defined arguments with embedded hypotheses and arguments containing information only, in terms of structured argumentation. I then combined those definitions with PrAFs and a human agent’s prior beliefs, using Bayes’ theorem, in order to provide a revised probability that incorporates the outputs from an intelligence analysis

and the acceptability of the available information and defeat of all other hypotheses. Four algorithms were proposed. The first three algorithms compute the likelihood of the acceptability of information and the defeat of all other hypotheses. However, they do this using different assumptions about the dependence between evaluated information arguments and the argument which contains the hypothesis of interest. The fourth algorithm takes the output from the first three and computes the probability of a hypothesis given the information and defeat of all other hypotheses. I then provided a set of abstract examples to show how the topology of a PrAF, the inducement probabilities of arguments and attacks, and an intelligence analyst's prior beliefs affect posterior estimates. The chapter concluded with results from experiments on the Operation Fortitude-South argumentation framework – which suggested the Axis' command should have considered the Normandy and Pas de Calais hypotheses as equally likely, given the information that was available – to demonstrate that the approach could be employed in practice. As the results from the Operation Fortitude-South scenario were acquired using a high performance computing cluster, I also presented some strong scaling speedup and weak scaling efficiency results, for two parallel implementations of one of the proposed algorithms. The algorithms proposed in this chapter are intended as a proof of concept to provide support for intelligence analysts by revising their prior beliefs using the outputs from probabilistic argumentation. The refined probabilities output by the algorithms are not meant to replace the analyst's analysis, but should act as a guide and ensure that they have made rational, probabilistic assessments prior to the dissemination of their findings to stakeholders and decision makers, in accordance with the suggestion made by Zlotnick [251] in Section 2.4.2.

With regard to future work, there are many avenues. The incorporation of the argumentation scheme for the representation of hypotheses in Chapter 3 with the formalism presented in this chapter, for the Operation Fortitude-South scenario, is an obvious extension. Moreover, extending the scheme proposed in Chapter 3 in order to include the joint probability of defeat of all other hypotheses might provide some interesting insights about an analyst's reasoning. Investigating scenarios with more than two hypotheses would also be interesting. Although I assume that evidence can be represented through an information argument with an inducement probability equal to unity, relaxing the restriction to the ASPIC⁺ framework so that it includes both strict and defeasible rules is also intriguing. Another attractive possibility could be to compute posteriors using two different inducement probabilities for each argument and relation within a PrAF, corresponding to the distinct numeric ranges in the Probability (or Uncertainty) Yardstick, as introduced in Section 2.1.4, in order to provide analysts with an uncertainty bound on the posteriors output by the technique in this chapter. A further,

interesting idea to extend this work would be to employ Dung's original argumentation framework [70] in the calculation of the likelihood so as to reduce the computational complexity of the proposed approach. Finally, as the proposed algorithms are intended for use within real-world scenarios, studying both static and dynamic task allocation, as well as hybrid distributed-shared memory parallelism, is important.

In this chapter, a method of prior probability refinement for hypotheses was proposed. In the next chapter, Dung's abstract argumentation framework [70] is employed for sensitivity analysis in order to provide benefit to analysts by using the rational, coherence of computational argumentation to identify diagnostic arguments.

Chapter 6

On Diagnostic Arguments in Abstract Argumentation

In this chapter, I draw upon the notion of sensitivity analysis which is used in step seven of the ACH [101], as discussed in Section 2.1.5.2, and employ abstract argumentation for this task. I set out a novel technique, called the *Diagnostic Argument Identifier*, which can identify *diagnostic* arguments. The algorithm defined to do this task employs a labelling-based approach in order to compute acceptance probabilities between dichotomous sets of arguments, which are then used in the calculation of change in mutual information across partitions of argument labels, measured before and after the removal of an argument of interest. I present a running example for an abstract argumentation framework to show how the method works and a software application to demonstrate that the results are easily obtainable and provide useful insights. To confirm that the technique can be extended to real-world scenarios, I present results obtained from the Operation Fortitude-South argumentation framework, which was introduced at the beginning of Part II. These results were acquired through use of a high performance computing cluster and as such, I present a hybrid, distributed-shared memory parallel implementation of the algorithms, and some strong scaling speedup results as well. I argue that the proposed algorithmic approach systematises and reduces the subjectivity of sensitivity analysis; thus, yielding benefits to intelligence analysts – or any other expert working within a decision or deliberation setting – who need to objectively reevaluate the dependence of their set of conclusions on observed data present within an analysis. The chapter concludes with a summary and discussion about whether the approach is capable of aiding intelligence analysts in the

task of sensitivity analysis, as well as suggesting avenues for future work.

6.1 Introduction

As discussed in Chapter 2, intelligence analysts possess a number of structured analytical techniques [173] which guide them in the evaluation of both complex and novel scenarios, during the *analysis* phase of the *Intelligence Cycle*. One such tool is the ACH [101], where one of the most crucial steps, according to Heuer Jr., is sensitivity analysis. Recall that during sensitivity analysis, step seven of the ACH in Section 2.1.5.2, analysts are tasked with the following:

“Analyse the sensitivity of the conclusions to identify how dependent the likelihood of each hypothesis is on the items of information, evidence, and assumptions.”

Both Heuer Jr. in [101] and Pherson and Heuer Jr. in [173] advise the analyst to sequentially remove each row entry from the ACH matrix in order to establish whether the likelihood of their conclusions change as result. In this context, a *diagnostic* data point is one where its removal from the analysis changes the conclusions drawn. However, there are a couple of issues with this approach, both of which arise from the fact we are humans, with cognitive biases, who can potentially fall victim to a myriad of intuitive traps and lapses in judgement. The first issue pertains to the ACH itself. For example, as discussed in Section 2.4.3, in [152], Murukannaiah *et al.*’s comparison of the ACH with their technique, Arg-ACH, confirmed that participants who used the standard ACH evaluated hypotheses using individual row entries. As I discussed in Chapter 2, a row entry, when considered in isolation, might not provide support for, or conflict with, a hypothesis. However, when row entries are combined, the analyst may identify such supports or conflicts, and the scheme proposed for the representation of hypotheses in Chapter 3 draws upon this idea. This notion is the same for sensitivity analysis also. The removal of a row entry from the ACH may not seem to change the likelihood of a hypothesis because the likelihood is computed using a coarse count which does not include the uncertainty in information, evidence, and assumptions, and the uncertainty associated with their sources. Moreover, as I demonstrate in Chapter 5, row entries can attack each other such that they indirectly support or attack a hypothesis. Thus, simply removing a row entry from the ACH does not capture these subtle changes in the acceptability of row entries, which alter the likelihood of hypotheses.

The second problem is that while analysts may remove a row entry from the ACH matrix in accordance with the procedure, it is incredibly difficult for a human to dis-

count an item of information and act as though it never existed. The *availability heuristic*, proposed by Tversky and Kahneman, is where the likelihood of an event is determined “by the ease with which instances or associations could be brought to mind” [218, p. 208]. Even though a given row entry is physically removed from the ACH matrix, that same row entry may be present within the analyst’s mind and therefore, undoubtedly impact their assessment of whether that entry was diagnostic.

Although the ACH does its best to formalise hypothesis evaluation, sensitivity analysis is a cognitively challenging task for the analyst because it is asking them to do something which they are not predisposed to do. Thus, the process of sensitivity analysis, in its current state, could fall victim to cognitive biases and an analyst’s subjectivity.

To combat those limitations, this chapter makes two contributions, which are listed below:

- The first is an evaluation-based approach to probabilistic argumentation, which uses the set of labellings discovered by a semantics, to calculate joint and marginal argument acceptance probabilities of partitions of arguments and their labels, over a set of labellings.
- The second is a novel technique, called the Diagnostic Argument Identifier (DAI), comprised of two computational tasks that apply the equations from the first contribution to quantify *diagnosticity scores* of arguments within abstract argumentation frameworks. The change in a framework’s evaluation is quantified after the sequential removal of each argument, emulating the task of sensitivity analysis. The technique should alleviate the reliance on human effort and reduce cognitive biases, through use of an algorithmic approach.

The remainder of the chapter is structured as follows. In Section 6.2, Dung’s abstract argumentation frameworks are briefly reintroduced. In Section 6.3, the two contributions of the chapter are made. The first contribution, the method for calculating the joint and marginal probability of partitions of argument labels, is described in Section 6.3.1. The calculation of the distribution of change in mutual information across partitions of argument labels is explained in Section 6.3.2. The way in which abstract argumentation was employed for sensitivity analysis is outlined in Section 6.3.3. Sections 6.3.1, 6.3.2, and 6.3.3 include a running example that demonstrates the approach. The computational approach for the DAI, the second contribution, is made concrete in the algorithms proposed in Section 6.3.4. In Section 6.4, results from the DAI are presented and discussed for: the arguments in the running example, in Section 6.4.1; a software application, in Section 6.4.2; and the Operation Fortitude-South

argumentation framework, in Section 6.4.3. A heuristic which rendered the parallel implementations of the DAI capable of producing results for the real-world example is described in Section 6.4.3.1, and some strong scaling speedup results are presented in Section 6.4.3.2. Work related to the DAI is considered in Section 6.5. The chapter concludes, in Section 6.6, with a summary of the contribution and some avenues for future research.

6.2 Preliminaries

Recall that a Dung abstract argumentation framework \mathcal{G} was defined in Section 2.3.1 as a pair $(\mathcal{A}, \mathcal{R})$ which contains a set of arguments \mathcal{A} and a binary attack relation $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$. The argument a_1 is said to attack a_2 if and only if $(a_1, a_2) \in \mathcal{R}$ and $a_1, a_2 \in \mathcal{A}$. As per Definitions 2.2 and 2.3, the semantics compute *admissible* sets of arguments that have, respectively, two characteristics: *conflict-freeness*; and *acceptability*.

The algorithm proposed in this chapter employs labelling-based methods and preferred semantics to enable the computation of probabilities from semantic evaluation. Recall the definition of a *complete labelling* from Definition 2.6 in Section 2.3.1. As per Definition 2.7 in the same section, a labelling is *preferred* if and only if it is a *complete labelling* with a maximal (with respect to set inclusion) set of arguments labelled IN.

6.3 Diagnostic Argument Identifier

In this section, the DAI is formally introduced. Two contributions are made, namely: the labelling-based argument acceptance probability equations for partitions of argument labels, derived from a set of probability spaces; and the DAI, a technique capable of identifying the most critical arguments within a Dung argumentation framework. To make the approach concrete, four algorithms are presented in Section 6.3.4. Algorithms 6.1 and 6.2 are, respectively, serial and parallel implementations that compute the distribution of mutual information across subsets of argument labels in an initial argumentation framework, and they are presented in Section 6.3.4.1. Algorithms 6.3 and 6.4 are, again, serial and parallel implementations that compute the change in mutual information after the sequential removal of each argument from a framework, and they are presented in Section 6.3.4.2.

6.3.1 Probability Spaces

Consider first the probability spaces that can be derived from the labellings of a Dung argumentation framework. Let $\mathcal{G} = (\mathcal{A}, \mathcal{R})$ be an argumentation framework containing N arguments \mathcal{A} and $|\mathcal{R}|$ relations \mathcal{R} . The set of arguments is ordered using a function $f(\mathcal{A}) : \mathcal{A} \rightarrow \mathbf{A}$, where $\mathbf{A} = (a_1, \dots, a_N)$ is an ordered vector of arguments. It is assumed there is a function $g(\mathbf{A}) : \mathbf{A} \rightarrow \mathcal{L}_M$ which assigns all arguments $a_i \in \mathbf{A}$ to a set of labelling vectors, such that

$$\mathcal{L}_M = \{\mathbf{L}_i\}_{i=1}^M \quad (6.1)$$

where M is the number of labelling vectors, and \mathbf{L}_i is the i -th labelling vector containing N argument labels, such that $\mathbf{L}_i = (l_1, \dots, l_N)$, where $l_j \in \mathbf{L}_i$ is the label of the argument $a_j \in \mathbf{A}$ and $l_j \in \text{LAB}$, such that $\text{LAB} = \{\text{IN}, \text{OUT}, \text{UND}\}$.

The set of arguments \mathcal{A} is partitioned into two sets, named \mathcal{A}_ϕ and \mathcal{A}_ψ , using Definition 6.1.

Definition 6.1 For an argumentation framework $\mathcal{G} = (\mathcal{A}, \mathcal{R})$, let the partitions of \mathcal{A} be $\mathcal{A}_\phi \subseteq \mathcal{A}$ and $\mathcal{A}_\psi \subseteq \mathcal{A}$ such that they are closed under union, where $\mathcal{A}_\phi \cup \mathcal{A}_\psi = \mathcal{A}$, and intersection, where $\mathcal{A}_\phi \cap \mathcal{A}_\psi = \emptyset$. The dichotomous sets \mathcal{A}_ϕ and \mathcal{A}_ψ are complements such that $\mathcal{A}_\phi^c = \mathcal{A} \setminus \mathcal{A}_\phi$ and $\mathcal{A}_\psi^c = \mathcal{A} \setminus \mathcal{A}_\psi$, where $\mathcal{A}_\phi^c = \mathcal{A}_\psi$ and $\mathcal{A}_\psi^c = \mathcal{A}_\phi$.

Both subsets \mathcal{A}_ϕ and \mathcal{A}_ψ are mapped to argument vectors through the function f , such that

$$f(\mathcal{A}_\phi) : \mathcal{A}_\phi \rightarrow \mathbf{A}_\phi = \{(a_1, \dots, a_{|\mathcal{A}_\phi|}) \mid \forall a_i \in \mathcal{A} \text{ where } a_i \notin \mathcal{A}_\psi\} \text{ and} \quad (6.2)$$

$$f(\mathcal{A}_\psi) : \mathcal{A}_\psi \rightarrow \mathbf{A}_\psi = \{(a_1, \dots, a_{|\mathcal{A}_\psi|}) \mid \forall a_j \in \mathcal{A} \text{ where } a_j \notin \mathcal{A}_\phi\}. \quad (6.3)$$

The partitioned vectors \mathbf{A}_ϕ and \mathbf{A}_ψ are mapped to corresponding sets of labelling vectors through the same function g , such that $g(\mathbf{A}_\phi) : \mathbf{A}_\phi \rightarrow \mathcal{L}_\phi$ and $g(\mathbf{A}_\psi) : \mathbf{A}_\psi \rightarrow \mathcal{L}_\psi$, respectively. It follows that

$$\mathcal{L}_\phi = \{\mathbf{L}_{\phi,i}\}_{i=1}^{M_\phi} \text{ and} \quad (6.4)$$

$$\mathcal{L}_\psi = \{\mathbf{L}_{\psi,i}\}_{i=1}^{M_\psi} \quad (6.5)$$

where $M_\phi \leq M$, $M_\psi \leq M$, $\mathcal{L}_\phi \subseteq \mathcal{L}_M$, $\mathcal{L}_\psi \subseteq \mathcal{L}_M$, and $\mathbf{L}_{\phi,i}$ and $\mathbf{L}_{\psi,i}$ are the i -th labelling vectors for \mathbf{A}_ϕ and \mathbf{A}_ψ , respectively.

Example 6.1 Consider a Dung argumentation framework \mathcal{G} , as shown in Figure 6.1, with arguments \mathcal{A} and relations \mathcal{R} , such that

$$\mathcal{A} = \{p, q, r, s, t\} \text{ and}$$

$$\mathcal{R} = \{(q, p), (r, q), (s, q), (r, s), (s, r), (r, t), (t, r), (s, t), (t, s)\}.$$

The evaluation of the framework \mathcal{G} , under preferred semantics, produces three labellings, as shown in Table 6.1. Following Definition 6.1, the set of arguments \mathcal{A} is split into dichotomous sets, where $\mathcal{A}_\phi = \{p, q\}$ and $\mathcal{A}_\psi = \{r, s, t\}$. Using the function f , the sets of arguments \mathcal{A} , \mathcal{A}_ϕ and \mathcal{A}_ψ are mapped to the argument vectors $\mathbf{A} = (p, q, r, s, t)$, $\mathbf{A}_\phi = (p, q)$ and $\mathbf{A}_\psi = (r, s, t)$, which enables the creation of sets of labelling vectors \mathcal{L}_M , \mathcal{L}_ϕ and \mathcal{L}_ψ , as shown in Equations 6.6, 6.7 and 6.8, respectively.

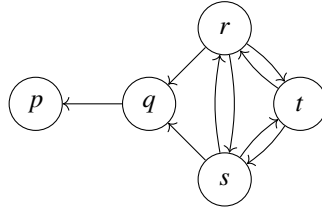


Figure 6.1: Example of a Dung argumentation framework \mathcal{G} .

Table 6.1: The preferred labellings of the framework displayed in Figure 6.1.

Labellings, $\mathcal{L}(\mathcal{A})$	p	q	r	s	t
$\mathcal{L}_1(\mathcal{A})$	IN	OUT	IN	OUT	OUT
$\mathcal{L}_2(\mathcal{A})$	IN	OUT	OUT	IN	OUT
$\mathcal{L}_3(\mathcal{A})$	OUT	IN	OUT	OUT	IN

$$g(\mathbf{A}) : \mathbf{A} \rightarrow \mathcal{L}_M = \left\{ \begin{pmatrix} \text{IN} \\ \text{OUT} \\ \text{IN} \\ \text{OUT} \\ \text{OUT} \end{pmatrix}, \begin{pmatrix} \text{IN} \\ \text{OUT} \\ \text{OUT} \\ \text{IN} \\ \text{OUT} \end{pmatrix}, \begin{pmatrix} \text{OUT} \\ \text{IN} \\ \text{OUT} \\ \text{OUT} \\ \text{IN} \end{pmatrix} \right\} \quad (6.6)$$

$$g(\mathbf{A}_\phi) : \mathbf{A}_\phi \rightarrow \mathcal{L}_\phi = \left\{ \begin{pmatrix} \text{IN} \\ \text{OUT} \end{pmatrix}, \begin{pmatrix} \text{OUT} \\ \text{IN} \end{pmatrix} \right\} \quad (6.7)$$

$$g(\mathbf{A}_\psi) : \mathbf{A}_\psi \rightarrow \mathcal{L}_\psi = \left\{ \begin{pmatrix} \text{IN} \\ \text{OUT} \\ \text{OUT} \end{pmatrix}, \begin{pmatrix} \text{OUT} \\ \text{IN} \\ \text{OUT} \end{pmatrix}, \begin{pmatrix} \text{OUT} \\ \text{OUT} \\ \text{IN} \end{pmatrix} \right\} \quad (6.8)$$

Let $(\Omega_\phi, \mathcal{F}_\phi, P_\phi)$ and $(\Omega_\psi, \mathcal{F}_\psi, P_\psi)$ be probability spaces, where Ω_ϕ and Ω_ψ are sample spaces, \mathcal{F}_ϕ and \mathcal{F}_ψ are event spaces, and P_ϕ and P_ψ are functions such that $P_\phi : \mathcal{F}_\phi \rightarrow (0, 1]$ and $P_\psi : \mathcal{F}_\psi \rightarrow (0, 1]$ for the partitions \mathcal{A}_ϕ and \mathcal{A}_ψ , respectively. This chapter considers two random variables, X_ϕ and X_ψ , that are real-valued measurable functions, $X_\phi : \Omega_\phi \rightarrow \mathbb{R}$ and $X_\psi : \Omega_\psi \rightarrow \mathbb{R}$, that map results from the sample spaces, Ω_ϕ and Ω_ψ , to numerical values; thus, modelling a random experiment which is the set of labellings output from semantic evaluation of an argumentation framework.

Definition 6.2 (Random Vector). Let (Ω, \mathcal{F}, P) be a probability space where the random vector $\mathbf{X} : \Omega \rightarrow \mathbb{R}$ is a measurable function. The random vector \mathbf{X} contains random variables $\mathbf{X} = (X_\phi, X_\psi)$ defined on two probability spaces $X_\phi : \Omega_\phi \rightarrow \mathbb{R}$ and $X_\psi : \Omega_\psi \rightarrow \mathbb{R}$.

Before semantic evaluation and given the constraints of argumentation, the number of possible elements (or labelling vectors) in the sample space Ω is the number of unique combinations of argument labels, such that

$$|\Omega| = |\text{LAB}|^N \quad (6.9)$$

where $|\Omega|$ is the number of potential elements in the sample space Ω , N is the number of arguments, and $|\text{LAB}|$ is either two or three, depending on whether a user employs two- or three-valued semantics. After semantic evaluation, the set of possible labelling vectors is reduced to a set containing M labelling vectors that were observed, such that $\Omega = \mathcal{L}_M$ and $|\Omega| = M$.

The spaces $(\Omega_\phi, \mathcal{F}_\phi, P_\phi)$ and $(\Omega_\psi, \mathcal{F}_\psi, P_\psi)$ are measurable ones, where $\mathcal{F}_\phi \otimes \mathcal{F}_\psi$ is the smallest product σ -field of potential subsets of $\Omega_\phi \times \Omega_\psi \supseteq \Omega$, containing all events of the form $\mathbf{L}_{\phi,i} \times \mathbf{L}_{\psi,j}$ where $\mathbf{L}_{\phi,i} \in \mathcal{F}_\phi$ and $\mathbf{L}_{\psi,j} \in \mathcal{F}_\psi$. The event space \mathcal{F} is a σ -algebra which contains the powerset of all elements in the product space $\mathcal{F}_\phi \otimes \mathcal{F}_\psi$, including the empty set and the set of all events. However, after semantic evaluation, the set of combinations of labelling vectors for each partition is, again, reduced to sets of realised events, where $\mathcal{L}_\phi \otimes \mathcal{L}_\psi \subset \mathcal{F}_\phi \otimes \mathcal{F}_\psi$ is the smallest subset of events from which non-zero probabilities are calculated, and \mathcal{L}_ϕ and \mathcal{L}_ψ are shown in Equations 6.4 and 6.5, respectively.

The observation of the i -th outcome is denoted $x^{(i)} \in \Omega$ which is the i -th labelling vector $x^{(i)} \in \mathcal{L}_M$, such that $x^{(i)} = (x_\phi^{(i)}, x_\psi^{(i)}) = (\mathbf{L}_{\phi,i}, \mathbf{L}_{\psi,i}) \in \Omega$. The outcomes from different partitions and labellings can be observed, where $x_\phi^{(i)}, x_\psi^{(j)} \in \Omega$ are the i -th and j -th labelling vector for the partitions \mathbf{A}_ϕ and \mathbf{A}_ψ , respectively, where $i \neq j$. With a slight abuse of notation, the probability of realising the i -th and j -th labelling vector

of \mathbf{A}_ϕ and \mathbf{A}_ψ in the sample space Ω is denoted as $P(x_\phi^{(i)})$ or $P(x_\psi^{(j)})$, which refers to $P(X_\phi = x_\phi^{(i)})$ or $P(X_\psi = x_\psi^{(j)})$, respectively.

There are two probabilities that concern the calculation of mutual information between partitions of argument labels, namely the joint and marginal probabilities of labels of arguments in each segment.

6.3.1.1 Joint Probability

The joint probability of the observation of both partitions' argument labels, across the i -th and j -th outcome in the sample space, is computed through a normalised count, such that

$$P(x_\phi^{(i)}, x_\psi^{(j)}) = \frac{1}{M} \sum_{k=1}^M \mathbb{I}_{x_\phi^{(i)}=x_\phi^{(k)}; x_\psi^{(j)}=x_\psi^{(k)}} \quad (6.10)$$

where \mathbb{I}_A returns a value of unity if and only if A is true.

Due to the distinct nature of labellings, there will only be one labelling vector in the sample space Ω that contains the same arrangement of argument labels as the labels for arguments in each partition. Thus, there will be M pairs of labellings vectors which produce non-zero joint probabilities across the product space of realised argument labels in each segment. Consider a list \mathcal{M} containing $|\mathcal{M}|$ distinct i - j pairs, where $|\mathcal{M}| = M_\phi \times M_\psi$, from the observed product space $\mathcal{L}_\phi \otimes \mathcal{L}_\psi$, such that we compute $|\mathcal{M}|$ joint probabilities. Each joint probability is the reciprocal of the number of labellings, if and only if both observed outcomes feature in the same labelling vector across the sample space Ω , and for any other i - j pair the probability is zero, as shown in Equation 6.11.

$$P(x_\phi^{(i)}, x_\psi^{(j)}) = \begin{cases} \frac{1}{M}, & \text{iff } \exists m \in \{1, \dots, M\} \text{ such that } i = i(m) \text{ and } j = j(m) \\ 0, & \text{otherwise.} \end{cases} \quad (6.11)$$

It follows that the list \mathcal{M} contains M i - j pairs of labelling vectors with a joint probability greater than zero, corresponding to the number of times that the sample space Ω contained those distinct segments of argument labels, or outcomes $x^{(m)} = (x_\phi^{(m)}, x_\psi^{(m)}) \in \Omega$.

