On Searching Explanatory Argumentation Graphs

Régis Riveret

Commonwealth Scientific and Industrial Research Organisation, Australia

ABSTRACT
Cases or examples can be often explained by the interplay of arguments in favour or against their outcomes. This paper addresses the problem of finding explanations for a collection of cases where an explanation is a labelled argumentation graph consistent with the cases, and a case is represented as a statement labelling. The focus is on semi-abstract argumentation graphs specifying attack and subargument relations between arguments, along with particular complete argument labellings taken from probabilistic argumentation where arguments can be excluded. The framework and computational considerations lead to an iterated local iterative deepening depth-first search to collect 'interesting' semi-abstract argumentation graphs consistent with a collection of statement labellings. Finally, the framework is illustrated and evaluated with a proof-of-concept implementation.

KEYWORDS
Explainable artificial intelligence, argumentation frameworks, probabilistic argumentation.

1. Introduction

A variety of models exists to represent and reason upon cases/examples in artificial intelligence. Cases can take multiple forms, ranging from combinations of unstructured data to well-defined data vectors of unambiguous features. Reasoning upon cases is also diverse. For example, reasoning can be performed by comparing aspects of cases (e.g. as in analogical reasoning), but it may be also performed via the construction of a synthetic account of the cases (e.g. a model or theory or some rules). The focus of this paper is on reasoning via synthetic accounts of collections of cases, where any case is a bivalent labelling of a set of statements (i.e. a labelling specifying which statement is accepted or not).

A 'synthetic account' refers here to the synthesis of a collection of cases into one single account that can be eventually used to explain or predict cases. Predictions contrast with explanations in that predictions are ex ante and explanations are ex post. Before a case is exposed, a prediction can be performed. Once the case has been exposed (possibly through a prediction), an explanation can be given. An explanatory synthetic account is meant to provide possible explanations for a number of cases, and often such an account paves the way for a predictive account. Explanations are not necessarily predictive. However, when explanations back predictions, they allow users to better understand and trust the predictions. Cf. diverse conceptions in philosophy e.g. (Bechtel and Abrahamsen, 2005; Hempel and Oppenheim, 1948; Lipton, 2001), psychology (Keil, 2005; Lombrozol, 2006) or (explainable) artificial intelligence (Doran, Schulz, and Besold, 2017; Doshi-Velez and Kim, 2017; Freitas, 2014; Lipton, 2016). In this paper, I will focus on explanatory synthetic accounts, i.e. the synthesis of accounts from a given collection of cases, such that each account can be used to explain the considered cases.

The original publication is available at www.tandfonline.com - https://doi.org/10.1080/11663081.2020.1766248

To get persuasive explanations, explanatory synthetic accounts can be built using elements of argumentation. It is quite natural to attempt to explain (the contingent outcome of) a given case by putting forward some arguments which relate various statements in the case. This process can be repeated for every case in a collection, and general rules can be then induced from all the advanced arguments. Such a process pertains to the field of argumentation. More generally, argumentation addresses defeasible statements raised on the basis of partial, uncertain and possibly conflicting pieces of information. It has been a traditional concern for philosophy and legal theory since ancient times, and has now become a focus for research in artificial intelligence (Atkinson, Baroni, Giacomin, Hunter, Prakken, Reed, Simari, Thimm, and Villata, 2017; Bench-Capon and Dunne, 2007). However, only a few works have investigated the automatic induction of explanatory synthetic accounts as argument-based accounts.

Let us outline how some elements of argumentation can be straightforwardly related to explanatory synthetic accounts. When a synthetic account is provided through an argument-based account, see e.g. (Bench-Capon, 2002; Bench-Capon and Sartor, 2003), *argumentation graphs* (or frameworks) (Dung, 1995) may be a useful abstract basis to determine the acceptability status of statements in a case. Given an argumentation graph, arguments (in favour or against the outcome of a case) can be labelled according to some acceptance *argument labellings* (Baroni, Caminada, and Giacomin, 2011); and from argument labellings, statements (e.g. the outcome of a case) can be labelled according to acceptance *statement labellings* (Baroni and Riveret, 2019). In this view, a case can be seen as a statement labelling, and an account of different cases as an argumentation graph along with its labellings that are consistent with the cases.

The other way around, i.e. the synthesis of argumentation graphs from statement labellings, is also important for reasoning with cases. Yet it has attracted less attention, cf. work on problems of determining graphs to account for some given argument labellings e.g. (Dunne, Dvorák, Linsbichler, and Woltran, 2015; Dyrkolbotn, 2014; Linsbichler, Pührer, and Strass, 2016; Niskanen, Wallner, and Jarvisalo, 2016, 2019; Pührer, 2015; Riveret, 2016; Riveret and Governatori, 2016). Instead of manually providing an argumentation graph (or some premises from which an argumentation graph can be built), there could be tools to automatically extract such a graph from a collection of cases. Once an argumentation graph is induced from a collection of cases, an agent could exploit arguments from the graph and determine the acceptance of some statements in order to explain cases. The work reported here deals with the automated synthesis of argumentation graphs from a given collection of cases, such that any argumentation graph can be utilised to explain every case in the collection.

Example 1.1. A medical doctor has a large collection of cases about disease c whose diagnosis is unresolved. Hidden in the data, patterns emerge. The disease often appears with symptom d and it never occurs with symptom a in isolation; but if symptom a co-occurs with symptom b then the disease may be there. Symptoms b and d never co-occur and if one appears then the other one is absent. To obtain clear diagnosis guidelines from the observed cases, arguments and their relations can be drawn in a picture that can actually be called an argumentation graph. How can the synthesis of such argument-based guidelines for diagnoses be automated?

Datasets can be inconsistent, and to deal with such datasets, argumentation models are appealing. One can put forward argument labelling semantics according to which an argumentation graph can entail specific sets of acceptable arguments, so that every set of arguments can account for a particular case. These semantics can be then embedded in probabilistic argumentation models so that a probability value is attached with every set of conclusions to reflect, for example, frequencies of features in the dataset. We endorse these ideas in the paper through the model of probabilistic labellings (Riveret, Baroni, Gao, Governatori, Rotolo, and

Sartor, 2018) to learn argumentation graphs from inconsistent datasets.

As it turns out that multiple argumentation graphs can be generated to explain a collection of cases (as shown later), a source of inspiration is Epicurus' Principle of Multiple Explanations. According to this principle, if several explanations are consistent with the observed data then we should retain them all. However, some graphs may be better or more interesting than others. For example, Ockham's Principle of Parsimony (according to which 'Entities should not be multiplied beyond necessity') suggests retaining parsimonious graphs only. By and large, the question 'What is a good explanation?' is elusive (Bechtel and Abrahamsen, 2005; Doran et al., 2017; Doshi-Velez and Kim, 2017; Freitas, 2014; Hempel and Oppenheim, 1948; Keil, 2005; Lipton, 2001, 2016; Lombrozol, 2006) and goes far beyond the work reported here. For our practical purposes, some criteria are nevertheless canvassed in this paper to gauge the (relative) 'interestingness' or 'goodness' of argumentation graphs. Given any dataset and using some criteria of interestingness, we consequently tackle the problem of finding, within a computational budget, a most interesting order of argumentation graphs to explain cases in the dataset.

To find a most interesting order of argumentation graphs, it is proposed to use an iterated local search. The proposed search can be metaphorically viewed as the iterative quest of a scientist for interesting synthetic accounts of data. At each iteration, the scientist starts from an account and slightly modifies it, then performs experiments with the modified accounts to elicit more interesting ones which can become the start of new (re)search investigations. When an iteration does not lead to any new interesting accounts, the scientist can backtrack to previous positions. As practical matters, the search allows to track and interact with the construction of such graphs. These aspects of the search are nevertheless not exploited in this paper, leaving them to future developments.

Contribution. The paper addresses the problem of finding explanatory synthetic accounts of collections of cases, where such accounts are argumentation graphs consistent with the cases and any case is a statement labelling. More specifically, the problem is to find, within a computational budget, a most interesting order of semi-abstract argumentation graphs consistent with the dataset, in other words, a particular sorting of relevant graphs. The problem is tackled with a novel combination of argument and statement labelling semantics congruent with probabilistic argumentation. These labelling semantics are then employed in an iterated local search of argumentation graphs which can be viewed as mimicking scientific enquiries. The local search is an iterative deepening depth-first search oriented by a simple heuristics.

The problem is closely related to problems in fields of machine learning such as structure learning and rule learning, and the proposed approach is inspired by works in these fields. Instead of classic constructs such as rules for predictive purposes, the investigation works on argument and statement labelling semantics as used in probabilistic argumentation, and the focus is on synthesis of argumentation graphs for explaining cases. Hence, the contribution aims at opening and exploring a novel argument-based perspective on synthetic constructions for explanatory purposes, rather than competing with the state-of-the-art in related fields of data mining or machine learning.

Outline. Section 2 further motivates the undertaking. Section 3 introduces the probabilistic semi-abstract argumentation setting. Section 4 formally defines the problem. Section 5 discusses consistency of argumentation graphs and Section 6 proposes various criteria to evaluate the interestingness of argumentation graphs consistent with a dataset. An iterated local search is put forward in Section 7 to find interesting argumentation graphs, along with heuristics to guide the local search. The framework is illustrated and evaluated with a proof-of-concept implementation in Section 9, and it is related to other work in Section 10, before the paper's conclusion.

2. Some Motivations

Motivations to synthesise semi-abstract argumentation graphs from datasets within the framework of probabilistic argumentation can be partitioned into (i) motivations for argumentation graph synthesis from datasets, (ii) motivations to do it through probabilistic argumentation. and (iii) motivations for the synthesis of semi-abstract argumentation graphs.

2.1. Motivations for argumentation graph synthesis from datasets

A variety of formal models exists to capture specific aspects of the complexity of argumentation activities. For instance, rule-based or logic-based argument models (Modgil and Prakken, 2014; Toni, 2014) can be integrated with formal approaches for the evaluation of the acceptance status of arguments, possibly at a more abstract level (Baroni, Governatori, and Riveret, 2016; Dung, 1995). Moreover, argument acceptance can be projected on the statements corresponding to their conclusions, in order to assess their acceptance in turn. In general, multiple aspects and their respective models need to be combined to get a satisfactory coverage of the full picture.

The first activity in an argumentation process is often the production of a set of arguments (possibly built from a set of premises) (Besnard, Garcia, Hunter, Modgil, Prakken, Simari, and Toni, 2014). Attack or support relations can be also established leading to argumentation graphs, dialogues or other structures. While there has been a long-lasting effort in making tools to automatically construct argumentation structures from a set of premises and to compute the acceptance of the considered arguments (Cerutti, Gaggl, Thimm, and Wallner, 2017), the automatic production of arguments or arguments' premises is a more recent avenue of research.

A remarkable thread of research concerns 'argument mining' from textual resources (Lawrence and Reed, 2020; Lippi and Torroni, 2016). However, it is equally possible to think of extracting arguments from collections of cases/examples made of attribute-value pairs. After all, arguments are often built from rules, and there exists a substantial amount of work concerning rule learning from examples (Fürnkranz, Gamberger, and Lavrač, 2014). Thus it may be worth investigating how arguments could be learnt from examples. Yet, such resources have hardly been considered for extracting arguments and their relations. This paper addresses the automatic synthesis of abstract argumentation graphs from collections of cases.

2.2. Motivations for argumentation graph synthesis with probabilistic argumentation

A collection of cases can be 'consistent' or 'inconsistent'. In machine learning literature, a consistent dataset usually refers to a set of cases where, for every case, the same conditions lead to the same outcome, while an inconsistent dataset includes cases where the same conditions lead to different outcomes. However, the qualification of an inconsistent dataset can be misleading because we could understand that some inconsistent cases should not hold altogether or may refer to noisy settings, whereas these cases may merely reflect some underlying non-deterministic phenomena. For this reason, instead of referring to consistent or inconsistent datasets, we will talk of deterministic or non-deterministic datasets in this paper.

To account for non-deterministic datasets, argumentation models are appealing. One can put forward semantics according to which an argumentation graph can entail different sets of acceptable conclusions, so that every set of conclusions can account for a particular case. These semantics can be then embedded in probabilistic argumentation models so that probability values are attached to arguments and conclusions in order to reflect the frequencies of some features in the dataset. In this view, probability values are interpreted as frequencies, thus giving a frequentist interpretation and a valuation method for probability values in probabilistic argumentation. Consistent with these ideas, a model of probabilistic labellings (Riveret et al., 2018) is adopted in the paper to 'learn' argumentation graphs from non-deterministic datasets.

The fundamental problem of learning a logical structure from cases in a probabilistic setting has been notably addressed in statistical relational learning (SRL) (Getoor and Taskar, 2007). SRL is at the crossroad of formalisms for logical reasoning, principled probabilistic and statistical approaches, and machine learning. It has successful integrations, see e.g. Probabilistic Relational Models (Getoor, Friedman, Koller, Pfeffer, and Taskar, 2007) and Markov Logic Networks (Richardson and Domingos, 2006). In these approaches, logics are used to structure in a qualitative manner the probabilistic relations amongst entities. Typically, (a subset of) first-order logic formally represents a qualitative knowledge describing the structure of a domain in a general manner (using universal quantification), while techniques from graphical models (Koller and Friedman, 2009), such as Bayesian networks (Getoor et al., 2007) or Markov networks (Richardson and Domingos, 2006), are applied to handle probability measures on the structured entities. Although argumentation plays no role in these approaches, and they concentrate on capturing data in its relational form (while we are only dealing with an abstract and flat data representation induced by our setting of abstract argumentation), one can be inspired by these combinations of graphical models and logic-based systems, and propose frameworks for probabilistic argumentation benefiting from the use of graphical models too. For example, a probabilistic abstract argumentation setting in (Riveret, Korkinof, Draief, and Pitt, 2015a; Riveret, Pitt, Korkinof, and Draief, 2015b), which uses the same labelling framework for probabilistic argumentation employed in the paper, underlies a combination of abstract argumentation and the graphical model of Boltzmann machines. By learning the abstract argumentation graph and by combining it with some probabilistic graphical models, the learnt explanatory argumentation model would not only be used to explain the outcomes, it could also be employed to predict outcomes.

2.3. Motivations for semi-abstract argumentation graph synthesis

In the probabilistic argumentation setting which is used in this paper, an important postulate is that the event of an expressed argument necessarily occurs along with the event of its expressed subarguments. In that regard, standard abstract argumentation graphs (Dung, 1995) lack a subargument relation to cater for this contention at an abstract level. To address this lacuna, we rely on so-called semi-abstract argumentation graphs (Riveret et al., 2018) which are basically networks of arguments connected by attack and subargument relations. In addition, it is assumed that any argument has a conclusion which is a statement, so that we can determine statement acceptance statuses wrt argument acceptance statuses. Hence, instead of bare abstract argumentation graphs, the work reported here is actually on semi-abstract argumentation graphs where every argument has a conclusion.

Such semi-abstract argumentation graphs inherently have a graphical representations which can appear as an appealing 'explanation interface'. Then, by attaching marginal probability values to arguments, probabilistic argumentation graphs can be drawn and appreciated by end-users, while making a straightforward connection with other works on probabilistic argumentation graphs/frameworks (Hunter, 2013; Polberg and Hunter, 2018) along with a subargument relation (Riveret et al., 2018).

The subargument relation in semi-abstract argumentation graphs can be understood as a *support* relation, and the support relation is a salient feature of *bipolar argumentation* (Cayrol and Lagasquie-Schiex, 2013). For this reason, semi-abstract argumentation graphs may be also called 'bipolar argumentation graphs', as long as the support relation in such bipolar graphs is interpreted as a subargument relation and any argument is assumed to have a conclusion. Furthermore, such a bipolar setting would have to enjoy all the (probabilistic) characteristics of the semi-abstract setting as exposed in the present paper.

Finally, the synthesis of semi-abstract argumentation graphs allows the overall induction problem to be broken down into smaller pieces. Instead of frontally facing the induction of detailed synthetic accounts (such as a set of rules or a theory), we can, for example, first address the synthesis of abstract graphs, and then look at the induction of more detailed accounts from the graphs. As we abstract from any particular underlying logic-based framework, we also prepare the ground for possible graph instantiations with different models of structured argumentation (Besnard et al., 2014), especially those which can be understood in terms of argumentation graphs. Hence, a semi-abstract approach may help to better solve the overall synthesis problem for different structured argumentation frameworks.

3. Probabilistic Argumentation Setting

The probabilistic argumentation setting relies on probabilistic labellings (Riveret et al., 2018) of semi-abstract argumentation graphs (and subgraphs) which are a slight development of abstract argumentation graphs (Dung, 1995). On the basis of semi-abstract argumentation graphs, arguments are labelled using argument labellings where arguments can be omitted, and statements are eventually labelled following bivalent statement labellings.

3.1. Semi-abstract argumentation graphs

The overall framework rests on a language which is a set of statements. Over a language, contraries and contradictories can be specified. Yet, the specification of contrary or contradictory statements can appear quite elusive in computational models of argument, see e.g. (Baroni, Giacomin, and Liao, 2015). In addition, one may prefer to endorse a basic and well-established conception in classical logic according to which contraries cannot be 'true' together, while contradictories are such that one of them is 'true' if and only if the others are 'false'. We adopt this classical conception, and we focus on contrary and contradictory binary relations regrouped into so-called 'language graphs'. Such language graphs are not usually explicitly exposed in computational models of argument: they are advanced here because they will be useful for our purposes.

Definition 3.1. Given a language Ψ, a **language graph** is a tuple $\langle \Phi, contrar, contrad \rangle$ where $\Phi \subseteq \Psi$ is a set of statements, $contrar \subseteq \Phi \times \Phi$ is a binary symmetric contrary relation, and $contrad \subseteq \Phi \times \Phi$ is a binary symmetric contradictory relation such that $contrar \cap contrad = \emptyset$.

Example 3.2. Relevant elements of the language in Example 1.1 can be captured by the language graph $\langle \{\mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{d}\}, \emptyset, \{(\mathsf{b}, \mathsf{d}), (\mathsf{d}, \mathsf{b})\} \rangle$ which specifies that b and d are contradictories.

Notation 1. Let $\mathfrak{L} = \langle \Phi, contrar, contrad \rangle$ be a language graph, we may denote Φ, $contrar$ and $contrad$ as $\Phi_{\mathfrak{L}}$, $contrar_{\mathfrak{L}}$ and $contrad_{\mathfrak{L}}$, respectively.

In the remainder, underlying language and language graphs may be left implicit.

Any statement of a language graph can be the conclusion of an argument. Any argument is associated with a unique identifier to distinguish arguments with equal conclusions. The set of identifiers and statement conclusions are not necessarily disjoint here, cf. (Baroni, Giacomin, and Liao, 2018), since identifiers will have no effects on the acceptance status of arguments and their conclusions.

Definition 3.3. Given a language Ψ and a set of argument identifiers $\mathcal{I}$, an **argument** is a tuple $\langle id, \phi \rangle$ where $id \in \mathcal{I}$ is the unique identifier of the argument and $\phi \in \Psi$ is the conclusion of the argument.

Notation 2. The conclusion of an argument $A = \langle id, \phi \rangle$ is denoted $\mathrm{con}(A)$, i.e. $\mathrm{con}(A) = \phi$.

In the probabilistic argumentation setting which is used in this paper, an important postulate is that the event of an argument necessarily occurs along with the event of its subarguments. In that regard, classic abstract argumentation graphs (Dung, 1995) lack a subargument relation to cater for this contention at an abstract level. To address this lacuna, we rely on so-called semi-abstract argumentation graphs (Riveret et al., 2018) featuring subargument and attack relations, cf. (Cayrol and Lagasquie-Schiex, 2013; Cohen, Gottifredi, García, and Simari, 2014; Dung and Thang, 2014; Prakken, 2014) for similar settings.

Definition 3.4. A **semi-abstract argumentation graph** is a tuple $\langle \mathcal{A}, \rightsquigarrow, \Mapsto \rangle$ where $\mathcal{A}$ is a set of arguments, $\rightsquigarrow \subseteq \mathcal{A} \times \mathcal{A}$ is a binary attack relation, $\Mapsto \subseteq \mathcal{A} \times \mathcal{A}$ is a binary direct subargument relation.

Notation 3. Let $\mathcal{G} = \langle \mathcal{A}, \rightsquigarrow, \Mapsto \rangle$ be a semi-abstract argumentation graph. We may denote the set of arguments $\mathcal{A}$, and the relations $\rightsquigarrow$ and $\Mapsto$ as $\mathcal{A}_\mathcal{G}$, $\rightsquigarrow_\mathcal{G}$ and $\Mapsto_\mathcal{G}$ respectively.

The 'direct subargument' relation may be called a 'support' relation, and thus when $A \Mapsto B$ we may say 'A supports B' instead of 'A is a direct subargument of B'. As support relationships are essential in bipolar argumentation (Cayrol and Lagasquie-Schiex, 2013), semi-abstract argumentation graphs may be also called 'bipolar argumentation graphs'. This holds on condition that the support relation in such bipolar graphs is understood as a direct subargument relation and any argument is assumed to have a conclusion, and that such graphs feature all the characteristics of semi-abstract argumentation graphs as introduced in the present work. However, the terminology is relatively inconsequential here, and in the remainder 'semi-abstract argumentation graphs' may be simply called argumentation graphs for the sake of brevity.

Furthermore, the direct subargument relation should not be confused with a more general subargument relation as defined below.

Definition 3.5. Given an argumentation graph $\mathcal{G}$, its **subargument relation** is a binary relation $\Longmapsto_\mathcal{G} \subseteq \mathcal{A}_\mathcal{G} \times \mathcal{A}_\mathcal{G}$ such that $A \Longmapsto_\mathcal{G} B$ iff $A \Mapsto_\mathcal{G} B$ or $\exists C \in \mathcal{A}_\mathcal{G}$ such that $A \Longmapsto_\mathcal{G} C$ and $C \Mapsto_\mathcal{G} B$.

Arguments without subarguments are called here assumptive arguments, and an assumption is the conclusion of an assumptive argument.

Definition 3.6. An argument is an **assumptive argument** iff its set of subarguments is empty.

Often a case exposes some facts rather than assumptions. Without entering into sophisticated epistemic discussions, a fact can be seen as a piece of information backed by sufficient evidence while an assumption does/may not refer to any evidence. To avoid overloading the framework, and because a fact can be assumed too (in the sense that an evidence can be assumed), the term assumption is used in the paper to also encompass facts.

As such, some semi-abstract argumentation graphs may not appear 'well-formed' since, for instance, an argument A can attack an argument B without attacking the arguments supported by B. For this reason, well-formed semi-abstract argumentation graphs are considered next.

Definition 3.7. A semi-abstract argumentation graph $\langle \mathcal{A}, \rightsquigarrow, \Mapsto \rangle$ is a **well-formed semi-abstract argumentation graph** iff:

- the relation $\Mapsto$ is acyclic and antireflexive, and
- if an argument A attacks an argument B, and B is a direct subargument of an argument C, then A attacks C.

In a well-formed semi-abstract argumentation graph, an argument can attack and support another argument, as sometimes allowed in rule-based argumentation frameworks. In the remainder, argumentation graphs are assumed semi-abstract and well-formed, unless specified otherwise.

Argumentation graphs can be induced by a set of statements, and arguments can have contrary or contradictory conclusions. In that regard, an argumentation graph and a language graph can be consistent or congruent.

Definition 3.8.

- An argumentation graph $\mathcal{G}$ is an **argumentation graph induced by a set of statements** Φ iff every statement in Φ is the conclusion of at least one argument in $\mathcal{A}_{\mathcal{G}}$ and the conclusion of every argument in $\mathcal{A}_{\mathcal{G}}$ is in Φ, i.e. $\Phi = \{\phi \mid \phi = \mathrm{con}(A), A \in \mathcal{A}_{\mathcal{G}}\}$.
- An argumentation graph $\mathcal{G}$ is **strictly induced** by Φ iff $\mathcal{G}$ is induced by Φ and $|\Phi| = |\mathcal{A}_{\mathcal{G}}|$.