Example 6.2 Continuing the running example. The sets of labelling vectors \mathcal{L}_ϕ and \mathcal{L}_ψ , presented in Equations 6.7 and 6.8, were employed to compute the joint probability of labels for arguments in each partition. The product space $\mathcal{L}_\phi \otimes \mathcal{L}_\psi$ contains $|\mathcal{M}| = M_\phi \times M_\psi$ combinations of potential labelling vectors based on the unique vectors in each partition, where $|\mathcal{M}| = 6$ in this example. Using both Equations 6.10 and

6.11, and Table 6.1, it is easy to see that the joint probability of first, second and last i - j pair from the product space $\mathcal{L}_\phi \otimes \mathcal{L}_\psi$, shown in Equation 6.12, is equal to $\frac{1}{3}$, which is $\frac{1}{M}$ in this example. The joint probability of the third, fourth, and fifth i - j pair in Equation 6.12 is equal to zero in both Equations 6.10 and 6.11 because those events were not observed as labelling vectors in the sample space Ω .

$$\mathcal{L}_\phi \otimes \mathcal{L}_\psi = \left\{ \begin{pmatrix} \text{IN} \\ \text{OUT} \\ \text{IN} \\ \text{OUT} \\ \text{OUT} \end{pmatrix}, \begin{pmatrix} \text{IN} \\ \text{OUT} \\ \text{OUT} \\ \text{IN} \\ \text{OUT} \end{pmatrix}, \begin{pmatrix} \text{IN} \\ \text{OUT} \\ \text{OUT} \\ \text{OUT} \\ \text{IN} \end{pmatrix}, \begin{pmatrix} \text{OUT} \\ \text{IN} \\ \text{IN} \\ \text{OUT} \\ \text{OUT} \end{pmatrix}, \begin{pmatrix} \text{OUT} \\ \text{IN} \\ \text{OUT} \\ \text{IN} \\ \text{OUT} \end{pmatrix}, \begin{pmatrix} \text{OUT} \\ \text{IN} \\ \text{OUT} \\ \text{OUT} \\ \text{IN} \end{pmatrix} \right\} \quad (6.12)$$

6.3.1.2 Marginal Probability

The second probability required to compute the mutual information is the marginal probability of each unique labelling vector in the spaces \mathcal{L}_ϕ and \mathcal{L}_ψ . Again, this computation turns out to be another normalised count across a vector subspace, counting how many times $X_\phi = x_\phi^{(i)}$ or $X_\psi = x_\psi^{(i)}$ was observed in the sample space Ω , as shown in Equations 6.13 and 6.14, respectively.

$$P(x_\phi^{(i)}) = \frac{1}{M} \sum_{j=1}^M \mathbb{I}_{x_\phi^{(i)} = x_\phi^{(j)}} \quad (6.13)$$

$$P(x_\psi^{(i)}) = \frac{1}{M} \sum_{j=1}^M \mathbb{I}_{x_\psi^{(i)} = x_\psi^{(j)}} \quad (6.14)$$

Example 6.3 Consider again the running example from Figure 6.1 and the partition \mathbf{A}_ϕ with its set of distinct labelling vectors \mathcal{L}_ϕ , as presented in Equation 6.7. There are two events within the space \mathcal{L}_ϕ where marginal probabilities can be calculated. Using Equation 6.13, the marginal probabilities of the first $\mathbf{L}_{\phi,1} \in \Omega$ and second $\mathbf{L}_{\phi,2} \in \Omega$ event are equal to $\frac{2}{3}$ and $\frac{1}{3}$, respectively.

6.3.2 Mutual Information

As I discussed in Section 6.1, sensitivity analysis assesses the dependence of the probability of hypotheses on the information employed in the evaluation those hypotheses. Row entries are sequentially removed from an ACH matrix by an analyst in order to understand whether their probabilistic judgements change as a result. To do this, the analyst must first arrive at an initial set of conclusions. Employing argumentation for sensitivity analysis is no different. The amount of information that can be inferred from

an initial argumentation framework's set of labelling vectors should be quantified, thus, enabling the measurement of the extent to which removing a given argument from a framework affects the distribution of labels between partitions of other arguments. In this chapter, this change is measured through the computation of mutual information between partitions of argument labels before and after the removal of an argument. The mutual information is a symmetric function that quantifies the conditional dependence between two random variables, determining the amount of information communicated, on average, about one random variable through observation of another [200, 59]. The mutual information between pairs of argument labels across the realised product space $\mathcal{L}_\phi \otimes \mathcal{L}_\psi$, where \mathcal{L}_ϕ and \mathcal{L}_ψ are defined Equations 6.4 and 6.5, respectively, is shown in Equation 6.15.

$$I(X_\phi; X_\psi) = \sum_{x_\phi^{(i)} \in \mathcal{L}_\phi} \sum_{x_\psi^{(j)} \in \mathcal{L}_\psi} P(x_\phi^{(i)}, x_\psi^{(j)}) \log \left(\frac{P(x_\phi^{(i)}, x_\psi^{(j)})}{P(x_\phi^{(i)})P(x_\psi^{(j)})} \right) \quad (6.15)$$

The units of mutual information depend on the base of the logarithm used in the calculation. For the purposes of this thesis, information will be measured in *bits* which is the logarithm to the base of two.

The mutual information, presented in Equation 6.15, is simplified by conducting one summation over $|\mathcal{M}|$ pairs of outcomes, $x_\phi^{(i)}$ and $x_\psi^{(j)}$, which feature in the realised product space $\mathcal{L}_\phi \otimes \mathcal{L}_\psi$, where $|\mathcal{M}| = M_\phi \times M_\psi$, as shown in Equation 6.16.

$$I(X_\phi; X_\psi) = \sum_{(x_\phi^{(i)}, x_\psi^{(j)}) \in \mathcal{L}_\phi \otimes \mathcal{L}_\psi} P(x_\phi^{(i)}, x_\psi^{(j)}) \log \left(\frac{P(x_\phi^{(i)}, x_\psi^{(j)})}{P(x_\phi^{(i)})P(x_\psi^{(j)})} \right) \quad (6.16)$$

Many of the addends in Equation 6.16 will equal zero for i - j pairs of outcomes in the product space that do not feature in the sample space Ω . The only non-zero contributions to the summation will be i - j pairs that are observed in both the sample space Ω and the product space $\mathcal{L}_\phi \otimes \mathcal{L}_\psi$. Thus, Equation 6.16 can be simplified further to only include outcomes $x_\phi^{(i)}$ and $x_\psi^{(j)}$ that have a joint probability greater than zero, as in Equation 6.17.

$$I(X_\phi; X_\psi) = \sum_{(x_\phi^{(i)}, x_\psi^{(j)}) \in \mathcal{L}_\phi \otimes \mathcal{L}_\psi, P(x_\phi^{(i)}, x_\psi^{(j)}) > 0} P(x_\phi^{(i)}, x_\psi^{(j)}) \log \left(\frac{P(x_\phi^{(i)}, x_\psi^{(j)})}{P(x_\phi^{(i)})P(x_\psi^{(j)})} \right) \quad (6.17)$$

In light of this constraint and using Equation 6.11, it is easy to see that the only non-zero summands to the mutual information will be from contributions where the joint probability of pairs of outcomes from the product space feature in the same labelling

from the sample space. For the sake of computational efficiency, Equation 6.11 is substituted into Equation 6.17 so that the mutual information is, therefore, reduced to a single sum over the set of labellings for this particular problem setting, such that

$$I(X_\phi; X_\psi) = \sum_{x^{(m)} \in \Omega} \frac{1}{M} \log \left(\frac{1}{M P(x_\phi^{(m)}) P(x_\psi^{(m)})} \right) \quad (6.18)$$

where $x^{(m)} = (x_\phi^{(m)}, x_\psi^{(m)})$, and $\Omega \subset \mathcal{L}_\phi \otimes \mathcal{L}_\psi$ is the space within which non-zero joint probabilities are found.

As mentioned earlier, the mutual information is a symmetric function which means there are $\frac{2^{|\mathcal{A}|}}{2}$ partitions that produce unique mutual information scores. Therefore, the mutual information is computed between divisions of arguments for up to half the powerset to completely explore the distribution of information communicated across the sets of labelling vectors within an initial argumentation framework.

Example 6.4 Turning back to the running example. Recall that $\mathbf{A}_\phi = (p, q)$ and $\mathbf{A}_\psi = (r, s, t)$. The mutual information between each segment's set of labelling vectors was found to be $I(X_\phi; X_\psi) = 0.9 \text{ bits}$ (1 d.p.), under preferred semantics. We see that observing the labels of, say, the arguments in \mathbf{A}_ϕ tells us 0.9 *bits* of information about the labels of arguments in \mathbf{A}_ψ , and vice versa.

6.3.3 Sensitivity Analysis

In the previous subsection, a method for computing the distribution of information between partitions of argument labels was proposed. The mutual information scores from the initial argumentation framework \mathcal{G} quantify how much one can infer from the set of labelling vectors resulting from semantic evaluation of a graph. In this subsection, the way within which argumentation was employed for sensitivity analysis is explained. To conduct sensitivity analysis on a Dung argumentation framework, each argument $a \in \mathcal{A}$, and any relation containing that argument, is sequentially removed from the initial argumentation framework, creating a sensitive framework \mathcal{G}^a as stated in Definition 6.3. The sensitive graph is then evaluated, using the same semantics chosen earlier and again, the distribution of mutual information between partitions of argument labels is computed. The change in mutual information between segments before and after the removal of the argument a is calculated, enabling the quantification of how much the initial set of conclusions changed as a result of the removal of an argument.

Definition 6.3 For an argumentation framework $\mathcal{G} = (\mathcal{A}, \mathcal{R})$ undergoing sensitivity analysis, the graph $\mathcal{G}^a = (\mathcal{A}^a, \mathcal{R}^a)$ is referred to as a *sensitive* framework that does not

include the argument a , the argument of interest, where $\mathcal{A}^a = \mathcal{A} \setminus a$ such that $\mathcal{A}^a \subset \mathcal{A}$ and $a \notin \mathcal{A}^a$. For a relation $r \in \mathcal{R}$, if the argument a features in that relation, i.e. $a \in r$, then it is removed from the set of relations, such that

$$\mathcal{R}^a = \{ r \mid \forall r \text{ where } r \neq (a, b) \vee (b, a) \text{ and } a, b \in \mathcal{A} \}. \quad (6.19)$$

It is important for the reader to note that this chapter only considers the removal of one argument from an initial argumentation framework while conducting sensitivity analysis. However, this approach could be extended to remove more than one argument from an initial graph to understand how this affects the results output from sensitivity analysis.

Notation 6.1 The two partitions of \mathcal{A}^a within the sensitive graph \mathcal{G}^a that obey Definition 6.1 are denoted as \mathcal{A}_ϕ^a or \mathcal{A}_ψ^a .

Again, the set of sensitive arguments \mathcal{A}^a is ordered using the function $f(\mathcal{A}^a) : \mathcal{A}^a \rightarrow \mathbf{A}^a$, such that $\mathbf{A}^a = (a_1, \dots, a_{N-1})$ is a vector of arguments. The vector of sensitive arguments \mathbf{A}^a is mapped to a set of labelling vectors through the function $g(\mathbf{A}^a) : \mathbf{A}^a \rightarrow \mathcal{L}_M^a$, such that

$$\mathcal{L}_M^a = \{\mathbf{L}_i^a\}_{i=1}^{M^a} \quad (6.20)$$

where M^a is the number of labelling vectors output from semantic evaluation of the sensitive framework \mathcal{G}^a , and \mathbf{L}_i^a is the i -th labelling vector containing $N - 1$ argument labels, such that $\mathbf{L}_i^a = (l_1^a, \dots, l_{N-1}^a)$, where $l_j^a \in \mathbf{L}_i^a$ is the label of the argument $a_j \in \mathbf{A}^a$ and $l_j^a \in \text{LAB}$ and $\text{LAB} = \{\text{IN}, \text{OUT}, \text{UND}\}$.

The sets \mathcal{A}_ϕ^a and \mathcal{A}_ψ^a are mapped to the argument vectors \mathbf{A}_ϕ^a and \mathbf{A}_ψ^a through

$$f(\mathcal{A}_\phi^a) : \mathcal{A}_\phi^a \rightarrow \mathbf{A}_\phi^a = \{(a_1, \dots, a_{|\mathcal{A}_\phi^a|}) \mid \forall a_i \in \mathcal{A}^a \text{ where } a_i \notin \mathcal{A}_\psi^a\} \text{ and} \quad (6.21)$$

$$f(\mathcal{A}_\psi^a) : \mathcal{A}_\psi^a \rightarrow \mathbf{A}_\psi^a = \{(a_1, \dots, a_{|\mathcal{A}_\psi^a|}) \mid \forall a_j \in \mathcal{A}^a \text{ where } a_j \notin \mathcal{A}_\phi^a\}. \quad (6.22)$$

The sets of distinct labelling vectors corresponding to the partitions \mathbf{A}_ϕ^a and \mathbf{A}_ψ^a are found using the function g , such that

$$g(\mathbf{A}_\phi^a) : \mathbf{A}_\phi^a \rightarrow \mathcal{L}_\phi^a = \{\mathbf{L}_{\phi,i}^a\}_{i=1}^{M_\phi^a} \text{ and} \quad (6.23)$$

$$g(\mathbf{A}_\psi^a) : \mathbf{A}_\psi^a \rightarrow \mathcal{L}_\psi^a = \{\mathbf{L}_{\psi,i}^a\}_{i=1}^{M_\psi^a} \quad (6.24)$$

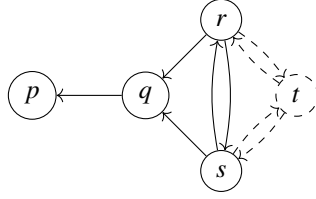
where $M_\phi^a \leq M^a$, $M_\psi^a \leq M^a$, $\mathcal{L}_\phi^a \subseteq \mathcal{L}_M^a$, $\mathcal{L}_\psi^a \subseteq \mathcal{L}_M^a$, and $\mathbf{L}_{\phi,i}^a$ and $\mathbf{L}_{\psi,i}^a$ are the i -th labelling vectors for the partitions \mathbf{A}_ϕ^a and \mathbf{A}_ψ^a , respectively.

The sensitive argumentation framework is evaluated using the same semantics as earlier in order to realise the labelling vectors for arguments in the vectors \mathbf{A}_ϕ^a and \mathbf{A}_ψ^a , enabling the computation of argument acceptance probabilities and thus, mutual information between partitions of arguments within a given sensitive graph.

Notation 6.2 Let \mathcal{G}^a be a sensitive framework with the argument a removed, as per Definition 6.3, and \mathbf{A}_ϕ^a and \mathbf{A}_ψ^a be two sensitive argument vectors, mapped from the dichotomous sets \mathcal{A}_ϕ^a and \mathcal{A}_ψ^a that follow Definition 6.1. The random vector $\tilde{\mathbf{X}} = (\tilde{X}_\phi, \tilde{X}_\psi)$ is a measurable function from a probability space $(\Omega^a, \mathcal{F}^a, P^a)$ where $\tilde{\mathbf{X}}$ maps elements from the sensitive sample space $\Omega^a = \mathcal{L}_M^a$ to events $\mathcal{L}_\phi^a \otimes \mathcal{L}_\psi^a \subset \mathcal{F}^a$.

Example 6.5 Back to the running example. The result from a mutual information calculation is presented for the sensitive graph \mathcal{G}^t , displayed in Figure 6.2. To start, Definition 6.3 was employed in the instantiation of \mathcal{G}^t with arguments $\mathcal{A}^t = \{p, q, r, s\}$ and relations $\mathcal{R}^t = \{(q, p), (r, q), (s, q), (r, s), (s, r)\}$. The sensitive partitions were equal to $\mathcal{A}_\phi^t = \{p, q\}$ and $\mathcal{A}_\psi^t = \{r, s\}$ because that was the same as the original partitions (i.e., \mathcal{A}_ϕ and \mathcal{A}_ψ in Example 6.1) where neither \mathcal{A}_ψ^t nor \mathcal{A}_ϕ^t contained the argument t . The dichotomous sets of arguments \mathcal{A}_ϕ^t and \mathcal{A}_ψ^t were mapped to the argument vectors $\mathbf{A}_\phi^t = (p, q)$ and $\mathbf{A}_\psi^t = (r, s)$, respectively, through the function f . Preferred semantics were then used to evaluate \mathcal{G}^t in order to observe the set labellings for each partition. The set of preferred labellings of \mathcal{G}^t are presented in Table 6.2. The realisation of argument labels enabled the mapping of each argument in \mathbf{A}_ϕ^t and \mathbf{A}_ψ^t to their respective sets of labelling vectors \mathcal{L}_ϕ^t and \mathcal{L}_ψ^t , through the function g . The mutual information $I(\tilde{X}_\phi, \tilde{X}_\psi)$ between the labelling vectors for this partition of sensitive arguments was equal to 0.0 *bits* (1 d.p.). This result is intuitive. The only arguments that caused a difference in the labellings of \mathcal{G}^t were the arguments r and s . However, both r and s were contained within the set \mathcal{A}_ψ^t . The arguments p and q were in the set \mathcal{A}_ϕ^t and their argument labels remained the same across both labellings of the sensitive graph. An important point to note here is that the mutual information will always equal zero when all the arguments in a partition are sceptically labelled because it is not possible to communicate any information about the other partition's acceptability when there is no diversity across a segment's set of labelling vectors.

The total number of combinations of \mathcal{A}_ϕ^a and \mathcal{A}_ψ^a that produce unique mutual information results for a sensitive framework \mathcal{G}^a is equal to $\frac{2^{|\mathcal{A}^a|}}{2}$, which is half the number of mutual information scores that can be calculated using the labellings from the initial graph \mathcal{G} , i.e. $\frac{2^{|\mathcal{A}^a|}}{2} = \frac{1}{2} \times \frac{2^{|\mathcal{A}|}}{2}$.

Figure 6.2: A sensitive argumentation framework \mathcal{G}^t Table 6.2: The preferred labellings of the \mathcal{G}^t displayed in Figure 6.2.

Labellings, $\mathcal{L}(\mathcal{A}^t)$	p	q	r	s
$\mathcal{L}_1(\mathcal{A}^t)$	IN	OUT	IN	OUT
$\mathcal{L}_2(\mathcal{A}^t)$	IN	OUT	OUT	IN

Once all the unique mutual information scores have been computed for $\frac{2^{|\mathcal{A}|}}{2}$ and $\frac{2^{|\mathcal{A}^a|}}{2}$ partitions in the initial and sensitive frameworks, respectively, the *diagnosticity score* is computed. The *diagnosticity score* measures the change in partitions' mutual information before and after the removal of an argument of interest and is defined in Definition 6.4.

Definition 6.4 The *diagnosticity score* is the change in mutual information before (calculated using the labelling vectors \mathcal{L}_ϕ and \mathcal{L}_ψ for all arguments in \mathcal{A}_ϕ and \mathcal{A}_ψ from \mathcal{G}) and after (calculated using \mathcal{L}_ϕ^a and \mathcal{L}_ψ^a for all arguments in \mathcal{A}_ϕ^a and \mathcal{A}_ψ^a from \mathcal{G}^a) the removal of the argument a , the argument of interest. Equation 6.25 computes the *diagnosticity score* when the argument a was removed from either \mathcal{A}_ϕ or \mathcal{A}_ψ .

$$\mathcal{D}(\mathcal{A}_\phi, \mathcal{A}_\psi; \mathcal{A}_\phi^a \vee \mathcal{A}_\psi^a) = I(\mathbf{X}_\phi; \mathbf{X}_\psi) - I(\tilde{\mathbf{X}}_\phi; \tilde{\mathbf{X}}_\psi) \quad (6.25)$$

An interesting point to note about Equation 6.25 is the sign. A positive diagnosticity score implies that there was more information communicated, on average, between partitions in the initial argumentation framework \mathcal{G} , and removing the argument a resulted in a loss of information between argument labels for a given a partition. A diagnosticity score of zero indicates that the removal of the argument a did not change the amount of information that could be inferred from the argument labels of the initial and sensitive frameworks. A score below zero indicates that, after the removal of the argument a , there was more information that could be inferred between partitions of the argument labels within the sensitive argumentation graph \mathcal{G}^a .

Example 6.6 Returning to the running example for the last time. We now have all that we need to quantify the impact that removing the argument t had on the acceptability of arguments within the partitions in the initial and sensitive frameworks. The set of labelling vectors of \mathbf{A}_ϕ and \mathbf{A}_ψ were employed to calculate the mutual information between X_ϕ and X_ψ in the initial graph (Example 6.4). The set of labelling vectors of \mathbf{A}_ϕ^t and \mathbf{A}_ψ^t were also used to compute the mutual information between \tilde{X}_ϕ and \tilde{X}_ψ in the sensitive framework \mathcal{G}^t (Example 6.5). Using Definition 6.4, the diagnosticity score $\mathcal{D}(\mathcal{A}_\phi, \mathcal{A}_\psi; \mathcal{A}_\phi^t, \mathcal{A}_\psi^t)$ for the aforementioned partitions turns out to be 0.9 *bits* (1 d.p.). As the diagnosticity score was above zero, this indicates that there was more mutual information between the argument labels of \mathbf{A}_ϕ and \mathbf{A}_ψ in the initial graph than there was amongst the labels of \mathbf{A}_ϕ^t and \mathbf{A}_ψ^t in the sensitive framework, across the set of labelling vectors; thus, confirming a loss of information between both \mathbf{A}_ϕ and \mathbf{A}_ψ 's set of labelling vectors, through the removal of t , the argument of interest.

6.3.4 Pseudo-code

The DAI is a technique that can be divided into two computational tasks, as shown in Figure 6.3. First, the user must instantiate an argumentation framework \mathcal{G} and choose a semantics \mathcal{S} to evaluate it. Then, they run Algorithm 6.1 (or 6.2) to compute the distribution of mutual information scores across half the powerset of arguments in the initial framework \mathcal{G} . After this, Algorithm 6.3 (or 6.4) takes the mutual information scores, along with their corresponding partitions, and the initial argumentation framework in order to conduct sensitivity analysis, which results in a diagnosticity vector comprised of $\frac{2^{|\mathcal{A}|}}{2}$ diagnosticity scores for each argument in the framework. It is important to note here that Algorithms 6.1 and 6.2 are, respectively, serial and parallel implementations that take the same input and produce the same output. Likewise, Algorithms 6.3 and 6.4 produce the same result and are serial and parallel implementations, respectively. All algorithms were implemented in C++ [106]. Both parallel algorithms employed a hybrid distributed-shared memory parallelism that used the Message Passing Interface (MPI) [146] and OpenMP [159], and they also require a user to input the number of nodes \mathcal{P} and threads \mathcal{T} . Some strong scaling speedup results for the parallel implementations are presented in Section 6.4.3.2.

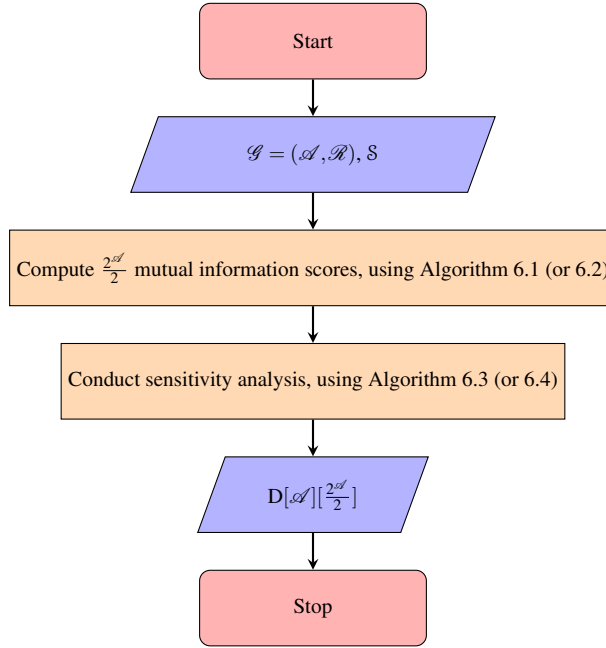


Figure 6.3: A flowchart depicting the order within which the algorithms for the DAI are executed.

6.3.4.1 Distribution of Mutual Information Across the Initial Framework

For an initial argumentation framework, the process by which the distribution of mutual information can be computed for up to half the powerset of arguments is formally presented in Algorithms 6.1 and 6.2. The algorithms require as input a Dung argumentation framework \mathcal{G} and a semantics \mathcal{S} that is capable of producing more than one labelling. The output from the algorithms is an array, of length $\frac{2^{|\mathcal{A}|}}{2}$, that contains unique mutual information scores for the initial framework and the partitions used in the computation of each result.

6.3.4.2 Sensitivity Analysis

The serial and parallel pseudo-codes for sensitivity analysis are, respectively, presented in Algorithms 6.3 and 6.4. The algorithms take as input: an array MI_s comprised of mutual information scores obtained from their corresponding partitions; a Dung argumentation framework \mathcal{G} ; and a semantics \mathcal{S} capable of producing more than one labelling. The algorithm returns a *diagnosticity vector* $D[\mathcal{A}][\frac{2^{|\mathcal{A}|}}{2}]$, which contains $\frac{2^{|\mathcal{A}|}}{2}$ diagnosticity scores for every argument a in the initial graph.

Algorithm 6.1: Compute the distribution of MI across half the powerset of partitions of argument labels within the initial argumentation framework.

Input : An AF $\mathcal{G} = (\mathcal{A}, \mathcal{R})$;
 A semantics S .
Output: An array $MI_s[\frac{2^{\mathcal{A}}}{2}]$ to store all the unique mutual information scores and their corresponding partitions.

```

1  $\mathcal{L}(\mathcal{A})$  // Evaluate initial AF  $\mathcal{G}$  with chosen semantics  $S$ .
2  $\mathbf{A} = f(\mathcal{A}) : \mathcal{A} \rightarrow \mathbf{A}$  // Create a vector of arguments for all arguments in the
   initial AF.
3  $\Omega = g(\mathbf{A}) : \mathbf{A} \rightarrow \mathcal{L}_M$  // Set the sample space to be equal to the set of labelling
   vectors from the initial AF.
4  $M = |\mathcal{L}_M|$  // Save the number of labellings from the initial AF.
5  $MI_s \leftarrow \text{array}(MI_s, \frac{2^{\mathcal{A}}}{2})$  // Declare the size of the initial AF's MI results array.
6 for  $\mathcal{A}_\phi \in \frac{2^{\mathcal{A}}}{2}$  do
7    $\mathcal{A}_\psi = \mathcal{A} \setminus \mathcal{A}_\phi$  // Create the other set  $\mathcal{A}_\psi$  (Definition 6.1).
8    $\mathbf{A}_\phi = f(\mathcal{A}_\phi) : \mathcal{A}_\phi \rightarrow \mathbf{A}_\phi$  and  $\mathbf{A}_\psi = f(\mathcal{A}_\psi) : \mathcal{A}_\psi \rightarrow \mathbf{A}_\psi$  // Create the argument vectors
   for  $\mathcal{A}_\phi$  and  $\mathcal{A}_\psi$ .
9    $\mathcal{L}_\phi = g(\mathbf{A}_\phi) : \mathbf{A}_\phi \rightarrow \mathcal{L}_\phi$  and  $\mathcal{L}_\psi = g(\mathbf{A}_\psi) : \mathbf{A}_\psi \rightarrow \mathcal{L}_\psi$  // Create the sets of labelling
   vectors (Equations 6.4 & 6.5).
10   $I(X_\phi, X_\psi) = 0$  // Set the MI equal to zero for the current partitions of
   the initial AF.
11  for  $(x_\phi^{(m)}, x_\psi^{(m)}) \in \Omega$  do
12    /* Iterate through labellings in the  $\Omega$  where both  $x_\phi^{(m)}$  and  $x_\psi^{(m)}$ 
       occur. */
13     $P(x_\phi^{(m)}), P(x_\psi^{(m)})$  // Equations 6.13 and 6.14, respectively.
14     $I(X_\phi, X_\psi) += \frac{1}{M} \log \left( \frac{1}{MP(x_\phi^{(m)})P(x_\psi^{(m)})} \right)$  // Compute the addend of the MI between
       segments for the  $m$ -th labelling vector of the initial AF
       (Equation 6.18).
15   $MI_s \leftarrow \{\mathcal{A}_\phi, \mathcal{A}_\psi, I(X_\phi, X_\psi)\}$  // Append the result to the array of initial MI
   calculations.
16 return  $MI_s$ 
```

Algorithm 6.2: Hybrid distributed-shared memory parallelism for Algorithm 6.1.