Definition 3.9. An argumentation graph $\mathcal{G}$ and a language graph $\mathfrak{L}$ are **consistent** (**congruent** resp.) iff for every $A, B \in \mathcal{A}_{\mathcal{G}}$:

- if $(\mathrm{con}(A), \mathrm{con}(B)) \in contrar_{\mathfrak{L}}$ then $A \rightsquigarrow_{\mathcal{G}} B$ or (‘*and*’ resp.) $B \rightsquigarrow_{\mathcal{G}} A$, and
- if $(\mathrm{con}(A), \mathrm{con}(B)) \in contrad_{\mathfrak{L}}$ then $A \rightsquigarrow_{\mathcal{G}} B$ or (‘*and*’ resp.) $B \rightsquigarrow_{\mathcal{G}} A$.

Hence, if an argumentation graph and a language graph are congruent then they are also consistent. The distinction between consistent and congruent graphs is used later to characterise some particular graphs.

When labelling arguments and synthesising argumentation graphs, we will employ the concept of subgrahs. Amongst subgraphs, we distinguish ‘argument subgraphs’ and ‘spanning subgraphs’. An argument subgraph $\mathcal{H}$ is a subgraph of a graph $\mathcal{G}$ such that $\mathcal{H}$ is induced by a set of arguments $\mathcal{A}_{\mathcal{H}}$, i.e. a subgraph which has exactly the attacks and supports that appear in $\mathcal{G}$ over the same set of arguments. An attack-support subgraph of $\mathcal{G}$ is a subgraph whose attacks and supports are subsets of the attacks and supports of $\mathcal{G}$, and such that all arguments of the original graph remain. Hence an attack-support subgraph can be called a *spanning* subgraph, but the term ‘attack-support’ will better fit other notions later in the paper.

Definition 3.10. Let $\mathcal{G}$ denote an argumentation graph.

- A **subgraph** $\mathcal{H}$ **of** $\mathcal{G}$ is an argumentation graph $\mathcal{H} = \langle \mathcal{A}_{\mathcal{H}}, \rightsquigarrow_{\mathcal{H}}, \Mapsto_{\mathcal{H}} \rangle$, where $\mathcal{A}_{\mathcal{H}} \subseteq \mathcal{A}_{\mathcal{G}}$ and $\rightsquigarrow_{\mathcal{H}} \subseteq \rightsquigarrow_{\mathcal{G}}$ and $\Mapsto_{\mathcal{H}} \subseteq \Mapsto_{\mathcal{G}}$.
- An **argument subgraph** $\mathcal{H}$ **of** $\mathcal{G}$ **induced by a set of arguments** $\mathcal{A}_{\mathcal{H}} \subseteq \mathcal{A}_{\mathcal{G}}$ is an argumentation graph such that $\mathcal{H} = \langle \mathcal{A}_{\mathcal{H}}, \rightsquigarrow_{\mathcal{G}} \cap (\mathcal{A}_{\mathcal{H}} \times \mathcal{A}_{\mathcal{H}}), \Mapsto_{\mathcal{G}} \cap (\mathcal{A}_{\mathcal{H}} \times \mathcal{A}_{\mathcal{H}}) \rangle$.
- An **attack-support subgraph** $\mathcal{H}$ **of** $\mathcal{G}$ **induced by an attack relation** $\rightsquigarrow_{\mathcal{H}} \subseteq \rightsquigarrow_{\mathcal{G}}$ **and a direct subargument relation** $\Mapsto_{\mathcal{H}} \subseteq \Mapsto_{\mathcal{G}}$ is an argumentation graph such that $\mathcal{H} = \langle \mathcal{A}_{\mathcal{G}}, \rightsquigarrow_{\mathcal{H}}, \Mapsto_{\mathcal{H}} \rangle$.

In the rest of the paper, we may omit to mention the set of arguments or the direct subargument relation used to induce a subgraph.

Amongst all the argument subgraphs of an argumentation graphs, we will be particularly interested by subgraphs such that every argument is included along with all its subarguments.

In our jargon, we are interested in subargument-complete subgraphs.

Definition 3.11. An argument subgraph $\mathcal{H}$ of an argumentation graph $\mathcal{G}$ is **subargument-complete** iff for every argument $A \in \mathcal{A}_{\mathcal{H}}$ if $B \Rightarrow_{\mathcal{G}} A$ then $B \in \mathcal{A}_{\mathcal{H}}$.

Example 3.12. Referring to the example in Section 1, the subtle diagnosis of the disease may turn out to be actually synthesised into an argumentation graph G as displayed in Figure 1, where $\mathcal{A}_{\mathsf{G}} = \{\mathsf{A}, \mathsf{B}, \mathsf{C1}, \mathsf{C2}, \mathsf{D}\}$, $\rightsquigarrow_{\mathsf{G}} = \{(\mathsf{A}, \mathsf{C2}), (\mathsf{B}, \mathsf{D}), (\mathsf{D}, \mathsf{B}), (\mathsf{B}, \mathsf{C2}), (\mathsf{D}, \mathsf{C1})\}$ and $\Rightarrow_{\mathsf{G}} = \{(\mathsf{A}, \mathsf{C1}), (\mathsf{B}, \mathsf{C1}), (\mathsf{D}, \mathsf{C2})\}$. In words, arguments A, B and D are assumptive arguments. Arguments A and B are direct subarguments of C1. Argument D is a direct subargument of C2. Argument A attacks argument C2 (by undercutting it for example). Arguments B and D attack each other, and thus B attacks C2, and D attacks C1. Each argument has a conclusion, here $\mathrm{con}(\mathsf{A}) = \mathsf{a}$, $\mathrm{con}(\mathsf{B}) = \mathsf{b}$, $\mathrm{con}(\mathsf{C1}) = \mathsf{c}$, $\mathrm{con}(\mathsf{C2}) = \mathsf{c}$, and $\mathrm{con}(\mathsf{D}) = \mathsf{d}$. The argumentation graph is well-formed and it is congruent with the language graph $\langle\{\mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{d}\}, \emptyset, \{(\mathsf{b}, \mathsf{d}), (\mathsf{d}, \mathsf{b})\}\rangle$.

Two argument subgraphs and an attack-support subgraph of graph G shown in Figure 1 are drawn in Figure 2. Graph (a) is a subargument-complete subgraph of graph G, while graph (b) is not because argument B is not included though it is a subargument of C1.

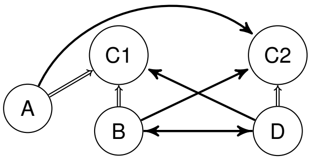


Figure 1.: A well-formed argumentation graph.

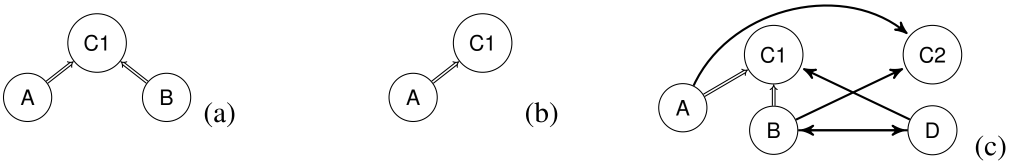


Figure 2.: Two argument subgraphs (a-b) and an attack-support subgraph (c).

□

Two argumentation graphs may have the same number of arguments which have equal conclusions and which are connected in the same way. They may only differ in the set of argument identifiers (see Definition 3.3). Hence, such argumentation graphs are not formally equal, they are isomorphic.

Definition 3.13. Two argumentation graphs $\mathcal{G}$ and $\mathcal{G}'$ are isomorphic iff there exists a bijection $f : \mathcal{A}_{\mathcal{G}} \rightarrow \mathcal{A}_{\mathcal{G}'}$ such that, for any $A, B \in \mathcal{A}_{\mathcal{G}}$,

- $A \rightsquigarrow_{\mathcal{G}} B$ iff $f(A) \rightsquigarrow_{\mathcal{G}'} f(B)$, and
- $A \Rightarrow_{\mathcal{G}} B$ iff $f(A) \Rightarrow_{\mathcal{G}'} f(B)$, and
- $\mathrm{con}(A) = \mathrm{con}(f(A))$.

Given two argumentation graphs $\mathcal{G}$ and $\mathcal{G}$' which are isomorphic through a bijection f, we may say, for the sake of simplicity, that a set of arguments $\mathcal{A} \subseteq \mathcal{A}_{\mathcal{G}}$ is a subset of a set of arguments $\mathcal{B} \subseteq \mathcal{A}_{\mathcal{G}'}$ (or equivalently $\mathcal{B}$ is a superset of $\mathcal{A}$) iff for every argument A in $\mathcal{A}$, there is an argument B in $\mathcal{B}$ such that $f(A) = B$. The notion of isomorphism is used later to compare argumentation graphs.

To recap, instead of abstract argumentation graphs, we work on semi-abstract argumentation graphs, and in particular on well-formed semi-abstract argumentation graphs. We have defined subgraphs, (subargument-complete) argument subgraphs and attack-support subgraphs, which can be used to label arguments, and later to synthesise graphs.

3.2. Argument labellings

Given an argumentation graph, the sets of arguments that are accepted or not are determined using some semantics. For our purposes, we adopt complete semantics (Dung, 1995) by labelling arguments as in the labelling approach to probabilistic argumentation (Riveret et al., 2018) – a slight probabilistic adaptation of the labellings reviewed in (Baroni et al., 2011). So, we distinguish $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}\}$-labellings and $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}, \textsf{OFF}\}$-labellings. In a $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}\}$-labelling, each argument is associated with one label which is either IN, OUT, UND, respectively meaning that the argument is accepted, rejected, or undecided. In a $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}, \textsf{OFF}\}$-labelling, the label OFF indicates that the argument is omitted or not considered, for example when it is not put forward in a dialogue for a case.

Definition 3.14. Let $\mathcal{G}$ be an argumentation graph.

- A **$\{\textsf{IN}, \textsf{OUT}, \textsf{UND}\}$-labelling** of $\mathcal{G}$ is a total function $L : \mathcal{A}_\mathcal{G} \to \{\textsf{IN}, \textsf{OUT}, \textsf{UND}\}$.
- A **$\{\textsf{IN}, \textsf{OUT}, \textsf{UND}, \textsf{OFF}\}$-labelling** of $\mathcal{G}$ is a total function $L : \mathcal{A}_\mathcal{G} \to \{\textsf{IN}, \textsf{OUT}, \textsf{UND}, \textsf{OFF}\}$.

Notation 4.

- For any label $l \in \{\textsf{IN}, \textsf{OUT}, \textsf{UND}, \textsf{OFF}\}$, $l(L)$ stands for $\{A \mid L(A) = l\}$. For instance $\textsf{IN}(L) = \{A \mid L(A) = \textsf{IN}\}$.
- A $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}\}$-labelling L is represented as a tuple $\langle \textsf{IN}(L), \textsf{OUT}(L), \textsf{UND}(L)\rangle$, and a $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}, \textsf{OFF}\}$-labelling L as a tuple $\langle \textsf{IN}(L), \textsf{OUT}(L), \textsf{UND}(L), \textsf{OFF}(L)\rangle$.

Most $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}\}$-labellings studied in the literature are 'complete' (Baroni et al., 2011), and we will work with such complete labellings in the remainder of the paper.

Definition 3.15. A **complete $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}\}$-labelling** of an argumentation graph $\mathcal{G}$ is a $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}\}$-labelling such that for every argument A in $\mathcal{A}_\mathcal{G}$ it holds that:

(1) A is labelled IN iff all attackers of A are labelled OUT, and
(2) A is labelled OUT iff A has an attacker labelled IN.

An argumentation graph may have several complete $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}\}$-labellings, and the labelling of arguments of a graph can be further specified, see e.g. (Baroni et al., 2011). We will work with complete $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}\}$-labellings.

When some arguments are omitted or not expressed by the concerned arguers, as it can happen in the upcoming probabilistic argumentation setting, we have $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}, \textsf{OFF}\}$-labellings where only expressed arguments can effectively attack other arguments. Hence, we can have complete $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}, \textsf{OFF}\}$-labellings.

Definition 3.16. Let $\mathcal{G}$ denote an argumentation graph and $\mathcal{H}$ a subargument-complete argument subgraph of $\mathcal{G}$. A **complete $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}, \textsf{OFF}\}$-labelling** of $\mathcal{G}$ is a $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}, \textsf{OFF}\}$-labelling where:

- every argument in $\mathcal{A}_\mathcal{H}$ is labelled as in a complete $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}\}$-labelling of $\mathcal{H}$,
- every argument in $\mathcal{A}_\mathcal{G} \backslash \mathcal{A}_\mathcal{H}$ is labelled OFF.

Example 3.17. A complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}\}$-labelling and a complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling (amongst others) are illustrated in Figure 3 and Figure 4 respectively.

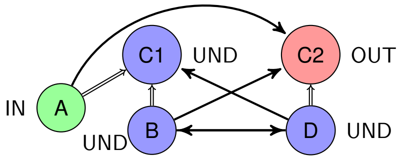


Figure 3.

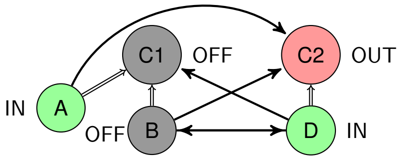


Figure 4.

□

Other labelling semantics can be certainly employed, see e.g. (Baroni et al., 2011). In order to avoid overloading the work reported here, we will focus on the above-mentioned labellings, and other labelling semantics are left to future investigations.

3.3. Statement labellings

In our argumentation context, cases are formalised as statement labellings. Given a set of statements pertaining to some language, a labelling of this set is a (preferably total) function associating any statement with a label. Diverse specifications for statement labellings are possible (Baroni and Riveret, 2019; Baroni et al., 2016). As we are interested in featured-based cases where features are Boolean-valued (as we will see soon), we consider bivalent labellings according to which a statement is either accepted or not, without further sophistication. If a statement is accepted then we label it ‘y’, otherwise it is labelled ‘n’.

Definition 3.18. Let Φ be a set of statements. A **bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling** of Φ is a total function $K : \Phi \rightarrow \{\mathsf{y}, \mathsf{n}\}$.

Notation 5. The statements labelled in a labelling K, i.e. the domain of K, is denoted Φ_K.

Following (Baroni and Riveret, 2019), if statements are labelled relatively to a particular argument labelling, then we have a statement ‘acceptance’ labelling.

Definition 3.19.

- Let $\mathcal{L}$ be a set of $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings, Φ a set of statements. An **acceptance bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling of Φ and from $\mathcal{L}$** is a total function $K : \mathcal{L}, \Phi \rightarrow \{\mathsf{y}, \mathsf{n}\}$ such that for every $L \in \mathcal{L}$, $\phi \in \Phi$, $K(L, \phi) = \mathsf{y}$ iff $\exists A \in \mathsf{IN}(L) : \mathrm{con}(A) = \phi$.
- Let $\mathcal{L}$ be a set of $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings of an argumentation graph $\mathcal{G}$. An **acceptance bivalent** $\{\mathsf{y}, \mathsf{n}\}$**-labelling of $\mathcal{G}$ and from $\mathcal{L}$** is an acceptance bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling of $\Phi = \{\phi \mid \phi = \mathrm{con}(A), A \in \mathcal{A}_{\mathcal{G}}\}$ and from $\mathcal{L}$.

Specific $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings can give rise to specific acceptance bivalent $\{\mathsf{y}, \mathsf{n}\}$-labellings. For our ends, we may say that we have *complete* bivalent $\{\mathsf{y}, \mathsf{n}\}$-labellings of $\mathcal{G}$ and from $\mathcal{L}$ if $\mathcal{L}$ is a set of *complete* $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings of $\mathcal{G}$. Eventually, an acceptance bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling may be simply called a bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling.

Example 3.20. Figure 5 shows possible complete bivalent $\{\mathsf{y}, \mathsf{n}\}$-labellings from complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$ labellings (amongst others) of the argumentation graph in Figure 1. □

Notation 6. We may omit to specify the set of labellings $\mathcal{L}$ attached to an acceptance bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling when the context raises no ambiguities. Eventually, a bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling

A	B	C1	D	C2
IN	UND	UND	UND	OUT
IN	UND	UND	UND	OFF
IN	IN	IN	OFF	OFF
IN	UND	OFF	UND	OUT
IN	UND	OFF	UND	OFF
IN	IN	OFF	OFF	OFF
IN	OFF	OFF	IN	OUT
IN	OFF	OFF	IN	OFF
IN	OFF	OFF	OFF	OFF
OFF	UND	OFF	UND	UND
OFF	UND	OFF	UND	OFF
OFF	IN	OFF	OFF	OFF
OFF	OFF	OFF	IN	IN
OFF	OFF	OFF	IN	OFF
OFF	OFF	OFF	OFF	OFF

a	b	c	d
y	n	n	n
y	n	n	n
y	y	y	n
y	n	n	n
y	n	n	n
y	y	n	n
y	n	n	y
y	n	n	y
y	n	n	n
n	n	n	n
n	n	n	n
n	y	n	n
n	n	y	y
n	n	n	y
n	n	n	n

Figure 5.: Complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings (amongst others), and corresponding acceptance bivalent $\{\mathsf{y}, \mathsf{n}\}$-labellings. Each row is a labelling, with the obvious meaning.

K or an acceptance bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling K can be represented as a tuple $\langle \mathsf{y}(K), \mathsf{n}(K)\rangle$ with the obvious meaning.

Consistency can be defined between bivalent $\{\mathsf{y}, \mathsf{n}\}$-labellings and language graphs to reflect the classical conception of contraries and contradictories: accordingly, contraries cannot be labelled y together (but they can be labelled n together), while contradictories are such that one of them is labelled y if and only if the other is labelled n.

Definition 3.21.

- **A bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling K and a language graph $\mathfrak{L}$ are consistent** iff
  - for every pair $(\phi, \beta) \in contrar_{\mathfrak{L}}$, if $K(\phi) = \mathsf{y}$ then $K(\beta) = \mathsf{n}$, and
  - for every pair $(\phi, \beta) \in contrad_{\mathfrak{L}}$, $K(\phi) = \mathsf{y}$ iff $K(\beta) = \mathsf{n}$.
- **A collection of bivalent $\{\mathsf{y}, \mathsf{n}\}$-labellings $\mathcal{K}$ and a language graph $\mathfrak{L}$ are consistent** iff for every labelling $K \in \mathcal{K}$, the labelling K and the language graph $\mathfrak{L}$ are consistent.

Consistency can be also defined between bivalent $\{\mathsf{y}, \mathsf{n}\}$-labellings and argumentation graphs or $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings.

Definition 3.22. Let $\mathcal{L}$ be a set of $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings of an argumentation graph $\mathcal{G}$.

- **A bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling K and a $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling L are consistent** iff for every statement $\phi \in \Phi_K$, $K(\phi) = K(L, \phi)$ where $K(L, \phi)$ is an acceptance bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling of Φ_K and from $\{L\}$.
- **A bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling K and an argumentation graph $\mathcal{G}$ are consistent** wrt $\mathcal{L}$ iff there exists a complete labelling $L \in \mathcal{L}$ such that K and L are consistent.
- **A collection of bivalent $\{\mathsf{y}, \mathsf{n}\}$-labellings $\mathcal{K}$ and an argumentation graph $\mathcal{G}$ are consistent** wrt $\mathcal{L}$ iff for every labelling $K \in \mathcal{K}$, the labelling K and graph $\mathcal{G}$ are consistent wrt $\mathcal{L}$.

For the sake of brevity, bivalent $\{\mathsf{y}, \mathsf{n}\}$-labellings and argumentation graphs may be said to be consistent without mentioning any specific set of $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings.

Example 3.23. The language graph $\langle\{\mathsf{a},\mathsf{b},\mathsf{c},\mathsf{d}\},\emptyset,\{(\mathsf{b},\mathsf{d}),(\mathsf{d},\mathsf{b})\}\rangle$ is consistent with the collection of bivalent $\{\mathsf{y},\mathsf{n}\}$-labellings in Figure 6. The collection of bivalent $\{\mathsf{y},\mathsf{n}\}$-labellings and the argumentation graph $\mathcal{G}$ in Figure 1 are consistent wrt the complete $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labellings of $\mathcal{G}$. □

a	b	c	d
y	y	y	n
y	y	n	n
y	n	n	y
n	y	n	n
n	n	y	y
n	n	n	y

Figure 6.: A collection of bivalent $\{\mathsf{y},\mathsf{n}\}$-labellings.

In this context, an explanation for a bivalent $\{\mathsf{y},\mathsf{n}\}$-labelling can be viewed as an argumentation graph along with one of its complete $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labelling to produce the bivalent labelling. The underlying graph is defined as an explanatory argumentation graph. In other words, an explanatory argumentation graph for a bivalent $\{\mathsf{y},\mathsf{n}\}$-labelling K (for a collection $\mathcal{K}$ resp.) is basically an argumentation graph consistent with K (with $\mathcal{K}$ resp.).

Labels from $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labellings allow more fine-grained explanations than $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND}\}$-labellings. When using $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND}\}$-labelling semantics, the fact that a statement is labelled n may be explained by showing that all its supporting arguments are labelled OUT or UND. However, a statement may be labelled n simply because there are no arguments put forward to support the statement. In this case, all potential arguments in favour of the statement should be rather labelled OFF. Such an important distinction can be carried through $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labelling semantics.

Example 3.24. If there is a case for which disease c is diagnosed without the manifestation of symptom a, then no arguments supporting a may have to be advanced. Such arguments may be simply omitted and labelled OFF.

Eventually, the combination of specific $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labelling and bivalent $\{\mathsf{y},\mathsf{n}\}$-labelling semantics allows single argumentation graphs to account for collections of $\{\mathsf{y},\mathsf{n}\}$-labellings that bare complete $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND}\}$-labelling semantics may be unable to account for. For example, no complete $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND}\}$-labellings of the argumentation graph in Figure 1 can account for the $\{\mathsf{y},\mathsf{n}\}$-labelling $\langle\{\mathsf{b}\},\{\mathsf{a},\mathsf{c},\mathsf{d}\}\rangle$, whereas this statement labelling is covered by multiple $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labellings of the graph. While some $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND}\}$-labelling semantics might be used to address this issue (if such labelling semantics are accepted), the focus in the paper is on the combination of complete $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labelling and bivalent $\{\mathsf{y},\mathsf{n}\}$-labelling semantics. Yet, the use of $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labellings can be also justified from a wider perspective provided by probabilistic argumentation, as put forward next.

3.4. Probabilistic argumentation setting

Due to a variety of interests, the combination of argumentation and probability theory is approached in the literature in various ways, see e.g. (Hunter and Thimm, 2017; Riveret et al., 2018; Verheij, 2017a). For our purposes, we employ the framework of probabilistic labellings as set forth in (Riveret et al., 2018), in particular because it relies on explicit probability spaces which facilitate formal reasoning on properties of argument and statement labellings, without any particular assumptions on probabilistic independence.

In the approach of probabilistic labellings (Riveret et al., 2018), a sample space is a set of specific labellings of an argumentation graph. At this stage of the investigation, we do not have to commit to a particular sample space, but as an illustration and to prepare forthcoming discussions on these matters, we consider here the variant where the set of complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings of any argumentation graph is the sample space of so-called probabilistic labelling frames.

Definition 3.25. A **probabilistic complete** $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$**-labelling frame** based on an argumentation graph $\mathcal{G}$ is a tuple $\langle \mathcal{G}, \langle \Omega, F, P\rangle\rangle$ where $\langle \Omega, F, P\rangle$ is a probability space such that:

- the sample space Ω is the set of complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings of $\mathcal{G}$;
- the σ-algebra F is the power set of Ω, i.e. $F = \mathrm{pow}(\Omega)$;
- the function P from F to $[0, 1]$ is a probability distribution satisfying Kolmogorov axioms.

Example 3.26. Figure 7 illustrates a probabilistic complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling frame of the argumentation graph from Figure 1 with an arbitrary probability distribution (labellings with probability zero are not displayed).

A	B	C1	D	C2	P
IN	IN	IN	OFF	OFF	2/15
IN	IN	OFF	OFF	OFF	5/15
IN	OFF	OFF	IN	OUT	1/15
IN	OFF	OFF	IN	OFF	2/15
OFF	IN	OFF	OFF	OFF	3/15
OFF	OFF	OFF	IN	IN	1/15
OFF	OFF	OFF	IN	OFF	1/15

Figure 7.: Complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$**-labellings and arbitrary probability values.**

□

When a probabilistic argumentation frame is employed, we do not have to commit to a particular interpretation of probability values (such as classical, frequentist or Bayesian views on these matters). Nevertheless, we can adopt a frequentist interpretation, and thus the probability distribution can be understood here as the probabilistic account of a collection of labellings. To facilitate the discussion, let us use the following notation for the multiplicity of an element in a collection.

Notation 7. The number of occurrences of an element A in a collection $\mathcal{A}$ is denoted $m_{\mathcal{A}}(A)$.

Given a non-empty collection of $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings $\mathcal{L}$, any labelling can be thus associated with a probability value $P_{\mathcal{L}}(\{L\})$ such that:

$$P_{\mathcal{L}}(\{L\}) = \frac{m_{\mathcal{L}}(L)}{|\mathcal{L}|}. \tag{1}$$

We can also devise a random variable $L_{\mathcal{A}}$ to capture the probability $P_{\mathcal{L}}(L_{\mathcal{A}} = l)$ over a collection of $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings $\mathcal{L}$ that every argument in a set of arguments $\mathcal{A}$ is labelled $l \in \{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$:

$$P_{\mathcal{L}}(L_{\mathcal{A}} = l) = \sum_{L \in \mathcal{L} : \mathcal{A} \subseteq l(L)} P_{\mathcal{L}}(\{L\}). \tag{2}$$

On this basis, we may draw ‘probabilistic argumentation graphs’ where every argument is presented with its marginal probability of being labelled l, without having to assume any independence assumption, cf. (Hunter, 2013; Li, Oren, and Norman, 2012) along with critical analyses e.g. (Bistarelli and Mantadelis, 2019; Riveret et al., 2018). Such probabilistic argumentation graphs provide an appealing graphical synthetic overview of probabilistic labelling frames (see Example 3.27 for an illustration). In the remainder of the paper, we provide probabilistic argumentation graphs where only the marginal probability of being labelled IN is indicated.

Example 3.27. A probabilistic argumentation graph corresponding to the probabilistic argumentation frame featured in Figure 7 is displayed in Figure 8, where every argument is associated with its marginal probability of being labelled IN (values are rounded off).

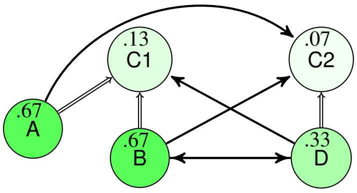


Figure 8.: A probabilistic argumentation graph.

□

Probabilistic argumentation graphs are not essential to the very purposes of this paper. However, such a probabilistic argumentation graph can provide a nice view on a dataset in terms of marginal probabilities attached to arguments of the graph meant to account for the dataset.

Furthermore, such graphs present connections between the use of probabilistic labellings in the present undertaking and other works in probabilistic argumentation, see (Riveret et al., 2018). In particular, they show an alternative to ‘probabilistic argumentation frameworks’ (Hunter, 2013; Li et al., 2012) where subargument support relationships are not accounted for, or follow-ups on probabilistic bipolar argumentation frameworks (Fazzinga, Flesca, and Furfaro, 2018; Polberg and Hunter, 2018) featuring other types of supports. We can also note that early works on probabilistic argumentation frameworks (Hunter, 2013; Li et al., 2012) looked at scenarios where arguments are assumed probabilistically independent, and follow-ups relaxed such an assumption (Fazzinga, Flesca, and Furfaro, 2016, 2019), cf. (Dung and Thang, 2010; Régis, Rotolo, and Sartor, 2012; Riveret et al., 2015a). The work reported here using probabilistic labellings does not rely on the assumption of arguments which are probabilistically independent. Probabilistic argumentation graphs also belong to a preceding thread of work on probabilistic argumentation where, for example, arguments/statements are associated with probability values, and the probability of the acceptance of arguments/statements is sought or have to be sought from the probability of expressed arguments or premises which are played in favour or against the acceptance of the arguments/statements, see e.g. (Riveret, Rotolo, Sartor, Prakken, and Roth, 2007; Riveret, Prakken, Rotolo, and Sartor, 2008). In contrast, the present undertaking directly learns such probabilities from considered datasets.

To sum up the section, given a well-formed semi-abstract argumentation graph, arguments are assumed to be labelled following specific $\{$IN, OUT, UND, OFF$\}$-labellings, and from it, statements are labelled following the bivalent $\{$y, n$\}$-labelling semantics. Given an argumentation graph, a set of specific $\{$IN, OUT, UND, OFF$\}$-labellings can be the sample space of probabilistic argumentation frames, and probabilistic argumentation graphs can be drawn in order to get a synthetic overview of probabilistic argumentation frames.

4. Defining the Problem

In this section, the problem of 'finding explanatory synthetic accounts of collections of cases' as previously introduced is more formally specified.

4.1. Problem input

Referring to some conventions in machine learning, we may distinguish 'attribute-based' and 'feature-based' examples (Fürnkranz et al., 2014). An attribute-based example is a set of attribute-value pairs, where the set of values of an attribute can be finite or infinite and possibly uncountable (as in case of real values). A feature-based example is a set of pairs feature-value where a feature is Boolean-valued and describes the presence or absence of certain properties of an attribute-based example. Typically, a feature-based example is built from an attribute-based example.

To fit our argumentation setting, features are here called statements, and examples are cases. Hence, attribute-based cases correspond to attribute-based examples, and feature-based cases correspond to feature-based examples (every statement corresponds to a feature). As the construction of features is often a problem of its own, we directly consider featured-based cases, that are simply called cases in the remainder. For our purposes, a case is thus a bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling, and a collection of cases is a dataset.

Definition 4.1. A **case** is a bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling (of a set of statements).

Definition 4.2. A **dataset** is a collection of cases (of a set of statements).

Henceforth, we assume that cases are finite cases over a non-empty set of statements, and datasets are finite non-empty datasets, such that all the cases in a dataset are bivalent $\{\mathsf{y}, \mathsf{n}\}$-labellings over the same set of statements.

Notation 8. The set of statements labelled in a dataset $\mathcal{K}$ is denoted $\Phi_{\mathcal{K}}$.

Given a dataset $\mathcal{K}$, any statement may be considered as either a 'premise' or a 'target'; leading to the definition of $\{\mathsf{premise}, \mathsf{target}\}$-labellings of a dataset. For our explanatory ends, the target is an 'explanandum' while the premises are the explanans. In other words, a target is a statement to be explained by using statements from a considered set of premises.

Definition 4.3. A $\{\mathsf{premise}, \mathsf{target}\}$-labelling of a dataset $\mathcal{K}$ is a partial function $\pi : \Phi_{\mathcal{K}} \to \{\mathsf{premise}, \mathsf{target}\}$.

Notation 9. A $\{\mathsf{premise}, \mathsf{target}\}$-labelling π may be expressed as a tuple $\langle \mathsf{premise}(\pi), \mathsf{target}(\pi) \rangle$.

Such $\{\mathsf{premise}, \mathsf{target}\}$-labellings are used later to constrain search spaces by assuming that premises are conclusions of assumptive arguments possibly supporting other arguments.

A dataset along with a $\{\mathsf{premise}, \mathsf{target}\}$-labelling may have cases where premises have equal acceptance statuses and targets are differently accepted. In this paper, such a dataset is said to be non-deterministic (or not deterministic).

Definition 4.4. Given a dataset $\mathcal{K}$ and a $\{\mathsf{premise}, \mathsf{target}\}$-labelling π of $\mathcal{K}$, the dataset $\mathcal{K}$ is **non-deterministic wrt** π iff there exist two cases K_1 and K_2 in $\mathcal{K}$ such that for every statement $\beta \in \mathsf{premise}(\pi)$, $K_1(\beta) = K_2(\beta)$, and there exists a statement $\phi \in \mathsf{target}(\pi)$ such that $K_1(\phi) \neq K_2(\phi)$.

Example 4.5. Any dataset including all the bivalent labellings in Figure 5 is non-deterministic wrt the $\{\mathsf{premise}, \mathsf{target}\}$-labelling $\langle\{\mathsf{a}, \mathsf{b}, \mathsf{d}\}, \{\mathsf{c}\}\rangle$. □

As to the terminology, in the literature, when a non-deterministic dataset has inconsistencies amongst its cases, the dataset is usually said to be inconsistent. However, the term 'inconsistent' can be questioned to characterise, for example, outcomes of stochastic systems (or non-deterministic systems in the physical sense) where the same conditions may naturally result into different outcomes. For instance, the outcomes of repeatedly tossing a coin are not naturally qualified as inconsistent. Since the labellings adopted here from probabilistic argumentation can be used to argue about some non-deterministic systems, we may prefer to say, in this context, that a dataset is non-deterministic rather than inconsistent.

Besides a dataset, a language graph can be another input of the problem. Often, a language graph may be induced from the given dataset, but the graph may be directly supplied as an input, even if it is incomplete in some sense. Such input language graphs can include some contrary or contradictory relations amongst statements which can be formally asserted as background or prior knowledge without having to induce them from datasets. For example, features such as $\mathsf{age} < 60$ and $\mathsf{age} \geq 60$ can be trivially asserted as contradictory, and it would be silly to ignore such pieces of information. Hence, in this paper, any input includes a language graph which is consistent with the given dataset.

Eventually, some prior knowledge can be also asserted as input through a (possibly empty) set of semi-abstract argumentation graphs, where every graph is consistent with the input dataset and language graph. One may see an overlap between the input language graph and the input set of argumentation graphs, however a graph may indicate attacks or supports between arguments which are not captured in the language graph.

To recap, cases are formalised as bivalent $\{\mathsf{y}, \mathsf{n}\}$-labellings, and any dataset is a collection of cases. Our problem input includes a dataset over a finite set of statements pertaining to some language, a language graph consistent with the dataset, and a set of background argumentation graphs consistent with the dataset and the language graph. The problem is defined next.

4.2. Problem definition

The paper addresses the problem of finding explanatory synthetic accounts of collections of cases, where such accounts are argumentation graphs consistent with the cases and any case is a statement labelling. To begin with, it is easy to see that for any collection of bivalent $\{\mathsf{y}, \mathsf{n}\}$-labellings, there exists a trivial argumentation graph which is consistent with the collection.

Definition 4.6. An argumentation graph $\mathcal{G}$ is a **trivial argumentation graph** of a dataset $\mathcal{K}$ iff $\mathcal{G}$ is induced by $\Phi_{\mathcal{K}}$ and $\rightsquigarrow_{\mathcal{G}} = \emptyset$ and $\Mapsto_{\mathcal{G}} = \emptyset$.

Example 4.7. A trivial argumentation graph of the dataset in Figure 5 is drawn in Figure 9.

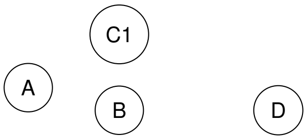


Figure 9.: A trivial argumentation graph of the dataset given in Figure 5.

□

Theorem 4.8. *For any dataset $\mathcal{K}$, any trivial argumentation graph of $\mathcal{K}$ is consistent with $\mathcal{K}$.*

Proof. For every case K in $\mathcal{K}$, let L be the complete $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}, \textsf{OFF}\}$-labelling of $\mathcal{G}$ such that $\textsf{IN}(L) = \{A \mid A \in \mathcal{A}_{\mathcal{G}}, \mathrm{con}(A) = \phi, \phi \in \mathsf{y}(K)\}$, $\textsf{OUT}(L) = \emptyset$, $\textsf{UND}(L) = \emptyset$ and $\textsf{OFF}(L) = \{A \mid A \in \mathcal{A}_{\mathcal{G}}, \mathrm{con}(A) = \phi, \phi \in \mathsf{n}(K)\}$. By Definition 3.22, labellings L and K are consistent. Therefore, argumentation graph $\mathcal{G}$ and dataset $\mathcal{K}$ are consistent. □

Given any dataset $\mathcal{K}$, any trivial argumentation graph of $\mathcal{K}$ is consistent with $\mathcal{K}$. Therefore, there exists an argumentation graph consistent with any given dataset.

Theorem 4.9. *For any dataset $\mathcal{K}$, there exists an argumentation graph consistent with $\mathcal{K}$.*

Proof. For any dataset $\mathcal{K}$, there exists a trivial argumentation graph $\mathcal{G}$ of $\mathcal{K}$. By Theorem 4.8, $\mathcal{G}$ is consistent with $\mathcal{K}$. Therefore, for any dataset $\mathcal{K}$, there exists an argumentation graph consistent with $\mathcal{K}$. □

Actually, there exist an infinity of argumentation graphs consistent with any given dataset $\mathcal{K}$, because any trivial argumentation graph of $\mathcal{K}$ is (trivially) consistent with $\mathcal{K}$, and there exist an infinity of trivial argumentation graphs of $\mathcal{K}$ which differ in the number of arguments (and argument identifiers), see also alternative graphs in e.g. (Dyrkolbotn, 2014; Linsbichler et al., 2016). However, such graphs may not be the most interesting graphs in terms of explanations. For this reason, we are after more 'interesting' graphs whose consistency is not trivial. Interestingness is a quite relative property, and varied qualities can be conceived to characterise interesting graphs. We identify several qualities in the next sections, leaving 'interestingness' as an abstract property for now in the problem definition.

As an infinity of graphs are trivially consistent with any dataset, we may reconsider Epicurus' Principle of Multiple Explanations (according to which if several explanations are consistent with the observed data then we should retain them all) through a weaker version of the principle by narrowing the investigation to a finite set of interesting argumentation graphs. Eventually, the set of interesting graphs may be ordered, leading thus to the problem of finding a most interesting order of argumentation graphs to explain cases.

Given:	a dataset $\mathcal{K}$, and a language graph $\mathfrak{L}$ consistent with $\mathcal{K}$, and a set of argumentation graphs consistent with $\mathcal{K}$ and $\mathfrak{L}$, and a finite computational budget,
find:	a most interesting order of argumentation graphs consistent with $\mathcal{K}$ and $\mathfrak{L}$.

The problem definition may appear unsuitable for noisy datasets where, for example, erroneous cases are present. For some noisy datasets, it may not be recommended to search for argumentation graphs which are consistent with the input dataset and language graph, essentially because the argumentation graphs could also be used to explain erroneous cases as if they were correct (as in overfitting, i.e. when explanatory graphs fits the data too well). To overcome this issue, the requirements on consistency can be relaxed and supplanted by some various evaluation criteria, such as a significance score above a requested threshold. Nevertheless, some types of noise can be adequately addressed by the framework (as we will see in Section 9), and variants of the problem definition to deal with more types of noise is left to future work.

By yielding a most interesting order of argumentation graphs, an end-user can make further (discretionary) tradeoffs within another set of criteria to elicit a graph. For example, the returned order of argumentation graphs can be further ordered with other criteria and one can select a particular graph, presumably a maximal graph in the order, and then induce a theory or a set of rules from it. As the problem of eliciting a particular argumentation graph and the problem of constructing a theory from an argumentation graph are different from the problem addressed in this paper, they are left to future research.

To get the argumentation graphs we are looking for, we can use a brute force approach. Let $\mathcal{C}$ denote any complete argumentation graph induced by the set of statements Φ labelled in the dataset, i.e. a graph where all the arguments attack and support each other, and such that every statement in Φ is the conclusion of at least one argument in $\mathcal{A}_{\mathcal{C}}$. Any interesting argumentation graph is necessarily a well-formed attack-support subgraph of a complete argumentation graph $\mathcal{C}$ induced by Φ. Thus, a brute force approach for our problem is to start from a complete argumentation graph $\mathcal{C}$ induced by the set of observed statements, and then assess every subgraph of $\mathcal{C}$ along with its bivalent $\{\mathsf{y}, \mathsf{n}\}$-labellings to see which well-formed attack-support subgraphs are interesting. The procedure can be then repeated with different complete graphs (by varying the number of arguments). However this brute force approach is of course not efficient, and thus alternatives are called for. In this paper, the problem is viewed as a search problem, and thus we will investigate a search algorithm with particular attention to the search space and ways to reduce it.

5. Maxconsistent Labellings

Ideally, a necessary condition for an argumentation graph to be part of any acceptable explanation is that the graph is consistent with the considered cases. For this reason, we need a way to evaluate consistency. In this section, the evaluation of argumentation graphs consistency is initiated through particular labellings making graphs consistent ‘as much as possible’ with the given cases.

5.1. Maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$*-labellings*

Given an argumentation graph and a case, there may exist no complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings consistent with the case, and thus we may instead attempt to label the graph to make it consistent with the case as much as possible. To do so, the focus in this paper is on so-called ‘maxconsistent’ $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings. Such (novel) labellings are obtained by labelling IN every argument whose conclusion is in $\mathsf{y}(K)$, as long as all its subarguments are labelled IN. Accordingly, we can characterise the arguments labelled IN in a $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling which is maxconsistent with a case by means of the fixed point of a ‘maxconsistent characteristic function’ as defined below.

Definition 5.1. Let K be a case and $\mathcal{G}$ an argumentation graph. The **maxconsistent characteristic function** of $\mathcal{G}$ wrt K is a function $F_{\mathcal{G},K} : \mathrm{pow}(\mathcal{A}_{\mathcal{G}}) \rightarrow \mathrm{pow}(\mathcal{A}_{\mathcal{G}})$ such that $F_{\mathcal{G},K}(\mathcal{A}) = \{A \mid A \in \mathcal{A}_{\mathcal{G}}$, and $\mathrm{con}(A) \in \mathsf{y}(K)$, and $\forall B \in \mathcal{A}_{\mathcal{G}}$: if $B \Mapsto_{\mathcal{G}} A$, then $B \in \mathcal{A}\}$.

Example 5.2. Let G denote the graph drawn in Figure 10, and K the case $\langle\{\mathsf{a}, \mathsf{b}, \mathsf{c}\}, \{\mathsf{d}\}\rangle$. It holds that $F_{\mathsf{G},\mathsf{K}}(\emptyset) = \{\mathsf{A}, \mathsf{B}\}$ and $F^2_{\mathsf{G},\mathsf{K}}(\emptyset) = \{\mathsf{A}, \mathsf{B}, \mathsf{C1}\}$, and finally $F^3_{\mathsf{G},\mathsf{K}}(\emptyset) = F^2_{\mathsf{G},\mathsf{K}}(\emptyset)$. Thus the function $F_{\mathsf{G},\mathsf{K}}$ has a fixed point which is the set of arguments $\{\mathsf{A}, \mathsf{B}, \mathsf{C1}\}$.

The characteristic function $F_{\mathcal{G},K}$ is monotonic, and if an argument is included in $F^i_{\mathcal{G},K}(\emptyset)$, then it is also included in $F^j_{\mathcal{G},K}(\emptyset)$ with $i \leq j$; consequently there exists a unique fixed point $\mathcal{A}^* = F^i_{\mathcal{G},K}(\emptyset)$.

Lemma 5.3 (Existence)**.** *For any case K and any argumentation graph $\mathcal{G}$, there exists a fixed*

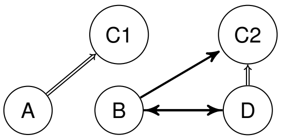


Figure 10.

□

point $\mathcal{A}^* = F^i_{\mathcal{G},K}(\emptyset)$ ($0 \leq i$) of the maxconsistent characteristic function of $\mathcal{G}$ wrt K.

Lemma 5.4 (Uniqueness)**.** *For any case K and any argumentation graph $\mathcal{G}$, the fixed point $\mathcal{A}^* = F^i_{\mathcal{G},K}(\emptyset)$ ($0 \leq i$) of the maxconsistent characteristic function of $\mathcal{G}$ wrt K is unique.*

Definition 5.5. Let K be a case, $\mathcal{G}$ an argumentation graph, and $\mathcal{A}^* = F^i_{\mathcal{G},K}(\emptyset)$ $(0 \leq i)$ the fixed point of the maxconsistent characteristic function of $\mathcal{G}$ wrt K. A $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling L of $\mathcal{G}$ is **maxconsistent** with K (or K-maxconsistent) iff $\mathsf{IN}(L) = \mathcal{A}^*$.

Example 5.6. The labellings $\langle\{\mathsf{A}\}, \{\mathsf{C2}\}, \{\mathsf{B}, \mathsf{D}, \mathsf{C1}\}, \emptyset\rangle$ and $\langle\{\mathsf{A}\}, \emptyset, \emptyset, \{\mathsf{B}, \mathsf{D}, \mathsf{C1}, \mathsf{C2}\}\rangle$ of the argumentation graph in Figure 1 are both maxconsistent with the case $\langle\{\mathsf{a}, \mathsf{c}\}, \{\mathsf{b}, \mathsf{d}\}\rangle$. □

Given an argumentation graph $\mathcal{G}$ and a case K, a K-maxconsistent $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling of $\mathcal{G}$ determines the arguments labelled IN, and missing pieces of information regard the labelling of arguments not labelled IN. For any argument which is not labelled IN, the argument is either labelled OUT or UND or OFF. To address this uncertainty, and continuing our attention paid to complete labellings, we can retain complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings which are maxconsistent with a case. Following our nomenclature, given a case K, an argumentation graph $\mathcal{G}$, and the fixed point $\mathcal{A}^* = F^i_{\mathcal{G},K}(\emptyset)$ $(0 \leq i)$ of the maxconsistent characteristic function of $\mathcal{G}$ wrt K, we say that a $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling L of $\mathcal{G}$ is a K-maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling if, and only if, $\mathsf{IN}(L) = \mathcal{A}^*$ and L is a complete labelling of $\mathcal{G}$.

Given a case K, an argumentation graph may have no K-maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings or even no K-maxconsistent conflict-free $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings (in the sense that there may exist attacks amongst arguments labelled IN). For example, if the argumentation graph is strictly induced by the statements labelled in the case, and two of its arguments attack each other while their conclusions are labelled y in case K, then no K-maxconsistent $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling of $\mathcal{G}$ is complete. Hence, a case K may not be explainable by a particular graph and its K-maxconsistent $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings, and such a graph may be thus discarded to explain the case.

To further specify how arguments are labelled, we may assess a 'minomit' complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling maxconsistent with the case at hand, i.e. a labelling minimising omitted arguments, but it is equally possible to consider a 'maxomit' complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling, i.e. a labelling maximising omitted arguments.

Definition 5.7. Let $\mathcal{G}$ be an argumentation graph and $\mathcal{L}$ a set of $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings of $\mathcal{G}$.

- A $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling L is a **minomit $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling** of $\mathcal{G}$ wrt $\mathcal{L}$ iff $\mathsf{OFF}(L)$ is minimal amongst all $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings in $\mathcal{L}$.
- A $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling L is a **maxomit $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling** of $\mathcal{G}$ wrt $\mathcal{L}$ iff $\mathsf{OFF}(L)$ is maximal amongst all complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings of $\mathcal{G}$ in $\mathcal{L}$.

In the remainder, instead of referring to minomit or maxomit $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings (of an argumentation graph) wrt a set of K-maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings (of the graph), we may simply refer to minomit or maxomit K-maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings (of the graph). Similar shortcuts apply to other types of minomit or maxomit $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings. Hence, for example, given a case K, amongst the K-maxconsistent $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings, we may retain K-maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings. And amongst K-maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings, we can keep minomit or maxomit $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings to yield minomit or maxomit K-maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings, as illustrated in Example 5.8.

Example 5.8. Figure 11 shows a maxomit complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling which is maxconsistent with the case $\langle\{\mathsf{a}, \mathsf{c}\}, \{\mathsf{b}, \mathsf{d}\}\rangle$; the minomit counterpart is in Figure 12.

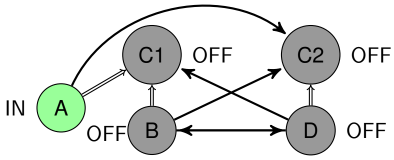


Figure 11.: Maxomit.