Input : An AF $\mathcal{G} = (\mathcal{A}, \mathcal{R})$;
 A semantics \mathcal{S} ;
 A number of processors \mathcal{P} ;
 A number of threads \mathcal{T} .

Output: An array $MIs[\frac{2^{\mathcal{A}}}{2}]$ to store all the unique mutual information scores and their corresponding partitions.

1 Do Lines 1 to 5 in Algorithm 6.1

2 $Tasks_Per_Processor = \lfloor \frac{2^{\mathcal{A}}}{2} \rfloor \times \frac{1}{\mathcal{P}}$ // Compute the number of mutual information calculations per processor

3 **do across** \mathcal{P} **processors**

4 $MIs(\mathcal{P}_i) \leftarrow \text{array}(MIs(\mathcal{P}_i), Tasks_Per_Processor)$

5 **for** $\mathcal{A}_\phi \in \frac{2^{\mathcal{A}}}{2} [\mathcal{P}_i : \mathcal{P}_i + Tasks_Per_Processor]$ **do**

6 $\mathcal{A}_\psi = \mathcal{A} \setminus \mathcal{A}_\phi$ $\mathbf{A}_\phi = f(\mathcal{A}_\phi) : \mathcal{A}_\phi \rightarrow \mathbf{A}_\phi$ and $\mathbf{A}_\psi = f(\mathcal{A}_\psi) : \mathcal{A}_\psi \rightarrow \mathbf{A}_\psi$

7 $\mathcal{L}_\phi = g(\mathbf{A}_\phi) : \mathbf{A}_\phi \rightarrow \mathcal{L}_\phi$ and $\mathcal{L}_\psi = g(\mathbf{A}_\psi) : \mathbf{A}_\psi \rightarrow \mathcal{L}_\psi$

8 $I(\mathbf{X}_\phi, \mathbf{X}_\psi) = 0$

9 $Tasks_Per_Thread = \lfloor \Omega \rfloor \times \frac{1}{\mathcal{T}}$ // Compute the number of mutual information calculations per processor

10 **do with** \mathcal{T} **threads**

11 /* Conduct a parallel reduction sum to compute the mutual information for a given partition */

12 **for** $(x_\phi^{(m)}, x_\psi^{(m)}) \in \Omega[\mathcal{T}_i : \mathcal{T}_i + Tasks_Per_Thread]$ **do**

13 $P(x_\phi^{(m)}), P(x_\psi^{(m)})$

14 $I(\mathbf{X}_\phi, \mathbf{X}_\psi) += \frac{1}{M} \log \left(\frac{1}{MP(x_\phi^{(m)})P(x_\psi^{(m)})} \right)$

15 $MIs(\mathcal{P}_i) \leftarrow \{ \mathcal{A}_\phi, \mathcal{A}_\psi, I(\mathbf{X}_\phi, \mathbf{X}_\psi) \}$

16 Send $MIs(\mathcal{P}_i)$ to main processor \mathcal{P}_1

17 **if** Processor is \mathcal{P}_1 **then**

18 **forall** \mathcal{P}_i **do**

19 $MIs \leftarrow MIs(\mathcal{P}_i)$ // Receive each processor's initial mutual information calculations, the corresponding partitions, and append the result to the MIs array

20 **return** MIs

Algorithm 6.3: Conduct sensitivity analysis on all arguments within an initial argumentation framework \mathcal{G} .

Input : An array $MIs[\frac{\gamma^{\mathcal{A}}}{2}]$ to store all the unique mutual information scores;
 An AF $\mathcal{G} = (\mathcal{A}, \mathcal{R})$;
 A semantics S .

Output: A diagnosticity vector $D[\mathcal{A}][\frac{\gamma^{\mathcal{A}}}{2}]$.

```

1   $D[\mathcal{A}][\frac{\gamma^{\mathcal{A}}}{2}]$  // Declare the size of diagnosticity vector  $D$ .
2  for  $a \in \mathcal{A}$  do
3      /* Begin sensitivity analysis where  $a$  is the argument of interest. */
4       $\mathcal{G}^a = (\mathcal{A}^a, \mathcal{R}^a)$  // Instantiate the sensitive AF (Definition 6.3).
5       $\mathcal{L}(\mathcal{A}^a)$  // Evaluate the sensitive AF  $\mathcal{G}^a$  with chosen semantics  $S$ .
6       $\mathbf{A}^a = f(\mathcal{A}^a) : \mathcal{A}^a \rightarrow \mathbf{A}^a$  // Create the argument vector for the set of all
       sensitive arguments.
7       $\Omega^a = g(\mathbf{A}) : \mathbf{A} \rightarrow \mathcal{L}_M^a$  // Set the sample space to be equal to the sensitive
       AF's set of labelling vectors.
8       $M^a = |\mathcal{L}_M^a|$  // Save the number of labellings of the sensitive AF.
9      for  $\mathcal{A}_\phi, \mathcal{A}_\psi, I(\mathcal{A}_\phi, \mathcal{A}_\psi) \in MIs$  do
10         /* Iterate through partitions and MI results from the initial AF
             $\mathcal{G}$ . */
11         if  $a \in \mathcal{A}_\phi$  then
12              $\mathcal{A}_\phi^a = \mathcal{A}_\phi \setminus a, \mathcal{A}_\psi^a = \mathcal{A}_\psi$  // Create  $\mathcal{A}_\phi^a$  and  $\mathcal{A}_\psi^a$  by removing  $a$  from  $\mathcal{A}_\phi$ 
              (Definitions 6.1 & 6.3).
13         else if  $a \in \mathcal{A}_\psi$  then
14              $\mathcal{A}_\phi^a = \mathcal{A}_\phi, \mathcal{A}_\psi^a = \mathcal{A}_\psi \setminus a$  // Create  $\mathcal{A}_\phi^a$  and  $\mathcal{A}_\psi^a$  by removing  $a$  from  $\mathcal{A}_\psi$ 
              (Definitions 6.1 & 6.3).
15          $\mathbf{A}_\phi^a = f(\mathcal{A}_\phi^a) : \mathcal{A}_\phi^a \rightarrow \mathbf{A}_\phi^a, \mathbf{A}_\psi^a = f(\mathcal{A}_\psi^a) : \mathcal{A}_\psi^a \rightarrow \mathbf{A}_\psi^a$  // Create the sensitive AF's
              argument vectors.
16          $\mathcal{L}_\phi^a = g(\mathbf{A}_\phi^a) : \mathbf{A}_\phi^a \rightarrow \mathcal{L}_\phi^a, \mathcal{L}_\psi^a = g(\mathbf{A}_\psi^a) : \mathbf{A}_\psi^a \rightarrow \mathcal{L}_\psi^a$  // Create the sensitive sets of
              labelling vectors.
17          $I(\tilde{\mathbf{X}}_\phi, \tilde{\mathbf{X}}_\psi) = 0$  // Set the MI equal to zero for the current partitions
              of the sensitive AF.
18         for  $(\tilde{x}_\phi^{(\tilde{m})}, \tilde{x}_\psi^{(\tilde{m})}) \in \Omega^a$  do
19             /* Iterate through the sensitive sample space  $\Omega^a$  */
20              $P(\tilde{x}_\phi^{(\tilde{m})}), P(\tilde{x}_\psi^{(\tilde{m})})$  // Equations 6.13 and 6.14, respectively.
21              $I(\tilde{\mathbf{X}}_\phi, \tilde{\mathbf{X}}_\psi) += \frac{1}{M^a} \log \left( \frac{1}{M^a P(\tilde{x}_\phi^{(\tilde{m})}) P(\tilde{x}_\psi^{(\tilde{m})})} \right)$  // Compute the addend of the MI
              between segments for the  $\tilde{m}$ -th labelling vector of the
              sensitive AF (Equation 6.18).
22         if  $a \in \mathcal{A}_\phi$  then
23              $\mathcal{D}(\mathcal{A}_\phi, \mathcal{A}_\psi, \mathcal{A}_\phi^a) = I(\mathbf{X}_\phi, \mathbf{X}_\psi) - I(\tilde{\mathbf{X}}_\phi, \tilde{\mathbf{X}}_\psi)$  // Compute the diagnosticity score
              (Equation 6.25)
24              $D[a] \leftarrow \{a, \mathcal{A}_\phi, \mathcal{A}_\psi, \mathcal{D}(\mathcal{A}_\phi, \mathcal{A}_\psi, \mathcal{A}_\phi^a)\}$  // Append the result to the
              diagnosticity vector  $D$ 
25         else if  $a \in \mathcal{A}_\psi$  then
26              $\mathcal{D}(\mathcal{A}_\phi, \mathcal{A}_\psi, \mathcal{A}_\psi^a) = I(\mathbf{X}_\phi, \mathbf{X}_\psi) - I(\tilde{\mathbf{X}}_\phi, \tilde{\mathbf{X}}_\psi)$  // Compute the diagnosticity score
              (Equation 6.25)
27              $D[a] \leftarrow \{a, \mathcal{A}_\phi, \mathcal{A}_\psi, \mathcal{D}(\mathcal{A}_\phi, \mathcal{A}_\psi, \mathcal{A}_\psi^a)\}$  // Append the result to the
              diagnosticity vector  $D$ 
28 return  $D$ 

```

Algorithm 6.4: Hybrid distributed-shared memory parallel implementation of Algorithm 6.3.

Input : An array MI_s which contains partitions and their mutual information scores;
 An AF $\mathcal{G} = (\mathcal{A}, \mathcal{R})$;
 A semantics S ;
 A number of processors \mathcal{P} ;
 A number of threads \mathcal{T} .

Output: A diagnosticity vector $D[\mathcal{A}][\frac{2^{\mathcal{A}}}{2}]$.

```

1   $D[\mathcal{A}][\frac{2^{\mathcal{A}}}{2}]$  // Declare the size of diagnosticity vector  $D$ .
2  for  $a \in \mathcal{A}$  do
3      Do Lines 4 to 8 in Algorithm 6.3
4       $Tasks\_Per\_Processor = |\mathcal{M}Is| \times \frac{1}{\mathcal{P}}$  // Compute the number of mutual information
      calculations per processor
5      do across  $\mathcal{P}$  processors
6           $\Delta MI_s(\mathcal{P}_i) \leftarrow \text{array}(\Delta MI_s(\mathcal{P}_i), Tasks\_Per\_Processor)$ 
7          for  $\mathcal{A}_\phi, \mathcal{A}_\psi, I(\mathcal{A}_\phi, \mathcal{A}_\psi) \in MI_s[\mathcal{P}_i : \mathcal{P}_i + Tasks\_Per\_Processor]$  do
8              Do Lines 12 to 16 in Algorithm 6.3
9               $I(\tilde{X}_\phi, \tilde{X}_\psi) = 0$ 
10              $Tasks\_Per\_Thread = |\Omega^a| \times \frac{1}{\mathcal{T}}$  // Compute the number of mutual
             information calculations per processor
11             do with  $\mathcal{T}$  threads
12                 for  $(\tilde{x}_\phi^{(\tilde{m})}, \tilde{x}_\psi^{(\tilde{m})}) \in \Omega^a[\mathcal{T}_i : \mathcal{T}_i + Tasks\_Per\_Thread]$  do
13                      $P(\tilde{x}_\phi^{(\tilde{m})}), P(\tilde{x}_\psi^{(\tilde{m})})$ 
14                      $I(\tilde{X}_\phi, \tilde{X}_\psi) += \frac{1}{M^a} \log \left( \frac{1}{M^a P(\tilde{x}_\phi^{(\tilde{m})}) P(\tilde{x}_\psi^{(\tilde{m})})} \right)$ 
15             if  $a \in \mathcal{A}_\phi$  then
16                  $\mathcal{D}(\mathcal{A}_\phi, \mathcal{A}_\psi, \mathcal{A}_\phi^a) = I(X_\phi, X_\psi) - I(\tilde{X}_\phi, \tilde{X}_\psi)$ 
17                  $\Delta MI_s(\mathcal{P}_i) \leftarrow \{a, \mathcal{A}_\phi, \mathcal{A}_\psi, \mathcal{D}(\mathcal{A}_\phi, \mathcal{A}_\psi, \mathcal{A}_\phi^a)\}$ 
18             else if  $a \in \mathcal{A}_\psi$  then
19                  $\mathcal{D}(\mathcal{A}_\phi, \mathcal{A}_\psi, \mathcal{A}_\psi^a) = I(X_\phi, X_\psi) - I(\tilde{X}_\phi, \tilde{X}_\psi)$ 
20                  $\Delta MI_s(\mathcal{P}_i) \leftarrow \{a, \mathcal{A}_\phi, \mathcal{A}_\psi, \mathcal{D}(\mathcal{A}_\phi, \mathcal{A}_\psi, \mathcal{A}_\psi^a)\}$ 
21             Send  $\Delta MI_s(\mathcal{P}_i)$  to main processor  $\mathcal{P}_1$ 
22         if Processor is  $\mathcal{P}_1$  then
23             forall  $\mathcal{P}_i$  do
24                  $D[a] \leftarrow \Delta MI_s(\mathcal{P}_i)$  // Receive each processor's change in mutual
                 information calculations and append the result to the
                 diagnosticity vector
25 return  $D$ 

```

6.4 Experimental Evaluation

In the previous section, the DAI was formally introduced. In this section, results from experiments on an abstract example, a software application that can execute the DAI, and the Operation Fortitude-South argumentation framework are presented. The purpose of these results is to show that the DAI is capable of identifying diagnostic arguments, the arguments that change an argumentation framework's evaluation the most, in both abstract and real-world settings. Moreover, the goal of presenting a software application that can run the DAI is to show that while there are many results output by the technique, the results are still easy to understand and insightful.

In the following subsection, the outputs from the DAI for the argumentation framework \mathcal{G} from the running example in Figure 6.1 are presented and discussed.

6.4.1 Results from the Running Example

The initial argumentation framework in Figure 6.1 was employed in the evaluation of the DAI in an abstract setting because it was assumed to possess a similar topology to graphs found within intelligence settings (i.e., many symmetric attacks instantiated because of conflicting information). The initial argumentation framework \mathcal{G} in Figure 6.1 was input into Algorithm 6.1 and the distribution of initial mutual information scores, along with their corresponding partitions, were output. Then, the initial argumentation framework, mutual information scores, and partitions were input into Algorithm 6.3, resulting in a diagnosticity vector, which contained $\frac{2^{|\mathcal{A}|}}{2}$ (i.e., 16) diagnosticity scores for each argument. Both algorithms employed preferred semantics to discover the set of preferred labellings from which mutual information was calculated. The diagnosticity scores for the 16 partitions, for each of the removed arguments, is presented in Figure 6.4.

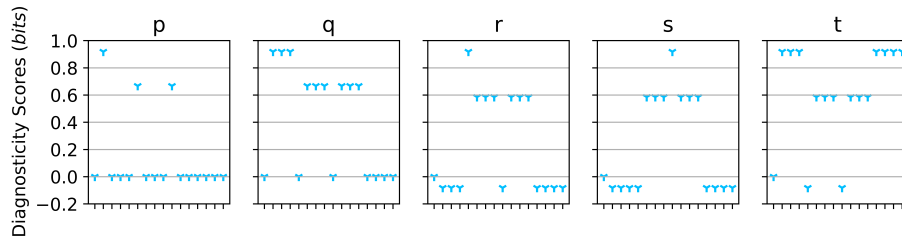


Figure 6.4: The diagnosticity scores for the 16 partitions of labelling vectors for all arguments within the Dung-style argument system \mathcal{G} in Figure 6.1, discovered using preferred semantics.

A violin plot is presented in Figure 6.5 which groups together the diagnosticity scores of each argument in Figure 6.4 to show the distribution of change in mutual information after the sequential removal of each argument from the framework. The mean, average absolute, and median diagnosticity scores are also included in Figure 6.5. The goal of the DAI is to identify arguments that produce the most change in an argumentation framework's evaluation, after their removal. Some of the diagnosticity scores in Figure 6.4 were discovered to be below zero, which meant that there was more that could be inferred about a given partition's arguments' labels in the sensitive framework. The average of absolute diagnosticity scores enabled the comparison of both positive and negative scores in order to quantify the magnitude change in mutual information caused by the removal of an argument of interest.

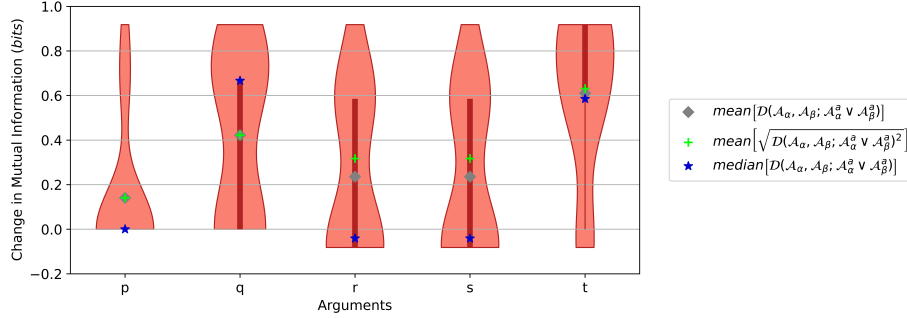


Figure 6.5: A violin plot showing the distribution of *diagnosticity scores*, as well as the median, mean and average absolute change in mutual information, for partitions of argument labels after the removal of each argument from the initial argumentation framework \mathcal{G} , under preferred semantics.

The argument t had the largest mean, absolute average, and median diagnosticity scores, as demonstrated in Figure 6.5. Thus, one could argue that the argument t was the most diagnostic because its removal caused the most change in what could be inferred about argument labels. This result is intuitive because the argument t had symmetric attacks between the arguments r and s in the initial framework and removing t reduces the number of labellings from three to two, as shown in Tables 6.1 and 6.2, respectively.

The argument q had the second largest set of diagnosticity scores, as shown in Figure 6.5. Although q only attacked one argument (i.e., q attacks p in Figure 6.1) and is attacked by two arguments (i.e. the arguments r and s), its removal does not reduce the number of labellings. While one might think that the diagnosticity scores should

be lower accordingly, the acceptability of the argument q depends on the acceptance of the argument t . Thus, the removal of the argument q produces, on average, larger changes in mutual information because it can only be labelled IN when the argument t is also labelled IN. The fact that q and t are conditionally dependent and their diagnosticity scores are greater than p , r , and s is somewhat synonymous with the notion of preferred semantics. The means and median change in mutual information for the arguments q and t were closer to unity and the same metrics for the arguments p , r , and s were closer to zero, such that the diagnosticity vector provides a numeric quantification of preferred labellings, whose outputs are usually sets of categorical variables.

Another interesting point to note is that the removal of the argument q from the initial argumentation framework would change the label of the argument p so that it was sceptically labelled IN within the sensitive graph \mathcal{G}^q . In this instance, for the sensitive framework \mathcal{G}^q and partitions where p was the only argument within a segment (i.e., $\mathcal{A}_\phi^q = \{p\}$ or $\mathcal{A}_\psi^q = \{p\}$), the mutual information scores would always equal zero because there can be no information communicated between partitions when one of the sets has no diversity in argument labels.

The arguments r and s share the same distribution of change in mutual information, as shown in Figure 6.4, because there is a topological symmetry between both arguments in the initial argumentation framework \mathcal{G} , as per Figure 6.1. The median diagnosticity score for both arguments was below zero which indicates that a lot more information was communicated between partitions of labelling vectors in the sensitive graphs \mathcal{G}^r and \mathcal{G}^s after the removal of r and s , respectively.

To compute the diagnosticity vector, the DAI must iterate through half the powerset of arguments in the initial and sensitive frameworks, which is computationally expensive. To understand whether other methods could be employed in the quantification of argument importance, two graph topological metrics, namely the degree and eigenvalue centrality, are presented in Table 6.3. The degree centrality counts the number of direct connections that a node has, whereas the eigenvalue centrality measures how well connected a node's links are. Starting with the degree centrality, the arguments r and s had the highest score, followed by the argument t , then q , and finally p . The arguments r and s had a total of five edges, argument t had four, q had three, and p had one in the initial framework \mathcal{G} in Figure 6.1. One quick glance at this result might indicate that the diagnosticity is correlated with how connected an argument is within a given framework. However, on closer inspection, the argument q has a lower degree centrality than, say, the argument r , which was more connected, and yet q had a higher average absolute diagnosticity score, as in Figure 6.5. Now consider the eigenvalue centrality in Table 6.3. The arguments q , r , s , and t shared the largest eigenvalue cen-

trality and the argument p had the lowest, which is intuitive since removing p produces no tangible change in mutual information in Figures 6.4 and 6.5. Removing arguments r , s and t results in two labellings of the sensitive frameworks \mathcal{G}^r , \mathcal{G}^s , and \mathcal{G}^t , respectively. Whereas, the removal of q does not change the number of labellings, but the argument p does end up being sceptically labelled IN across all of the labellings of \mathcal{G}^q (and \mathcal{G}^t because q cannot ever be labelled IN when t is removed). The argument t was the most diagnostic argument within the initial argumentation framework, according to the results output by the DAI. However, both the degree and eigenvalue centrality do not indicate this because they are topological metrics, whilst the diagnosticity vector depends on the outputs from semantic evaluation of the initial and sensitive frameworks. The DAI employs an evaluation-based method for quantifying the change in an initial graph's set of conclusions, after the removal of an argument of interest, which means that the importance of arguments is calculated as a function of change in a logical agent's set of conclusions, not graph topology.

Table 6.3: The degree and eigenvalue centrality for the arguments in the initial AF \mathcal{G} .

Metric	p	q	r	s	t
Degree Centrality	0.250	0.750	1.250	1.250	1.000
Eigenvalue Centrality	0.243	0.485	0.485	0.485	0.485

The results from an experiment on the initial argumentation framework from the running example demonstrate that the DAI is capable of quantifying the importance of arguments after their removal. The DAI employs an evaluation-based approach in this quantification which means that it produces results that other, less computationally expensive metrics, such as the degree and eigenvalue centrality, cannot achieve. Not only does the DAI fully explain the change that removing an argument has on an argumentation framework, it captures the notion of conditional dependence between arguments encapsulated by the semantics. This subsection has shown that the DAI enables users to understand how sensitive a set of conclusions are to the removal of arguments, thus, providing a notion of robustness, which intelligence analysts would ordinarily have to decipher themselves.

6.4.2 A Software Application

In this subsection, an application which can execute the DAI and visualise the diagnosticity vector is presented. The purpose of the proposed application is to show that

although there are many diagnosticity scores output by the technique, the results can be easy to obtain, whilst also being interpretable and intuitive for the analyst.

The home screen of the application is presented in Figures 6.6 and 6.8. Analysts can choose to instantiate a Dung argumentation framework using the software itself (Figure 6.8). Or, the analyst can import an argument analysis stored within the AIF and the programme will transform it into an ASPIC⁺ argumentation theory and then to a Dung argument system, using the procedure described in Section 2.3.2.3 (Figure 6.6). The arguments and sub-arguments instantiated using the ASPIC⁺ framework are also provided for full explainability and to ensure that users can view the internal structure of arguments abstracted to Dung’s formalism (Figure 6.7). Once a Dung-style argument system has been instantiated, the user can plot and evaluate the graph using many of the well-known semantics, and the labellings are displayed in a table so that the user can understand each argument’s acceptability. In Figures 6.6 and 6.7, the AIF data for the Operation Fortitude-South argumentation framework was imported and converted to a Dung-style argument graph, and subsequently plotted and evaluated using complete semantics. Whereas, in Figure 6.8, the argumentation framework \mathcal{G} , presented in Figure 6.1 and used throughout the running examples in Sections 6.3.1, 6.3.2, and 6.3.3, was instantiated using the application and evaluated using preferred semantics.