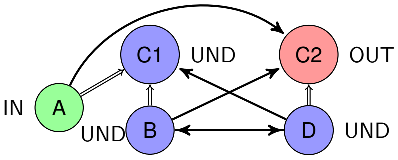


Figure 12.: Minomit.

□

Arguably, minomit K-maxconsistent complete labellings have more explanatory potential then maxomit counterparts where arguments not labelled IN are simply labelled OFF. Nevertheless, it is often convenient to work with maxomit complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings for computational purposes, because they can be straightforwardly conceived and efficiently computed. This holds especially in the search of interesting graphs, but once interesting graphs are found then minomit labellings can be used for explanatory ends. Furthermore, if there exists a maxomit complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling maxconsistent with a case then it is unique.

Theorem 5.9 (Uniqueness)**.** *For any case K and argumentation graph $\mathcal{G}$, if there exists a maxomit K-maxconsistent complete* $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$*-labelling L of $\mathcal{G}$, then L is unique.*

Proof. By assuming there exist two entities that both satisfy the condition, and logically deducing their equality. Let L_1 and L_2 denote two maxomit K-maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings of $\mathcal{G}$. The labellings L_1 and L_2 are K-maxconsistent, and thus $\mathsf{IN}(L_1) = \mathsf{IN}(L_2)$. The labellings L_1 and L_2 are maxomit, thus, by Definition 5.7, $\mathsf{OFF}(L_1)$ and $\mathsf{OFF}(L_2)$ are maximal amongst all K-maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings of $\mathcal{G}$. Consequently, $\mathsf{OUT}(L_1) = \emptyset$ and $\mathsf{OUT}(L_2) = \emptyset$ respectively, and therefore $\mathsf{OUT}(L_1) = \mathsf{OUT}(L_2)$; similarly $\mathsf{UND}(L_1) = \mathsf{UND}(L_2)$. As $\mathsf{OFF}(L_1) = \mathcal{A}_{\mathcal{G}} \backslash (\mathsf{IN}(L_1) \cup \mathsf{OUT}(L_1) \cup \mathsf{UND}(L_1))$ and $\mathsf{OFF}(L_2) = \mathcal{A}_{\mathcal{G}} \backslash (\mathsf{IN}(L_1) \cup \mathsf{OUT}(L_2) \cup \mathsf{UND}(L_2))$, we have $\mathsf{OFF}(L_1) = \mathsf{OFF}(L_2)$. Since $\mathsf{IN}(L_1) = \mathsf{IN}(L_2)$, $\mathsf{OUT}(L_1) = \mathsf{OUT}(L_2)$, $\mathsf{UND}(L_1) = \mathsf{UND}(L_2)$ and $\mathsf{OFF}(L_1) = \mathsf{OFF}(L_2)$, we have $L_1 = L_2$. Therefore, if there exists a maxomit K-maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling L of $\mathcal{G}$, then L is unique. □

A complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling which is maxconsistent with a case may not be consistent with the case (see Example 5.8) and thus their relative inconsistency can be evaluated. In that regard, we devise later a 'consistency score' function associating any graph and case with a score value, and we will exploit this score in some heuristics of a search for interesting explanatory graphs.

5.2. Maxconsistent bivalent $\{\mathsf{y},\mathsf{n}\}$*-labellings*

On the basis of maxconsistent $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labellings, we can then define acceptance maxconsistent bivalent $\{\mathsf{y},\mathsf{n}\}$-labellings.

Definition 5.10. An acceptance bivalent $\{\mathsf{y},\mathsf{n}\}$-labelling K' of an argumentation graph $\mathcal{G}$ is **maxconsistent** with a case K (or K-maxconsistent) iff K' is the bivalent $\{\mathsf{y},\mathsf{n}\}$-labelling from a singleton $\{L\}$ such that L is a K-maxconsistent $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labelling of $\mathcal{G}$.

We can note that, as a slight alternative to the above definition, the y-labelled statements may be directly defined wrt the fixed point of the maxconsistent characteristic function of $\mathcal{G}$ wrt K. And thus a K-maxconsistent bivalent $\{\mathsf{y},\mathsf{n}\}$-labelling could be equally defined wrt the fixed point.

Following our nomenclature, a *complete* bivalent $\{\mathsf{y},\mathsf{n}\}$-labelling K' of an argumentation graph $\mathcal{G}$ is K-maxconsistent if, and only if, K' is the bivalent $\{\mathsf{y},\mathsf{n}\}$-labelling from a *complete* $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labelling $\{L\}$ of $\mathcal{G}$ such that L is K-maxconsistent. Given a case K, there may exist no K-maxconsistent complete bivalent $\{\mathsf{y},\mathsf{n}\}$-labellings of an argumentation graph $\mathcal{G}$, because there may exist no K-maxconsistent complete $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labelling of $\mathcal{G}$. However, if there exists a K-maxconsistent complete $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labelling of $\mathcal{G}$, then there exists a K-maxconsistent complete bivalent $\{\mathsf{y},\mathsf{n}\}$-labelling of $\mathcal{G}$.

Proposition 1 (Existence)**.** *For any case K and argumentation graph $\mathcal{G}$, if there exists a K-maxconsistent complete* $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$*-labelling of $\mathcal{G}$, then there exists a K-maxconsistent complete bivalent* $\{\mathsf{y},\mathsf{n}\}$*-labelling of $\mathcal{G}$.*

Proposition 2 (Uniqueness)**.** *For any case K and argumentation graph $\mathcal{G}$, if there exists a K-maxconsistent complete* $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$*-labelling of $\mathcal{G}$, then the K-maxconsistent complete bivalent* $\{\mathsf{y},\mathsf{n}\}$*-labelling of $\mathcal{G}$ is unique.*

Proof. By Lemma 5.4, the fixed point $\mathcal{A}^* = F^i_{\mathcal{G},K}(\emptyset)$ $(0 \leq i)$ of the maxconsistent characteristic function of $\mathcal{G}$ wrt K is unique. Any K-maxconsistent complete $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labelling L is such that $\mathsf{IN}(L) = \mathcal{A}^*$. Therefore, there exists a unique K-maxconsistent complete bivalent $\{\mathsf{y},\mathsf{n}\}$-labelling of $\mathcal{G}$. □

Example 5.11. The bivalent $\{\mathsf{y},\mathsf{n}\}$-labelling $\langle\{\mathsf{a}\},\{\mathsf{b},\mathsf{c},\mathsf{d}\}\rangle$ of the argumentation graph in Figure 1 is the unique $\{\mathsf{y},\mathsf{n}\}$-labelling which is maxconsistent with the case $\langle\{\mathsf{a},\mathsf{c}\},\{\mathsf{b},\mathsf{d}\}\rangle$.□

5.3. Probabilistic argumentation setting

Let us introduce some terminology to indicate those argumentation graphs and complete $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labellings which are maxconsistent with a dataset rather than a case.

Definition 5.12.

- An **argumentation graph $\mathcal{G}$ is maxconsistent with a dataset $\mathcal{K}$** (or $\mathcal{K}$-maxconsistent) iff for every case K in $\mathcal{K}$, there exists a K-maxconsistent complete $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labelling of $\mathcal{G}$.
- Given a $\mathcal{K}$-maxconsistent argumentation graph $\mathcal{G}$, the **$\mathcal{K}$-maxconsistent set $\mathcal{L}$ of (complete resp.)** $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$**-labellings of $\mathcal{G}$** is the union of the sets of (complete resp.) $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labellings maxconsistent with any case in $\mathcal{K}$, i.e.

  $$\mathcal{L} = \textstyle\bigcup_{K\in\mathcal{K}} \mathcal{L}_K$$

  where $\mathcal{L}_K$ is the set of K-maxconsistent (complete resp.) $\{\mathsf{IN},\mathsf{OUT},\mathsf{UND},\mathsf{OFF}\}$-labellings of

$\mathcal{G}$.

We can now posit (novel) probabilistic complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling frames wrt any given dataset.

Definition 5.13. Let $\mathcal{K}$ be a dataset, and $\mathcal{G}$ a $\mathcal{K}$-maxconsistent argumentation graph. A **probabilistic complete** $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$**-labelling frame** based on argumentation graph $\mathcal{G}$ and **wrt dataset** $\mathcal{K}$ is a tuple $\langle \mathcal{K}, \mathcal{G}, \langle \Omega, F, P\rangle\rangle$ where $\langle \Omega, F, P\rangle$ is a probability space such that:

- the sample space Ω is the $\mathcal{K}$-maxconsistent set of complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings of $\mathcal{G}$;
- the σ-algebra F is the power set of Ω, i.e. $F = \mathrm{pow}(\Omega)$;
- the function P from F to $[0, 1]$ is a probability distribution satisfying the Kolmogorov axioms.

Other types of probabilistic frames can be certainly proposed, they are left to future investigations. The point here is that $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}\}$-labellings or $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings and associated probabilistic frames as reviewed in Section 3 can be adapted to the present purposes, and novel problems in argumentation can imply the conception of novel argumentation labellings and probabilistic frames, as evidenced above.

To recap the section, K-maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings and K-maxconsistent $\{\mathsf{y}, \mathsf{n}\}$-labellings have been devised. Given a case K, if an argumentation graph has a maxomit K-maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling, then the labelling is unique, but it may not be consistent with the case. Eventually, we can note that a case can have multiple argumentation graphs having a K-maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling which is consistent with the case, and for this reason we may elicit and order argumentation graphs, as investigated next.

6. On (Ordering) Interesting Argumentation Graphs

Given a dataset, we want to find out an interesting order of argumentation graphs consistent with the dataset, and thus we have to define what is meant by an ‘interesting order’. As mentioned earlier, there is an infinity of argumentation graphs consistent with any given dataset, and for practical reasons, it may not be possible to return and present this infinity of graphs in a compact representation, and some graphs may not be very interesting. Hence, we can seek some criteria to elicit particular argumentation graphs.

First, we may be tempted to use common evaluation measures from machine learning. For example, accuracy, precision or recall may be considered. As we are in a probabilistic setting, we may actually prefer a logloss or some statistical distances. However, we are looking for argumentation graphs for *explanatory* purposes, not for predictive purposes. Thus, instead of common machine learning measures of relevance for evaluate predictive models, we are after criteria to elicit ‘interesting’ or ’good’ explanations.

The general question ‘What is a good explanation?’ can be quite elusive (Bechtel and Abrahamsen, 2005; Doran et al., 2017; Doshi-Velez and Kim, 2017; Freitas, 2014; Hempel and Oppenheim, 1948; Keil, 2005; Lipton, 2001, 2016; Lombrozol, 2006). Nevertheless that should not stop us to canvass criteria for our practical purposes, as long as these criteria can be discussed and lead to acceptable implementations. Forthcoming criteria of interestingness are proposed because they turned out, as evidenced later in Section 9, to provide decent results from experiments along with reasonable operational performances.

6.1. On interesting argumentation graphs

Before ordering interesting argumentation graphs, such interesting graphs must be characterised. To characterise them, various properties can be advanced and they can be diversely organised. In this paper, proposed properties are regrouped into the following criteria around the construction stages of our probabilistic setting: (i) 'super well-formedness' which regards the structural construction of argumentation graphs, (ii) 'parsimony' which includes labelling considerations, and (iii) 'frequency' which is concerned with probabilistic aspects.

6.1.1. Super well-formedness

Well-formedness of an argumentation graph does not exhaust possible structural constraints on arguments, attack and support relations. In particular, well-formedness does not cater for structural properties constitutive of explanations deemed to be good or interesting. In the following, some further possible structural properties are proposed to characterise interesting argumentation graphs, while keeping in mind their potential to reduce the search space.

Succinctness. First, we may prefer argumentation graphs such that every argument and its subarguments have distinct conclusions.

Definition 6.1. An argumentation graph $\mathcal{G}$ is **succinct** iff $\mathcal{G}$ is subargument-succinct and support-succinct, where

- $\mathcal{G}$ is **subargument-succinct** iff for every argument A in $\mathcal{A}_{\mathcal{G}}$ no subargument of A has a conclusion equal to the conclusion of A.
- $\mathcal{G}$ is **support-succinct** iff for every argument A in $\mathcal{A}_{\mathcal{G}}$ no distinct subarguments of A have an equal conclusion.

Succinctness can be related to various properties in structure argumentation, such as for example the minimality constraint in studies of deductive arguments (Besnard and Hunter, 2014). We also have to remark that there exist argumentation frameworks which do not consider succinctness or similar ideas. For example, in accrual of arguments, see e.g. (Lucero, Chesñevar, and Simari, 2009; Prakken, 2005), one can conceive that distinct arguments with the same conclusion accrue into a structure or argument which can appear more persuasive for its claim. Such ideas are not considered in the present work, and hence we will focus on so-called succinct argumentation graphs. Now, other properties related to succinctness could be advanced, however, succinctness as defined above will turn to be sufficient for our purposes.

Attack-subargument concordance. We may also discard argumentation graphs where there is an attack between an argument and any of its subargument. Such attacks are actually very interesting in some sense, but, in our context, they may overload end-users. Furthermore, for practical matters, attack-subargument concordance allows to greatly reduce the search space. For this reason, attack-subargument concordance is considered here.

Definition 6.2. An argumentation graph $\mathcal{G}$ is **attack-subargument concordant** iff

- its attack and subargument relations are disjoint, i.e. $\rightsquigarrow_{\mathcal{G}} \cap \Longmapsto_{\mathcal{G}} = \emptyset$, and
- its transposed attack relation and its subargument relation are disjoint, i.e. $\rightsquigarrow_{\mathcal{G}}^{-1} \cap \Longmapsto_{\mathcal{G}} = \emptyset$.

Example 6.3. The argumentation graphs in Figure 13 are not attack-subargument concordant.

As illustrated in Figure 13 (a), we can remark that if an argumentation graph is well-formed and there exists an attack between subarguments of an argument, then the graph is not attack-subargument concordant. Consequently, if a graph is well-formed, then attack-subargument concordance prohibits any argument whose subarguments attack each other, for example when

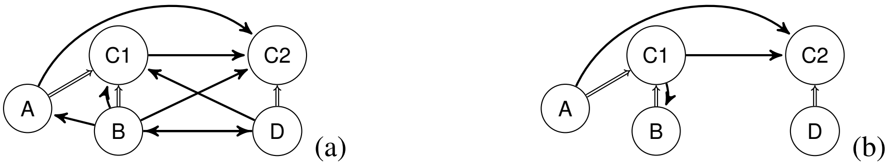


Figure 13.

□

subarguments support contradictory statements, cf. consistency constraints in studies of deductive arguments (Besnard and Hunter, 2014). Nevertheless, as illustrated in Figure 13 (b), an argumentation graph may not be attack-subargument concordant, and have no attacks between subarguments of any arguments.

To ease the discussion later, above-mentioned properties are encompassed under the definition of ‘super well-formed’ argumentation graphs.

Definition 6.4. A semi-abstract argumentation graph $\mathcal{G}$ is **super well-formed** iff $\mathcal{G}$ is

(1) well-formed, and
(2) succinct, and
(3) attack-subargument concordant.

These structural properties are endorsed together because they greatly reduce the search space. In particular, these properties can be easily ensured by checking the structure of any potentially interesting argumentation graph. They also turned out to allow an implementation of the proposed framework with decent results as evidenced later in the experimental evaluation. As alluded to, they can be related to various aspects in structured arguments, and in-depth studies of such relationships are left to future work. We may also note that some of these properties may not be viewed as necessary in diverse argumentation frameworks from the literature, but such frameworks have often little or no consideration for learning argumentation graphs and related implementations. In the remainder, we thus discard argumentation graphs which are not super well-formed, and retain those which are super well-formed.

6.1.2. Parsimony

Endorsing Ockham’s Principle of Parsimony according to which ‘Entities must not be multiplied beyond necessity’, we may define necessary arguments and ‘argument-parsimonious’ argumentation graphs consistent with a given dataset.

Definition 6.5. Let $\mathcal{G}$ be an argumentation graph consistent with a dataset $\mathcal{K}$. A set of arguments $\mathcal{A} \subseteq \mathcal{A}_{\mathcal{G}}$ is a **necessary set of arguments** wrt $\mathcal{K}$ iff the argument subgraph of $\mathcal{G}$ induced by $\mathcal{A}_{\mathcal{G}} \backslash \mathcal{A}$ is not consistent with $\mathcal{K}$.

Definition 6.6. An argumentation graph $\mathcal{G}$ consistent with a dataset $\mathcal{K}$ is an **argument-parsimonious argumentation graph** wrt $\mathcal{K}$ iff every non-empty set of arguments $\mathcal{A} \subseteq \mathcal{A}_{\mathcal{G}}$ is necessary wrt $\mathcal{K}$.

Example 6.7. Let $\mathcal{K}$ denote the dataset given in Figure 5. The argumentation graph in Figure 14 where $\mathrm{con}(\mathsf{C1}) = \mathrm{con}(\mathsf{C3})$ is not argument-parsimonious wrt $\mathcal{K}$, since either singleton $\{\mathsf{C1}\}$ or $\{\mathsf{C3}\}$ is unnecessary.

□

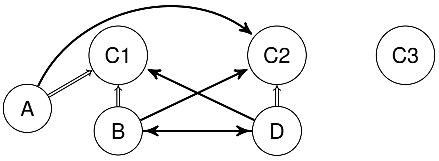


Figure 14.

Note that if an argumentation graph is strictly induced by the statements labelled in a given dataset, then the graph is argument-parsimonious wrt the dataset.

A set of arguments may be necessary, yet not very interesting. Like the simplification engendered from the disjunction elimination rule in classical logic, one may prefer simpler argumentation graphs than those graphs where a necessary pair of arguments is such that they both have an equal conclusion, and the conclusions of their supporting arguments differ by a pair of contradictory statements. Such a set of arguments is said to be unabridged.

Notation 10. Let $\mathcal{G}$ be a semi-abstract argumentation graph and A any argument in $\mathcal{A}$. We denote the set of conclusions of supporting arguments of A as $\mathrm{SubCon}(A)$, i.e. $\mathrm{SubCon}(A) = \{\phi \mid \phi = \mathrm{con}(B)$ and $B \Mapsto_{\mathcal{G}} A\}$.

Definition 6.8. Let $\mathcal{K}$ be a dataset, $\mathfrak{L}$ a language graph consistent with $\mathcal{K}$, and $\mathcal{G}$ an argumentation graph. A set of arguments $\{A_i, A_j\} \subseteq \mathcal{A}_{\mathcal{G}}$ is **unabridged** wrt $\mathcal{K}$ and $\mathfrak{L}$ iff

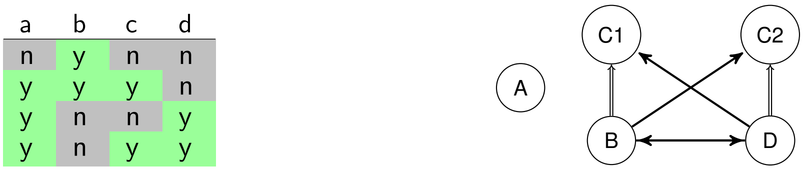
- the conclusions of arguments A_i and A_j are equal, i.e. $\mathrm{con}(A_i) = \mathrm{con}(A_j)$, and
- $\mathrm{SubCon}(A_i) = \Phi \cup \{\phi_i\}$, $\mathrm{SubCon}(A_j) = \Phi \cup \{\phi_j\}$ and $(\phi_i, \phi_j) \in contrad_{\mathfrak{L}}$, and
- the argument subgraph of $\mathcal{G}$ induced by $\mathcal{A}_{\mathcal{G}} \backslash \{A_i, A_j\}$ is not argument-parsimonious wrt $\mathcal{K}$.

Referring to arguments in Definition 6.8, for any case where the conclusion ϕ of A_i and A_j is labelled y and the statements in Φ are conclusions of arguments labelled IN, since at least one of the statements ϕ_i and ϕ_j is labelled y, then for any of such a case, either A_i or A_j can be labelled IN. In that regard, it would be simpler to work with one argument A whose conclusion is ϕ and the set of conclusions of their supporting arguments is Φ, i.e. $\mathrm{SubCon}(A) = \Phi$.

An argumentation graph without unabridged sets of arguments is said to be abridged.

Definition 6.9. Let $\mathcal{K}$ be a dataset, $\mathfrak{L}$ a language graph consistent with $\mathcal{K}$. An argumentation graph $\mathcal{G}$ is an **abridged argumentation graph** wrt $\mathcal{K}$ and $\mathfrak{L}$ iff there exist no unabridged sets of arguments in $\mathcal{A}_{\mathcal{G}}$ wrt $\mathcal{K}$ and $\mathfrak{L}$.

Example 6.10. Let us assume the argumentation graph and the dataset in Figure 15 where b and d are contradictory statements. The argumentation graph is more complex than the trivial argumentation graph of the dataset, yet its support relationships do not carry any sort of interesting information: the graph is not abridged. One may prefer a trivial argumentation graph which is simpler and does not carry less information.



a	b	c	d
n	y	n	n
y	y	y	n
y	n	n	y
y	n	y	y

Figure 15.: A dataset and an unabridged argumentation graph which is consistent with the dataset.

□

Eventually, an argumentation graph is said to be parsimonious if, and only if, it is argument-parsimonious and abridged.

Definition 6.11. Let $\mathcal{K}$ be a dataset, and $\mathfrak{L}$ a language graph consistent with $\mathcal{K}$. An argumentation graph $\mathcal{G}$ is a **parsimonious argumentation graph** wrt $\mathcal{K}$ and $\mathfrak{L}$ iff $\mathcal{G}$ is argument-parsimonious wrt $\mathcal{K}$ and abridged wrt $\mathcal{K}$ and $\mathfrak{L}$.

Hence, given a dataset and a language graph consistent with the dataset, we will look for argumentation graphs which are parsimonious wrt the dataset and the language graph.

6.1.3. Frequency

To further elicit argumentation graphs and reduce the search space, frequency values can be gauged, similarly as in the mining of association rules where the rules are evaluated in relation to their frequency (also called support) values (Agrawal, Imieliński, and Swami, 1993). The idea is that if an argument is not frequently labelled IN to cover a dataset then this argument is not frequently used to explain the cases, and thus it may be discarded.

Definition 6.12. Let $\mathcal{K} = (K_1, \ldots, K_N)$ be a dataset, and $\mathcal{L} = (L_1, \ldots, L_N)$ a collection of $\{\text{IN}, \text{OUT}, \text{UND}, \text{OFF}\}$-labellings of an argumentation graph such that every labelling $L_i \in \mathcal{L}$ is a K_i-maxconsistent complete $\{\text{IN}, \text{OUT}, \text{UND}, \text{OFF}\}$-labelling. The **frequency of an argument** A wrt $\mathcal{K}$, denoted $\text{freq}(A, \mathcal{K})$, is such that:

$$\text{freq}(A, \mathcal{K}) = P_{\mathcal{L}}(L_{\{A\}} = \text{IN}).$$

Thus, for any given dataset $\mathcal{K}$ and argument A, it holds that $\text{freq}(A, \mathcal{K}) \in [0, 1]$.

The frequency of an argumentation graph may be then defined on the basis of the frequency of its arguments. Different definitions are possible, preferably such that the graph frequency can be efficiently computed. For the sake of simplicity, we can adopt the idea that if an argument is discarded when it is not frequently used (i.e. not frequently labelled IN), then a graph may be discarded when any of its arguments has a frequency value which is deemed too low, i.e. below some frequency threshold. On this basis, the frequency of a graph may be defined as the minimal frequency of any of its arguments.

However, it is arguable that only the frequency values of non-assumptive arguments should be used to obtain the frequency of a graph. Let us suppose that the frequencies of assumptive and non-assumptive arguments are considered to compute the frequency of a graph: if some statements are rarely labelled y in a dataset and if we are looking for frequent argumentation graphs, then no graphs may be given because some assumptive arguments may not be sufficiently frequent. By considering only the frequencies of non-assumptive arguments, any statement which is rarely labelled y can be supported by an assumptive argument, whatever the frequency threshold.

Accordingly, the frequency of any graph is defined as the minimal frequency amongst all the frequency values of non-assumptive arguments.