Once an argumentation framework has been instantiated, using either the method provided within the application or argument data stored within the AIF, and the semantics chosen, the user can run the DAI by clicking a button, as shown in Figures 6.6 and 6.8. The software then employs Algorithms 6.1 and 6.3 to compute $\frac{2|\mathcal{A}|}{2}$ diagnosticity scores for every argument within the initial argumentation framework. The output from those algorithms is a diagnosticity vector which describes the average change in what can be inferred about labelling vectors and their argument labels as a result of the sequential removal of arguments. The application generates a new window that displays the results within the diagnosticity vector, as demonstrated in Figure 6.9. The window contains a violin plot, which shows the distribution of diagnosticity scores (top right), and a plot presenting the mean, median, and average absolute diagnosticity scores (bottom right), for each argument within the initial argumentation framework. The user can investigate the diagnosticity scores of each argument further by clicking on an option button (top left). The application then displays a table (bottom left) that contains all diagnosticity scores for the chosen argument, the partitions before and after the removal of the argument of interest, and the initial and sensitive mutual information results for each segment. The diagnosticity scores for the selected argument are also plotted (middle top) to allow for easier comparison of the $\frac{2|\mathcal{A}|}{2}$ scores. Upon clicking on a row in the table (highlighted in red), a specific diagnosticity result can be explored

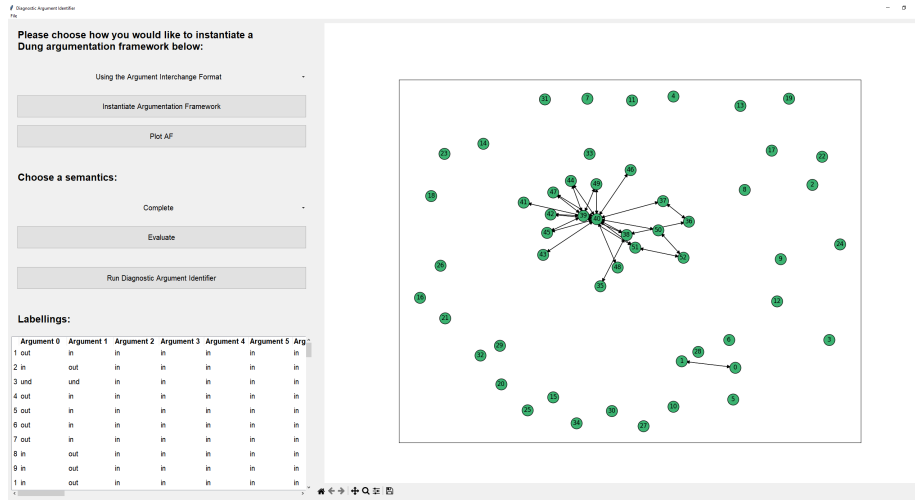


Figure 6.6: A screenshot of the home screen of the DAI application. The Operation Fortitude-South AIF data was imported, transformed, and instantiated into a Dung argumentation framework. The graph was plotted (right) and evaluated using complete semantics to discover the complete labellings (bottom left).

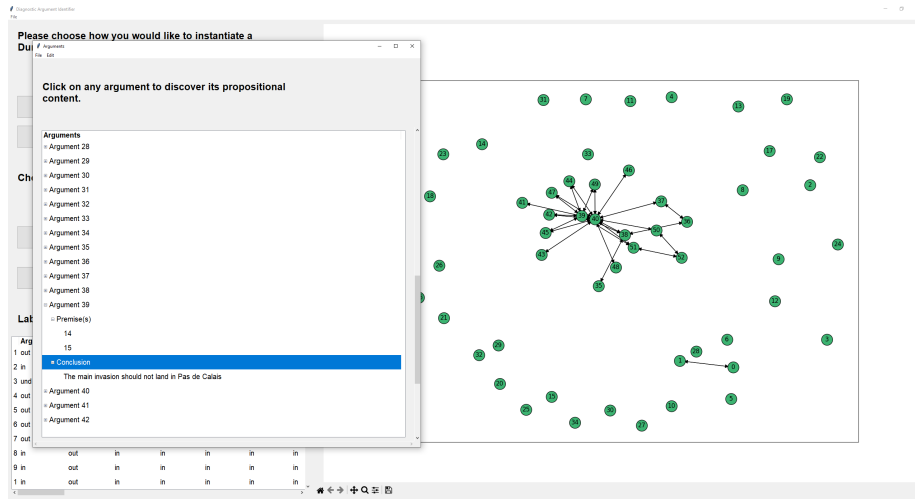


Figure 6.7: A screenshot of the DAI application with the Operation Fortitude-South argumentation framework (right), and the corresponding arguments and sub-arguments instantiated using an ASPIC⁺ argumentation system (left).

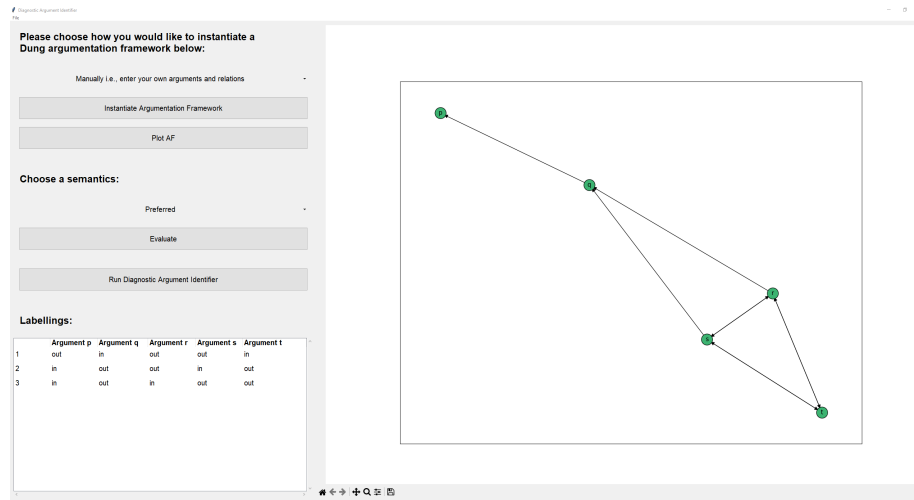


Figure 6.8: A screenshot of the home screen of the DAI, where the argumentation framework \mathcal{G} from Figure 6.1 was instantiated using the application, plotted (right), and evaluated to discover the preferred labellings (bottom left).

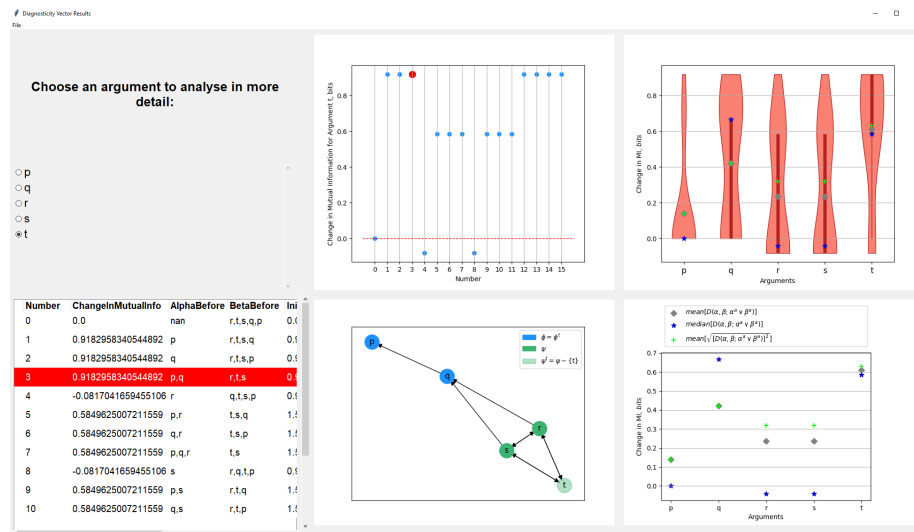


Figure 6.9: A screenshot of a window from the DAI application that displays the results from the diagnosticity vector.

in more detail. The plot containing an argument's diagnosticity scores (middle top) changes and the selected row's change in mutual information is, again, highlighted in

red in the chart. The initial argumentation framework is also displayed (middle bottom). Arguments are coloured in either blue or green to denote the partition they were part of, and the removed argument is slightly more transparent. The diagnosticity score that was analysed in more detail in Figure 6.9 is the same as the result produced by the running example which culminated in Example 6.6.

6.4.3 Operation Fortitude-South

The diagnosticity vector for the arguments within the Operation Fortitude-South argumentation framework was obtained by executing Algorithms 6.2 and 6.4 on a high performance computing cluster¹. The distribution of diagnosticity scores, as well as the mean, average absolute, and median score, for all arguments within the scenario, are presented in Figure 6.10. Argument 39 had the largest means and median. One could argue that Argument 39 was the most diagnostic because it caused the most change in mutual information after its removal from the initial framework. As stated in Appendix B.4, Argument 39 had the following inference:

Arg 39: [Arg 14, Arg 15] \rightarrow The main invasion should not land in Pas de Calais.

Arg 14: If the Allies bring about an invasion of Pas de Calais, then there will be no opportunities for expansion – e.g., no link-up with Operation Dragoon.

Arg 15: No opportunities for expansion with Operation Dragoon will result in a bad outcome for the Allies.

It is indeed interesting that Argument 39 caused the most change in mutual information after its removal because its conclusion supports the Normandy hypothesis indirectly, through the negation of the Pas de Calais hypothesis. Moreover, Argument 39 highlights the importance of strengthening a military effort by combining forces with another operation, Operation Dragoon, which was an Allied invasion plan for Southern France that started on 15th August 1944. Had the Axis' command been provided with the knowledge of the importance of Argument 39, they may have reconsidered and correctly forecasted that the Allies wanted to recapture France by attacking from both the North and South in Normandy and Provence, respectively.

To obtain results on the Operation Fortitude-South argumentation framework, the DAI was executed on a high performance computing cluster using a distributed-shared

¹The high performance computing cluster employed was called Scafell Pike: <https://www.hartree.stfc.ac.uk/technologies/high-performance-computing/> (Accessed on: 02.10.2024).

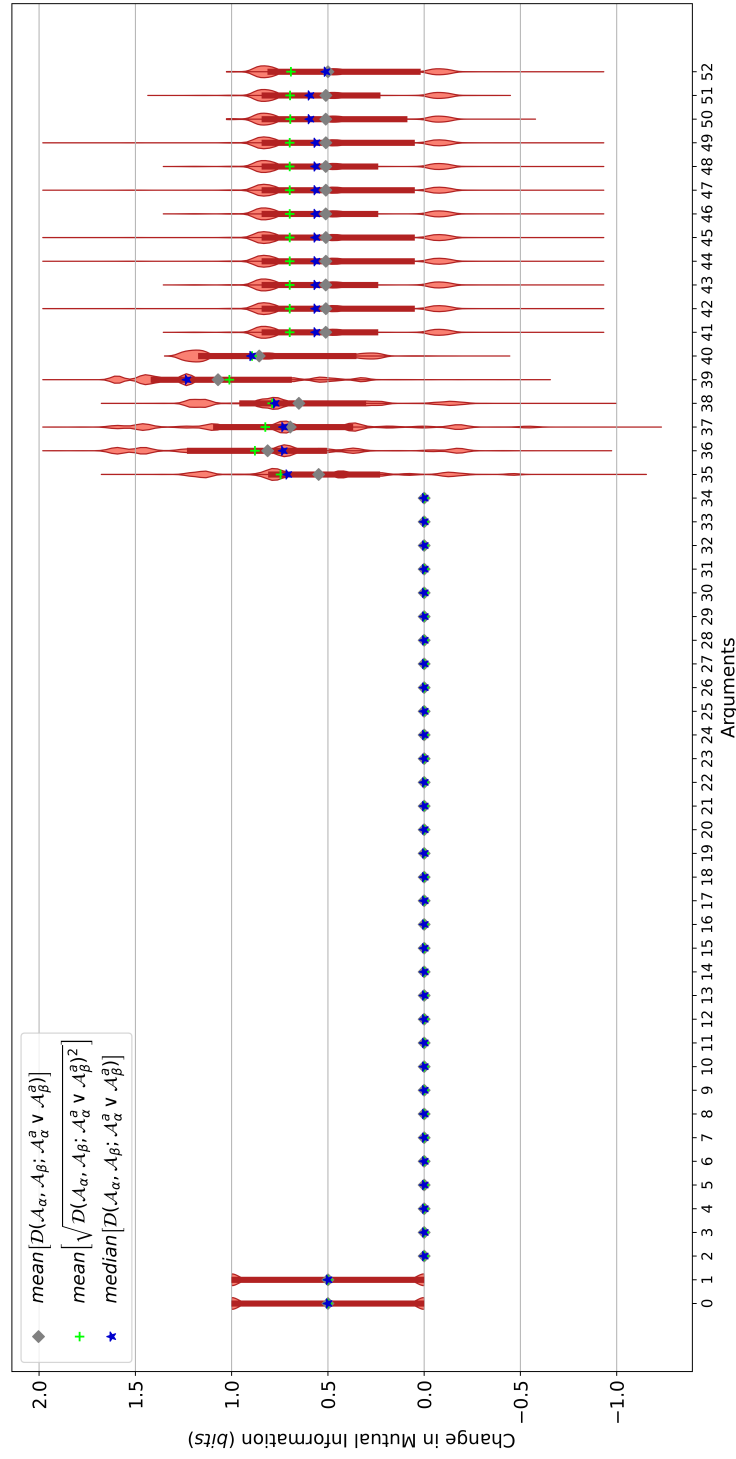


Figure 6.10: The distribution of change in mutual information, as well as the mean, absolute average, and median diagnosticity scores, for all the arguments in the Operation Fortitude-South argumentation framework, found using preferred semantics.

memory approach, which is made clear in Algorithms 6.2 and 6.4. The DAI is able to fully quantify the change in an argumentation framework's evaluation by computing the change in mutual information across partitions of arguments, up to half the power-set. While this approach works for small problems, the number of mutual information calculations doubles with every argument instantiated within an initial argumentation framework. To address this, two heuristics were noted and one was employed to reduce the number of computations required to acquire the diagnosticity vector for the Operation Fortitude-South framework. The heuristics are discussed in the next subsection.

6.4.3.1 Heuristics for Real-World Problems

Recall from Example 6.5 that when all the argument labels in a partition are all sceptically labelled, the mutual information between that segment and another is equal to zero. This was observed to be the case for all segments that did not possess any diversity in arguments labels. Moreover, when a partition contains at least two arguments – one that was sceptically labelled and one with more than one type of label, across the set of labellings – it is the probabilities calculated with arguments possessing different labels that contribute to the mutual information sum. Knowing this, the problem space from which mutual information is calculated in the initial and sensitive argumentation frameworks can be reduced at runtime. The two approaches that could be employed go as follows. For the first method, arguments that are neither attacked nor attack any other argument will always be labelled IN in the labellings of both the initial and sensitive argumentation frameworks, unless they were removed during sensitivity analysis. As such, sceptically labelled arguments can be omitted from the partitions of arguments from which mutual information is calculated. For example, consider Figure 6.6, after transforming the AIF to the ASPIC⁺ framework and then to a Dung argument system, Arguments 2 to 34 possess no edges². Those arguments are the set of ordinary premises in the knowledge base of an ASPIC⁺ argumentation theory and will always be labelled IN. Including those arguments in the set of arguments that are to be partitioned increases the number of combinations of partitions without providing any more insights pertaining to an argument's diagnosticity. For instance, the $\frac{2^{|A|}}{2}$ mutual information calculations in the original Operation Fortitude-South argumentation framework with all arguments included is equal to 4.5×10^{15} . Whereas, if Arguments 2 to 34 were not included, then there is only 524,288 calculations to fully explore the distribution of mutual information, which is a significant decrease in the number of computations

²Refer to Appendix B.4 and B.5 for a more detailed look at the internal structure of arguments and the Dung argumentation framework instantiated from the Operation Fortitude-South scenario, respectively.

whilst also obtaining the same result.

While the first approach considers arguments that possess no edges, it does not account for arguments that may be sceptically labelled OUT or UND. To resolve this one could instead evaluate the initial argumentation framework and then check to see whether any arguments are sceptically labelled, and omit those arguments from the set of mutual information calculations. However, the first approach was employed to obtain results from the Operation Fortitude-South framework and the second method is left as a task for future work.

6.4.3.2 Strong Scaling Speedup

As previously mentioned, in order to obtain results from the Operation Fortitude-South argumentation framework, Algorithms 6.1 and 6.3 were parallelised using a hybrid distributed-shared memory approach which resulted in Algorithms 6.2 and 6.4, respectively. To understand whether the parallel implementations were, in fact, faster than their serial counterparts some strong scaling speedup results are presented in this subsection.

Recall from Chapter 5 that the strong scaling speedup is governed by Amdahl's law [6] such that the speedup is limited by the portion of a code which cannot be parallelised. The strong scaling speedup measures the decreased execution time of an algorithm as a function of the number of processors, for a fixed-problem size. Strong scaling speedup is calculated as follows

$$\text{Strong scaling speedup} = \frac{t(1)}{t(\mathcal{P})} \quad (6.26)$$

where $t(1)$ and $t(\mathcal{P})$ are the execution times for, respectively, one and \mathcal{P} processors.

Recall also from Chapter 5 that the ideal speedup and execution time should be

$$\text{Ideal speedup} = \mathcal{P} \text{ and ideal execution time} = \frac{t(1)}{\mathcal{P}}, \quad (6.27)$$

where \mathcal{P} is the number of processors.

Experimental Setup. The results presented in this subsection were obtained on a cluster of 846 *2x 128 Intel Xeon Gold E5-6142*³ CPUs, each with 16 cores, a clock speed of 2.5GHz (up to 3.7GHz), and 192Gb of RAM. Algorithms 6.2 and 6.4 were implemented in C++ [106] and parallelised using *MPI* [146] and *OpenMP* [159], for

³<https://ark.intel.com/content/www/us/en/ark/products/120487/intel-xeon-gold-6142-processor-22m-cache-2-60-ghz.html> (Accessed on: 02.10.2024)

distributed and shared memory parallelism, respectively. An example of the hybrid distributed-shared memory architecture employed is presented in Figure D.1 in Appendix D.1. The strong scaling experiments measured speedup across 1, 2, 10, 25, 50, 100, and 200 nodes, each with either 1, 16, or 32 threads.

The execution time and strong scaling speedup results for the Operation Fortitude-South argumentation framework are presented in Figure 6.11. The execution time decreased and the speedup increased with an increasing number of processors. The results demonstrated good scaling when $\mathcal{P} = 1, 5, 10, 25, 50$ and 100, and acceptable scaling when $\mathcal{P} = 200$, across all threads. The largest speedup was equal to 99 and observed when $\mathcal{P} = 200$ and 32 threads. There was a negligible difference between execution times and speedup for 1, 16 and 32 threads across all values of \mathcal{P} , which shows that there was no tangible benefit when offloading the probability and mutual information calculations to OpenMP (see Lines 10 to 14 in Algorithm 6.2 and Lines 11 to 14 in Algorithm 6.4). The reason for this is not so intuitive but pertains to the number of labellings discovered after semantic evaluation. When the number of labelling vectors are small in number, the overhead of shared memory parallelisation outweighs the decreased execution time such that there is no real benefit observed from the parallelism. The hybrid distributed-shared memory parallelism proposed in this chapter should produce much better results than the ones presented in Figure 6.11 when computing the diagnosticity vector for complex argumentation frameworks with many labellings. However, when increasing the complexity of an argumentation framework, the speedup will be limited by the length of time taken to compute the semantics, yet studying this is a task left for future work.

6.5 Related Work

The prior literature on probabilistic argumentation can be divided into two parts, namely the epistemic [212, 213] and constellation [69, 133] approaches, and the method of probability calculation proposed in this chapter is distinct from both. For instance, the proposed algorithms do not employ probability functions in the evaluation of argumentation frameworks, as with the epistemic approach, nor do they iterate through all permutations of sub-graphs to compute a probabilities, as with the constellation approach, which was used to refine an analyst's prior beliefs about hypotheses in Chapter 5. There was one, similar work that employed the frequency of individual argument labels to compute marginal probabilities which were subsequently used in combination with a Markov network to semantically evaluate an argumentation framework [176].

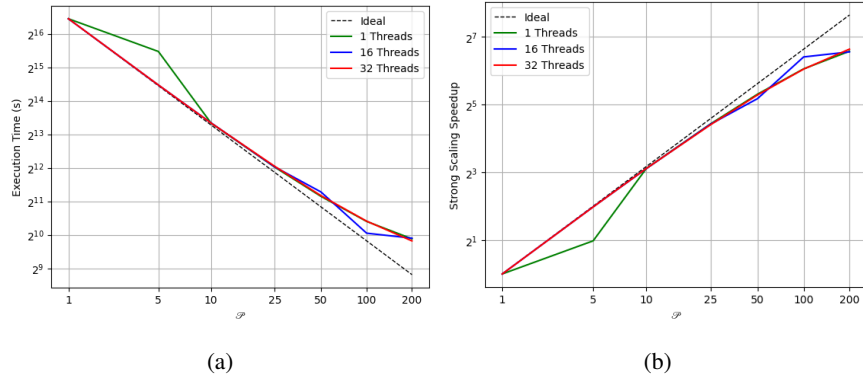


Figure 6.11: (a) The execution time and (b) strong scaling speedup for the execution of Algorithms 6.2 and 6.4 for the Operation Fortitude-South argumentation framework, where $\frac{2^{|A|}}{2} = 524,288$ in the initial graph.

However, the method of probability computation proposed in this chapter differs from that work because probabilities are computed for sets of arguments that contain a range of either one to all arguments, and their labels. The formalised probability calculations, presented in Equations 6.10, 6.11, 6.13 and 6.14, were the first contribution of this chapter and are the first attempt within the literature to employ partitions of arguments and their labels, over a set of labellings, to compute joint and marginal probabilities.

To the best of found knowledge, there have been only two proposals before this one which employed argumentation for sensitivity analysis, the second contribution of this chapter. The first proposal combines argumentation and Markov random fields to quantify the sensitivity of items of information [1]. The second proposal evaluates the sensitivity of initial weights assigned to arguments, within the context of inverse argumentation, and considers whether changes in an argument's weight affects the acceptability degree of other arguments on a continuous scale, computed using gradual semantics [160]. The DAI, the technique proposed in this chapter, relies solely on Dung's original and seminal framework and the outputs from it, computed using a semantics, to calculate probabilities, the distribution of mutual information in the initial and sensitive graphs, and the change thereof caused by the removal of arguments. The technique can fully describe what can be inferred about arguments and their labels, whilst also quantifying the dependence, sensitivity, and robustness of an argumentation framework's conclusions based upon the arguments that the graph is comprised of. The DAI does not rely on the assignment of weights or probabilities and is a purely evaluation-based approach that utilises the rationality of computational agents in order

to conduct sensitivity analysis. This approach ensures that the outputs from sensitivity analyses includes logical reasoning about all the arguments found within an assessment, which may or may not be the case when the same task is completed by a human agent.

Another closely aligned set of literature is the work conducted on argument strength. Strength has been represented previously through the arbitrary assignment of a *weights* to arguments [9]. Ranking-based semantics [47, 7, 179, 10, 32, 91, 31] also consider argument strength and more pertinently, they produce a rank order of arguments based on their relative strength, on an ordinal scale, such that the strength of arguments can be compared. The DAI is very much similar to this approach because the diagnosticity vector fully captures the relative importance of arguments by measuring how the distribution of mutual information scores changes after the removal of arguments.

One could argue that the strength of an argument could be described as a function of the labellings where that argument is acceptable such that its strength depends on whether it can defend itself against attack, if it is attacked at all. However, the sets of arguments that are critical to an argumentation framework's evaluation is a different notion. In [33], Booth *et al.* study the sets of arguments whose acceptability is critical for evaluation other arguments, such that observing the labels of those arguments resolves the acceptability of others. The DAI is capable of identifying those critical arguments numerically through the quantification of change in the distribution of mutual information across partitions of arguments. For instance, in Figure 6.5, the arguments r , s , and t seem to determine the labels of arguments p and q . Thus, the DAI is another step towards resolving the set of critical arguments, which are arguments that an analyst should study in more detail in order to understand whether their reasoning and resulting probabilistic judgements are robust.

One final, interesting set of work pertains to the value of the addition of arguments to argumentation frameworks. As discussed in Chapter 2, in [189], the outputs from Li, Oren, and Norman's probabilistic argumentation frameworks [133] were combined with a utility function to quantify the value of added arguments in order to aid in RFI generation and intelligence collection, two fundamental parts of the *Intelligence Cycle*. The proposal was then extended to a simple, real-world example in [190] to demonstrate that the method has practical benefit. The notion of adding arguments into argumentation frameworks and measuring the resulting change is an interesting idea that, with some slight alterations, the DAI is capable of. While the addition of arguments is a task left for future work, the objective from such a procedure could sensibly be to aid in RFI generation, maximise the amount of confusion in the labellings of an argumentation framework within, say, a deception operation, or more broadly speaking, to

identify the most important arguments to attack within a debate.

6.6 Conclusions and Future Work

The DAI, a novel approach presented in this chapter, emulates sensitivity analysis by employing abstract argumentation. Arguments were partitioned into dichotomous sets and mapped to argument vectors. After evaluation, the resulting labellings were transformed into labelling vectors, enabling the computation of joint and marginal acceptance probabilities of partitions of argument labels, derived from a set of probability spaces. Those probabilities were then employed in the calculation of mutual information for segments of arguments for up to half of the powerset, providing a thorough quantification of what could be inferred from a set labellings. Arguments were then sequentially removed from an initial argumentation framework and the distribution of mutual information was calculated for partitions of argument labels within sensitive graphs. The change in mutual information as a result of the removal of an argument was represented by the diagnosticity score for each partition, and the scores for all partitions were then stored in a diagnosticity vector. The approach can be split into two computational tasks. Four algorithms, two serial and two parallel implementations, were provided to make the method concrete. Results from an abstract example, a software application to visualise the results, and the Operation Fortitude-South scenario were presented and discussed. The DAI was shown to be capable of identifying the most diagnostic arguments, whilst also numerically quantifying notions of the outputs from a semantics. The technique provides a holistic quantification of the sensitivity, dependence, and robustness of an argumentation framework's evaluation using arguments that were removed. Such a tool should provide benefit to intelligence analysts by algorithmically identifying diagnostic arguments found within an analysis, and guide where to focus their efforts.

Future work could, for instance, look to: extend the DAI to include more flavour-some argumentation frameworks; incorporate the heuristic in Section 6.4.3.1 so that the approach excludes all sceptically labelled arguments from its computations; and measure the strong scaling speedup for argumentation frameworks with a myriad of labellings. While the technique presented in this chapter was intended for use within intelligence settings, another avenue for future work could investigate different domains in which the DAI could be applied. The technique functions within scenarios that contain conflicting information, where a reasoner wishes to rank the relative importance of arguments based on the dependence of their conclusions on each argument.

For instance, the DAI could aid in decision and deliberation problems, or in the analysis of debates. Removing multiple arguments is interesting and should be considered because it might be the case that the removal of one argument on its own is not diagnostic, but the elimination of two or more is. Finally, the addition of arguments is intriguing and may help support the intelligence community in RFI generation and intelligence collection in the future.

Part III

Summary of Contributions

The work presented in this thesis has demonstrated that AI tools, techniques, and algorithms could be employed to provide analysts with reasoning support within defence, security, and intelligence settings. A number of proposals were made that concentrated on hypothesis representation and generation in Part I and hypothesis evaluation in Part II. In this part, I summarise the contributions made throughout this thesis and discuss how the work proposed in each chapter addresses the research question and goals defined in Chapter 1. I then provide some possible areas for future work, focusing mainly on how to combine the techniques proposed in each chapter to provide analysts with superior reasoning systems.

Chapter 7

Conclusions and Future Direction

In this chapter, I offer a summary of the contributions made by the work presented throughout this thesis and some potential avenues for future work.

7.1 Summary of Contributions

Recall the aim of this thesis, as defined in Chapter 1, was to endeavour to answer the following research question:

By what means may AI be employed for hypothesis generation and evaluation within defence, security, and intelligence settings?

This thesis has considered two fundamental aspects within the *analysis* stage of the *Intelligence Cycle*, namely; hypothesis representation and generation in Part I; and hypothesis evaluation in Part II. In Section 1.2, a number of objectives were laid out in an attempt to answer the question posed. I now summarise the contributions and detail how the proposals posited in Chapters 3 to 6 address each of the objectives within this thesis.

Part I was comprised of Chapters 3 and 4. Consider first the proposal made for an argumentation scheme from multiple causes to effect in Chapter 3. The scheme is a characterisation of presumptive reasoning and should provide intelligence analysts with a concrete method for representing their reasoning about items of information,

evidence, and assumptions that support hypotheses. Drawing upon Walton's argument from cause to effect [236], the proposed scheme extends its level of specificity so that it can be employed within intelligence settings where analysts must reason with probabilities and under uncertainty. The scheme is accompanied by a set of critical questions which could aid the analyst in RFI generation, or the instantiation of attacks within abstract formalisms. The scheme is defeasible in nature and can be revised as new information comes to light, or as the probabilities and uncertainties within the scheme change. Moreover, the scheme is presumptive, in that the premises are believed to be true – within a probability range chosen by the analyst and corresponding to the quantities defined in the Probability Yardstick in Section 2.1.4 – unless there are other propositions that they are in conflict with, which would change the likelihood of a given proposition. In presenting the proposal for an argument from multiple causes to effect, I have addressed the first goal of this research:

“To provide a theory of representation for hypotheses that are supported by information, evidence, and assumptions that accounts for the defeasible nature of the reasoning, probability, and uncertainty inherent within intelligence analyses.”

Chapter 4 examined a number of state-of-the-art large language models and employed a bespoke dataset of hypotheses (labels) supported by information (features) to assess whether such a technology was capable of generating a diverse set of plausible hypotheses. The dataset contained intelligence reports concerning the current conflagration between Russia and Ukraine. Large language models were shown to be capable of generating hypotheses within an intelligence setting. Whether those hypotheses were *good*, according to Pherson and Heuer Jr.'s definition in Chapter 1, is questionable. Moreover, as the results demonstrated, the majority of the completions from models had a high probability of hallucination, so whether large language models can be reliably used for hypothesis generation is unanswered. Language models perform well at tasks for which they have been trained such as many of the tasks that I listed in Section 4.1. However, they perform relatively poorly at reasoning tasks, and did perform poorly whilst conducting the abstractive reasoning task in Chapter 4. The work conducted in this chapter illustrated that while large language models can generate hypotheses, whether current models can generate a diverse set of *good* and reliable (i.e., without hallucination) for more than one intelligence scenario is, respectively, questionable and unanswered, thus, addressing the second research goal:

“To investigate the feasibility of employing large language models for hypothesis

generation within an intelligence domain.”

Moving onto to the proposals made in Part II of this thesis. In Chapter 5, a formal method was proposed for the refinement of intelligence analysts’ prior beliefs about the probability of hypotheses, using the outputs from semantic evaluation of probabilistic argumentation frameworks [133] and Bayes’ theorem [18]. The technique enables the incorporation of uncertainty in the probabilistic judgements about hypotheses, whilst also allowing analysts to assign a confidence level to each information argument and argument with an embedded hypothesis instantiated from an analysis. The approach was shown to produce both sensible and intuitive outputs from frameworks instantiated within both abstract and practical settings. The approach defined in Chapter 5 addresses the third research goal:

“To design and develop a formal account capable of refining an intelligence analyst’s initial probabilistic judgements by conditioning their prior beliefs on the acceptability of available information. The approach should be tested on a real-world example to assess the potential performance of the proposal in practice.”

Chapter 6 proposed a technique that employed argumentation for sensitivity analysis, step seven of the ACH (Section 2.1.5.2). The approach uses the outputs from semantic evaluation of Dung’s abstract argumentation frameworks [70] in order to compute the probability of partitions of argument labels that were subsequently employed in mutual information [200, 59] calculations, before and after the removal of an argument of interest. The change in the distribution of mutual information between segments of argument labels was assumed to be synonymous with the notion of an argument’s diagnosticity. The proposal explores all the partitions that produce unique mutual information scores, providing a holistic quantification of the sensitivity, robustness, and dependence of an argumentation framework’s conclusions to change caused by the removal of arguments. The DAI is an evaluation-based approach that uses the rationality, coherence, and logic of computational argumentation to identify diagnostic arguments. The approach should alleviate the reliance on human effort whilst analysts conduct sensitivity analyses. The diagnosticity vector output by the DAI was assessed using an abstract example, a software application to show that results from the approach are easy for the analyst to obtain and visualise, and the Operation Fortitude-South argumentation framework, a real-world example. Chapter 6, therefore, answers the last goal of this research:

“To design and develop explainable algorithms which quantify the sensitivity, robustness, and dependence of the conclusions drawn from an intelligence analysis based on the arguments present within it, and evaluate the proposed approach using a real-world example from the intelligence domain.”

Finally, in the introduction to Part II, the methodology for the manual argument mining procedure undertaken on data from a real-world deception scenario, called Operation Fortitude-South, was presented. Arguments from the example were extracted from the ACH matrix in Appendix B.1.5 and represented using a number of argumentation schemes, and the critical questions were employed to instantiate attacks between those arguments. The annotated argument data was stored in the AIF, which enabled the acceptability of arguments to be discovered through the translation of the AIF to an ASPIC⁺ argumentation theory and then to Dung argument system, using the process described in Section 2.3.2.3. The introduction to this part confirmed that analysis stored within an ACH matrix can be transformed to refined argument data (Section 2.3.3.1) to enable argumentation to be used on real-world scenarios, partially addressing the last two goals of this thesis.

7.2 Future Directions

The results presented throughout this thesis have provided a number of interesting avenues for future work and in this section, I discuss some of those possible areas.

Future work from this thesis can be divided into two categories, namely: work pertaining to the extension of the research within each chapter; and the combination of the techniques in each chapter to create superior reasoning systems for intelligence analysts. Here, I only consider the latter as it is the most interesting, and because of the diverse range of techniques proposed throughout this thesis. Moreover, many of the ideas to extend the work at the chapter-level are included within the conclusions of each chapter. However, I will reiterate some of the previously mentioned avenues from individual chapters when discussing how the work from this thesis might be combined, for the reader's benefit.

While the tools and techniques proposed within this thesis were designed, developed, and tested in isolation, they do not need to be implemented nor applied independently. First and foremost, as the *finished intelligence product* is a fundamental part of the *Intelligence Cycle*, the proposals for hypothesis generation in Chapter 4, prior probability refinement in Chapter 5, and sensitivity analysis in Chapter 6 could be, re-

spectively, employed to: generate hypotheses that analysts evaluate; revise the resulting probabilistic judgements using the outputs from semantic evaluation of PrAFs; and identify diagnostic arguments. The outputs from all three aforesaid techniques could be included with intelligence reports. Moreover, the argumentation scheme proposed in Chapter 3 could also be included with an intelligence report in order to help stakeholders, commanders, and decision makers understand ASAs' reasoning through the representation of propositions and inference from real-world data within intelligence analyses.

It is also easy to think about how the argumentation scheme for an argument from multiple causes to effect, as proposed in Chapter 3, and the technique for the refinement of analysts' prior beliefs about hypotheses, as put forth in Chapter 5, could be combined. Hypotheses instantiated as conclusions from real-world argument data could be represented by the scheme proposed in Chapter 3, which might produce better posterior probabilities than the ones presented for the Operation Fortitude-South argumentation framework in Section 5.4.2. Although the aforementioned avenue regards how the scheme in Chapter 3 could enhance the proposal in Chapter 5, the technique in Chapter 5 can improve the probabilistic judgements instantiated within the scheme in Chapter 3. For instance, in Section 3.1.4, I describe how, due to the nature of intelligence analyses, it is extremely challenging to reliably compute the joint probability of information, evidence, assumptions, and a hypothesis, and the conditional probability of the information, evidence, and assumptions conditioned on a hypothesis. However, the proposal for the refinement of priors, made in Chapter 5, could sensibly be employed to compute those quantities. Furthermore, as I argued in Section 5.3.5, the likelihood of an information argument given the acceptability of the hypothesis of interest can be quantified using the probability of non-existence and defeat of all other hypotheses. Thus, the addition of another proposition within the argument from multiple causes to effect that contains an assertion about the defeat of all other hypotheses, and the probability of that event, might help to make the scheme's reasoning even more explicit than it already is.

Recall from Section 4.2.2.1 that the intelligence reports published on the X, formerly Twitter, account of the United Kingdom's Ministry of Defence contained strategic and operational hypotheses, and tactical hypotheses were not observed within the dataset, most likely because of their level of classification. In [225], the United States' Department of Defence define strategic, operational, and tactical intelligence and describe how tactical intelligence should inform operational objectives, and operational intelligence should also inform strategic objectives. As I briefly stated in Chapter 4, the relationship between between strategic, operational, and tactical hypotheses seemingly

forms a tree structure, where tactical hypotheses that are evaluated using tactical intelligence could aid in the evaluation of operational hypotheses which, when evaluated using operational intelligence, help in the assessment of strategic hypotheses and their probabilities. Although defining such a relationship is a task left for future work, there has been some work within the computational argumentation community that considers hierarchical reasoning in legal cases using abstract dialectical frameworks [35], where the *outcome* from a case (i.e., the root or level zero of the tree) is determined by the *issues* (i.e., level one of the tree) and the acceptability of a set of abstract *factors* (i.e., the leaves or level two of the tree) of a given case [3, 2, 16]. Mapping tactical, operational, and strategic hypotheses to this formalism would be no small undertaking. However, through close collaboration with the intelligence community, a number argumentation schemes for domain- and scenario-independent hypotheses, for each type of hypothesis at all levels of the hierarchy, could be created, allowing the outcome of a strategic hypothesis to be determined. Nevertheless, there would be issues with this approach because hypotheses contained within *finished intelligence products* are required to be accompanied by a qualitative probability range, as discussed in Section 2.1.4. The proposed formalism would be required to incorporate both probability and uncertainty, a task that abstract dialectical frameworks are not capable of in their current state of development. One solution to this would be to extend the capabilities of abstract dialectical frameworks so that the formalism includes notions of probability at each node and edge within the graph. Another approach might instead be to move away from abstract dialectical frameworks and study whether the probabilistic outputs from analysis of tactical, operational, and strategic hypotheses could be revised, using the approach in Chapter 5, and the tree structure instantiated within a Bayesian network [111] in order to understand whether tactical and operational hypotheses could be employed in the evaluation of strategic ones. This approach would assume that each level of the tree is its own sub-analysis, where an analyst has instantiated and evaluated an ACH matrix at each level of the tree hierarchy, transformed the ACH matrix to an argumentation framework, and arrived at tentative conclusions about the prior probability of each hypothesis (i.e., each node within the tree) so that those probabilities can be refined, using the technique in Chapter 5.

Finally, I have said little throughout this thesis about the types of data instantiated within arguments for a hypothesis or its negation. Intelligence analysts reason using multi-modal data. In Section 2.1.2, I outline the five types of intelligence which are usually collected using one of the five disciplines of intelligence collection [139]. For example, IMINT analysts collect and reason with images, HUMINT agents observe events and report them using natural language, and SIGINT analysts intercept signals

from communication and non-communication devices, to name a few pertinent examples. Moreover, much of the information that we, as humans, consume and converse about is multi-modal. Moving forward, the field of computational models of argument as a whole should study argumentation that is comprised of more than just natural language corpora because, as is often the case within intelligence settings, the propositions instantiated as premises within arguments might not always take the form of a string.

The avenues identified for future work in this section are just some of the potential options resulting from the work presented in this thesis. While this section discussed how one might combine the different techniques in each of the chapters of this thesis, there are also a number of interesting chapter-level extensions that would benefit from further study. The results presented throughout this thesis demonstrate that both the theoretical and practical work from the field of computational argumentation are showing much promise for use within defence, security, and intelligence settings.

Bibliography

- [1] Y. Tang, N. Oren, and K. Sycara. “Markov Argumentation Random Fields”. In: *Proc. of AAAI Conference on Artificial Intelligence*. 2016.
- [2] L. Al-Abdulkarim. “Representation of case law for argumentative reasoning”. PhD thesis. University of Liverpool, 2017.
- [3] L. Al-Abdulkarim, K. Atkinson, and T. Bench-Capon. “A methodology for designing systems to reason with legal cases using abstract dialectical frameworks”. In: *Artificial Intelligence and Law* 24 (2016), pp. 1–49.
- [4] A. V. Aho. “Algorithms for Finding Patterns in Strings”. In: *Algorithms and Complexity*. Ed. by J. v. Leeuwen. Handbook of Theoretical Computer Science. Amsterdam: Elsevier, 1990, pp. 255–300.
- [5] T. Alhindi and D. Ghosh. ““Sharks are not the threat humans are”: Argument Component Segmentation in School Student Essays”. In: *Proc. of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*. Ed. by J. Burstein et al. Association for Computational Linguistics, 2021, pp. 210–222.
- [6] G. M. Amdahl. “Validity of the single processor approach to achieving large scale computing capabilities”. In: *Proceedings of the April 18-20, 1967, Spring Joint Computer Conference*. AFIPS. Atlantic City, New Jersey: Association for Computing Machinery, 1967, pp. 483–485.
- [7] L. Amgoud and J. Ben-Naim. “Ranking-Based Semantics for Argumentation Frameworks”. In: *Scalable Uncertainty Management*. Vol. 8078. Springer, 2013, pp. 134–147.
- [8] L. Amgoud and C. Cayrol. “A Reasoning Model Based on the Production of Acceptable Arguments”. In: *Annals of Mathematics and Artificial Intelligence* 34 (2002), pp. 197–215.

-
- [9] L. Amgoud and C. Cayrol. “Integrating preference orderings into argument-based reasoning”. In: *Proc. of Qualitative and Quantitative Practical Reasoning*. Vol. 1244. Springer, 1997, pp. 159–170.
 - [10] L. Amgoud et al. “Ranking arguments with compensation-based semantics”. In: *Proc. of Knowledge Representation and Reasoning*. Cape Town, South Africa: AAAI Press, 2016, pp. 12–21.
 - [11] U. o. D. Argumentation Research Group School of Computing. *The Argument Interchange Format (AIF) Specification*. Retrieved from: <https://www.arg-tech.org/wp-content/uploads/2011/09/aif-spec.pdf>. (Accessed on: 25.06.2024). 2011.
 - [12] Aristophanes and D. MacDowell. *Wasps*. Clarendon paperbacks. Clarendon Press, 1988. ISBN: 9780198144656.
 - [13] Aristotle. *On Sophistical Refutations*. ReadHowYouWant.com, Limited, 2006.
 - [14] U. S. Army. “Infantry Platoon and Squad”. In: *ATP 3-21.8* (2016).
 - [15] K. Atkinson and T. Bench-Capon. “Abstract Argumentation Scheme Frameworks”. In: *Artificial Intelligence: Methodology, Systems, and Applications*. Ed. by D. Dochev, M. Pistore, and P. Traverso. Berlin, Heidelberg: Springer, 2008, pp. 220–234.
 - [16] K. Atkinson and T. Bench-Capon. “ANGELIC II: An Improved Methodology for Representing Legal Domain Knowledge”. In: *Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law*. Braga, Portugal: Association for Computing Machinery, 2023, pp. 12–21.
 - [17] P. Baroni and M. Giacomin. “Semantics of Abstract Argument Systems”. In: *Argumentation in Artificial Intelligence*. Ed. by G. Simari and I. Rahwan. Boston, MA, USA: Springer, 2009, pp. 25–44.
 - [18] T. Bayes. “An essay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, F. R. S. communicated by Mr. Price, in a letter to John Canton, A. M. F. R. S”. In: *Philosophical Transactions of the Royal Society of London* 53 (1763), pp. 370–418.
 - [19] T. Bench-Capon and H. Prakken. “Using Argument Schemes for Hypothetical Reasoning in Law”. In: *Artificial Intelligence and Law* 18 (2010), pp. 153–174.
 - [20] T. J. M. Bench-Capon. “Persuasion in Practical Argument Using Value-based Argumentation Frameworks”. In: *Journal of Logic and Computation* 13.3 (2003), pp. 429–448.

-
- [21] J. Bentahar, B. Moulin, and M. Bélanger. “A taxonomy of argumentation models used for knowledge representation”. In: *Artificial Intelligence Review* 33 (2010), pp. 211–259.
- [22] R. A. Best Jr. “Intelligence Information: Need-to-Know vs. Need-to-Share”. In: *Congressional Research Service: Report for Congress* (2011).
- [23] F. Bex et al. “On logical specifications of the Argument Interchange Format”. In: *Journal of Logic and Computation* 23.5 (2012), pp. 951–989.
- [24] F. Bex, T. Bench-Capon, and K. Atkinson. “Did he jump or was he pushed?”. In: *Artificial Intelligence and Law* 17 (2009), pp. 79–99.
- [25] F. Bex et al. “A hybrid formal theory of arguments, stories and criminal evidence”. English. In: *Artificial Intelligence and Law* 18.2 (June 2010). Copyright 2010 Elsevier B.V., All rights reserved., pp. 123–152. ISSN: 0924-8463. DOI: 10.1007/s10506-010-9092-x.
- [26] D. Billman et al. “Collaborative intelligence analysis with CACHE and its effects on information gathering and cognitive bias”. In: *Human Computer Interaction Consortium Workshop*. Vol. 35. Snow Mountain, CO, 2006.
- [27] O. Biran and O. Rambow. “Identifying Justifications in Written Dialogs by Classifying Text as Argumentative”. In: *International Journal of Semantic Computing* 5 (2011), pp. 363–381.
- [28] J. Bommarito et al. *GPT as Knowledge Worker: A Zero-Shot Evaluation of (AI)CPA Capabilities*. 2023. DOI: <https://doi.org/10.48550/arXiv.2301.04408>.
- [29] M. Bommarito II and D. M. Katz. *GPT Takes the Bar Exam*. 2022. DOI: 10.48550/arXiv.2212.14402.
- [30] A. Bondarenko, F. Toni, and R. Kowalski. “An Assumption-based Framework for Non-monotonic Reasoning”. In: *Proc. of the International Workshop on Logic Programming and Non-monotonic Reasoning*. Ed. by L. Pereira and A. Nerode. 1993, pp. 171–189.
- [31] E. Bonzon et al. “A Parametrized Ranking-Based Semantics Compatible with Persuasion Principles”. In: *Argument and Computation* 12.1 (2021), pp. 49–85.
- [32] E. Bonzon et al. “Argumentation Ranking Semantics Based on Propagation”. In: *Proc. of Computational Models of Argument*. Vol. 287. Frontiers in Artificial Intelligence and Applications. Postdam, Germany: IOS Press, 2016, pp. 139–150.

- [33] R. Booth et al. “Complexity Properties of Critical Sets of Arguments”. In: *Proc. of Computational Models of Argument*. Vol. 266. Frontiers in Artificial Intelligence and Applications. Scottish Highlands, UK: IOS Press, 2014, pp. 173–184.
- [34] B. Brewin. *What’s Brewin: Now it’s ‘responsibility to provide’*. Retrieved from: <https://www.nextgov.com/cybersecurity/2008/04/whats-brewin-now-its-responsibility-to-provide/41967/>. (Accessed on: 22.01.2022). 2008.
- [35] G. Brewka and S. Woltran. “Abstract dialectical frameworks”. In: *Proc. of Knowledge Representation and Reasoning*. Toronto, Ontario, Canada: AAAI Press, 2010, pp. 102–111.
- [36] K. Budzynska and C. Reed. “Argumentum Ad Alia: argument structure of arguing about what others have said”. In: *Synthese* 201 (2023), pp. 1–29.
- [37] K. Budzynska and C. Reed. “Speech acts of argumentation: Inference anchors and peripheral cues in dialogue”. In: *Proc. of the Workshop on Computational Models of Natural Argument*. 2011, pp. 3–10.
- [38] K. Burns. “Bayesian HELP: Assisting Inferences in All-Source Intelligence”. In: *Papers from the 2015 AAAI Fall Symposium*. 2015.
- [39] K. Burns. *Integrated Cognitive-neuroscience Architectures for Understanding Sensemaking: Phase 2 Challenge Problem Design and Test Specification*. Tech. rep. MITRE Technical Report, MTR140412, 2014.
- [40] E. Cabrio and S. Villata. “A Natural Language Bipolar Argumentation Approach to Support Users in Online Debate Interactions”. In: *Argument and Computation* 4.3 (2013), pp. 209–230.
- [41] M. Caminada. “An algorithm for computing semi-stable semantics”. In: *In Proc. of the Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*. Springer Verlag, Berlin, 2007, pp. 222–234.
- [42] M. Caminada. “On the Issue of Reinstatement in Argumentation”. In: *Logics in Artificial Intelligence*. Ed. by M. Fisher et al. Berlin, Heidelberg: Springer, 2006, pp. 111–123.
- [43] M. Caminada and D. M. Gabbay. “A Logical Account of Formal Argumentation”. In: *Studia Logica: An International Journal for Symbolic Logic* 93.2/3 (2009), pp. 109–145.

-
- [44] Canadian Forces Intelligence Command. *Aide-Mémoire on Intelligence Analysis Tradecraft*. Department of National Defence, Canada, 2015.
- [45] L. Cao. “AutoAM: An End-To-End Neural Model for Automatic and Universal Argument Mining”. In: *Advanced Data Mining and Applications*. Ed. by X. Yang et al. Springer Nature Switzerland, 2023, pp. 517–531.
- [46] Careers In The Military. *All-Source Intelligence Specialists*. Retrieved from: <https://www.careersinthemilitary.com/career-detail/all-source-intelligence-specialists>. (Accessed on: 22.03.2022).
- [47] C. Cayrol and M. C. Lagasquie-Schiex. “Graduality in Argumentation”. In: *Journal of Artificial Intelligence Research* 23 (2005), pp. 245–297.
- [48] C. Cayrol and M. C. Lagasquie-Schiex. “On the Acceptability of Arguments in Bipolar Argumentation Frameworks”. In: *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*. 2005, pp. 378–389.
- [49] T. Chakrabarty, V. Padmakumar, and H. He. “Help me write a Poem - Instruction Tuning as a Vehicle for Collaborative Poetry Writing”. In: *Proc. of the Conference on Empirical Methods in Natural Language Processing*. Ed. by Y. Goldberg, Z. Kozareva, and Y. Zhang. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, 2022, pp. 6848–6863.
- [50] W. Chang et al. “Restructuring structured analytic techniques in intelligence”. In: *Intelligence and National Security* 33.3 (2018), pp. 337–356.
- [51] M. Chen et al. *Evaluating Large Language Models Trained on Code*. 2021. DOI: <https://doi.org/10.48550/arXiv.2107.03374>.
- [52] C. Chesñevar et al. “Towards an argument interchange format”. In: *The Knowledge Engineering Review* 21.4 (2006), pp. 293–316.
- [53] R. M. Clark. *Intelligence Collection*. Washington, DC: CQ Press, 2013.
- [54] C. von Clausewitz. *On War (Book One): Edited and Translated by Michael Howard and Peter Paret*. Princeton University Press. Princeton, New Jersey, 1832.
- [55] M. L. Cobo, D. C. Martínez, and G. R. Simari. “Acceptability in Timed Frameworks with Intermittent Arguments”. In: *Artificial Intelligence Applications and Innovations*. Ed. by L. Iliadis, I. Maglogiannis, and H. Papadopoulos. Berlin, Heidelberg: Springer, 2011, pp. 202–211.

- [56] M. L. Cobo, D. C. Martínez, and G. R. Simari. “An approach to timed abstract argumentation”. In: *Proc. of International Workshop of Non-monotonic Reasoning*. Vol. 2010. Toronto, Canada. 2010.
- [57] M. L. Cobo, D. C. Martínez, and G. R. Simari. “On Admissibility in Timed Abstract Argumentation Frameworks”. In: *Proc. of the European Conference on Artificial Intelligence*. IOS Press, 2010, pp. 1007–1008.
- [58] S. Coulthart. “Why do analysts use structured analytic techniques? An in-depth study of an American intelligence agency”. In: *Intelligence and National Security* 31.7 (2016), pp. 933–948.
- [59] T. Cover and J. Thomas. “Elements of Information Theory”. In: John Wiley & Sons, 2005. Chap. Entropy, Relative Entropy, and Mutual Information, pp. 13–55.
- [60] G. D’Aniello et al. “Application of Granular Computing and Three-way decisions to Analysis of Competing Hypotheses”. In: *Proc. of the IEEE International Conference on Systems, Man, and Cybernetics*. 2016, pp. 1650–1655.
- [61] J. Dagdelen et al. “Structured information extraction from scientific text with large language models”. In: *Nature Communications* 15 (1418 2024).
- [62] L. Dalcín, R. Paz, and M. Storti. “MPI for Python”. In: *Journal of Parallel and Distributed Computing* 65.9 (2005), pp. 1108–1115.
- [63] Defence Science and Technology Laboratory. *eleatics: Knowledge Representation and Reasoning*. Retrieved from: <https://dstl.github.io/eleatics/>. (Accessed on: 15.12.2022).
- [64] *Department of Defence: News Briefing - Secretary Rumsfeld and General Myers*. Retrieved from: <http://archive.defense.gov/Transcripts/Transcript.aspx?TranscriptID=2636>. (Accessed on: 30.09.2020).
- [65] J. Devlin et al. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *Proc. of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Ed. by J. Burstein, C. Doran, and T. Solorio. Vol. 1. Minneapolis, Minnesota: Association for Computational Linguistics, 2019, pp. 4171–4186.
- [66] L. Donovan. *Strategic Deception: Operation Fortitude*. Lucknow Books, 2014.
- [67] M. C. Duke. “Probability and confidence: How to improve communication of uncertainty about uncertainty in intelligence analysis”. In: *Journal of Behavioral Decision Making* 37.1 (2024).