Definition 6.13. Let $\mathcal{G}$ be an argumentation graph, and $\mathcal{A} \subseteq \mathcal{A}_{\mathcal{G}}$ its set of non-assumptive arguments. The **frequency** of $\mathcal{G}$ wrt a dataset $\mathcal{K}$, denoted $\mathrm{freq}(\mathcal{G}, \mathcal{K})$, is such that if $\mathcal{A} \neq \emptyset$ then

$$\mathrm{freq}(\mathcal{G}, \mathcal{K}) = \min_{A \in \mathcal{A}} \mathrm{freq}(A, \mathcal{K})$$

else $\mathrm{freq}(\mathcal{G}, \mathcal{K}) = 1$.

Hence, for any given dataset $\mathcal{K}$ and any argumentation graph $\mathcal{G}$, it holds that $\mathrm{freq}(\mathcal{G}, \mathcal{K}) \in [0, 1]$.

Example 6.14. Suppose the argumentation graphs and the dataset $\mathcal{K}$ in figures 16, 17, and 18.

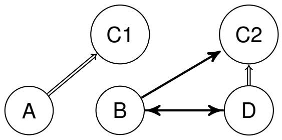


Figure 16.: G′.

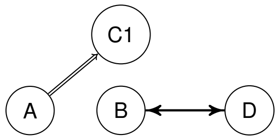


Figure 17.: G″.

a	b	c	d
y	n	n	n
y	y	y	n
y	n	n	y
n	n	n	n
n	y	n	n
n	n	y	y
n	n	n	y

G′	A	B	C1	D	C2
	IN	OFF	OFF	OFF	OFF
	IN	IN	IN	OFF	OFF
	IN	OFF	OFF	IN	OFF
	OFF	OFF	OFF	OFF	OFF
	OFF	IN	OFF	OFF	OFF
	OFF	OFF	OFF	IN	IN
	OFF	OFF	OFF	IN	OFF

G″	A	B	C1	D
	IN	OFF	OFF	OFF
	IN	IN	IN	OFF
	IN	OFF	OFF	IN
	OFF	OFF	OFF	OFF
	OFF	IN	OFF	OFF
	OFF	OFF	OFF	IN
	OFF	OFF	OFF	IN

Figure 18.: A dataset (on the left, where each row is a case) and corresponding maxomit complete {IN, OUT, UND, OFF}-labellings of the argumentation graphs drawn in figures 16 and 17 such that these labellings are maxconsistent with cases of the dataset.

For argumentation graphs G′ and G″, we have the following frequencies:

G′ arg.	A	B	D	C1	C2	G″ arg.	A	B	D	C1
$\mathrm{freq}(\cdot, \mathcal{K})$	$3/7$	$2/7$	$3/7$	$1/7$	$1/7$	$\mathrm{freq}(\cdot, \mathcal{K})$	$3/7$	$2/7$	$3/7$	$1/7$

Hence $\mathrm{freq}(\mathsf{G}', \mathcal{K}) = 1/7$ and $\mathrm{freq}(\mathsf{G}'', \mathcal{K}) = 1/7$. □

As in data mining where an association rule is discarded if its frequency (or support) is not above a given threshold, a non-assumptive argument and thus its argumentation graph can be discarded if the frequency is not above a given threshold; such a threshold is denoted $freq$ in the remainder. A graph whose frequency is above the threshold is said to be a frequent graph.

Definition 6.15. Let $freq$ denote a real number in $[0, 1]$. Given a dataset $\mathcal{K}$, an argumentation graph $\mathcal{G}$ is a **frequent argumentation graph** wrt $\mathcal{K}$ and a frequency $freq$ iff $\mathrm{freq}(\mathcal{G}, \mathcal{K}) \geq freq$.

For the sake of simplicity, instead of saying that an argumentation graph is frequent wrt a dataset $\mathcal{K}$ and a frequency $freq$, we may simply say that the graph is frequent wrt $\mathcal{K}$ without mentioning the frequency $freq$ which is left as a background parameter.

The use of frequency thresholds implies that if an argument is not frequently labelled IN then it is discarded. In particular, if a non-assumptive argument A has its set of subarguments included in the set of subarguments of another argument B, and arguments A and B are labelled IN when the subarguments are labelled IN, then the frequency of A is superior or equal to the frequency of B. For this reason, a frequency threshold over non-assumptive arguments can also be used to discard arguments whose subarguments are deemed too numerous. If arguments with multiple subarguments or rare argumentation patterns are looked for, then the frequency threshold can be lowered.

Whatever the frequency threshold, we may also acknowledge the pathological situation where an argument is supported by an assumptive subargument which has a frequency one. If a dataset features a statement which is labelled y in every case of the dataset, then any assumptive argument whose conclusion is this statement can be a subargument of any other argument, but such a support provides no informational value. To avoid such situations, we will discard argumentation graphs which are not *concise*, as defined next.

Definition 6.16. An argumentation graph $\mathcal{G}$ is a **concise argumentation graph** wrt a dataset $\mathcal{K}$ iff for every non-assumptive argument A in $\mathcal{A}_{\mathcal{G}}$, there exist no assumptive subargument B of A such that the frequency of B wrt $\mathcal{K}$ equals one (i.e. $\text{freq}(B, \mathcal{K}) = 1$).

Hence, given a dataset and a language graph consistent with the dataset, we will look for argumentation graphs which are frequent and concise wrt the dataset.

6.1.4. Interesting argumentation graphs

As previously mentioned, the investigation is narrowed to interesting argumentation graphs, where interestingness is defined relative to our ends. Firstly, super well-formedness can be put forward, so that, given a dataset, we will retain super well-formed argumentation graphs which are consistent with the dataset, and discard argumentation graphs which are not super well-formed. Furthermore, we can retain parsimonious super well-formed graphs which are frequent (wrt a frequency threshold) and concise. Such graphs are said here to be interesting.

Definition 6.17. Let $\mathcal{K}$ denote a dataset, $\mathfrak{L}$ a language graph consistent with $\mathcal{K}$, and $\mathcal{G}$ an argumentation graph consistent with $\mathcal{K}$ and $\mathfrak{L}$. The argumentation graph $\mathcal{G}$ is an **interesting argumentation graph** wrt $\mathcal{K}$ and $\mathfrak{L}$ iff

- $\mathcal{G}$ is super well-formed, and
- $\mathcal{G}$ is parsimonious wrt $\mathcal{K}$ and $\mathfrak{L}$, and
- $\mathcal{G}$ is frequent wrt $\mathcal{K}$ (and a frequency threshold parameter), and
- $\mathcal{G}$ is concise wrt $\mathcal{K}$.

Again, interestingness can be differently defined in any other investigations. The above-mentioned properties can be discussed, others can be proposed and they can be differently organised. For example, one may have found some omitted logic connections between the elicited properties, and thus some sort of axiomatic system of interestingness may be put forward. Yet the proposed properties are sufficient for the present purposes, and thus, given a dataset and a language graph consistent with the dataset, we are looking for argumentation graphs which are interesting (as defined above) wrt the dataset and the language graph.

6.2. On ordering interesting argumentation graphs

Once a set of interesting argumentation graphs is identified, we can order the graphs. To do this, we can firstly use a partial order through a refinement relation, and then fully order the most refined argumentation graphs through a confidence measure.

6.2.1. Refinement of argumentation graphs

Given two distinct argumentation graphs, if one graph is a subgraph of the other (the supergraph) then the supergraph may have more attack or support relationships than the subgraph (but not vice-versa). Intuitively, drawing attack and support relationships is what we are looking for in our quests of interesting graphs, and we may thus assume that the more a graph displays attacks and supports, the more it is interesting in some sense. Consequently, if our two argumentation graphs are interesting wrt a dataset, then we may prefer the supergraph to the subgraph.

To order graphs of a set of argumentation graphs, we may thus use a preference relation according to which a graph $\mathcal{G}$ is preferred to a graph $\mathcal{H}$ if $\mathcal{H}$ is isomorphic to a subgraph of $\mathcal{G}$. For our purposes, such a relation will be called a graph refinement relation, and we can conceive three types of refinements, each type corresponding to a notion of subgraph (it will be argued very soon that only attack-support refinements are relevant).

Definition 6.18. Let $\mathcal{G}$ and $\mathcal{H}$ denote two argumentation graphs such that $\mathcal{G}$ and $\mathcal{H}$ are not isomorphic. Argumentation graph $\mathcal{G}$ is more

- **argument-attack-support refined** than $\mathcal{H}$ iff $\mathcal{H}$ is isomorphic to any subgraph of $\mathcal{G}$;
- **argument refined** than $\mathcal{H}$ iff $\mathcal{H}$ is isomorphic to any argument subgraph of $\mathcal{G}$;
- **attack-support refined** than $\mathcal{H}$ iff $\mathcal{H}$ is isomorphic to any attack-support subgraph of $\mathcal{G}$.

Given a set of argumentation graphs, we can order it on the basis of graph refinement, resulting thus into a *partially* (refinement-) ordered set of argumentation graphs. And because a partially refinement-ordered set of argumentation graphs can appear too large for human uses, we can distinguish the ‘most refined’ argumentation graphs, yielding a refined set of arguments. Since three types of refinements have been identified, namely argument-attack-support refinements and argument refinements and attack-support refinements, three types of refined sets of arguments can be straightforwardly defined.

Definition 6.19. Let $\mathfrak{G}$ be a set of argumentation graphs.

- $\mathfrak{G}$ is **argument-attack-support refined** iff there exists no argumentation graph $\mathcal{G}$ and $\mathcal{H}$ in $\mathfrak{G}$ such that $\mathcal{G}$ is more argument-attack-support refined than $\mathcal{H}$.
- $\mathfrak{G}$ is **argument refined** iff there exists no argumentation graph $\mathcal{G}$ and $\mathcal{H}$ in $\mathfrak{G}$ such that $\mathcal{G}$ is more argument refined than $\mathcal{H}$.
- $\mathfrak{G}$ is **attack-support refined** iff there exists no argumentation graph $\mathcal{G}$ and $\mathcal{H}$ in $\mathfrak{G}$ such that $\mathcal{G}$ is more attack-support refined than $\mathcal{H}$.

Example 6.20. Let $\mathsf{H1}$, $\mathsf{G1}$, $\mathsf{H2}$ and $\mathsf{G2}$ denote the argumentation graphs drawn in figures 19, 20, 21 and 22, respectively. The set $\{\mathsf{H1}, \mathsf{H2}, \mathsf{G1}, \mathsf{G2}\}$ is not attack-support refined, whereas the set $\{\mathsf{G1}, \mathsf{G2}\}$ is attack-support refined. □

Intuitively, the more a graph is refined, the more it is interesting. In that regard, argument-attack-support refinements can appear problematic if an interesting graph has actually a subgraph which is the pattern we are looking for and its consistency has to be asserted. This would typically occur for very noisy datasets where multiple noise statements are often labelled y. Argument refinement is also problematic because it can yield a large refined set of graphs. Hence, the conceptions of argument-attack-support refinements and argument refinements are discarded, and, as a middle way, only attack-support refinements are considered in the paper.

In the remainder, we may simply say refinements instead of attack-support refinement, and, amongst a partially (refinement-) ordered set of argumentation graphs which are consistent with a dataset, we will retain the most refined graphs and discard others.

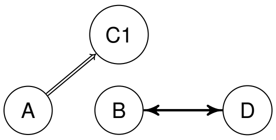


Figure 19.: H1.

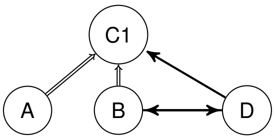


Figure 20.: G1.

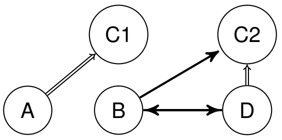


Figure 21.: H2.

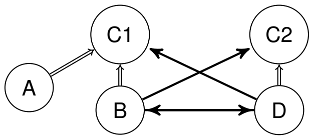


Figure 22.: G2.

6.2.2. Confidence

To fully order (a most refined set of) argumentation graphs, we can use some scalar measures of interestingness of the graphs, and again various measures can be proposed. For the sake of simplicity, we may gauge 'confidence' values which are similar to well-established confidence measures as used in the mining of association rules to evaluate the interestingness of such rules (Agrawal et al., 1993). In the context of our probabilistic argumentation setting, the confidence of an argument can be designed to indicate how frequently an argument is labelled IN when its subarguments are labelled IN, measuring so the confidence that an argument is accepted (and thus expressed) when all its subarguments are accepted.

Definition 6.21. Let $\mathcal{K} = (K_1, \ldots, K_N)$ be a dataset, and $\mathcal{L} = (L_1, \ldots, L_N)$ a collection of $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings of an argumentation graph $\mathcal{G}$ such that every labelling L_i is a K_i-maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling. The **confidence of an argument** A wrt $\mathcal{K}$, denoted $\mathrm{conf}(A, \mathcal{K})$, is such that:

$$\mathrm{conf}(A, \mathcal{K}) = \frac{P_{\mathcal{L}}(L_{\{A\}} = \mathsf{IN})}{P_{\mathcal{L}}(L_{\mathrm{sub}(A)} = \mathsf{IN})},$$

where $\mathrm{sub}(A)$ is the set of direct subarguments of A, i.e. $\mathrm{sub}(A) = \{B \mid B \mapsto\!\!\!\Rightarrow_{\mathcal{G}} A\}$.

We can remark that, for any argument A, if the argument is assumptive, then $P_{\mathcal{L}}(L_{\mathrm{sub}(A)} = \mathsf{IN}) = 1$ by Equation 2, and thus $\mathrm{conf}(A, \mathcal{K}) = P_{\mathcal{L}}(L_{\{A\}} = \mathsf{IN})$. If the argument is not assumptive and $P_{\mathcal{L}}(L_{\mathrm{sub}(A)} = \mathsf{IN}) = 0$ then, like the undefined probability of an event conditioned on any event with zero probability, the confidence of such an argument is left undefined.

Similarly as the frequency of a graph, the confidence of a graph can be defined in various ways. We may prefer graphs where arguments are associated with high confidence values instead of low values. Accordingly, the confidence of a graph is simply defined as the minimal confidence amongst all the defined confidence values of non-assumptive arguments.

Definition 6.22. Let $\mathcal{G}$ be an argumentation graph, and $\mathcal{A} \subseteq \mathcal{A}_{\mathcal{G}}$ its set of non-assumptive arguments such that for every argument $A \in \mathcal{A}$ the confidence value of A is defined. The **confidence** of $\mathcal{G}$ wrt a dataset $\mathcal{K}$, denoted $\mathrm{conf}(\mathcal{G}, \mathcal{K})$, is such that if $\mathcal{A} \neq \emptyset$ then

$$\mathrm{conf}(\mathcal{G}, \mathcal{K}) = \min_{A \in \mathcal{A}}(\mathrm{conf}(A, \mathcal{K}))$$

else $\mathrm{conf}(\mathcal{G}, \mathcal{K}) = 0$.

Hence, for any given dataset $\mathcal{K}$ and argumentation graph $\mathcal{G}$, it holds that $\mathrm{conf}(\mathcal{G}, \mathcal{K}) \in [0, 1]$.

Example 6.23. Suppose the argumentation graphs and the dataset $\mathcal{K}$ in figures 16, 17, and 18. Graphs G' and G'' have the following confidence values:

G' arguments	A	B	D	C1	C2
$\mathrm{conf}(\cdot, \mathcal{K})$	$3/7$	$2/7$	$3/7$	$1/3$	$1/3$

G'' arguments	A	B	D	C1
$\mathrm{conf}(\cdot, \mathcal{K})$	$3/7$	$2/7$	$3/7$	$1/3$

Consequently, $\mathrm{conf}(\mathsf{G}', \mathcal{K}) = 1/3$ and $\mathrm{conf}(\mathsf{G}'', \mathcal{K}) = 1/3$. □

Given a set of argumentation graphs, we can order it with a total order relation $\leq$ on the set of graphs by comparing the confidence values of the graphs, resulting thus into a totally confidence-ordered set of argumentation graphs.

Definition 6.24. A **confidence-order** on a set of argumentation graphs $\mathfrak{G}$ wrt a dataset $\mathcal{K}$ is a binary relation $\leq \subseteq \mathfrak{G} \times \mathfrak{G}$, such that for all $\mathcal{G}_1, \mathcal{G}_2 \in \mathfrak{G}$, $\mathcal{G}_1 \leq \mathcal{G}_2$ iff $\mathrm{conf}(\mathcal{G}_1, \mathcal{K}) \leq \mathrm{conf}(\mathcal{G}_2, \mathcal{K})$.

In this view, a graph is meant to be preferred to another graph if its confidence is higher than the confidence of the latter. Hence, the confidence-order reflects the idea that we shall prefer graphs where arguments are accepted rather than discarded (i.e. labelled OUT, UND or OFF) when their subarguments are accepted. Eventually, given a confidence-ordered set of argumentation graphs, we might prefer the argumentation graph(s) with the highest confidence, but such a choice is left here at the discretion of the end-users.

6.2.3. Most interesting order of argumentation graphs

Interesting graphs can be ordered by combining refinement and confidence criteria, thereby leading to confidence-orders of refined sets of interesting argumentation graphs, simply called 'interesting orders of argumentation graphs'.

Definition 6.25. Let $\mathcal{K}$ be a dataset, $\mathfrak{L}$ a language graph consistent with $\mathcal{K}$. An **interesting order of argumentation graphs** wrt $\mathcal{K}$ and $\mathfrak{L}$ is a confidence-ordered refined non-empty set of interesting argumentation graphs wrt $\mathcal{K}$ and $\mathfrak{L}$.

Given a refined non-empty set of interesting graphs (e.g. found in some search), the confidence-order can apply to any subsets of the set. As we are looking for the maximal order (wrt set inclusion), we will focus on the confidence order over the whole set, i.e. the 'most interesting order' of graphs.

Definition 6.26. Let $\mathcal{K}$ be a dataset, $\mathfrak{L}$ a language graph consistent with $\mathcal{K}$. Given a non-empty set $\mathfrak{G}^*$ of refined sets of argumentation graphs, an interesting order $\preceq$ of argumentation graphs wrt $\mathcal{K}$ and $\mathfrak{L}$ is a **most interesting order** wrt $\mathfrak{G}^*$ iff $\preceq$ is an interesting order on a set $\mathfrak{X}$ in $\mathfrak{G}^*$ such that $\mathfrak{X}$ is not a strict subset of any set in $\mathfrak{G}^*$.

The definition also applies to provide a partial order amongst results of different searches. If a search returns a refined set of argumentation graphs $\mathfrak{G}$ and another search returns another set $\mathfrak{H}$ such that $\mathfrak{H}$ is a subset of $\mathfrak{G}$, then the interesting order of $\mathfrak{G}$ is the most interesting order.

To recap the section, criteria concerning super well-formedness, parsimony and frequency have been set forth to elicit interesting argumentation graphs. Then a refinement relation amongst graphs have been proposed to identify most refined interesting argumentation graphs, which can be eventually fully ordered wrt their confidence values. Of course, other definitions of interesting graphs or interesting orders of such graphs can be given. The framework can be changed here (without necessarily impacting much the problem definition) and variants are left to future investigations.

7. An Iterated Local Iterative Deepening Depth-First Search

Given a dataset and a language graph consistent with the dataset, the problem is about finding a most interesting order of argumentation graphs wrt the dataset and language graph (Section 4). To address this problem, we can first collect interesting argumentation graphs and then order the graphs.

To collect interesting argumentation graphs, search strategies in rule learning (Fürnkranz et al., 2014) are a compelling source of inspiration. Various search strategies can be set up. First we can endorse basic search strategies such as a breadth-first search or a depth-first search. A depth-first search may be preferred wrt a breadth-first search to get a reasonable space complexity. However, a depth-first search may be stuck into branches of uninteresting argumentation graphs before collecting interesting graphs (if any are found). An iterative deepening depth-first search may then be preferred, but the amount of time to collect interesting argumentation graphs in such an iterative search may be a practical limitation. This prohibits bare depth-first search or iterative deepening variants. As an alternative to collect interesting graphs, we will investigate an iterated local search where interesting argumentation graphs are sought by modifying any graphs which are potentially interesting.

The search space can be immense. It has been reduced to particular super well-formed argumentation graphs induced by the statements labelled in the given dataset: we set up a $\{\mathsf{premise}, \mathsf{target}\}$-labelling, so that any argumentation graph of the search space is such that every argument is either a 'premise argument' (i.e. an argument whose conclusion is a premise) or a 'target argument' (i.e. an argument whose conclusion is a target and all its direct subarguments are premise arguments) or an assumptive argument whose conclusion is neither a premise nor a target. Amongst these super well-formed graphs, some graphs can be found to be both consistent with the dataset and interesting.

As any iterated local search, we have to define initial points for the search, a neighbourhood relation, the local search and possibly some heuristics. They are specified in the rest of the section.

7.1. Initial argumentation graph

For the sake of simplicity, the set of initial points for the search is reduced in this paper to a singleton including a particular parsimonious argumentation graph induced by the statement labelled in the dataset. This argumentation graph is such that its support relation is empty, while its attack relation is shaped to make it consistent with the dataset and congruent with the input language graph. The idea is that any arguments can attack each other as long as the contrary has not been shown, as suggested by Observation 1 and Proposition 3.

Observation 1. Let L be an $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling of an argumentation graph $\mathcal{G}$, and $K = K(L, \Phi)$ be the acceptance bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling of $\mathcal{G}$ and from $\{L\}$. There exist statements $\phi_1, \phi_2 \in \mathsf{y}(K)$ iff there exist arguments $A, B \in \mathcal{A}_{\mathcal{G}}$ such that $\phi_1 = \mathrm{con}(A)$, $\phi_2 = \mathrm{con}(B)$ and $A, B \in \mathsf{IN}(L)$.

Proposition 3. *Let L be a complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling of an argumentation graph $\mathcal{G}$. If arguments $A, B \in \mathsf{IN}(L)$ then argument A does not attack B, i.e. $A \not\rightsquigarrow_{\mathcal{G}} B$.*

Observation 1 and Proposition 3 indicate that, given a case K, if an argumentation graph has no attacks between arguments whose conclusions are labelled y in K then the graph can be viewed as a potential graph consistent with K. Accordingly, we distinguish graphs which are 'attack-consistent' with a dataset $\mathcal{K}$ and those which are 'attack-maxconsistent' with $\mathcal{K}$.

Definition 7.1. Let $\mathcal{K}$ be a dataset, and $\mathcal{G}$ an argumentation graph.

- $\mathcal{G}$ is **attack-consistent with** $\mathcal{K}$ iff for any arguments $A, B \in \mathcal{A}_{\mathcal{G}}$, if there exists a case $K \in \mathcal{K}$ such that $\mathrm{con}(A), \mathrm{con}(B) \in \mathsf{y}(K)$ then argument A does not attack B, i.e. $A \not\rightsquigarrow_{\mathcal{G}} B$.
- $\mathcal{G}$ is **attack-maxconsistent with** $\mathcal{K}$ iff $\mathcal{G}$ is attack-consistent with $\mathcal{K}$ and $\rightsquigarrow_{\mathcal{G}}$ is maximal (wrt set inclusion).

Example 7.2. The argumentation graph shown in Figure 23 is attack-maxconsistent with the dataset given in Figure 5.

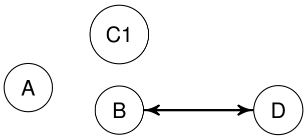


Figure 23.

□

Notation 11. If an argumentation graph is attack-consistent (attack-maxconsistent resp.) with a singleton $\{K\}$, then we may simply write that the graph is attack-consistent (attack-maxconsistent resp.) with case K.

If an argumentation graph is attack-consistent with a dataset and its support relation is not empty, then the graph may not be well-formed, e.g. if an argument A attacks an argument B, and B is a direct subargument of an argument C, then A may not attack C. For instance, the graph $\langle \{\mathsf{A}, \mathsf{B}, \mathsf{C1}, \mathsf{C2}, \mathsf{D}\}, \{(\mathsf{B}, \mathsf{D}), (\mathsf{D}, \mathsf{B})\}, \{(\mathsf{B}, \mathsf{C1}\}\rangle$ in Figure 24 is attack-(max)consistent with the dataset in Figure 5 and it is not well-formed (since D does not attack $\mathsf{C1}$). However, if the support relation is empty, then the argumentation graph is trivially well-formed.