-
- [68] K. A. Duncan and J. L. Wilson. *A multinomial-dirichlet model for analysis of competing hypotheses*. Vol. 28(6). 2008, pp. 1699–709.
- [69] P. Dung and P. Thang. “Towards (Probabilistic) Argumentation for Jury-based Dispute Resolution.” In: *Proc. of the Conference on Computational Models of Argument*. 2010, pp. 171–182.
- [70] P. M. Dung. “On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games”. In: *Artificial Intelligence* 77.2 (1995), pp. 321–357.
- [71] P. E. Dunne et al. “Weighted argument systems: Basic definitions, algorithms, and complexity results”. In: *Artificial Intelligence* 175.2 (2011), pp. 457–486.
- [72] R. Duthie, K. Budzynska, and C. Reed. “Mining Ethos in Political Debate”. In: *Proc. of the Conference on Computational Models of Argument*. Ed. by P. Baroni et al. Vol. 287. Frontiers in Artificial Intelligence and Applications. Netherlands: IOS Press, 2016, pp. 299–310.
- [73] J. Eckle-Kohler, R. Kluge, and I. Gurevych. “On the Role of Discourse Markers for Discriminating Claims and Premises in Argumentative Discourse”. In: *Proc. of the Conference on Empirical Methods in Natural Language Processing*. Ed. by L. Màrquez, C. Callison-Burch, and J. Su. Lisbon, Portugal: Association for Computational Linguistics, 2015, pp. 2236–2242.
- [74] W. Edwards et al. “Probabilistic Information Processing Systems: Design and Evaluation”. In: *IEEE Transactions on Systems Science and Cybernetics* 4.3 (1968), pp. 248–265.
- [75] F. H. van Eemeren and R. Grootendorst. *Speech Acts in Argumentative Discussions*. Berlin, New York: De Gruyter Mouton, 1984.
- [76] F. H. van Eemeren et al. *Handbook of Argumentation Theory*. Springer Dordrecht, 2014.
- [77] S. Eger, J. Daxenberger, and I. Gurevych. “Neural End-to-End Learning for Computational Argumentation Mining”. In: *Proc. of the Annual Meeting of the Association for Computational Linguistics*. Ed. by R. Barzilay and M.-Y. Kan. Vancouver, Canada: Association for Computational Linguistics, 2017, pp. 11–22.
- [78] V. W. Feng and G. Hirst. “Classifying arguments by scheme”. In: *Proc. of the Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Ed. by D. Lin, Y. Matsumoto, and R. Mihalcea. Portland, Oregon, USA: Association for Computational Linguistics, 2011, pp. 987–996.

- [79] C. E. Fisk. “The Sino-Soviet Border Dispute: A Comparison of the Conventional and Bayesian Methods for Intelligence Warning”. In: *Center for the Study of Intelligence, CIA Archives* (1996).
- [80] E. Florou et al. “Argument extraction for supporting public policy formulation”. In: *Proc. of the Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*. Ed. by P. Lendvai and K. Zervanou. Sofia, Bulgaria: Association for Computational Linguistics, 2013, pp. 49–54.
- [81] J. Fox, P. Krause, and M. Elvang-Gøransson. “Argumentation as a General Framework for Uncertain Reasoning”. In: *Uncertainty in Artificial Intelligence*. Ed. by D. Heckerman and A. Mamdani. Morgan Kaufmann, 1993, pp. 428–434.
- [82] J. A. Friedman, J. S. Lerner, and R. Zeckhauser. “Behavioral Consequences of Probabilistic Precision: Experimental Evidence from National Security Professionals”. In: *International Organization* 71.4 (2017), pp. 803–826.
- [83] A. Galassi, M. Lippi, and P. Torrioni. “Multi-Task Attentive Residual Networks for Argument Mining”. In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 31 (2023), pp. 1877–1892.
- [84] M. García-Villalba and P. Saint-Dizier. “A Framework to Extract Arguments in Opinion Texts”. In: *International Journal of Cognitive Informatics and Natural Intelligence* 6 (2012), pp. 62–87.
- [85] T. van Gelder. “The rationale for Rationale™”. In: *Law, Probability and Risk* 6.1-4 (2007), pp. 23–42.
- [86] T. Givón. *Topic Continuity in Discourse: A Quantitative Cross-language Study*. Typological studies in language. J. Benjamins Publishing Company, 1983.
- [87] A. Glaese et al. *Improving alignment of dialogue agents via targeted human judgements*. 2022. DOI: <https://doi.org/10.48550/arXiv.2209.14375>.
- [88] T. F. Gordon, H. Prakken, and D. Walton. “The Carneades model of argument and burden of proof”. In: *Artificial intelligence* 171.10-15 (2007), pp. 875–896.
- [89] J. E. Grimes. *The Thread of Discourse*. Mouton, 1975. ISBN: 9789027931641.
- [90] L. Groarke, C. Tindale, and L. Fisher. *Good Reasoning Matters!: A Constructive Approach to Critical Thinking*. Oxford University Press, 1997.
- [91] D. Grossi and S. Modgil. “On the graded acceptability of arguments in abstract and instantiated argumentation”. In: *Artificial Intelligence* 275 (2019), pp. 138–173.

-
- [92] J. L. Gustafson. “Reevaluating Amdahl’s law”. In: *Communications of the ACM* 31.5 (1988), pp. 532–533.
- [93] J. Haase and P. H. Hanel. “Artificial muses: Generative artificial intelligence chatbots have risen to human-level creativity”. In: *Journal of Creativity* 33.3 (2023), p. 100066.
- [94] I. Habernal and I. Gurevych. “Exploiting Debate Portals for Semi-Supervised Argumentation Mining in User-Generated Web Discourse”. In: *Proc. of the Conference on Empirical Methods in Natural Language Processing*. Ed. by L. Màrquez, C. Callison-Burch, and J. Su. Lisbon, Portugal: Association for Computational Linguistics, 2015, pp. 2127–2137.
- [95] S. Hamilton. *Blind Judgement: Agent-Based Supreme Court Modelling With GPT*. 2023. DOI: <https://doi.org/10.48550/arXiv.2301.05327>.
- [96] M. Hansen. *The Athenian Democracy in the Age of Demosthenes: Structure, Principles, and Ideology*. The ancient world. University of Oklahoma Press, 1999.
- [97] C. R. Harris et al. “Array programming with NumPy”. In: *Nature* 585.7825 (2020), pp. 357–362.
- [98] A. C. Hastings. “A Reformulation of the Modes of Reasoning in Argumentation”. PhD thesis. Evanston, Illinois, 1963.
- [99] A. Hautli-Janisz et al. “QT30: A Corpus of Argument and Conflict in Broadcast Debate”. In: *Proc. of the Language Resources and Evaluation Conference*. Ed. by N. Calzolari et al. Marseille, France: European Language Resources Association, 2022, pp. 3291–3300.
- [100] P. He, J. Gao, and W. Chen. *DeBERTaV3: Improving DeBERTa using ELECTRA-Style Pre-Training with Gradient-Disentangled Embedding Sharing*. 2021. DOI: <https://doi.org/10.48550/arXiv.2111.09543>.
- [101] R. Heuer Jr. *Psychology of Intelligence Analysis*. Center for the Study of Intelligence, 1999. ISBN: 9780160590351.
- [102] J. Hirschberg and D. Litman. “Empirical Studies on the Disambiguation of Cue Phrases”. In: *Computational Linguistics* 19.3 (1993). Ed. by J. Hirschberg, pp. 501–530.
- [103] D. Hitchcock. “Enthymematic Arguments”. In: *Informal Logic* 7.2 (1985).
- [104] M. Humphreys. *Aristophanes Clouds*. College series of Greek authors. Ginn & Company, 1892.

-
- [105] T. Irani. *Plato on the Value of Philosophy: The Art of Argument in the Gorgias and Phaedrus*. Cambridge University Press, 2017.
 - [106] ISO. *ISO/IEC 14882:2011 Information technology — Programming languages — C++*. Geneva, Switzerland: International Organization for Standardization, Feb. 2012, 1338 (est.) URL: http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=50372.
 - [107] K. Y. Iu and V. M.-Y. Wong. *ChatGPT by OpenAI: The End of Litigation Lawyers?* SSRN Scholarly Paper. Rochester, NY, 2023. DOI: 10.2139/ssrn.4339839. URL: <https://papers.ssrn.com/abstract=4339839>.
 - [108] M. Janier, J. Lawrence, and C. Reed. “OVA+: An Argument Analysis Interface”. In: *Proc. of the Conference on Computational Models of Argument*. Vol. 266. Frontiers in Artificial Intelligence and Applications. 2014, pp. 463–464.
 - [109] I. L. Janis. *Groupthink: Psychological Studies of Policy Decisions and Fiascoes*. Houghton Mifflin, 1982. ISBN: 9780395317044.
 - [110] A. Janjeva, A. Harris, and J. Byrne. “The Future of Open Source Intelligence for UK National Security”. In: *Royal United Services Institute for Defence and Security Studies* (2022).
 - [111] F. V. Jensen et al. *An introduction to Bayesian networks*. Vol. 210. UCL press London, 1996.
 - [112] Y. Jo et al. “A Cascade Model for Proposition Extraction in Argumentation”. In: *Proc. of the Workshop on Argument Mining*. Ed. by B. Stein and H. Wachsmuth. Florence, Italy: Association for Computational Linguistics, 2019, pp. 11–24.
 - [113] Y. Jo et al. “Classifying argumentative relations using logical mechanisms and argumentation schemes”. In: *Transactions of the Association for Computational Linguistics* 9 (2021), pp. 721–739.
 - [114] R. Johnston. *Analytic culture in the United States intelligence community: An ethnographic study*. Central for the Study of Intelligence, 2005.
 - [115] N. Jones. “Critical epistemology for Analysis of Competing Hypotheses”. In: *Intelligence and National Security* 33.2 (2018), pp. 273–289.
 - [116] D. Kahneman. *Thinking, Fast and Slow*. Penguin Books, London, 2011.
 - [117] C. W. Karvetski, D. R. Mandel, and D. Irwin. “Improving Probability Judgment in Intelligence Analysis: From Structured Analysis to Statistical Aggregation”. In: *Risk Analysis* 40 (5 2020), pp. 1040–1057.

-
- [118] C. W. Karvetski et al. “Structuring and analyzing competing hypotheses with Bayesian networks for intelligence analysis”. In: *Journal on Decision Processes* 1.3 (2013), pp. 205–231.
 - [119] E. Kasneci et al. “ChatGPT for good? On opportunities and challenges of large language models for education”. In: *Learning and Individual Differences* 103 (2023), p. 102274.
 - [120] D. Katz et al. “GPT-4 passes the bar exam”. In: *Philosophical Transactions of the Royal Society A* 382 (2024).
 - [121] M. Kienpointner. *Alltagslogik: Struktur und Funktion von Argumentationsmustern*. Problemata (Stuttgart). Frommann-Holzboog, 1992.
 - [122] J. Kocoń et al. “ChatGPT: Jack of all trades, master of none”. In: *Information Fusion* 99 (2023), p. 101861.
 - [123] T. Kondo et al. “Bayesian Argumentation-Scheme Networks: A Probabilistic Model of Argument Validity Facilitated by Argumentation Schemes”. In: *Proc. of the Workshop on Argument Mining*. Ed. by K. Al-Khatib, Y. Hou, and M. Stede. Punta Cana, Dominican Republic: Association for Computational Linguistics, 2021, pp. 112–124.
 - [124] P. J. Krause et al. “A Logic of Argumentation for Reasoning Under Uncertainty”. In: *Computational Intelligence* 11 (1995).
 - [125] J. Lawrence and C. Reed. “Argument mining using argumentation scheme structures”. In: *Proc. of Computational Models of Argument*. Ed. by P. Baroni, M. Stede, and T. Gordon. IOS Press, 2016, pp. 379–390.
 - [126] J. Lawrence and C. Reed. “Argument Mining: A Survey”. In: *Computational Linguistics* 45.4 (2020), pp. 765–818.
 - [127] J. Lawrence, J. Visser, and C. Reed. “BBC Moral Maze: Test Your Argument”. In: *Proc. of the Conference on Computational Models of Argument*. Frontiers in Artificial Intelligence and Applications. 2018.
 - [128] J. Lawrence et al. “AIFdb: Infrastructure for the Argument Web”. In: *Proc. of the Conference on Computational Models of Argument*. Vol. 245. 2012.
 - [129] P. Lehner et al. “Confirmation Bias in Complex Analyses”. In: *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans* 38 (2008), pp. 584–592.

- [130] M. Lenz et al. “Towards an Argument Mining Pipeline Transforming Texts to Argument Graphs”. In: *Proc. of the Conference on Computational Models of Argument*. Ed. by H. Prakken et al. Vol. 326. Frontiers in Artificial Intelligence and Applications. IOS Press, 2020, pp. 263–270.
- [131] N. Leone et al. “The DLV system for knowledge representation and reasoning”. In: *ACM Transactions on Computational Logic* 7.3 (2006), pp. 499–562.
- [132] R. Levy et al. “Context Dependent Claim Detection”. In: *Proc. of International Conference on Computational Linguistics*. Ed. by J. Tsujii and J. Hajic. Association for Computational Linguistics, 2014, pp. 1489–1500.
- [133] H. Li, N. Oren, and T. Norman. “Probabilistic Argumentation Frameworks”. In: *Theories and Applications of Formal Argumentation*. Vol. 7132. 2011, pp. 1–16.
- [134] D. Liga. “Argumentative Evidences Classification and Argument Scheme Detection Using Tree Kernels”. In: *Proc. of the Workshop on Argument Mining*. Ed. by B. Stein and H. Wachsmuth. Association for Computational Linguistics, 2019, pp. 92–97.
- [135] M. Lippi and P. Torroni. “Argumentation Mining: State of the Art and Emerging Trends”. In: *ACM Transactions on Internet Technology*. 16.2 (2016). ISSN: 1533-5399.
- [136] E. Liu, G. Neubig, and J. Andreas. *An Incomplete Loop: Instruction Inference, Instruction Following, and In-context Learning in Language Models*. 2024. DOI: <https://doi.org/10.48550/arXiv.2404.03028>.
- [137] R. Liu and N. B. Shah. *ReviewerGPT? An Exploratory Study on Using Large Language Models for Paper Reviewing*. 2023. DOI: <https://doi.org/10.48550/arXiv.1907.11692>.
- [138] Y. Liu et al. *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. 2019. DOI: <https://doi.org/10.48550/arXiv.1907.11692>.
- [139] M. M. Lowenthal and R. M. Clark. *The Five Disciplines of Intelligence Collection*. SAGE Publications, 2015. ISBN: 9781452217635.
- [140] Y. Luo et al. “Synthesizing Natural Language to Visualization (NL2VIS) Benchmarks from NL2SQL Benchmarks”. In: *Proc. of the International Conference on Management of Data*. Association for Computing Machinery, 2021, pp. 1235–1247.

-
- [141] E. Maegherman et al. “Test of the analysis of competing hypotheses in legal decision-making”. In: *Applied Cognitive Psychology* 35.1 (2021), pp. 62–70.
- [142] P. Manakul, A. Liusie, and M. Gales. “SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models”. In: *Proc. of the Conference on Empirical Methods in Natural Language Processing*. Ed. by H. Bouamor, J. Pino, and K. Bali. Association for Computational Linguistics, 2023, pp. 9004–9017.
- [143] T. Mayer, E. Cabrio, and S. Villata. “Transformer-based argument mining for healthcare applications”. In: *Proc. of the European Conference on Artificial Intelligence*. IOS Press, 2020, pp. 2108–2115.
- [144] S. Menini et al. “Never Retreat, Never Retract: Argumentation Analysis for Political Speeches”. In: *Proc. of the AAAI Conference on Artificial Intelligence*. Vol. 32. 1. 2018.
- [145] J. Mensonides et al. “Automatic Detection and Classification of Argument Components using Multi-task Deep Neural Network”. In: *Proc. of the International Conference on Natural Language and Speech Processing*. Ed. by M. Abbas and A. A. Freihat. Association for Computational Linguistics, 2019, pp. 25–33.
- [146] Message Passing Interface Forum. *MPI: A Message-Passing Interface Standard Version 4.0*. Retrieved from: <https://www.mpi-forum.org/docs/mpi-4.0/mpi40-report.pdf>. (Accessed on: 19.11.2024). 2021.
- [147] M. Miotto, N. Rossberg, and B. Kleinberg. “Who is GPT-3? An exploration of personality, values and demographics”. In: *Proc. of the Workshop on Natural Language Processing and Computational Social Science*. Ed. by D. Bamman et al. Association for Computational Linguistics, 2022, pp. 218–227.
- [148] P. Mirowski et al. “Co-Writing Screenplays and Theatre Scripts with Language Models: Evaluation by Industry Professionals”. In: *Proc. of the Conference on Human Factors in Computing Systems*. Association for Computing Machinery, 2023.
- [149] R. Mochales and M. Moens. “Argumentation Mining”. In: *Artificial Intelligence and Law* 19 (2011), pp. 1–22.
- [150] S. Modgil and H. Prakken. “The ASPIC+ framework for structured argumentation: a tutorial”. In: *Argument & Computation* 5.1 (2014), pp. 31–62.

-
- [151] M. Moens et al. “Automatic detection of arguments in legal texts”. In: *Proc. of the International Conference on Artificial Intelligence and Law*. Association for Computing Machinery, 2007, pp. 225–230.
 - [152] P. K. Murukannaiah et al. “Resolving goal conflicts via argumentation-based analysis of competing hypotheses”. In: *Proc. of the International Requirements Engineering Conference*. 2015, pp. 156–165.
 - [153] H. Naveed et al. *A Comprehensive Overview of Large Language Models*. 2024. DOI: <https://doi.org/10.48550/arXiv.2307.06435>.
 - [154] H. V. Nguyen and D. J. Litman. “Argument mining for improving the automated scoring of persuasive essays”. In: *Proc. of the AAAI Conference on Artificial Intelligence*. Vol. 32. AAAI Press, 2018.
 - [155] V. Niculae, J. Park, and C. Cardie. “Argument Mining with Structured SVMs and RNNs”. In: *Proc. of the Annual Meeting of the Association for Computational Linguistics*. Ed. by R. Barzilay and M.-Y. Kan. Association for Computational Linguistics, 2017, pp. 985–995.
 - [156] S. Nielsen and S. Parsons. “A Generalization of Dung’s Abstract Framework for Argumentation: Arguing with Sets of Attacking Arguments”. In: *Proc. of Argumentation in Multi-Agent Systems*. Vol. 4766. 2006, pp. 54–73.
 - [157] North Atlantic Treaty Organisation. “Allied Joint Doctrine for Intelligence Procedures”. In: *AJP-2.1* (2016).
 - [158] OpenAI. *Introducing ChatGPT*. Retrieved from: <https://openai.com/index/chatgpt/>. (Accessed on: 20.02.2024).
 - [159] OpenMP Architecture Review Board. *OpenMP Application Program Interface Version 3.0*. Retrieved from: <http://www.openmp.org/mp-documents/spec30.pdf>. (Accessed on: 19.11.2024). 2008.
 - [160] N. Oren et al. “Inverse Problems for Gradual Semantics”. In: *Proc. of the International Joint Conference on Artificial Intelligence*. 2022, pp. 2719–2725.
 - [161] N. Oren and T. J. Norman. “Semantics for Evidence-Based Argumentation”. In: *Proc. of the Conference on Computational Models of Argument*. Ed. by P. Besnard, S. Doutre, and A. Hunter. Vol. 172. Frontiers in Artificial Intelligence and Applications. IOS Press, 2008, pp. 276–284.
 - [162] A. F. Osborn. *Applied Imagination: Principles and Procedures of Creative Thinking*. Scribner, 1957.

-
- [163] *Oxford Dictionary and Thesaurus (Second Edition)*. Oxford University Press, 2007, p. 543.
- [164] Palo Alto Research Center. *ACH Software: Version 2.0.3*. 2006.
- [165] J. S. Park et al. *Generative Agents: Interactive Simulacra of Human Behavior*. 2023. DOI: <https://doi.org/10.48550/arXiv.2304.03442>.
- [166] J. Park and C. Cardie. “Identifying Appropriate Support for Propositions in Online User Comments”. In: *Proc. of the Workshop on Argumentation Mining*. Ed. by N. Green et al. Association for Computational Linguistics, 2014, pp. 29–38.
- [167] A. Pease et al. “Lakatos-style collaborative mathematics through dialectical, structured and abstract argumentation”. In: *Artificial Intelligence* 246 (2017), pp. 181–219.
- [168] A. Peldszus and M. Stede. “From Argument Diagrams to Argumentation Mining in Texts: A Survey”. In: *International Journal of Cognitive Informatics and Natural Intelligence* 7.1 (2013), pp. 1–31.
- [169] A. Peldszus and M. Stede. “Joint prediction in MST-style discourse parsing for argumentation mining”. In: *Proc. of the Conference on Empirical Methods in Natural Language Processing*. Ed. by L. Màrquez, C. Callison-Burch, and J. Su. Association for Computational Linguistics, 2015, pp. 938–948.
- [170] M. Pellert et al. “AI Psychometrics: Assessing the Psychological Profiles of Large Language Models Through Psychometric Inventories”. In: *Perspectives on Psychological Science* 19.5 (2024), pp. 808–826.
- [171] C. Perelman and L. Olbrechts-Tyteca. *The New Rhetoric: A Treatise on Argumentation*. University of Notre Dame Press, 1969. ISBN: 9780268001919.
- [172] R. Pherson. *Handbook of Analytic Tools and Techniques, 5th Edition*. Pherson Associates, LLC, 2018. ISBN: 9780979888090.
- [173] R. H. Pherson and R. J. Heuer Jr. *Structured Analytic Techniques for Intelligence Analysis*. SAGE Publications, 2019. ISBN: 9781506368924.
- [174] L. Polanyi. “A formal model of the structure of discourse”. In: *Journal of Pragmatics* 12.5 (1988), pp. 601–638.
- [175] S. Pope and A. Jøsang. “Analysis of Competing Hypotheses using Subjective Logic”. In: *Proc. of the International Command and Control Research and Technology Symposium*. 2005.

- [176] N. Potyka. “Abstract Argumentation with Markov Networks”. In: *Proc. of the European Conference on Artificial Intelligence*. Vol. 325. IOS Press, 2020, pp. 865–872.
- [177] H. Prakken. “An abstract framework for argumentation with structured arguments”. In: *Argument & Computation* 1.2 (2010), pp. 93–124.
- [178] H. Prakken, C. Reed, and D. Walton. “Argumentation schemes and generalisations in reasoning about evidence”. In: *Proc. of the International Conference on Artificial Intelligence and Law*. Association for Computing Machinery, 2003, pp. 32–41.
- [179] F. Pu et al. “Argument Ranking with Categoriser Function”. In: *Knowledge Science, Engineering and Management*. Vol. 8793. Springer, 2014, pp. 290–301.
- [180] I. Rahwan and C. Reed. “The Argument Interchange Format”. In: *Argumentation in Artificial Intelligence*. Ed. by G. Simari and I. Rahwan. Boston, MA: Springer US, 2009, pp. 383–402.
- [181] I. Rahwan, F. Zablith, and C. Reed. “Laying the foundations for a World Wide Argument Web”. In: *Artificial Intelligence* 171.10 (2007). Argumentation in Artificial Intelligence, pp. 897–921.
- [182] E. Razumovskaia et al. “Little Red Riding Hood Goes around the Globe: Crosslingual Story Planning and Generation with Large Language Models”. In: *Proc. of the Joint International Conference on Computational Linguistics, Language Resources and Evaluation*. Ed. by N. Calzolari et al. ELRA and ICCL, 2024, pp. 10616–10631.
- [183] C. Reed and G. Rowe. “Araucaria: Software for argument analysis, diagramming and representation”. In: *International Journal of Artificial Intelligence Tools* 14 (2004), pp. 961–979.
- [184] N. Reimers and I. Gurevych. “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks”. In: *Proc. of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2019, pp. 3982–3992.
- [185] R. Rinott et al. “Show Me Your Evidence - an Automatic Method for Context Dependent Evidence Detection”. In: *Proc. of the Conference on Empirical Methods in Natural Language Processing*. Ed. by L. Màrquez, C. Callison-Burch, and J. Su. Association for Computational Linguistics, 2015, pp. 440–450.

-
- [186] J. Robinson. “Distributed Hypothesis Generation and Evaluation”. In: *Online Handbook of Argumentation for Artificial Intelligence 3* (2022). Ed. by F. Castagna et al., pp. 38–42.
- [187] J. Robinson et al. “Identifying Diagnostic Arguments in Abstract Argumentation”. In: *Computational Models of Argument - Proceedings of COMMA 2024, Hagen, Germany, September 18-20, 2014*. Ed. by C. Reed, M. Thimm, and T. Rienstra. Vol. 388. Frontiers in Artificial Intelligence and Applications. IOS Press, 2024, pp. 359–360. DOI: 10.3233/FAIA240339. URL: <https://doi.org/10.3233/FAIA240339>.
- [188] J. Robinson et al. “On Diagnostic Arguments in Abstract Argumentation”. In: *Proc. of the International Workshop on Systems and Algorithms for Formal Argumentation co-located with International Conference on Computational Models of Argument*. Ed. by A. Borg et al. Vol. 3757. CEUR Workshop Proceedings. CEUR-WS.org, 2024, pp. 27–40.
- [189] T. Robinson. “Value of Information for Argumentation based Intelligence Analysis”. In: (2021). DOI: <https://doi.org/10.48550/arXiv.2102.08180>.
- [190] T. Robinson and L. Pardoe. “Value Based Collection in Intelligence Analysis”. In: *Proc. of the International Conference on Military Communication and Information Systems*. 2021, pp. 1–6.
- [191] J. Rogan and M. Baker. # 1617 - Mike Baker on The Joe Rogan Experience. Retrieved from: <https://open.spotify.com/episode/5g9cLclRbwQKNqWiVZtna?si=fc4cf7780ac045b1>. (Accessed on: 22.04.2022). Mar. 2021.
- [192] R. Ruiz-Dolz et al. “Transformer-based models for automatic identification of argument relations: A cross-domain evaluation”. In: *IEEE Intelligent Systems* (2021).
- [193] P. Saint-Dizier. “Processing natural language arguments with the <TextCoop> platform”. In: *Argument & Computation* 3.1 (2012), pp. 49–82.
- [194] J. Savelka et al. *Explaining Legal Concepts with Augmented Large Language Models (GPT-4)*. 2023. DOI: <https://doi.org/10.48550/arXiv.2306.09525>.
- [195] R. Schrag et al. “Probabilistic Argument Maps for Intelligence Analysis: Capabilities Underway”. In: *Proc. of the International Workshop on Computational Models of Natural Argument at the International Joint Conference on Artificial Intelligence*. Ed. by F. Bex, F. Grasso, and N. L. Green. Vol. 1876. CEUR Workshop Proceedings. 2016, pp. 16–21.

- [196] R. Schrag et al. “Probabilistic Argument Maps for Intelligence Analysis: Completed Capabilities.” In: *Proc. of the International Workshop on Computational Models of Natural Argument at the International Joint Conference on Artificial Intelligence*. Vol. 1876. CEUR Workshop Proceedings. 2016, pp. 34–39.
- [197] F. Schroff, D. Kalenichenko, and J. Philbin. “FaceNet: A unified embedding for face recognition and clustering”. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 815–823.
- [198] P. Schwartz. *The Art of the Long View*. A Currency book. Doubleday/Currency, 1991. ISBN: 9780385267311.
- [199] D. M. Schweiger, W. R. Sandberg, and J. W. Ragan. “Group approaches for improving strategic decision making: a comparative analysis of dialectical inquiry, devil’s advocacy, and consensus”. In: *Academy of Management Journal*. Vol. 29. 1986, pp. 51–71.
- [200] C. E. Shannon. “A Mathematical Theory of Communication”. In: *The Bell System Technical Journal* 27 (1948), pp. 379–423.
- [201] K. Shuster et al. *BlenderBot 3: a deployed conversational agent that continually learns to responsibly engage*. 2022. DOI: <https://doi.org/10.48550/arXiv.2208.03188>.
- [202] G. R. Simari and R. P. Loui. “A mathematical treatment of defeasible reasoning and its implementation”. In: *Artificial Intelligence* 53.2 (1992), pp. 125–157.
- [203] M. Snaith and C. Reed. “TOAST: online ASPIC+ implementation”. In: *Proc. of the International Conference on Computational Models of Argument*. Ed. by B. Verheij, S. Szeider, and S. Woltran. 2012, pp. 509–510.
- [204] L. Snidaro. “ChatGPT Act as an Intelligence Officer”. In: *Proc. of the IEEE International Workshop on Technologies for Defense and Security*. 2023, pp. 449–454.
- [205] K. Sparck Jones. “A statistical interpretation of term specificity and its application in retrieval”. In: *Document Retrieval Systems*. Taylor Graham Publishing, 1988, pp. 132–142.
- [206] *Spy Kids: Briefing on The Intelligence Cycle*. Retrieved from: <https://www.cia.gov/spy-kids/static/59d238b4b5f69e\0497325e49f0769acf/Briefing-intelligence-cycle.pdf>. (Accessed on: 07.05.2024).

-
- [207] C. Stab and I. Gurevych. “Identifying Argumentative Discourse Structures in Persuasive Essays”. In: *Proc. of the Conference on Empirical Methods in Natural Language Processing*. Ed. by A. Moschitti, B. Pang, and W. Daelemans. Association for Computational Linguistics, 2014, pp. 46–56.
- [208] K. E. Stanovich and R. F. West. “Individual differences in reasoning: Implications for the rationality debate?” In: *Behavioral and Brain Sciences* 23.5 (2000), pp. 645–665.
- [209] R. Taylor et al. *Galactica: A Large Language Model for Science*. 2022. DOI: <https://doi.org/10.48550/arXiv.2211.09085>.
- [210] G. Tecuci et al. “Intelligence Analysis as Agent-Assisted Discovery of Evidence, Hypotheses and Arguments”. In: *Proc. of the Advances in Intelligent Decision Technologies*. Ed. by G. Phillips-Wren et al. Springer, 2010, pp. 1–10.
- [211] The pandas development team. *pandas-dev/pandas: Pandas*. Version latest. Feb. 2020. DOI: 10.5281/zenodo.3509134. URL: <https://doi.org/10.5281/zenodo.3509134>.
- [212] M. Thimm. “A Probabilistic Semantics for Abstract Argumentation”. In: *Proc. of the European Conference on Artificial Intelligence*. Vol. 242. 2012, pp. 750–755.
- [213] M. Thimm and A. Hunter. “Probabilistic Argumentation with Incomplete Information”. In: *Proc. of the European Conference on Artificial Intelligence*. Vol. 263. Aug. 2014, pp. 1033–1034.
- [214] R. Thoppilan et al. *LaMDA: Language Models for Dialog Applications*. 2022. DOI: <https://doi.org/10.48550/arXiv.2201.08239>.
- [215] A. Toniolo et al. “Human-machine collaboration in intelligence analysis: An expert evaluation”. In: *Intelligent Systems with Applications* 17 (2023), p. 200151.
- [216] A. Toniolo et al. “Supporting reasoning with different types of evidence in intelligence analysis”. In: *Proc. of the International Conference on Autonomous Agents and Multiagent Systems*. Ed. by G. Weiss and P. Yolum. Vol. 2. International Foundation for Autonomous Agents and Multiagent Systems, 2015, pp. 781–789.
- [217] S. E. Toulmin. *The Uses of Argument*. Cambridge University Press, 1958.
- [218] A. Tversky and D. Kahneman. “Availability: A Heuristic for Judging Frequency and Probability”. In: *Cognitive Psychology* 5.2 (1973), pp. 207–232.

- [219] S. Tzu. *The Art of War*. Dover Military History, Weapons, Armor. Dover Publications, 2002. ISBN: 9780486425573.
- [220] United Kingdom Ministry of Defence. *Joint Doctrine Publication JDP 2-00, Understanding and Intelligence Support to Joint Operations*. Development, Concepts and Doctrine Centre, 2023.
- [221] United Kingdom's Department for Defence Intelligence. *Defence Intelligence – communicating probability*. Retrieved from: <https://www.gov.uk/government/news/defence-intelligence-communicating-probability>. (Accessed on: 03.06.2024).
- [222] United Kingdom's Department for Defence Intelligence. *Quick Wins for Busy Analysts*. United Kingdom's Ministry of Defence, 2016.
- [223] United States Center for the Study of Intelligence. *A Tradecraft Primer: Structured Analytic Techniques for Improving Intelligence Analysis*. U.S. Central Intelligence Agency, Center for the Study of Intelligence, 2009.
- [224] United States Congress. *S.2845 - Intelligence Reform and Terrorism Prevention Act of 2004*.
- [225] United States Department of Defence. "Dictionary of Military and Associated Terms". In: *JP 1-02* (2009).
- [226] United States Intelligence Community Staff (ICS), ed. *Glossary of Intelligence Terms and Definitions*. Washington, DC: ICS, 1989.
- [227] United States Interagency Operations Security Support Staff. *Intelligence Threat Handbook*. 2000.
- [228] United States Joint Chiefs of Staff. "Joint Intelligence: Joint Publication 2-0". In: *JP 2-0* (2022).
- [229] United States Office of the Director of National Intelligence. *Intelligence Community Directive 203*. Retrieved from: <https://fas.org/irp/dni/icd/icd-203.pdf>. (Accessed on: 03.06.2024).
- [230] United States: National Commission on Terrorist Attacks upon the United States. *The 9/11 Commission Report*. Washington: Government Printing Office, 2004, pp. 71–107.
- [231] M. Valtorta et al. "Extending Heuer's analysis of competing hypotheses method to support complex decision analysis". In: *Proc. of the International Conference on Intelligence Analysis*. 2005.

-
- [232] G. Van Rossum and F. L. Drake. *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009. ISBN: 1441412697.
- [233] A. Vaswani et al. “Attention is All you Need”. In: *Proc. of the Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017.
- [234] J. Visser et al. “Argumentation in the 2016 US presidential elections: annotated corpora of television debates and social media reaction”. In: *Lang Resources & Evaluation* 54 (2020), pp. 123–154.
- [235] D. Walton, C. Reed, and F. Macagno. *Argumentation Schemes*. Ed. by C. Reed and F. Macagno. New York: Cambridge University Press, 2008.
- [236] D. Walton. *A Pragmatic Theory of Fallacy*. Tuscaloosa and London: University of Alabama Press, 1995, pp. 140–141.
- [237] D. Walton. *A Pragmatic Theory of Fallacy*. Tuscaloosa and London: University of Alabama Press, 1995, pp. 155–156.
- [238] D. Walton. *A Pragmatic Theory of Fallacy*. Tuscaloosa and London: University of Alabama Press, 1995, p. 147.
- [239] D. Walton. *Argumentation Schemes for Presumptive Reasoning*. Mahwah, N.J.: Erlbaum, 1996.
- [240] D. Walton. *Legal Argumentation and Evidence*. University Park: Pennsylvania State University Press, 2002, p. 46.
- [241] D. Walton. *Practical Reasoning*. Savage, Md.: Rowman and Littlefield, 1990.
- [242] D. Walton. *Witness Testimony Evidence: Argumentation and the Law*. Cambridge University Press, 2007, p. 60.
- [243] M. Whitesmith. *Cognitive Bias in Intelligence Analysis: Testing the Analysis of Competing Hypotheses Method*. Edinburgh University Press, 2020.
- [244] J. H. Wigmore. *The process of proof as given by logic, psychology, and general experience and illustrated in judicial trials*. Second. Little, Brown & Company, 1931.
- [245] T. Wolf et al. “Transformers: State-of-the-Art Natural Language Processing”. In: *Proc. of the Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Ed. by Q. Liu and D. Schlangen. Online: Association for Computational Linguistics, Oct. 2020, pp. 38–45.
- [246] S. Wu et al. *BloombergGPT: A Large Language Model for Finance*. 2023. DOI: <https://doi.org/10.48550/arXiv.2303.17564>.

- [247] A. Z. Wyner et al. “Semi-Automated Argumentative Analysis of Online Product Reviews”. In: *Proc. of the Conference on Computational Models of Argument*. Ed. by B. Verheij, S. Szeider, and S. Woltran. Vol. 245. Frontiers in Artificial Intelligence and Applications. IOS Press, 2012, pp. 43–50.
- [248] A. Z. Wyner, T. J. M. Bench-Capon, and K. Atkinson. “Arguments, Values and Baseballs: Representation of Popov v. Hayashi”. In: *Proc. of the Annual Conference on Legal Knowledge and Information Systems*. Ed. by A. R. Lodder and L. Mommers. Vol. 165. Frontiers in Artificial Intelligence and Applications. IOS Press, 2007, pp. 151–160.
- [249] F. Yu, L. Quartey, and F. Schilder. *Legal Prompting: Teaching a Language Model to Think Like a Lawyer*. 2022. DOI: <https://doi.org/10.48550/arXiv.2212.01326>.
- [250] T. Zhang et al. *BERTScore: Evaluating Text Generation with BERT*. 2020. DOI: <https://doi.org/10.48550/arXiv.1904.09675>.
- [251] J. Zlotnick. “A Theorem for Prediction”. In: *Center for the Study of Intelligence, CIA Archives* (1967).
- [252] J. Zlotnick. “Bayes’ Theorem for Intelligence Analysis”. In: *Center for the Study of Intelligence, CIA Archives* (1996).
- [253] D. Zografistou et al. “ACH-Nav: Argument Navigation Using Techniques for Intelligence Analysis”. In: *Proc. of the Conference on Computational Models of Argument*. Frontiers in Artificial Intelligence and Applications. 2018.

Appendices

Appendix A

On Large Language Models for Hypothesis Generation

A.1 Unique Identifiers

Presented below is a list of unique identifiers that were employed in extraction of Tweets that potentially contained intelligence reports. The hashtag # denotes a number pertaining to the order within which Tweets were published. For instance, (1 / 5) would refer to Tweet number one out of five.

- (# / #)
- (#/#)
- (# of #)
- [#/#]
- /#
- Post # of #
- INTELLIGENCE UPDATE:

A.2 Example of Prompts

A.2.1 One-Shot (Unbiased)

The prompt presented below contains one, unbiased training example that was generated using Algorithm 4.1. The randomly selected training example is italicised and the target information is in bold text.

- ‘You are an Artificial Intelligence assistant for an intelligence analyst who is analysing the conflict between Russia and Ukraine. Your job is to generate a hypothesis using a set of information that you have received. In this context, a hypothesis is a sentence, it is testable and falsifiable, it is based on the information you received, and it predicts anticipated results clearly. Please generate one hypothesis based on the inputted information.’
- ‘*Information: Elements of the Ukrainian Armed Forces have advanced into the multi-layered main Russian defensive line east of the town of Robotyne. Ukrainian dismounted infantry forces are continuing to make gradual tactical advances against Russian positions and attrite Russian forces in the area. Ukrainian forces have also maintained pressure on Russian positions to the south of Bakhmut, making gradual gains between Klishchiivka and Adriivka.*’
- ‘**Information: Over the last two months, Russia has likely started forming up major new formations to add depth to its ground forces. These include the 25th Combined Arms Army. Since its invasion of Ukraine, Russia has mainly deployed mobilised reservists to back-fill established formations, or as part of territorial defence infantry regiments. It has rarely established new, all-arms organisations such as combined arms armies which are designed to be a self-sufficient force. An exception to this was the 3rd Army Corps created in summer 2022, which has generally performed poorly.**’
- ‘Hypothesis: ’

A.2.2 One-Shot (Biased)

The prompt presented below contains one, biased training example that was generated using Algorithm 4.2. Again, the training example is italicised and the target information is in bold text.

- ‘You are an Artificial Intelligence assistant for an intelligence analyst who is analysing the conflict between Russia and Ukraine. Your job is to generate a hypothesis using a set of information that you have received. In this context, a hypothesis is a sentence, it is testable and falsifiable, it is based on the information you received, and it predicts anticipated results clearly. Please generate one hypothesis based on the inputted information.’
- ‘*Information: Over the last two months, Russia has likely started forming up major new formations to add depth to its ground forces. These include the 25th Combined Arms Army. Since its invasion of Ukraine, Russia has mainly deployed mobilised reservists to back-fill established formations, or as part of territorial defence infantry regiments. It has rarely established new, all-arms organisations such as combined arms armies which are designed to be a self-sufficient force. An exception to this was the 3rd Army Corps created in summer 2022, which has generally performed poorly. Russia will likely deploy any new formation as a reserve force in Ukraine. However, in the longer term, Russia aspires to strengthen its forces facing North Atlantic Treaty Organisation.*’
- ‘**Information: Over the last two months, Russia has likely started forming up major new formations to add depth to its ground forces. These include the 25th Combined Arms Army. Since its invasion of Ukraine, Russia has mainly deployed mobilised reservists to back-fill established formations, or as part of territorial defence infantry regiments. It has rarely established new, all-arms organisations such as combined arms armies which are designed to be a self-sufficient force. An exception to this was the 3rd Army Corps created in summer 2022, which has generally performed poorly.**’
- ‘Hypothesis: ’

A.3 Examples of Model Completions

The GPT-3.5, 2 epochs model was provided with the same target information within a one-shot learning prompt, and tasked with generating the same target hypothesis. The target hypothesis for both the unbiased and biased one-shot prompts, respectively, displayed in Appendices A.2.1 and A.2.2 is presented below:

‘Russia will likely deploy any new formation as a reserve force in Ukraine, however, in the longer term, Russia aspires to strengthen its forces facing North Atlantic Treaty Organisation.’

The model completions for unbiased and biased one-shot learning when temperature was equal to 0.1 and unity is included below.

A.3.1 Unbiased Prompting

Starting with unbiased prompting. The prompt from Appendix A.2.1 was used for the generations in this subsection. The completions generated using both temperatures are provided below.

A.3.1.1 Completions, when temperature was equal to 0.1

1. ‘It is highly likely that the 25th Combined Arms Army is being formed to help address the Russian military’s chronic shortage of combat power.’
2. ‘It is highly likely that Russia has created the 25th Combined Arms Army to help address its ongoing manpower shortages.’
3. ‘It is highly likely that the 25th Combined Arms Army is being formed to help alleviate the strain on the Russian military.’
4. ‘It is highly likely that the 25th Combined Arms Army is being formed from a mix of mobilised reservists and regular forces.’
5. ‘It is highly likely that the 25th Combined Arms Army is being formed to help address Russia’s ongoing manpower shortages.’
6. ‘It is highly likely that Russia has created the 25th Combined Arms Army to help bolster its defences in the north of the country.’
7. ‘It is highly likely that Russia has formed the 25th Combined Arms Army to help bolster its defences in the Donbas region.’

8. 'It is highly likely that the 25th Combined Arms Army is being formed from a mix of mobilised reservists and regular army personnel.'
9. 'It is highly likely that the 25th Combined Arms Army is being formed to help Russia maintain a credible force in the Donbas region.'
10. 'It is highly likely that the 25th Combined Arms Army is being formed to help address the Russian military's ongoing shortage of trained personnel.'

A.3.1.2 Completions, when temperature was equal to unity

1. 'Russia is likely making this change to improve its ability to perform large-scale offensive operations against determined and well-armed opposition.'
2. 'It is possible that Russia's new combined arms army formations are too large, combining mobilised reservists with recruited contracted soldiers from Russia.'
3. 'The creation of the 25th Combined Arms Army shows that the Russian Ground Forces is still suffering from significant structural and operational problems nearly nine months into the intervention.'
4. 'It is highly likely that Russia will insert the 25th Combined Arms Army into the frontline following a period of training and to provide depth.'
5. 'This shift towards forming new composite formations suggests Russia likely faces continued difficulty finding sufficient experienced reservists to deploy to Ukraine.'
6. 'Straining its resources even further by forming up 25th Combined Arms Army is likely to have been a reluctant decision for Russia.'
7. 'The 25th Combined Arms Army is likely being formed in part because post-Summer 2022 mobilisations have probably added little combat power to the ground forces.'
8. 'Russia's attempt to create another large new formation is likely driven by a requirement for combat power, rather than desperation to provide additional forces to the front.'
9. 'Russian commander in Ukraine, Andrey Serdyukov, may be attempting to create a stronger organisation with a renewed guidance to be ready for combat and for operations at a higher operational tempo.'

10. ‘It is highly likely that Russia is creating these new army-level formations in response to the challenges it has experienced on the ground, and to support its anticipated strategy in Ukraine.’

A.3.2 Biased prompting

The biased prompt from Appendix A.2.2 was used for the generations in this subsection. The completions generated using both temperatures are provided below.

A.3.2.1 Completions, when temperature was equal to 0.1

1. ‘Russia will likely deploy any new formation as a reserve force in Ukraine.’
2. ‘Russia will likely deploy any new formation as a reserve force in Ukraine.’
3. ‘Russia will likely deploy any new formation as a reserve force in Ukraine.’
4. ‘Russia will likely deploy any new formation as a reserve force in Ukraine. However, in the longer term, Russia aspires to strengthen its forces facing North Atlantic Treaty Organisation.’
5. ‘Russia will likely deploy any new formation as a reserve force in Ukraine.’
6. ‘Russia will likely deploy any new formation as a reserve force in Ukraine. However, in the longer term, Russia aspires to strengthen its forces facing North Atlantic Treaty Organisation.’
7. ‘Russia will likely deploy any new formation as a reserve force in Ukraine.’
8. ‘Russia will likely deploy any new formation as a reserve force in Ukraine.’
9. ‘Russia will likely deploy any new formation as a reserve force in Ukraine.’
10. ‘Russia will likely deploy any new formation as a reserve force in Ukraine. However, in the longer term, Russia aspires to strengthen its forces facing North Atlantic Treaty Organisation.’

A.3.2.2 Completions, when temperature was equal to unity

1. ‘With its military confronted in eastern Ukraine and potentially facing off against North Atlantic Treaty Organisation in the future, it is likely getting increasingly difficult for Russia to resupply its forces in Ukraine with the necessary munitions, equipment and personnel.’

2. 'Without a major new wave of mandatory mobilisation, Russia is unlikely to find enough new troops to resource even one new army.'
3. 'Most mobilised reservists are only likely to be deployed within the region they were raised, unlike voluntary mobilisation which pulls from current or former full-time servicemen from across Russia.'
4. 'Without a major new wave of mandatory mobilisation, Russia is unlikely to find enough new troops to resource even one new army.'
5. 'Russia is likely finding it challenging to crew and stand up 25th Combined Arms Army, a formation it never planned for prior to the large-scale deployment of Russian ground forces to Ukraine.'
6. 'Russia will likely deploy any new formation as a reserve force in Ukraine. However, in the longer term, Russia aspires to strengthen its forces facing North Atlantic Treaty Organisation.'
7. 'These new formations are likely part of a reorganised and growing Russian invasion force.'
8. 'Russia's fielding of new units in Ukraine's east indicates that, despite persistent problems quickly mobilising and equipping wave after wave of reservists, Moscow intends to maintain strong pressure on Ukraine.'
9. 'Russian commanders are likely still reluctant to rely on reservists to secure the wider ground in Ukraine.'
10. 'Without a major new wave of mandatory mobilisation, Russia is unlikely to find enough new troops to resource even one new army.'

Appendix B

Operation Fortitude-South

In this appendix, I describe how the refined ACH matrix from the Operation Fortitude-South scenario was instantiated in a Dung argumentation framework. In Section B.1, I describe Alan Knox’s contribution, which was presented on Dstl’s *eleatics* website, in full. I also present: the list of information, evidence, and assumptions found on Dstl’s *eleatics* website in Section B.2; the AIF argument graph for the scenario in Section B.3; the arguments instantiated using the ASPIC⁺ framework in Section B.4; and the abstract argumentation framework that was translated from the AIF in Section B.5.

B.1 Dstl’s Contribution

Knox identified Operation Fortitude-South as an interesting scenario for further study within the context of the ACH and argumentation schemes. As previously discussed in Part II, Operation Fortitude-South was an Allied attempt to deceive the German military and make them believe that the Allied forces would invade Europe at the crossing between Dover and Pas de Calais, and any other attacks were meant to be thought of as diversionary attacks. The Allied nations were able to deceive the Germans using a number of different tactics, such as through the reporting of false information by double agents, and also positioning inflatable rubber equipment, dirt road tracks leading to nowhere, and the erection of empty cities of tents in southeastern England, to name a few [66, p. 13-18].

Knox loosely follows the ACH process for the Operation Fortitude-South scenario, making some slight alterations to the procedure which are discussed below. Recall from Part II that the analysis of the scenario was conducted from the perspective of

what Hilter and the German command would have believed was the most likely Allied invasion site. Operation Fortitude-South was so successful that Hilter still did not believe that the Normandy landings on D-Day, on 6th June 1944, was the main invasion site even a month after they had occurred, on the 8th July 1944 [66, p. 15].

In the following subsections, I now summarise the work conducted by Knox at Dstl for the Operation Fortitude-South example found on the *eleatics* website.

B.1.1 Step One

To begin their analysis, and in accordance with the first step within the ACH discussed in Section 2.1.5.2, Knox identified all the hypotheses for the Operation Fortitude-South scenario¹, which are listed below:

- *Pas de Calais*: The main invasion will land in Pas de Calais.
- *Normandy*: The main invasion will land in Normandy.
- *Brittany*: The main invasion will land in Brittany.
- *Cotentin Peninsula*: The main invasion will land on the Cotentin Peninsula.
- *Elsewhere*: The main invasion will land outside the range of close air support.

B.1.2 Step Two

After the generation of hypotheses, the second step of the ACH requires that analysts list the information, evidence, and assumptions. Knox identified seventeen items which are shown in the row names of Table B.1 and each acronym is defined in Appendix B.2.

At this point, Knox made a slight adaptation to the ACH process and reduced the number of hypotheses, from five to two, before instantiating the initial ACH matrix. Pherson and Heuer Jr. suggest that the diagnosticity matrix should be refined in step five [101, 173], after completing the initial analysis in step three. Knox then reduced the number of hypotheses using the two following assumptions:

- *Close Air Support*: The Allied invasion is required to have close air support.
- *Peninsula*: Forces landing on a peninsula can be cut off on a narrow front.

¹<https://Dstl.github.io/eleatics/argumentation/fortitude/hypotheses.xhtml> (Accessed on: 10.02.2023)

Knox reasoned using these assumptions and discounted both the *Brittany* and the *Contentin Peninsula* hypotheses because they were located on a peninsula. The *elsewhere* hypothesis was also removed due to lack of close air support. In reality, this refinement should have occurred in step five after debating instances where different analysts disagreed with the initial analysis in step four. However, this alteration to the ACH process most likely occurred because the analysis was conducted by one analyst within Dstl.

B.1.3 Step Three

Remember from Section 2.1.5.2 that step three of the ACH requires analysts to instantiate a matrix with hypotheses (as column headers) and items of information, evidence, and assumptions (as row headers). The consistency, or otherwise, of each row entry against each hypothesis is assigned by an analyst during analysis. In the running example, Knox populated an initial ACH matrix, which is displayed in Table B.1.

Knox argues that a number of cells within the matrix can be modelled through the instantiation of one of Walton's argumentation schemes² [235], which is a concept that has been explored previously by Murukannaiah *et al.* in [152] and discussed in more detail in Section 2.4.3. The schemes that Knox identified are presented in Table B.2. While Knox does list some schemes that might be employed within the ACH, they do not show how these schemes should be instantiated using the available data, which is a novel task and an extension of Dstl's contribution that is presented in Part II.