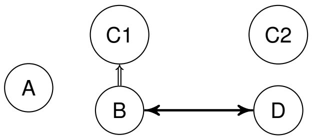


Figure 24.: Argumentation graph which is not well-formed.

As initial argumentation graph, we will take a 'simple argumentation graph' (defined below), which always exists.

Definition 7.3. An argumentation graph $\mathcal{G}$ is a **simple argumentation graph** of a dataset $\mathcal{K}$ iff $\mathcal{G}$ is strictly induced by $\Phi_{\mathcal{K}}$, and attack-maxconsistent with $\mathcal{K}$, and $\Rightarrow_{\mathcal{G}} = \emptyset$.

Example 7.4. Figure 23 displays a simple argumentation graph of the dataset in Figure 6. □

Theorem 7.5 (Existence)**.** *For any dataset $\mathcal{K}$, there exists a simple argumentation graph of $\mathcal{K}$.*

Proof. For any dataset $\mathcal{K}$, let us $\mathcal{G}$ denote an argumentation graph which is strictly induced by $\Phi_{\mathcal{K}}$. Make the relation $\rightsquigarrow_{\mathcal{G}}$ such that for any argument $A, B \in \mathcal{A}_{\mathcal{G}}$, if there exists a case

$K \in \mathcal{K}$ such that $\mathrm{con}(A), \mathrm{con}(B) \in \mathsf{y}(K)$ then argument A does not attack B, i.e. $A \not\rightsquigarrow_{\mathcal{G}} B$, otherwise $A \rightsquigarrow_{\mathcal{G}} B$. Consequently, by Definition 7.1, $\mathcal{G}$ is attack-maxconsistent. Make the relation $\Mapsto_{\mathcal{G}} = \emptyset$. Hence, $\mathcal{G}$ is strictly induced by $\Phi_{\mathcal{K}}$, and attack-maxconsistent with $\mathcal{K}$, and $\Mapsto_{\mathcal{G}} = \emptyset$, therefore $\mathcal{G}$ is a simple argumentation graph of $\mathcal{K}$. □

For any dataset $\mathcal{K}$, there exists actually an infinite number of simple argumentation graphs of $\mathcal{K}$: these graphs differ in their argument identifiers. Nevertheless, they are isomorphic with each other.

Theorem 7.6. *For any dataset $\mathcal{K}$, all simple argumentation graphs of $\mathcal{K}$ are isomorphic.*

Proof. Let $\mathcal{G}$ and $\mathcal{G}'$ be two simple argumentation graphs of $\mathcal{K}$. Let f denote the bijection between the sets of arguments $\mathcal{A}_{\mathcal{G}}$ and $\mathcal{A}_{\mathcal{G}'}$ such that $f(A) = A'$ iff $\mathrm{con}(A) = \mathrm{con}(A')$. Consequently, for any $A, B \in \mathcal{A}_{\mathcal{G}}$, $\mathrm{con}(A) = \mathrm{con}(f(A))$ and $\mathrm{con}(B) = \mathrm{con}(f(B))$, and $A \rightsquigarrow_{\mathcal{G}} B$ iff $f(A) \rightsquigarrow_{\mathcal{G}'} f(B)$. Moreover, for any $A, B \in \mathcal{A}_{\mathcal{G}}$, $\Mapsto_{\mathcal{G}} = \emptyset$ and $\Mapsto_{\mathcal{G}'} = \emptyset$, and thus $\Mapsto_{\mathcal{G}} = \Mapsto_{\mathcal{G}'}$. Therefore, the argumentation graphs $\mathcal{G}$ and $\mathcal{G}'$ are isomorphic. □

The initial argumentation graph of a search is any simple argumentation graph of the input dataset. This graph is consistent with the input dataset and the language graph (as requested by the problem definition). It is also interesting wrt the input dataset and language graph.

Theorem 7.7. *For any dataset $\mathcal{K}$, any simple argumentation graph of $\mathcal{K}$ is consistent with $\mathcal{K}$.*

Proof. Let $\mathcal{K}$ be any dataset, and $\mathcal{G}$ a simple argumentation graph of $\mathcal{K}$. For every case K in $\mathcal{K}$, let L be any K-maxconsistent complete $\{\textsf{IN}, \textsf{OUT}, \textsf{UND}, \textsf{OFF}\}$-labelling of $\mathcal{G}$. The labellings L and K are consistent. Therefore, graph $\mathcal{G}$ is consistent with dataset $\mathcal{K}$. □

Theorem 7.8. *For any dataset $\mathcal{K}$, any simple argumentation graph of $\mathcal{K}$ is consistent with any language graph consistent with $\mathcal{K}$.*

Proof. Let $\mathfrak{L}$ denote a language graph consistent with dataset $\mathcal{K}$, $\mathcal{G}$ a simple argumentation graph of $\mathcal{K}$, and A, B any arguments in $\mathcal{A}_{\mathcal{G}}$. Graph $\mathcal{G}$ is a simple argumentation graph of $\mathcal{K}$ and thus, by definition 7.3, $\mathcal{G}$ is attack-maxconsistent with $\mathcal{K}$. Graph $\mathcal{G}$ is attack-maxconsistent with $\mathcal{K}$ and thus, by Definition 7.1, if there exists a case $K \in \mathcal{K}$ such that $\mathrm{con}(A), \mathrm{con}(B) \in \mathsf{y}(K)$ then argument A does not attack B, i.e. $A \not\rightsquigarrow_{\mathcal{G}} B$, else $A \rightsquigarrow_{\mathcal{G}} B$. Language graph $\mathfrak{L}$ is consistent with dataset $\mathcal{K}$ and thus, by Definition 3.21, if $(\mathrm{con}(A), \mathrm{con}(B)) \in \mathit{contrar}_{\mathfrak{L}}$ or $(\mathrm{con}(A), \mathrm{con}(B)) \in \mathit{contrad}_{\mathfrak{L}}$ then $\mathrm{con}(A), \mathrm{con}(B) \notin \mathsf{y}(K)$, and thus $B \rightsquigarrow_{\mathcal{G}} A$ and $A \rightsquigarrow_{\mathcal{G}} B$. Therefore, by Definition 3.9, graph $\mathcal{G}$ is consistent with the language graph. □

Theorem 7.9. *Let $\mathcal{K}$ be a dataset and $\mathfrak{L}$ a language graph consistent with $\mathcal{K}$. Any simple argumentation graph of $\mathcal{K}$ is interesting wrt $\mathcal{K}$ and $\mathfrak{L}$.*

Proof. Let $\mathcal{G}$ denote a simple argumentation graph of dataset $\mathcal{K}$. Graph $\mathcal{G}$ is consistent with $\mathcal{K}$ and $\mathfrak{L}$ (theorems 7.7 and 7.8). It is well-formed, succinct, and attack-subargument concordant and thus, by Definition 6.4, it is super well-formed. It is also parsimonious wrt $\mathcal{K}$ and $\mathfrak{L}$, frequent wrt $\mathcal{K}$ (since its frequency is one, i.e. maximal), and concise (since the set of non-assumptive arguments is empty). Therefore, by Definition 6.17, $\mathcal{G}$ is interesting wrt $\mathcal{K}$ and $\mathfrak{L}$. □

Eventually, a simple argumentation graph of a dataset can be efficiently built by using Algorithm 1.

Algorithm 1 Computation of a simple argumentation graph of a dataset.

```
input A dataset $\mathcal{K}$,
Compute a trivial argumentation graph $\mathcal{G}$ strictly induced by $\Phi_{\mathcal{K}}$,
$\leadsto \;\leftarrow\; \mathcal{A}_{\mathcal{G}} \times \mathcal{A}_{\mathcal{G}}$,
for all $K$ in $\mathcal{K}$ do
    $\leadsto \;\leftarrow\; \leadsto \setminus \{(A, B) \mid (A, B) \in \leadsto \text{ and } \mathrm{con}(A), \mathrm{con}(B) \in \mathsf{y}(K)\}$
end for
return $\langle \mathcal{A}_{\mathcal{G}}, \leadsto, \emptyset \rangle$.
```

7.2. Neighbourhood

As the initial argumentation graph caters for attacks only (its support relation is empty), we can now focus on supports. We investigate here a graph neighbourhood built by adding a support between arguments of the graph or between an assumptive argument and a 'new' argument. Hence, the neighbourhood is partitioned into two types, neighbour graphs with exactly one new argument (extended neighbours), and neighbour graphs with no new arguments (restricted neighbours).

Definition 7.10. Let $\mathcal{K}$ be a dataset, $\mathfrak{L}$ a language graph consistent with dataset $\mathcal{K}$, π a $\{\mathsf{premise}, \mathsf{target}\}$-labelling of $\mathcal{K}$, $\mathcal{G}$ an argumentation graph consistent with $\mathfrak{L}$, and $A \in \mathcal{A}_{\mathcal{G}}$ an assumptive argument such that its conclusion is a premise, i.e. $\mathrm{con}(A) \in \mathsf{premise}(\pi)$.

- Let B be an argument in $\mathcal{A}_{\mathcal{G}}$ such that its conclusion is a target, i.e. $\mathrm{con}(B) \in \mathsf{target}(\pi)$, and A does not support B, i.e. $A \not\Mapsto_{\mathcal{G}} B$. An argumentation graph $\mathcal{H}$ is a **restricted neighbour of** $\mathcal{G}$ wrt $\mathcal{K}$ and $\mathfrak{L}$ iff
  (1) $\mathcal{A}_{\mathcal{H}} = \mathcal{A}_{\mathcal{G}}$, and
  (2) $\leadsto_{\mathcal{H}} = \leadsto_{\mathcal{G}} \cup \{(C, D) \mid C \leadsto_{\mathcal{H}} E \,\text{and}\, E \Longmapsto_{\mathcal{H}} D\} \cup \{(C, D) \mid C, D \in \mathcal{A}_{\mathcal{H}}$ and $(\mathrm{con}(C), \mathrm{con}(D)) \in \mathit{contrar}_{\mathfrak{L}} \cup \mathit{contrad}_{\mathfrak{L}}\}$, and
  (3) $\Mapsto_{\mathcal{H}} = \Mapsto_{\mathcal{G}} \cup \{(A, B)\}$, and
  (4) $\mathcal{H}$ is super well-formed, and
  (5) $\mathcal{H}$ is concise wrt $\mathcal{K}$.
- Let B be an argument *not* in $\mathcal{A}_{\mathcal{G}}$ such that its conclusion is a target, i.e. $\mathrm{con}(B) \in \mathsf{target}(\pi)$. An argumentation graph $\mathcal{H}$ is an **extended neighbour of** $\mathcal{G}$ wrt $\mathcal{K}$ and $\mathfrak{L}$ iff
  (1) $\mathcal{A}_{\mathcal{H}} = \mathcal{A}_{\mathcal{G}} \cup \{B\}$, and
  (2) $\leadsto_{\mathcal{H}} = \leadsto_{\mathcal{G}} \cup \{(C, D) \mid C \leadsto_{\mathcal{H}} E \,\text{and}\, E \Longmapsto_{\mathcal{H}} D\} \cup \{(C, D) \mid C, D \in \mathcal{A}_{\mathcal{H}}$ and $(\mathrm{con}(C), \mathrm{con}(D)) \in \mathit{contrar}_{\mathfrak{L}} \cup \mathit{contrad}_{\mathfrak{L}}\}$, and
  (3) $\Mapsto_{\mathcal{H}} = \Mapsto_{\mathcal{G}} \cup \{(A, B)\}$, and
  (4) $\mathcal{H}$ is super well-formed, and
  (5) $\mathcal{H}$ is concise wrt $\mathcal{K}$.
- An argumentation graph $\mathcal{H}$ is a **neighbour of** $\mathcal{G}$ wrt $\mathcal{K}$ and $\mathfrak{L}$ iff
  - ○ $\mathcal{H}$ is a restricted neighbour of $\mathcal{G}$ wrt $\mathcal{K}$ and $\mathfrak{L}$, or
  - ○ $\mathcal{H}$ is an extended neighbour of $\mathcal{G}$ wrt $\mathcal{K}$ and $\mathfrak{L}$.

In Definition 7.10, for any restricted neighbour, the first item specifies that the set of arguments is unchanged, whereas for an extended neighbour the first item adds a new argument to the set of arguments. For both types of neighbours, the second item specifies attacks so that the graph is well-formed and consistent with the language graph; the third updates the supports with a new support; the fourth and fifth require super well-formedness and conciseness respectively.

Well-formedness and conciseness are included here because these properties can be (quickly) checked without any (costly) pass through the dataset, while other properties requiring a pass through the dataset can be checked later in some particular order to optimise the search.

Since only assumptive arguments can support any other arguments in neighbour graphs, we are moving towards pretty flat structures. The search for deeper structures along with intermediary concepts would suggest a more sophisticated neighbourhood relation, which is left for future investigations.

Example 7.11. Suppose the dataset along with the $\{\mathsf{premise}, \mathsf{target}\}$-labelling $\langle\{\mathsf{a}, \mathsf{b}, \mathsf{d}\}, \{\mathsf{c}\}\rangle$ and the argumentation graph in Figure 25; the graph is attack-maxconsistent but not consistent with the dataset. The neighbours are in Figure 26; neighbour (c) is consistent with the dataset.

a	b	c	d
y	y	y	n
y	n	n	y
n	y	n	n
n	n	y	y
n	n	n	y

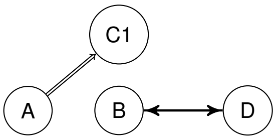


Figure 25.: A dataset and an argumentation graph.

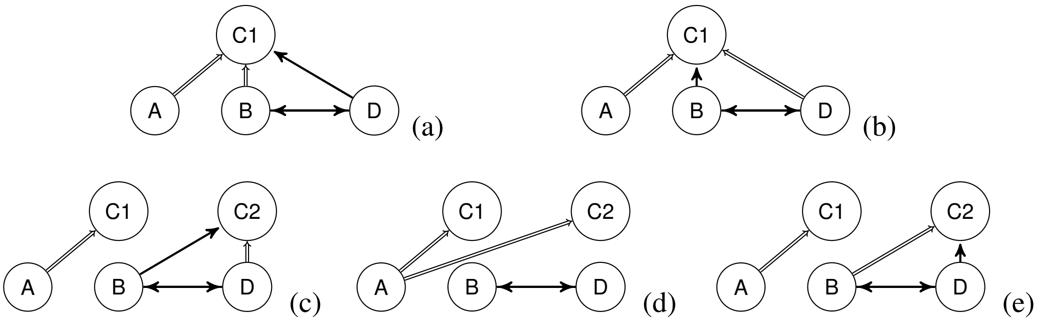


Figure 26.: Neighbours.

□

Eventually, we can flag interesting graphs in a neighbourhood. To see whether a graph is interesting, it can be evaluated wrt the input dataset and the language graph. The most demanding criteria are frequency and parsimony. The most brutal approach for checking frequency implies to compute, for every case K in the dataset, a K-maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling L of the graph, e.g. the unique maxomit K-maxconsistent complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling which can be easily computed. We can then check whether a graph is frequent or not. Checking parsimony is harder. In a brute force approach, every non-empty subset of arguments must be shown to be necessary. In more subtle approaches, some subsets can be reckoned necessary at the outset, this is the case of any subsets of assumptive arguments for example. As a work-around of parsimony checking, a pseudo-parsimony was used: if the graph was not found parsimonious in a given finite computational budget, then its parsimony was not proven, and thus it was discarded.

7.3. Local search and heuristics

The iterated local search can be viewed as the iterative quest of a scientist for interesting explanatory synthetic accounts of some data. At each iteration, the scientist starts from an argumentation graph and slightly modifies it, evaluates the modified graphs wrt the data, and elicits most interesting ones which can become the start of new (re)search investigations. When an iteration does not lead to any new interesting argumentation graph, the scientist backtracks to previous positions.

As local search, various options are possible, see e.g. (Fürnkranz et al., 2014; Gendreau and Potvin, 2010). We adopt an iterative deepening depth-first search, yielding thus an 'iterated local iterative deepening depth-first search'. The iterative deepening depth search iterates a depth-limited depth-first search with increasing depth limits until an interesting graph is found. As alluded to earlier, the main reason for a local deepening depth-first search is that such a search may be less likely to suffer from myopia than basic hill-climbing search approaches.

The overall search maintains a set of argumentation graphs which are found interesting, and a last-in first-out stack of argumentation graphs, called a frontier, which are graphs found interesting and whose all neighbours have not been explored. Frontier graphs are added to the stack one at a time, and the graph selected or erased at the frontier at any time is the last graph which was added. The search selects a frontier graph at the beginning of every iterative deepening depth-first search, and it backtracks to the next frontier graph when all of the neighbours of the first selection have been explored.

The depth-limited depth-first search looks for an interesting graph by building a neighbourhood tree, i.e. a tree where every node is an argumentation graph and the child of every node is a neighbour argumentation graph. The root node is a frontier graph. Then the depth-limited depth-first search can be eventually oriented by some heuristics, as suggested next.

On the basis of $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings of an argumentation graph $\mathcal{G}$ which are max-consistent with a dataset $\mathcal{K}$, we can evaluate the consistency of $\mathcal{G}$ with the dataset $\mathcal{K}$. To do so, we may first assess the consistency of $\mathcal{G}$ wrt any case in the dataset through a consistency score. For convenience, this score can be defined so that, given a case K, it is maximal when the difference is minimal between case K and the K-maxconsistent bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling of $\mathcal{G}$. Accordingly, we may use the following score, a Jaccard similarity coefficient comparing the sets $\mathsf{y}(K)$ and $\mathsf{y}(K')$.

Definition 7.12. Let K be a case, and K' the K-maxconsistent bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling of an argumentation graph $\mathcal{G}$. The **consistency score of argumentation graph $\mathcal{G}$ wrt case K**, denoted $S(\mathcal{G}, K)$, is such that if $\mathsf{y}(K) \cup \mathsf{y}(K') \neq \emptyset$ then

$$s(\mathcal{G}, K) = \frac{|\mathsf{y}(K) \cap \mathsf{y}(K')|}{|\mathsf{y}(K) \cup \mathsf{y}(K')|}.$$

else $s(\mathcal{G}, K) = 1$.

Then, the consistency wrt a dataset can be evaluated through the distribution $P_{\mathcal{K}}$ of bivalent $\{\mathsf{y}, \mathsf{n}\}$-labellings in the dataset. This distribution reflects the frequency of labellings in the collection:

$$P_{\mathcal{K}}(K) = \frac{m_{\mathcal{K}}(K)}{|\mathcal{K}|}. \tag{3}$$

On this basis, we can then adopt a score as in (Riveret and Governatori, 2016) that we may call the expected consistency of a graph $\mathcal{G}$ wrt a dataset $\mathcal{K}$.

Definition 7.13. The **expected consistency of an argumentation graph $\mathcal{G}$ wrt a dataset $\mathcal{K}$**, denoted $S(\mathcal{G}, \mathcal{K})$, is such that

$$S(\mathcal{G}, \mathcal{K}) = \sum_{K \in \mathcal{K}} P_{\mathcal{K}}(K) \cdot s(\mathcal{G}, K).$$

So, using notation from Definition 7.12, if for any case K in a dataset $\mathcal{K}$ it holds that $\mathsf{y}(K) = \mathsf{y}(K')$, then $s(\mathcal{G}, K) = 1$, and thus $S(\mathcal{G}, \mathcal{K}) = 1$, and graph $\mathcal{G}$ is consistent with $\mathcal{K}$.

Example 7.14. The consistency scores of argumentation graphs drawn in figures 27 and 28 are given wrt a dataset in Figure 29. Graph G' is consistent with the dataset, G'' is not.

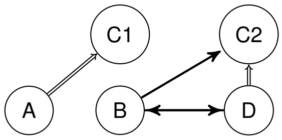


Figure 27.: G'.

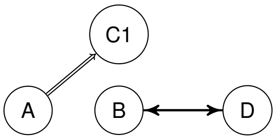


Figure 28.: G''.

a	b	c	d	G'	A	B	C1	D	C2	$s(\mathsf{G}', K)$	G''	A	B	C1	D	$s(\mathsf{G}'', K)$
y	n	n	n		IN	OFF	OFF	OFF	OFF	1		IN	OFF	OFF	OFF	1
y	y	y	n		IN	IN	IN	OFF	OFF	1		IN	IN	IN	OFF	1
y	n	n	y		IN	OFF	OFF	IN	OFF	1		IN	OFF	OFF	IN	1
n	n	n	n		OFF	OFF	OFF	OFF	OFF	1		OFF	OFF	OFF	OFF	1
n	y	n	n		OFF	IN	OFF	OFF	OFF	1		OFF	IN	OFF	OFF	1
n	n	y	y		OFF	OFF	OFF	IN	IN	1		OFF	OFF	OFF	IN	0.5
n	n	n	y		OFF	OFF	OFF	IN	OFF	1		OFF	OFF	OFF	IN	1

Figure 29.: A dataset (on the left, where each row is a case) and corresponding maxomit complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings of the argumentation graphs drawn in figures 27 and 28, such that these labellings are maxconsistent with cases of the dataset.

□

Different consistency scores or measures may be defined; we leave these points to other investigations. Nonetheless, the proposed scores have interesting properties when comparing the consistency of a graph and its subgraphs. In particular, to guide the search of argumentation graphs, a possible heuristics can be drawn by observing that the addition of supports to an argumentation graph may decrease the consistency score, but not necessarily, and that the addition of arguments can increase the score, as investigated below.

Lemma 7.15. *Let K be a case, and K' the K-maxconsistent bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling of an argumentation graph $\mathcal{G}$.*

$$\mathsf{y}(K') \subseteq \mathsf{y}(K).$$

Proof. let L denote any K-maxconsistent $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling of $\mathcal{G}$. For any statement $\phi \in \Phi_K$, if $K'(\phi) = \mathsf{y}$ then that there exists an argument $A \in \mathcal{A}_{\mathcal{G}}$ such that $\mathrm{con}(A) = \phi$ and $L(A) = \mathsf{IN}$ (by Definition 3.19). Moreover, if there exists an argument $A \in \mathcal{A}_{\mathcal{G}}$ such that $\mathrm{con}(A) = \phi$ and $L(A) = \mathsf{IN}$ then $K(\phi) = \mathsf{y}$ (by definitions 5.1 and 5.5). Hence, for any statement $\phi \in \Phi_K$, if $K'(\phi) = \mathsf{y}$ then $K(\phi) = \mathsf{y}$. Therefore, $\mathsf{y}(K') \subseteq \mathsf{y}(K)$. □

Example 7.16. Suppose for instance the argumentation graphs G' and G'' in figures 27 and 28. Let $\mathsf{K} = \langle\{\mathsf{c}, \mathsf{d}\}, \{\mathsf{a}, \mathsf{b}\}\rangle$ be a given case. The K-maxconsistent bivalent $\{\mathsf{y}, \mathsf{n}\}$-labellings of G'

or G'' are $\mathsf{K}' = \langle\{\mathsf{c},\mathsf{d}\},\{\mathsf{a},\mathsf{b}\}\rangle$ and $\mathsf{K}'' = \langle\{\mathsf{d}\},\{\mathsf{a},\mathsf{b},\mathsf{c}\}\rangle$ respectively. Hence $\mathsf{y}(\mathsf{K}') \subseteq \mathsf{y}(\mathsf{K})$ and $\mathsf{y}(\mathsf{K}'') \subseteq \mathsf{y}(\mathsf{K})$ indeed. □

Lemma 7.17. *Let K be a case, $\mathcal{G}_1$ and $\mathcal{G}_2$ two argumentation graphs, and K_1 and K_2 the K-maxconsistent bivalent $\{\mathsf{y},\mathsf{n}\}$-labellings of $\mathcal{G}_1$ and $\mathcal{G}_2$ (respectively). If $|\mathsf{y}(K_1)| \leq |\mathsf{y}(K_2)|$ then $s(\mathcal{G}_1, K) \leq s(\mathcal{G}_2, K)$.*

Proof. From Definition 7.12,

$$s(\mathcal{G}_1, K) = \frac{|\mathsf{y}(K) \cap \mathsf{y}(K_1)|}{|\mathsf{y}(K) \cup \mathsf{y}(K_1)|} \quad \text{and} \quad s(\mathcal{G}_2, K) = \frac{|\mathsf{y}(K) \cap \mathsf{y}(K_2)|}{|\mathsf{y}(K) \cup \mathsf{y}(K_2)|}.$$

By Lemma 7.15, $\mathsf{y}(K_1) \subseteq \mathsf{y}(K)$ and $\mathsf{y}(K_2) \subseteq \mathsf{y}(K)$, and thus

$$s(\mathcal{G}_1, K) = \frac{|\mathsf{y}(K_1)|}{|\mathsf{y}(K)|} \quad \text{and} \quad s(\mathcal{G}_2, K) = \frac{|\mathsf{y}(K_2)|}{|\mathsf{y}(K)|}.$$

Therefore, if $|\mathsf{y}(K_1)| \leq |\mathsf{y}(K_2)|$ then $s(\mathcal{G}_1, K) \leq s(\mathcal{G}_2, K)$. □

Lemma 7.18. *Let $\mathcal{K}$ be a dataset, $\mathcal{G}$ be an argumentation graph, $\mathcal{G}_1$ a restricted neighbour of $\mathcal{G}$ wrt $\mathcal{K}$, and $\mathcal{G}_2$ an extended neighbour of $\mathcal{G}$ wrt $\mathcal{K}$, $K \in \mathcal{K}$ a case, K', K'_1 and K'_2 the K-maxconsistent bivalent $\{\mathsf{y},\mathsf{n}\}$-labellings of $\mathcal{G}$, $\mathcal{G}_1$ and $\mathcal{G}_2$ respectively.*

$$\mathsf{y}(K'_1) \subseteq \mathsf{y}(K') \subseteq \mathsf{y}(K'_2).$$

Proof. Let K denote any case in $\mathcal{K}$, and L, L_1 and L_2 any K-maxconsistent $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labellings of $\mathcal{G}$, $\mathcal{G}_2$ and $\mathcal{G}_2$ respectively.

Case $\mathsf{y}(K'_1) \subseteq \mathsf{y}(K')$. For any statement $\phi \in \Phi_K$, if $K'_1(\phi) = \mathsf{y}$ then that there exists an argument $X \in \mathcal{A}_{\mathcal{G}1}$ such that $\mathrm{con}(X) = \phi$ and $L_1(X) = \mathsf{IN}$ (by Definition 3.19). Moreover, if there exists an argument $X \in \mathcal{A}_{\mathcal{G}1}$ such that $\mathrm{con}(X) = \phi$ and $L_1(X) = \mathsf{IN}$ then there exists an argument $Y \in \mathcal{A}_{\mathcal{G}}$ such that $\mathrm{con}(Y) = \phi$ and $L(Y) = \mathsf{IN}$. If there exists an argument $Y \in \mathcal{A}_{\mathcal{G}}$ such that $\mathrm{con}(Y) = \phi$ and $L(Y) = \mathsf{IN}$ then $K'(\phi) = \mathsf{y}$ (by Definition 3.19). Hence, for any statement $\phi \in \Phi_K$, if $K'_1(\phi) = \mathsf{y}$ then $K'(\phi) = \mathsf{y}$. Therefore, $\mathsf{y}(K'_1) \subseteq \mathsf{y}(K')$.

Case $\mathsf{y}(K') \subseteq \mathsf{y}(K'_2)$. For any statement $\phi \in \Phi_K$, if $K'(\phi) = \mathsf{y}$ then that there exists an argument $X \in \mathcal{A}_{\mathcal{G}}$ such that $\mathrm{con}(X) = \phi$ and $L(X) = \mathsf{IN}$ (by Definition 3.19). Moreover, if there exists an argument $X \in \mathcal{A}_{\mathcal{G}}$ such that $\mathrm{con}(X) = \phi$ and $L(X) = \mathsf{IN}$ then there exists an argument $Y \in \mathcal{A}_{\mathcal{G}2}$ such that $\mathrm{con}(Y) = \phi$ and $L_2(Y) = \mathsf{IN}$. If there exists an argument $Y \in \mathcal{A}_{\mathcal{G}2}$ such that $\mathrm{con}(Y) = \phi$ and $L_2(Y) = \mathsf{IN}$ then $K'_2(\phi) = \mathsf{y}$ (by Definition 3.19). Hence, for any statement $\phi \in \Phi_K$, if $K'(\phi) = \mathsf{y}$ then $K'_2(\phi) = \mathsf{y}$. Therefore, $\mathsf{y}(K') \subseteq \mathsf{y}(K'_2)$. □

We can now show that the addition of supports with no new arguments to an argumentation graph may decrease the consistency score (but not necessarily), and such an addition cannot strictly increase the score. In contrast, the addition of supports with new arguments may increase the score, and it cannot strictly decrease the score.

Theorem 7.19. *Let $\mathcal{K}$ be a dataset, and $\mathcal{G}_1$ a restricted neighbour of an argumentation graph $\mathcal{G}$ wrt $\mathcal{K}$, such that both graphs are induced by the set of statements $\Phi_{\mathcal{K}}$.*

$$S(\mathcal{G}_1, \mathcal{K}) \leq S(\mathcal{G}, \mathcal{K}).$$

Proof. By Lemma 7.18, for any case $K \in \mathcal{K}$ it holds that $\mathsf{y}(K'_1) \subseteq \mathsf{y}(K')$ and thus $|\mathsf{y}(K'_1)| \leq |\mathsf{y}(K')|$. By Lemma 7.17, $s(\mathcal{G}_1, K) \leq s(\mathcal{G}, K)$. Therefore, $S(\mathcal{G}_1, \mathcal{K}) \leq S(\mathcal{G}, \mathcal{K})$. □

Theorem 7.20. *Let $\mathcal{K}$ be a dataset, and $\mathcal{G}_2$ an extended neighbour of an argumentation graph $\mathcal{G}$ wrt $\mathcal{K}$, such that both graphs are induced by the set of statements $\Phi_{\mathcal{K}}$.*

$$S(\mathcal{G}, \mathcal{K}) \leq S(\mathcal{G}_2, \mathcal{K}).$$

Proof. By Lemma 7.18, for any case $K \in \mathcal{K}$ it holds that $\mathsf{y}(K') \subseteq \mathsf{y}(K'_2)$ and thus $|\mathsf{y}(K')| \leq |\mathsf{y}(K'_2)|$. By Lemma 7.17, $s(\mathcal{G}, K) \leq s(\mathcal{G}_2, K)$. Therefore, $S(\mathcal{G}, \mathcal{K}) \leq S(\mathcal{G}_2, \mathcal{K})$. □

This investigation suggests some guidance on the addition of supports to an argumentation graph in a search of interesting graphs. Since we prefer attack-support refined argumentation graphs, we can first add supports with no new arguments to an argumentation graph which was previously found interesting. As Theorem 7.19 reflects that the addition of supports (with no new arguments) to an argumentation graph may decrease the consistency score, but not necessarily, we can favour the addition of supports which do not decrease the consistency score. If the resulting argumentation graph is not anymore consistent with the dataset, then we can try to recover consistency by adding a support with a new argument, since Theorem 7.20 shows that the addition of an argument (possibly along with a support) to an argumentation graph can increase the score. In other words, when the addition of a support decreases the score, we can seek to compensate it by adding arguments. These are simple ideas on which can be based some heuristics to guide the iterated local search, as implemented next.

8. Implementation

As the proposed search implies a good deal of backtracking (the local search backtracks to interesting argumentation graphs whose neighbourhood has been unexplored), a proof-of-concept was implemented in Prolog.

First, the collected interesting argumentation graphs can be refined and ordered to result into an interesting order of graphs, as proposed in the pseudocode of Algorithm 2.

Algorithm 2 Searching an interesting order of argumentation graphs to explain cases.

```
inputDataSet(Dataset).
inputLanguageGraph(LanguageG).
inputInterestingGraph(InterestingGs).

findInterestingGraphs(OutputInterestingOrderGs) :-
  iliddfSearch,
  collect(CollectedInterestingGs),
  refine(CollectedInterestingGs, RefinedInterestingGs),
  order(RefinedInterestingGs, OutputInterestingOrderGs).
```

The input dataset, language graph and set of interesting argumentation graphs are assumed to be asserted through the predicates `inputDataSet/1`, `inputLanguageGraph/1` and `inputInterestingGraph/1` respectively.

Querying `findInterestingGraphs(OutputInterestingOrderGs)` initiates the process, where `OutputInterestingOrderGs` is the output consisting of a most interesting order of argumentation graphs collected during the search. The overall search of a most interesting order goes into the following steps.

- `iliddfSearch/0` initiates an iterated local iterative deepening depth-first search,
- `collect/1` collects all the interesting graphs which have been found during the search,
- `refine/2` refines the interesting graphs,

- `order/2` orders the refined graphs resulting thus into a most interesting order of argumentation graphs wrt the considered dataset (and frequency threshold).

An iterated local iterative deepening depth-first search (iliddfSearch) can be implemented as exposed in Algorithm 3.

Algorithm 3 Iterated local iterative deepening depth-first search.

```
iliddfSearch :-
  stop_iliddfSearch.

iliddfSearch :-
  getFrontierGraph(G),
  iddfSearch(G), !
  iliddfSearch.

iliddfSearch :-
  eraseFrontierGraph,
  iliddfSearch.
```

Querying `iliddfSearch` initiates the search. Then the search loops through `iliddfSearch/0`, along with the following auxiliary predicates.

- `getFrontierGraph/1` gets the last recorded frontier graph which is interesting,
- `iddfSearch/1` begins an iterative deepening depth-first search. If this search finds no interesting graphs then it fails and the frontier graph is erased, before moving to another iterative search from the frontier graph which was previously recorded.

An iterated search continues until `stop_iliddfs` is satisfied, in particular when all the frontier graphs have been explored, or possibly when other conditions are satisfied, for example, when some computational budget is depleted.

In this context, an iterative deepening depth-first search (iddfSearch) can be implemented by iterating a depth-first search, as proposed in Algorithm 4.

Algorithm 4 An iterative deepening depth-first search.

```
iddfSearch(G1) :-
  limit(Depth),
  dfSearch(Depth, G1, []).
```

Querying `iddfSearch(InitG)` initiates the search to find interesting argumentation graphs, where `InitG` is an input (frontier) interesting argumentation graph. Then the search begins:

- `limit/1` determines the limit depth.
- `dfSearch/3` initiates an informed depth-limited depth-first search.

If no interesting graphs are found then the query fails.

An informed (heuristic) depth-limited depth-first search (dfSearch) is given in Algorithm 5. Querying `dfSearch(Depth, InitG, [])` initiates the search to find a graph, where `Depth` is the input limit depth, `InitG` is an input argumentation graph, and the third argument is instantiated to an empty list `[]` and indicates that no new interesting graphs have been found at the outset of the search. The search loops through `dfSearch/3`, and calls the following predicates.

- `generateNeighbourGraphs/2` generates neighbour graphs of the current graph `G1`, without new arguments or with one new argument (see Definition 7.10). As suggested by the heuristics in Subsection 7.3 on the addition of supports, the order of the clauses is such that restricted neighbours are explored first, with a possible backtrack to extended neighbours.

Algorithm 5 An informed (heuristic) depth-limited depth-first search.

```
1:   dfSearch(Depth, _, FoundInterestingGs) :-
2:    stop_dfSearch(Depth, _, FoundInterestingGs), !.
3:
4:  dfSearch(Depth, G1, _) :-
5:    generateNeighbourGraphs(G1, NeighbourGs),
6:    search(Depth, G1, NeighbourGs, G2),
7:    updateInterestingGraphs(G2, FoundInterestingGs),
8:    NewDepth is Depth-1,
9:    dfSearch(NewDepth, G2, FoundInterestingGs), !.
10:
11: generateNeighbourGraphs(G1, NeighbourGs):-
12:   restrictedNeighbourGraphs(G1, NeighbourGs).
13:
14: generateNeighbourGraphs(G1, NeighbourGs):-
15:   extendedNeighbourGraphs(G1, NeighbourGs).
16:
17: search(Depth, G1, NeighbourGs, G2) :-
18:   scoreGraphs(Depth, NeighbourGs, ScoredGs),
19:   selectGraph(ScoredGs, G2),
20:   heuristics(G1, G2),
21:   parsimonious(G2),
22:   frequent(G2).
23:
24: heuristics([G1, 1], [G2, Score2]) :-
25:     Score2 < 1, !.
26:
27: heuristics([G1, Score1], [G2, Score2]) :-
28:   Score2 >= Score1.
```

- `search/4` does the search amongst neighbour graphs by scoring, sorting and selecting interesting or promising graphs.
    - ○ `scoreGraphs/3` computes the consistency score of neighbour graphs wrt the given dataset; maxomit labellings were used to do so because they can be easily computed. It also sorts the graphs wrt their consistency scores. `scoreGraphs/3` can be also defined to stop scoring argumentation graphs has soon as a neighbour argumentation graph is found consistent, and then backtrack if this graph is not found promising wrt the heuristics.
    - ○ `selectGraph/1` retains a neighbour graph with the highest consistency score and allows backtracking in the search.
    - ○ `heuristics/2` further orients the depth-first search. The first clause (line 24) pursues the search with an argumentation graph which is inconsistent with the dataset if its parent is consistent with it; this typically occurs when a support is added, decreasing so the consistency score, see Theorem 7.19. The second clause (line 27) pursues the search with an argumentation graph as long as this graph is more consistent with the dataset than its parent; this typically occurs with the addition of a new argument increasing so the consistency score, see Theorem 7.20.
    - ○ `parsimonious/1` checks whether the selected neighbour graph is parsimonious.
    - ○ `frequent/1` ensures that the selected neighbour graph is frequent (wrt a given frequency threshold).
- `updateInterestingGraphs/2` takes as input an argumentation graph (here `G2`) and records it as an interesting graph on the top of the stack of frontier graphs. The list `FoundInterestingGs = [G2]` indicates whether an interesting graph has been found.

Eventually, `stop_dfSearch/2` stops the search wrt some conditions, for example when the limit depth or some computational budget has been reached, or as soon as an interesting argumentation graph has been found.

The computational burden of Algorithm 2 holds in the iterated local search as given in Algorithm 3. It is thus particularly important to assess some computational properties of Algorithm 3. First all, we can note that the algorithm terminates, since it stops when the computational budget is reached.

Theorem 8.1. *Algorithm 3 terminates.*

Proof. Algorithm 3 runs with a finite computational budget, and it stops when the computational budget is reached. □

Concerning completeness, Algorithm 3 is complete in the sense of search algorithms because it is guaranteed to find a solution for any input dataset. If more interesting graphs are looked after, then one can rely on the completeness of the local iterative deepening depth-first search.

Theorem 8.2. *Algorithm 3 is complete.*

Proof. If Algorithm 3 finds no interesting argumentation graphs wrt the input dataset and frequency threshold parameter, then it stands with the input graph, i.e. a simple argumentation graph of the dataset, which is consistent with the dataset (Theorem 7.7) and any input language graph which is consistent with the dataset (Theorem 7.8), and which is interesting wrt the input dataset and language graph (Theorem 7.9). □

However, the algorithm may not return all the interesting argumentation graphs wrt a given dataset and threshold. For example, the simple treatment of attacks does not allow to retrieve the exact argumentation graph drawn in Figure 1. A more sophisticated mechanism to induce attacks is left to future research.

As to soundness, any argumentation graph collected by Algorithm 3 is interesting wrt the considered dataset and language graph. Hence, the algorithm is sound.

Theorem 8.3. *Algorithm 3 is sound.*

Proof. The iterative deepening depth-first search (iddfSearch, Algorithm 4 which iterates over dfSearch), collects only argumentation graphs which are interesting wrt the given dataset and language graph. Consequently, the iterated local iterative deepening depth-first search (iliddfSearch, Algorithm 3 which iterates over iddfSearch by initiating the search from frontier graphs which are interesting wrt the given dataset and language graph, including a simple argumentation graph of the dataset) collects/returns only argumentation graphs which are interesting wrt the given dataset and language graph. □

As Algorithm 3 returns a sound solution to the problem even if it is interrupted before it ends. As any iterated local search, Algorithm 3 is actually an anytime algorithm.

Regarding complexity, Algorithm 3 is based on an iterated local search where the local search is an iterated deepening depth-first search (with no infinite branches), thus it inherits the computational complexity of such depth-first search (Korf, 1985). Regarding the verification of the interestingness of an argumentation graph, it is possible to do the verification with a computational budget. If the verification is not completed within the budget, then the graph is not proved interesting and it can be discarded. Now, beyond these computational properties, the search has to be evaluated in practice; some experiments are reported next.

9. Experiments

The evaluation of some knowledge extracted from data may not be easy when patterns in the data are initially unknown. Measures such as accuracy can be taken, but they may not reflect well the quality of the extracted knowledge. For example, in the legal domain, the quality of the extracted knowledge, in terms of intelligibility, is as important as accuracy measures. To avoid such issues, the work reported here was illustrated and evaluated with constructed datasets, whose properties were known and controlled, so that the effects of different properties on the induction of argumentation graphs could be more clearly assessed.

Considered datasets were built from routine cases as proposed in (Bench-Capon, 1993) about, without quotes, a fictional welfare benefit paid to pensioners to defray expenses for visiting a spouse in hospital. The conditions to obtain a benefit (represented by statement grant) are as follows.

(1) The person should be of pensionable age (60 for a woman, 65 for a man) (age), and
(2) the person should have four out of the last five paid contributions in relevant contribution years ($\mathsf{contrib}$), and
(3) the person should be a spouse of the patient (spouse), and
(4) the person should not be absent from the UK (uk), and
(5) the person should have capital resources not amounting to more than $3,000$ ($\mathsf{capital}$), and
(6) if the relative is an in-patient ($\mathsf{inPatient}$) the hospital should be within a certain distance ($\mathsf{inDistance}$); if an out-patient ($\mathsf{outPatient}$), beyond that distance ($\mathsf{outDistance}$).

The original data built on these conditions can be real or Boolean (Bench-Capon, 1993), whereas our argumentation setting deals with bivalent features only. For this reason, attribute-value pairs were mapped into propositions. This mapping can be achieved as in (Bench-Capon, Coenen, and Leng, 2000) in a preprocessing for the discovery of association rules, by using some domain knowledge to partition the domain of each feature. Instead of this mapping, attribute-value pairs were mapped into statements as given in the above items (e.g. age stands for a person of pensionable age), so that our approach can be better illustrated.

To capture the conditions to obtain a benefit in the context of the proposed approach, all the statements were considered premises, except grant and $\neg\mathsf{grant}$; and grant was the target. Premise statements also included 52 'noise' statements.

Every statement was the conclusion of an argument. Every noise statement was the conclusion of a so-called noise assumptive argument. Every statement which was not a noise statement was the conclusion of arguments denoted as follows: non-assumptive arguments were denoted as A^- and $\mathsf{A}_1^+, \mathsf{A}_2^+, \ldots, \mathsf{A}_n^+$ (different arguments with the same conclusion), and assumptive arguments as $\mathsf{B}^{-/+}$, $\mathsf{C}^{-/+}$, $\mathsf{D}^{-/+}$, $\mathsf{E}^{-/+}$, $\mathsf{F}^{-/+}$, $\mathsf{G}^{-/+}$, $\mathsf{H}^{-/+}$ such that:

$$
\begin{array}{ll}
\mathrm{con}(\mathsf{A}_i^+) = \mathsf{grant}; & \mathrm{con}(\mathsf{A}^-) = \neg\mathsf{grant}; \\
\mathrm{con}(\mathsf{B}^+) = \mathsf{age}; & \mathrm{con}(\mathsf{B}^-) = \neg\mathsf{age} \\
\mathrm{con}(\mathsf{C}^+) = \mathsf{contrib}; & \mathrm{con}(\mathsf{C}^-) = \neg\mathsf{contrib}; \\
\mathrm{con}(\mathsf{D}^+) = \mathsf{spouse}; & \mathrm{con}(\mathsf{D}^-) = \neg\mathsf{spouse}; \\
\mathrm{con}(\mathsf{E}^+) = \mathsf{uk}; & \mathrm{con}(\mathsf{E}^-) = \neg\mathsf{uk}; \\
\mathrm{con}(\mathsf{F}^+) = \mathsf{capital}; & \mathrm{con}(\mathsf{F}^-) = \neg\mathsf{capital}; \\
\mathrm{con}(\mathsf{G}^+) = \mathsf{inPatient}; & \mathrm{con}(\mathsf{G}^-) = \mathsf{outPatient}; \\
\mathrm{con}(\mathsf{H}^+) = \mathsf{inDistance}; & \mathrm{con}(\mathsf{H}^-) = \mathsf{outDistance}.
\end{array}
$$

On the basis of these arguments, the conditions for the welfare benefit can be modelled with different argumentation graphs, possibly leading to different datasets and thus experiments. Two sets of experiments were conducted to evaluate the approach. Every experiment was done with YAP Prolog 6.2.2 and Intel i7-4790 CPU @ 3.60 GHz.

Remark on evaluation. For every experiment, any returned argumentation graph can be anticipated to be 'interesting' and consistent with the input dataset (as requested by the problem definition). Besides consistency and interestingness criteria, we might also be tempted to use common machine learning evaluation measures such as accuracy, precision or recall. However, such measures are not appropriate for our explanatory purposes.

To better understand why common machine learning evaluation measures are not appropriate for our explanatory purposes, let us consider for example accuracy which equals $(tp + tn)/(tp + tn + fn + fp)$ where tp is the number of true positives (i.e. positives which are correctly predicted), tn the number of true negatives (i.e. negatives which are correctly predicted), fn the number of false negatives (i.e. negatives which are incorrectly predicted), fp the number of false positives (i.e. positives which are incorrectly predicted). For prediction purposes, predictions are made, then cases are fully revealed, and then accuracy can be computed. However, for our explanatory purposes, the cases are exposed first and then explained. Consequently, accuracy does not appear as an appropriate measure for explanatory purposes, since no predictions are made. Similarly, precision and recall are inappropriate, and the remark holds for any logloss or statistical distances (given we have a probabilistic setting).

Nevertheless, machine learning evaluation measures may be adjusted for explanatory purposes. Positives which are correctly (incorrectly resp.) *predicted* can be replaced by positives which are correctly (incorrectly resp.) *explained*, and similarly for negatives. On the basis of explanatory evaluation with this interpretation for true and false positives, and true and false negatives, it turns out that the explanatory accuracy, precision and recall are maximal. Indeed, since the problem definition request to find argumentation graphs which are consistent with the dataset, the number of false positives and negatives is zero ($fn = fp = 0$). Consequently, explanatory accuracy, precision and recall are maximal for every returned argumentation graph, in every forthcoming experiment. For this reason, in the rest of the paper, an alternative evaluation will regard the returned orders of explanatory argumentation graphs which are consistent with input datasets.

9.1. First set of experiments

Setting. In the first set of experiments, the welfare benefit along with its conditions were modelled 'by hand' with an argumentation graph including 52 noise arguments and whose argument subgraph induced by non-noisy arguments is shown in Figure 30. This argumentation graph is called the *source argumentation graph*, since it was the 'source' of different datasets. More specifically, different datasets of 600 cases were generated from this argumentation graph. Every dataset was characterised by:

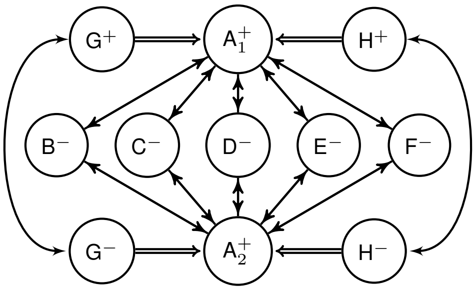


Figure 30.: Argument subgraph induced by non-noisy arguments for the first set of experiments. Attacks $G^+ \rightsquigarrow A_2^+$, $H^+ \rightsquigarrow A_2^+$, $G^- \rightsquigarrow A_1^+$ and $H^- \rightsquigarrow A_1^+$ are not shown for the sake of clarity.