Order of Battle (ORBAT). At this point, in between step three and four of the ACH, Knox again strays from the formal ACH process and conducts a sub-analysis, referred to as the ORBAT matrix³ and presented in Table B.3, grouping row entries which they believed were subordinate to others. They argue that it is not the existence of two military divisions, namely FUSAG and 21AG (see Appendix B.2), that was diagnostic. It was, however, where these forces were located that was important. Moreover, they argue that it was not the knowledge that General Patton (see Appendix B.2) was the most likely commander of FUSAG that was diagnostic, but it was in fact where Patton's army was situated that was diagnostic. Furthermore, Knox discusses the RECON and SIGINT (see Appendix B.2) row entries, arguing that they are data points which

²<https://Dstl.github.io/electrics/argumentation/fortitude/schemes.xhtml> (Accessed on: 10.02.2023)

³<https://dstl.github.io/electrics/argumentation/fortitude/orbat.xhtml> (Accessed on: 10.02.2023)

Table B.1: The initial ACH matrix for Operation Fortitude-South, where “C” and “I” denote row headers that are consistent and inconsistent with the hypotheses (column headers), respectively.

Information, Evidence, and Assumptions	Hypotheses	
	Normandy	Pas de Calais
PORT	I	C
EXPANSION	C	I
STRONGPOINT	C	I
21AG	C	C
21AGLOC	C	I
PATTON	I	C
FUSAG	C	C
FUSAGLOC	I	C
SPIES	I	C
SIGINT	I	C
SHORTEST	I	C
VSITES	I	C
RECON	I	C
AIRPLAN	I	C
PENINSULA	C	C
CAS	C	C

provide information about the location of FUSAG and 21AG. This insight results in another ACH matrix which contains information about the Order of Battle. The evaluated ORBAT matrix, shown in Table B.3, provides support for the *Pas de Calais* hypothesis because there were fewer row entries that were inconsistent with that hypothesis. ORBAT's support for the *Pas de Calais* hypothesis is represented by a row entry in the refined ACH matrix in Table B.4, which I discuss in Section B.1.5.

B.1.4 Step Four

Although step four of the ACH requires that analysts review where there is disagreement amongst themselves, the work produced by Knox at Dstl does not include this particular task, which is most likely because this analysis was conducted by one person. The ACH can be conducted using either a team of analysts or a single reasoner,

Table B.2: A table containing the argumentation schemes, which were identified by Dstl, corresponding to the row entries in the initial and refined ACH matrix for the Operation Fortitude-South example.

Argumentation Scheme	Information, Evidence and Assumptions
Argument from Consequences	SHORTEST, STRONGPOINT, EXPANSION, and VSITES
Argument from Established Rule	PORT, CAS, and PENINSULA
Argument from Evidence to a Hypothesis	SIGINT, ORBAT, and AIRPLAN
Argument from Position to Know	SPIES
Argument from Cause to Effect	FUSAGLOC, and 21AGLOC
Argument from Expert Opinion	PATTON

Table B.3: The ORBAT ACH matrix , where “C” and “I” denote row headers pertaining to the Order of Battle that are consistent and inconsistent with the hypotheses, respectively.

Information, Evidence, and Assumptions	Hypotheses	
	Normandy	Pas de Calais
21AGLOC	C	I
PATTON	I	C
FUSAGLOC	I	C
SPIES	I	C
SIGINT	I	C
RECON	I	C

so this alteration is in conformity with the ACH procedure [173].

B.1.5 Step Five

Step five of the ACH dictates that analysts must refine the set of hypotheses, removing or combining superfluous or overlapping hypotheses, respectively. However, Knox had already completed this reduction task in step two, as discussed in Section B.1.2, before the instantiation of the initial ACH matrix. The refined matrix is presented in Table B.4. The matrix contains fewer items of information, evidence, and assumptions than the initial matrix, shown in Table B.1. One interesting point to note here is that there seems to be some double counting of row entries due to the introduction of ORBAT within the refined matrix. For instance, both the ORBAT and refined matrix in Tables B.3 and B.4, respectively, contain the items SIGINT and SPIES. The inclusion of those items twice, whether intentional or otherwise, means that they bear more weight on the evaluation of hypotheses.

Table B.4: The refined ACH matrix for Operation Fortitude-South, where “C” and “I” denote row headers that are consistent and inconsistent with the hypotheses (column headers), respectively.

Information, Evidence, and Assumptions	Hypotheses	
	Normandy	Pas de Calais
PORT	I	C
EXPANSION	C	I
STRONGPOINT	C	I
SPIES	I	C
SIGINT	I	C
SHORTEST	I	C
AIRPLAN	I	C
ORBAT	I	C

B.1.6 Step Six

In line with step six of the process, the likelihood of hypotheses is found by counting the number of row entries which were inconsistent with each hypothesis such that the hypothesis that was the least inconsistent was the most likely. Knox drew tentative conclusions about the likelihood of both the *Normandy* and *Pas de Calais* hypothe-

ses, where they concluded that the latter was the most likely given the information, evidence, and assumptions available.

B.1.7 Step Seven

Recall that step seven of the ACH requires analysts to conduct sensitivity analysis where they sequentially remove each row entry and consider whether the likelihood of any of the hypotheses changes. Here, Knox reasoned with the items of information, evidence, and assumptions that feature within the analysis and concluded that *ORBAT* was ‘*probably the most critical*’⁴. This row entry was chosen as the most diagnostic because it is row entry which had accrued other items of information and evidence, thus, its removal significantly changes the likelihood of the *Pas de Calais* hypothesis.

B.1.8 Step Eight and Step Nine

In step eight of the ACH, analysts are required to report the hypotheses and their likelihoods, as well as any diagnostic row entries. Knox suggests that Toulmin’s argument schemata (see Figure 2.7 in Section 2.2) could be employed in a such a report as it is capable of expressing evidence (or *backing*) that supports a conclusion (or *claim*) through an inference. While it is possible to employ Toulmin’s seminal model of argument to represent hypotheses (as claims) which are supported by observations (as grounds), in accordance with Pherson and Heuer Jr.’s definition of a hypothesis (Chapter 1), Toulmin’s model is not fit for operational use within intelligence settings because of the issues described in Section 2.2.

Finally, in step nine of the ACH, analysts must identify things to observe as indicators of future development of a situation. Knox states that this task could have been completed if the ACH was invented prior to World War Two and the Allied nations had chosen to use the technique.

B.1.9 Concluding Remarks on Dstl’s Contribution

In summary, Knox demonstrates on Dstl’s *eleatics* website that the ACH can be combined with argumentation schemes for real-world scenarios. However, the assumption that each cell in an diagnosticity matrix can be the instantiation of one of Walton’s argumentation schemes is quite restrictive: hypotheses can be supported by a number of

⁴<https://dstl.github.io/eleatics/argumentation/fortitude/fortitude.xhtml> (Accessed on: 10.02.2023)

sequentially linked instantiations of schemes, as I discussed in the introduction to Part II.

Future work should study the combination of said techniques to understand whether the approach works in practice for real-world situations. The study could look somewhat similar to Murukannaiah *et al.*'s work in [152] but the data employed within the test should be from a real-world scenario. The incorporation of both trained intelligence analysts as test subjects, who employ the ACH and argumentation schemes and also the ACH alone as a control, should provide a better assessment of whether this approach is beneficial for analysts.

B.2 List of Information, Evidence, and Assumptions

Presented below is the list of information, evidence, and assumptions employed by Dstl for the ACH for the Operation Fortitude-South scenario, taken from Dstl's *eleatics* website⁵ [63]:

- **INVASION:** The build-up of US troops in England makes invasion inevitable. It's just a question of where and when.
- **PENINSULA:** Forces landing on a peninsula can be cut off on a narrow front.
- **CAS:** The invasion must have close air support.
- **PORT:** A major port will be needed to supply invasion forces.
- **STRONGPOINT:** The Pas de Calais is the strongest point of the Axis defences.
- **SIGINT:** Axis signals intelligence.
- **RECON:** Reports from Axis aerial reconnaissance.
- **SPIES:** Reports from Axis spies in England.
- **FUSAG:** First US Army Group. Apparently larger than 21st Army Group.
- **FUSAGLOC:** FUSAG is based in Southeast England
- **PATTON:** Patton is the most likely commander for an Allied invasion, and Patton commands FUSAG.

⁵<https://Dstl.github.io/eleatics/argumentation/fortitude/evidence.xhtml> (Accessed on: 10.02.2023)

- AIRPLAN: The Axis view of the Allied air plan.
- 21AG: A British headquarters formation, in command of two field armies and other supporting units, consisting primarily of the British Second Army and the First Canadian Army. Established in London during July 1943, under the command of Supreme Headquarters Allied Expeditionary Force.
- 21AGLOC: 21st Army Group located in southern England.
- SHORTEST: The Pas de Calais affords the shortest crossing from Britain to mainland Europe.
- VSITES: Pas de Calais is the nearest landing point to the V1 and V2 launch sites.
- EXPANSION: Better opportunities for expansion - e.g. link-up with Operation Dragoon.

B.3 The Refined ACH Matrix's AIF Argument Graph

The argument annotation, conducted using *OVA+* [108] and captured in the AIF, for the refined Operation Fortitude-South ACH matrix is presented in Figure B.1.

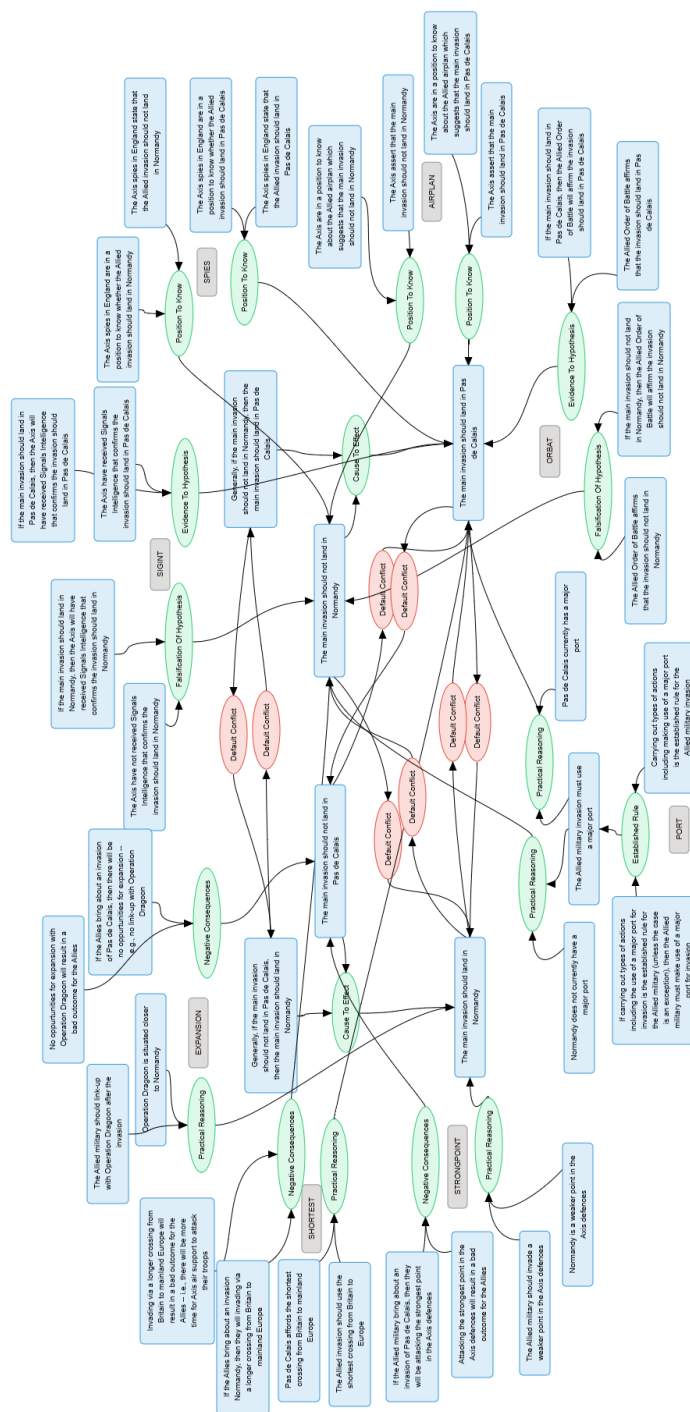


Figure B.1: The refined ACH matrix for the Operation Fortitude-South example which was annotated using *OVA*+ [108].

B.4 Arguments Instantiated Using The ASPIC⁺ Framework for the Refined ACH Matrix

Presented below are all the arguments that were instantiated from the refined ACH matrix, using the ASPIC⁺ framework.

- Arg 0: Generally, if the main invasion should not land in Pas de Calais, then the main invasion should land in Normandy
- Arg 1: Generally, if the main invasion should not land in Normandy, then the main invasion should land in Pas de Calais.
- Arg 2: If carrying out types of actions including the use of a major port for invasion is the established rule for the Allied military (unless the case is an exception), then the Allied military must make use of a major port for invasion.
- Arg 3: Carrying out types of actions including making use of a major port is the established rule for the Allied military invasion.
- Arg 4: Pas de Calais currently has a major port.
- Arg 5: Normandy does not currently have a major port.
- Arg 6: If the Allied military bring about an invasion of Pas de Calais, then they will be attacking the strongest point in the Axis defences.
- Arg 7: Attacking the strongest point in the Axis defences will result in a bad outcome for the Allies.
- Arg 8: The Allied military should invade a weaker point in the Axis defences.
- Arg 9: Normandy is a weaker point in the Axis defences.
- Arg 10: If the Allies bring about an invasion Normandy, then they will invading via a longer crossing from Britain to mainland Europe.
- Arg 11: Invading via a longer crossing from Britain to mainland Europe will result in a bad outcome for the Allies – i.e., there will be more time for Axis air support to attack their troops.
- Arg 12: The Allied invasion should use the shortest crossing from Britain to Europe.
- Arg 13: Pas de Calais affords the shortest crossing from Britain to mainland Europe.
- Arg 14: If the Allies bring about an invasion of Pas de Calais, then there will be no opportunities for expansion – e.g., no link-up with Operation Dragoon.
- Arg 15: No opportunities for expansion with Operation Dragoon will result in a bad outcome for the Allies.
- Arg 16: The Allied military should link-up with Operation Dragoon after the invasion.
- Arg 17: Operation Dragoon is situated closer to Normandy.
- Arg 18: If the main invasion should land in Normandy, then the Axis will have received Signals Intelligence that confirms the invasion should land in Normandy.
- Arg 19: The Axis have not received Signals Intelligence that confirms the invasion should land in Normandy.
- Arg 20: If the main invasion should land in Pas de Calais, then the Axis will have received Signals Intelligence that confirms the invasion should land in Pas de Calais.
- Arg 21: The Axis have received Signals Intelligence that confirms the invasion should land in Pas de Calais.

- Arg 22: The Axis spies in England are in a position to know whether the Allied invasion should land in Normandy.
- Arg 23: The Axis spies in England state that the Allied invasion should not land in Normandy.
- Arg 24: The Axis spies in England are in a position to know whether the Allied invasion should land in Pas de Calais.
- Arg 25: The Axis spies in England state that the Allied invasion should land in Pas de Calais.
- Arg 26: The Axis are in a position to know about the Allied airplan which suggests that the main invasion should land in Pas de Calais.
- Arg 27: The Axis assert that the main invasion should land in Pas de Calais.
- Arg 28: The Axis are in a position to know about the Allied airplan which suggests that the main invasion should not land in Normandy.
- Arg 29: The Axis assert that the main invasion should not land in Normandy.
- Arg 30: If the main invasion should land in Pas de Calais, then the Allied Order of Battle will affirm the invasion should land in Pas de Calais.
- Arg 31: The Allied Order of Battle affirms that the invasion should land in Pas de Calais.
- Arg 32: If the main invasion should not land in Normandy, then the Allied Order of Battle will affirm the invasion should not land in Normandy.
- Arg 33: The Allied Order of Battle affirms that the invasion should not land in Normandy.
- Arg 34: [Arg 2, Arg 3] → The Allied military invasion must use a major port.
- Arg 35: [Arg 6, Arg 7] → The main invasion should not land in Pas de Calais.
- Arg 36: [Arg 8, Arg 9] → The main invasion should land in Normandy.
- Arg 37: [Arg 10, Arg 11] → The main invasion should not land in Normandy.
- Arg 38: [Arg 12, Arg 13] → The main invasion should land in Pas de Calais.
- Arg 39: [Arg 14, Arg 15] → The main invasion should not land in Pas de Calais.
- Arg 40: [Arg 16, Arg 17] → The main invasion should land in Normandy.
- Arg 41: [Arg 18, Arg 19] → The main invasion should not land in Normandy.
- Arg 42: [Arg 20, Arg 21] → The main invasion should land in Pas de Calais.
- Arg 43: [Arg 22, Arg 23] → The main invasion should not land in Normandy.
- Arg 44: [Arg 24, Arg 25] → The main invasion should land in Pas de Calais.
- Arg 45: [Arg 26, Arg 27] → The main invasion should land in Pas de Calais.
- Arg 46: [Arg 28, Arg 29] → The main invasion should not land in Normandy.
- Arg 47: [Arg 30, Arg 31] → The main invasion should land in Pas de Calais.
- Arg 48: [Arg 32, Arg 33] → The main invasion should not land in Normandy.
- Arg 49: [Arg 4, Arg 34] → The main invasion should land in Pas de Calais.
- Arg 50: [Arg 5, Arg 34] → The main invasion should not land in Normandy.
- Arg 51: [Arg 1, Arg 37, Arg 41, Arg 43, Arg 46, Arg 48, Arg 50] → The main invasion should land in Pas de Calais.
- Arg 52: [Arg 0, Arg 35, Arg 39] → The main invasion should land in Normandy.

B.5 The Refined ACH Matrix's Argumentation Framework

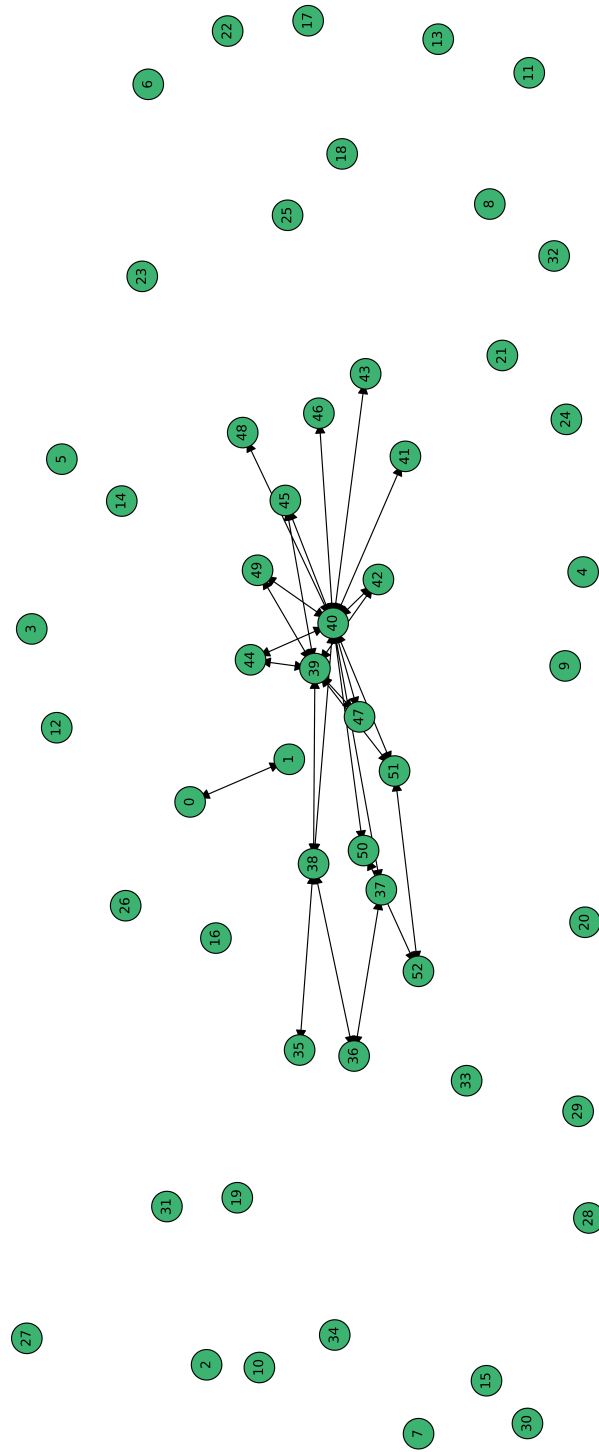


Figure B.2: The Dung argumentation framework from the refined Operation Fortitude-South ACH matrix.

Appendix C

On Human-Machine Bayesian Hypothesis Evaluation

C.1 Algorithms for Probabilistic Argumentation Frameworks

The pseudo-code for the inducement of a Dung-style argument graph, as described by Li, Oren, and Norman in [133], is formally presented in Algorithm C.1.

Algorithm C.1: An algorithm to induce a Dung argumentation framework [133].

Input : A $PrAF = (A, P_A, R, P_R)$.
Output: An induced argument graph $\mathcal{G}^I = (\mathcal{A}^I, \mathcal{R}^I)$.

```
1 Function INDUCE AF( $PrAF$ ) is
2    $\mathcal{A}^I = \emptyset$ 
3    $\mathcal{R}^I = \emptyset$ 
4   forall  $a_i \in \mathcal{A}$  do
5     Generate a random number  $r_{a_i} : r_{a_i} \in (0, 1] \subset \mathbb{R}$ 
6     if  $P_A(a) \geq r_{a_i}$  then
7        $\mathcal{A}^I = \mathcal{A}^I \cup \{a\}$ 
8   forall  $(a_i, a_j) \in \mathcal{R}$  do
9     Generate a random number  $r_{(a_i, a_j)} : r_{(a_i, a_j)} \in (0, 1] \subset \mathbb{R}$ 
10    if  $P_R((a_i, a_j)) \geq r_{(a_i, a_j)}$  then
11       $\mathcal{R}^I = \mathcal{R}^I \cup \{(a_i, a_j)\}$ 
12  return  $\mathcal{G}^I = (\mathcal{A}^I, \mathcal{R}^I)$ 
```

The semantic evaluation function, as defined in Definition 5.3, is presented in Algorithm C.2.

Algorithm C.2: The semantic evaluation function (Definition 5.3 in Section 5.2.2) [133].

Input : An induced argument graph $\mathcal{G}^I = (\mathcal{A}^I, \mathcal{R}^I)$;
 A set of arguments \mathcal{X} ;
 A semantics \mathcal{S} ;
 A count \mathcal{C} .

Output: An updated count \mathcal{C} if \mathcal{X} features in one of \mathcal{G}^I 's set of extensions.

```

1 Function SEMANTIC EVALUATION FUNCTION( $\mathcal{G}^I, \mathcal{X}, \mathcal{S}, \mathcal{C}$ ) is
2   if  $\zeta^{\mathcal{S}}(\mathcal{G}^I, \mathcal{X}) = \text{true}$  then
3      $\mathcal{C} = \mathcal{C} + 1$ 
4   return  $\mathcal{C}$ 

```

C.2 Distributed Memory Parallel Architecture

To obtain results from Algorithm 5.3 for the Operation Fortitude-South ACH matrix, distributed memory parallelism was employed on a high performance computing cluster. An example of such a distributed memory architecture is displayed in Figure C.1. The approach used up to 256 processors that possessed their own, private memory. Data was shared between CPUs through network communication.

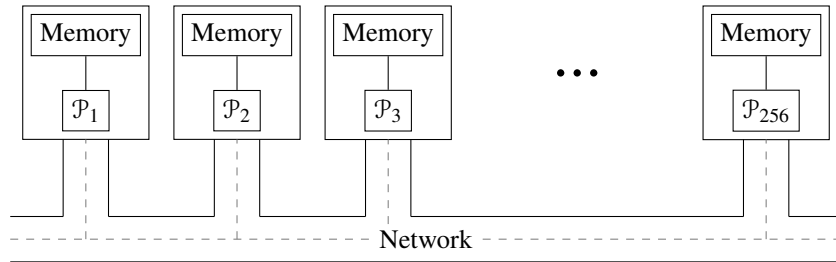


Figure C.1: Distributed memory architecture employed in the parallel execution of Algorithm 5.3.

Appendix D

On Diagnostic Arguments in Abstract Argumentation

D.1 Hybrid Distributed-Shared Memory Architecture

To obtain results from the DAI for the Operation Fortitude-South argumentation framework, a hybrid distributed-shared memory parallel architecture was employed on a high performance computing cluster. An example of such an architecture is displayed in Figure D.1.

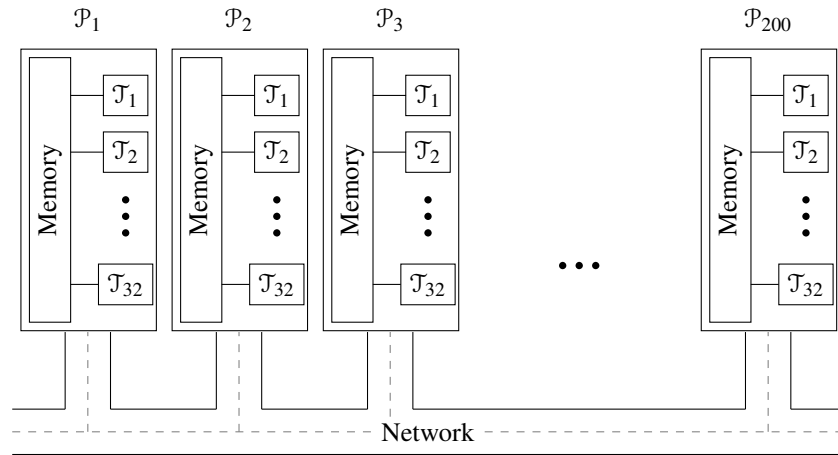


Figure D.1: An example of the hybrid distributed-shared memory parallel architecture, where each compute node \mathcal{P} had a total of 32 threads \mathcal{T} .