- the probability P_{noise} that any noise statement is labelled y in any case of the dataset. Thus, if $P_{\text{noise}} = 0$ for instance then every noise statement was labelled n in every case of the dataset, and if $P_{\text{noise}} = 1$ then every noise statement was always labelled y;

- the probability P_{error} that the welfare was not attributed in a case though it should have been (that can simulate discretionary or erroneous decisions). Thus, if $P_{\text{error}} = 0$ then the dataset is deterministic, otherwise it is non-deterministic.

Any deterministic dataset was such that grant was labelled y in 300 cases, and n in the 300 other cases. Accordingly, a database characterised by $P_{\text{error}} = 0.5$ was such that grant was labelled y in about 150 cases, and n in the 450 other cases.

For each dataset, Algorithm 2 was run to find an interesting order of argumentation graphs wrt the dataset and a frequency threshold $freq = 0.01$, and with a computational budget of 300 seconds. The maximal limit depth for any iterative deepening depth-first search was 10.

Results. For each dataset, the search returned an interesting order of argumentation graphs as specified in Section 6. Of course, we cannot show all the graphs in the returned sets. Instead, we give in Table 1 the 'position' of the source argumentation graph in the returned ordered set for different values of P_{noise} and P_{error}. For example, for $P_{\text{noise}} = 0.5$ and $P_{\text{error}} = 0.4$, the position $0 : 4 : 5$ means that 5 graphs were returned, 0 graphs had a strictly superior confidence than the source graph, 4 graphs had a strictly inferior confidence; and thus we can conclude that the source graph was returned, and it is the graph with the highest confidence in the returned order. A position such as $4 : 0 : 4$ implies that the source argumentation graph was not included in the returned order. A position such as $0 : 0 : 2$ implies that the source argumentation graph was included in the returned order, and the second graph in the order had the same confidence. To ease the reading, an asterisk indicates that the source argumentation graph was included in the returned order.

Table 1.: Positions of the source argumentation graph. For each position i:j:k, the integer k indicates the number of graphs in the returned order, i (j resp.) the number of graphs which had a strictly superior (inferior resp.) confidence than the confidence of the source graph. An asterisk indicates that the source graph was included in the order.

		P_{noise}										
		0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
P_{error}	0.0	0:0:1*	0:0:1*	0:0:1*	0:0:1*	0:0:1*	0:0:1*	3:0:4*	0:3:4*	1:3:5*	3:0:3	0:0:1*
	0.1	0:0:1*	0:0:1*	0:0:1*	0:0:1*	0:0:1*	0:0:1*	0:3:4*	0:3:4*	1:3:5*	3:0:3	0:0:1*
	0.2	0:0:1*	0:0:1*	0:0:1*	0:0:1*	0:0:1*	1:3:5*	0:4:5*	1:2:4*	2:2:5*	4:0:4	0:0:1*
	0.3	0:0:1*	0:0:1*	0:0:1*	0:0:1*	0:0:1*	2:2:5*	0:4:5*	0:5:6*	1:3:5*	4:0:4	0:0:1*
	0.4	0:0:1*	0:0:1*	0:0:1*	0:0:1*	0:0:1*	0:4:5*	1:3:5*	2:1:4*	1:3:5*	4:0:4	0:0:1*
	0.5	0:0:1*	0:0:1*	0:0:1*	0:0:1*	0:0:1*	4:0:5*	3:1:5*	0:5:6*	2:3:6*	4:0:4	0:0:1*
	0.6	0:0:1*	0:0:1*	0:0:1*	0:0:1*	2:0:3*	3:0:4*	4:0:5*	2:3:6*	3:2:6*	4:0:4	0:0:1*
	0.7	0:0:1*	0:0:1*	0:0:1*	0:2:3*	0:0:3*	0:3:5*	0:2:4*	0:3:6*	0:4:5*	3:0:3	0:0:1*
	0.8	0:0:1*	0:0:1*	0:0:1*	0:0:2*	0:0:2*	0:0:4*	0:3:6*	1:0:5*	4:0:4	3:0:3	0:0:1*
	0.9	1:0:1	1:0:1	3:0:3	1:0:1	6:0:6	6:0:6	8:0:8	7:0:7	6:0:6	2:0:2	1:0:1
	1.0	1:0:1	1:0:1	1:0:1	1:0:1	1:0:1	1:0:1	1:0:1	1:0:1	1:0:1	1:0:1	1:0:1

Any order included no more than 8 argumentation graphs. Assuming that one can parse 8 argumentation graphs in a reasonable amount of time to elicit a convenient graph, this result shows that the problem definition where an order has to be returned is valuable in practice.

Non-trivial argumentation graphs could be found consistent with datasets which were non-deterministic. That results from the labelling semantics adopted from probabilistic argumentation. For deterministic datasets with noise parameter $P_{\text{noise}} \leq 0.8$, the search could find the source argumentation graph which could thus actually be used for prediction.

If one is looking for the source graph, we can note that it was included in the returned order for most datasets, in particular in every dataset where the noise parameters were such that

Table 2.: Runtimes in seconds for finding the source argumentation graph (if found).

	P_{noise}										
P_{error}	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0.0	0.047	1.407	1.656	1.938	2.126	2.436	2.453	2.639	2.78	-	0.39
0.1	0.047	1.25	1.468	1.735	1.938	2.108	2.374	2.531	2.578	-	0.39
0.2	0.062	1.25	1.438	1.577	1.75	1.986	2.078	2.265	2.423	-	0.329
0.3	0.046	1.048	1.375	1.452	1.593	1.766	1.859	1.906	2.203	-	0.313
0.4	0.062	0.953	1.156	1.345	1.468	1.608	1.735	1.828	1.939	-	0.328
0.5	0.047	0.86	1.031	1.11	1.313	1.5	1.454	1.515	1.781	-	0.265
0.6	0.016	0.781	0.876	1.001	1.063	1.189	1.249	1.298	1.608	-	0.281
0.7	0.047	0.579	0.749	0.828	0.859	1.015	1.079	1.093	1.907	-	0.235
0.8	0.016	0.578	0.656	0.703	0.734	0.812	0.859	0.859	-	-	0.219
0.9	-	-	-	-	-	-	-	-	-	-	-
1.0	-	-	-	-	-	-	-	-	-	-	-

$P_{\text{noise}} \leq 0.7$ and $P_{\text{error}} \leq 0.8$, or $P_{\text{noise}} \leq 0.8$ and $P_{\text{error}} \leq 0.7$. As expected, the source argumentation graph was not found for $P_{\text{error}} = 1$, and indeed it is difficult to conceive how the graph could be learnt with no positive examples of any benefit grants.

Runtimes to find the source argumentation graph are given in Table 2. In general, the higher the noise parameter P_{noise}, the longer the runtime to find the source graph. When the noise parameters were too high (e.g. $P_{\text{noise}} = 0.9$), the source argumentation graph could not be found. An exception is the extreme noise level at $P_{\text{noise}} = 1$, where the search space was dramatically reduced thanks to the conciseness of neighbourhoods.

By comparing tables 1 and 2, it can be remarked that for every dataset for which the source graph was found, such a graph was included in the final order of graphs. However, the source graph was not necessarily ordered as the greatest element, because a graph with a higher confidence value was found. This observation suggests that argumentation graphs may be better ordered by using different criteria than those proposed in the paper, possibly leading to further research in that regard.

As probabilistic argumentation graphs were mentioned in Section 3, we can note that such graphs can be drawn on the basis of any graphs in the returned order by computing the frequency of arguments wrt datasets. For instance, the source probabilistic argumentation graph for $P_{\text{noise}} = 0.5$ and $P_{\text{error}} = 0.4$ is drawn in Figure 31.

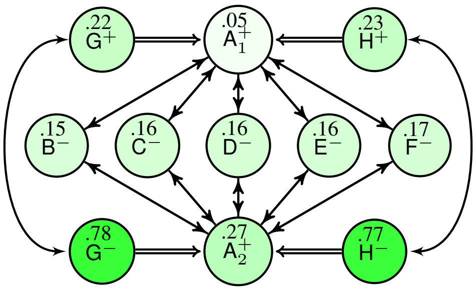


Figure 31.: A probabilistic source argumentation graph of the first set of experiments. Attacks $\mathsf{G}^+ \rightsquigarrow \mathsf{A}_2^+$, $\mathsf{H}^+ \rightsquigarrow \mathsf{A}_2^+$, $\mathsf{G}^- \rightsquigarrow \mathsf{A}_1^+$ and $\mathsf{H}^- \rightsquigarrow \mathsf{A}_1^+$ are not shown for the sake of clarity.

9.2. Second set of experiments

Setting. In a second set of experiments, the algorithm was executed on Boolean versions of the dataset from (Bench-Capon, 1993). Let us call it the original dataset. The dataset contains 2400 cases, and the search was performed on 1200 (training) cases. As in the first set of experiments, every version was characterised by the probability values P_{noise} and P_{error} with the same meaning as in the first set of experiments.

For each dataset, Algorithm 2 was executed with a computational budget of 300 seconds, and a frequency threshold $freq = 0.01$. The maximal limit depth for any iterative deepening depth-first search was 10.

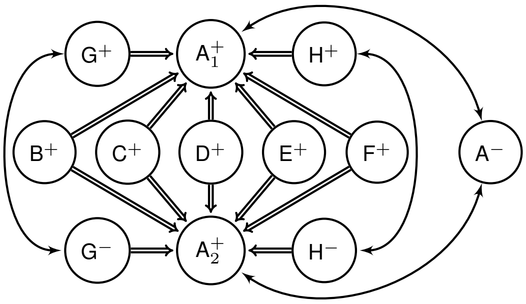


Figure 32.: Ideal argumentation graph. Attacks $\mathsf{G}^+ \rightsquigarrow \mathsf{A}_2^+$, $\mathsf{H}^+ \rightsquigarrow \mathsf{A}_2^+$, $\mathsf{G}^- \rightsquigarrow \mathsf{A}_1^+$ and $\mathsf{H}^- \rightsquigarrow \mathsf{A}_1^+$ are not shown for the sake of clarity.

Results. For each dataset, the results can be evaluated by the position of an 'ideal' argumentation graph which would fit perfectly well the dataset, such as the argumentation graph drawn in Figure 32, where the 52 noise assumptive arguments are not displayed due to the lack of space. Arguments B^- C^- D^- E^- and F^- do not appear because there were no assignments to their conclusions in the original dataset, while the argument A^- is included in the graph because its conclusion is a statement which was explicitly evaluated in the original dataset (this conclusion is an example of a statement which was neither a premise nor a target).

The position of the ideal argumentation graph in the output ordered set is given in Table 3.

Table 3.: Positions of the ideal argumentation graph drawn in Figure 32.

					P_{noise}							
		0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
P_{error}	0.0	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:1:4*	0:2:5*	0:1:2*
	0.1	0:0:1*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:3:4*	0:2:5*	0:1:2*
	0.2	0:0:1*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:3:4*	0:4:5*	0:1:2*
	0.3	0:0:1*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:3:4*	5:0:5	0:1:2*
	0.4	0:0:1*	1:0:2*	1:0:2*	1:0:2*	1:0:2*	1:0:2*	1:0:2*	1:0:2*	1:0:4*	5:0:5	1:0:2*
	0.5	0:0:1*	1:0:2*	1:0:2*	1:0:2*	1:0:2*	1:0:2*	1:0:2*	1:3:5*	3:1:5*	5:0:5	1:0:2*
	0.6	0:0:1*	1:0:2*	1:0:2*	1:0:2*	1:0:2*	1:0:2*	1:0:2*	1:3:5*	1:3:5*	5:0:5	1:0:2*
	0.7	0:0:1*	1:0:2*	1:0:2*	1:0:2*	1:0:2*	1:0:2*	1:0:2*	1:3:5*	1:4:6*	5:0:5	1:0:2*
	0.8	0:0:1*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:1:2*	0:2:6*	0:4:5*	4:1:6*	5:0:5	0:1:2*
	0.9	0:0:1*	1:0:2*	1:0:2*	1:0:2*	1:0:2*	1:0:5*	1:2:5*	1:1:6*	6:0:6	5:0:5	1:0:2*
	1.0	1:0:1	1:0:1	1:0:1	1:0:1	1:0:1	1:0:1	1:0:1	1:0:1	1:0:1	1:0:1	1:0:1

Table 4.: Runtimes in seconds for finding the ideal argumentation graph in Figure 32 (if found).

		P_{noise}									
P_{error}	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0.0	0.437	5.499	6.078	6.564	7.016	9.577	46.407	49.03	52.811	61.234	2.595
0.1	0.406	5.093	5.656	6.11	6.656	9.703	44.296	48.456	50.579	59.563	2.53
0.2	0.392	4.844	5.436	5.813	6.202	9.172	43.015	45.811	49.016	57.983	2.345
0.3	0.39	4.484	5.03	5.391	5.812	6.645	41.516	44.75	47.922	-	2.218
0.4	0.266	4.187	4.546	4.796	5.234	6.124	39.514	41.921	45.486	-	2.204
0.5	0.298	3.782	4.14	4.531	4.749	6.734	37.44	40.5	43.14	-	2.031
0.6	0.281	3.5	3.78	4.002	4.267	12.767	36.719	38.952	42.188	-	2.001
0.7	0.297	3.11	3.344	3.516	3.798	6.094	34.375	37.093	40.313	-	1.796
0.8	0.25	2.752	2.969	3.077	3.436	11.735	33.031	35.234	38.47	-	1.704
0.9	0.188	2.375	2.564	2.75	2.953	14.469	31.001	33.814	-	-	1.593
1.0	-	-	-	-	-	-	-	-	-	-	-

Any order returned no more than 6 argumentation graphs. Assuming that a user can parse 6 argumentation graphs in a reasonable amount of time, this result shows again that the problem definition where an order has to be returned is valuable in practice.

As in the first set of experiments, argumentation graphs could be found consistent with non-deterministic datasets. For deterministic datasets, the search could find the ideal argumentation graph, which could thus be used for prediction.

The ideal argumentation graph was included in the returned order in most cases, and in every dataset where the noise parameters were such that $P_{\text{noise}} \leq 0.7$ and $P_{\text{error}} \leq 0.9$, or $P_{\text{noise}} \leq 0.8$ and $P_{\text{error}} \leq 0.8$. As the first experiment and as expected, the ideal argumentation graph was not found for $P_{\text{error}} = 1$, for the same reason.

Runtimes to find the ideal argumentation graph are given in Table 4. Similarly as in the first set of experiments, the higher the noise P_{noise}, the longer the runtime to find the ideal graph, except for $P_{\text{noise}} = 1$. As we see in Table 4, when the different noises were too high, then the ideal argumentation graph could not be found. When the ideal argumentation graph was found, it was always included in the returned order of graphs, but not necessarily at the first position. Possible improvements for ordering argumentation graphs are left to future research.

Again, probabilistic argumentation graphs could be drawn on the basis of any graphs in the returned order by computing argument frequencies. For example, the resulting probabilistic ideal argumentation graph for $P_{\text{noise}} = 0.5$ and $P_{\text{error}} = 0.4$ is drawn in Figure 33.

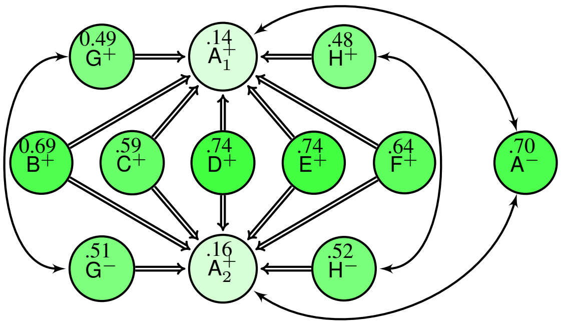


Figure 33.: Probabilistic ideal argumentation graph for $P_{\text{noise}} = 0.5$ and $P_{\text{error}} = 0.4$. Attacks $\mathsf{G}^{+} \rightsquigarrow \mathsf{A}_2^{+}$, $\mathsf{H}^{+} \rightsquigarrow \mathsf{A}_2^{+}$, $\mathsf{G}^{-} \rightsquigarrow \mathsf{A}_1^{+}$ and $\mathsf{H}^{-} \rightsquigarrow \mathsf{A}_1^{+}$ are not shown for the sake of clarity.

The ideal argumentation graph fits perfectly well the dataset. If one assumes a rule-based argumentation model where arguments are possibly built from defeasible rules, as e.g. done in (Riveret et al., 2018) and inspired by ASPIC^{+} (Modgil and Prakken, 2014), then one can straightforwardly propose the following defeasible rules to build arguments:

$$\begin{array}{rl}
 & \Rightarrow \neg\mathsf{grant}\\
\mathsf{age, contrib, spouse, uk, capital, inPatient, inDistance} & \Rightarrow \mathsf{grant}\\
\mathsf{age, contrib, spouse, uk, capital, outPatient, outDistance} & \Rightarrow \mathsf{grant}
\end{array}$$

These rules fit very well the grant conditions; the induction of theories from argumentation graphs is left to future research though.

To sum up the evaluation of the search to collect interesting argumentation graphs, its implemented algorithm returned an interesting order of argumentation graphs for any given dataset, even for non-deterministic datasets. For both sets of experiments, and although not reported here, every criterion defining interestingness had a positive effect on the results, in the sense that the omission of any criterion would degrade the results. Whatever the returned order, the argumentation graph with the highest confidence may be finally selected to explain the given dataset, but the ultimate selection may be also performed at the discretion of human agents. The number of argumentation graphs in the returned order did not exceed 8. That suggests that the problem definition where an order has to be returned and possible examined by a human is valuable in practice. The experiments also suggested that the returned argumentation graphs may be better ordered, possibly leading to further research in that regard.

10. Related Work

Datasets built on the conditions of the welfare benefits have been used in other works, see e.g. (Johnston and Governatori, 2003; Možina, Žabkar, Bench-Capon, and Bratko, 2005). Diverse machine learning techniques have been employed, and thus these techniques can be compared with the present approach. The main differences hold in that previous works attempted to induce other constructs such as rules for predictive purposes, whereas the work reported here employs labelling semantics taken from probabilistic argumentation and aims at the synthesis of argumentation graphs to explain cases. In that regard, the contribution aims at investigating a novel argument-based perspective on synthetic constructions for explanatory purposes rather than competing with the state-of-the-art in machine learning for predictive purposes.

While explanatory models may appear more humble than predictive models, the experiments have shown that generated explanations may lead to interesting representation results which can be compared to other results from works using variants of the welfare dataset taken from (Bench-Capon, 1993). For example, using Defeasible Logic for which an argumentation semantics is possible (Lam, Governatori, and Riveret, 2016), a refinement best-first search to induce defeasible theories (Johnston and Governatori, 2003) from the welfare dataset as booleanised in (Bench-Capon et al., 2000) yielded the following defeasible rules:

$$\begin{array}{rl}
 & \Rightarrow \neg grant\\
spouse, \neg absent, \neg age.It.60, \neg capital.gt\_3000 & \Rightarrow grant\\
distance.short, inpatient & \Rightarrow \neg grant
\end{array}$$

where rules are given in order of superiority, from weakest to strongest. In (Možina et al., 2005), an argument-based adaptation of CN2 (Clark and Niblett, 1989; Mozina, Zabkar, and Bratko, 2007) was used to produce a predictive model including the following rules:

(1) IF capital $>$ 2900 THEN qualified = no;

(2) IF age $\leq$ 59 THEN qualified = no;

(3) IF absent = yes THEN qualified = no;

(4) IF spouse = no THEN qualified = no;

(5) IF cont4 = no AND cont2 = no THEN qualified = no;
(6) IF inpatient = yes AND distance $>$ 735 THEN qualified = no;
(7) IF inpatient = no AND distance $\leq$ 735 THEN qualified = no;
(8) IF cont3 = no AND cont2 = no THEN qualified = no;
(9) IF cont5 = no AND cont3 = no AND cont1 = no THEN qualified = no;
(10) IF cont4 = no AND cont3 = no AND cont1 = no THEN qualified = no;
(11) IF cont5 = no AND cont4 = no AND cont1 = no THEN qualified = no.

None of these rules fully cover the given dataset, while our approach does for its explanatory purposes. However, the challenge was higher in (Johnston and Governatori, 2003; Možina et al., 2005) since these works used a trickier 'featurisation'. On that matter, the extension of the present approach to deal with continuous attributes and sophisticated features is left to future research. Overall, a remarkable difference between the present proposal and (Johnston and Governatori, 2003; Možina et al., 2005) is that their rule frameworks are not designed to deal with non-deterministic databases at the outset. As a consequence, such frameworks may lead to mixed results given non-deterministic datasets, see e.g. (Možina et al., 2005) on the robustness of argument-based rule learning in the face of erroneous decisions. On the contrary, the framework proposed here is designed on the basis of a labelling approach taken from probabilistic argumentation to properly tackle non-deterministic datasets, leading to relatively decent results as evidenced by the experiments.

Beyond these works which are evaluated on the same datasets, some other investigations in abstract argumentation can be related to the present proposal. Using the same setting for probabilistic argumentation, the closest works are those where the problem is to determine 'on-the-fly' an abstract argumentation graph to account for argument labellings or statement labellings taken from a stream of labellings (Riveret, 2016; Riveret and Governatori, 2016). In (Riveret, 2016) in particular, argument and statement labellings are taken from probabilistic argumentation and any statement can be supported by only one argument. In that regard, the approach taken in (Riveret, 2016) could be used in the work here to compute the simple argumentation graph of any input dataset.

Another related thread of research regards the 'synthesis' problem (Niskanen et al., 2016, 2019), i.e. the problem of building argumentation graphs that are semantically close to a collection of sets of accepted arguments (so called extensions). In contrast to this problem, the main input in the problem definition of the work reported here is a collection of statement labellings which have to be interpreted within a probabilistic (semi-abstract) argumentation setting where the support relation is essential. Instead of inducing one particular relevant argumentation graph, we look for a most interesting order of argumentation graphs. Furthermore, Niskanen et al. (2016, 2019) propose to solve the synthesis problem by using solutions based on Boolean optimization, while, in coherence with the fact that argumentation formalisms are typically inspired by 'natural argumentation', the proposed search of argumentation graphs as an iterated local search *may* better reflect human inquiries, thereby facilitating the human interaction with such a search.

The problem definition in the paper and the 'synthesis' problem can be also related to 'realisability' problems where the central question is whether a set of argument or statement interpretations (i.e. mappings assigning acceptance statuses to arguments/statements) can be realised by an argumentation graph (or related formalisms) along with some semantics (Dunne et al., 2015; Dyrkolbotn, 2014; Linsbichler et al., 2016; Pührer, 2015). An important difference with the work reported here is that no probabilistic setting is conceived in existing investigations of realisability, and no proper experimental evaluation has been realised.

As the proposed iterated local search uses a neighbourhood relation where the attack relation and supports relation of a graph are modified to fit a collection of labellings, it may be also related to 'enforcement' problems concerning modifications of an argumentation graph so

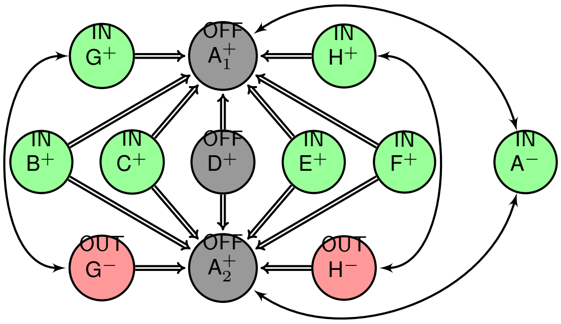


Figure 34.

that some arguments obtain particular status in the modified argumentation graph (possibly with a minimum amount of modifications) (Baumann, 2012; Baumann and Brewka, 2010; Coste-Marquis, Konieczny, Mailly, and Marquis, 2014; Coste-Marquis, Konieczny, Mailly, and Marquis, 2014; Coste-Marquis, Konieczny, Mailly, and Marquis, 2015). However, the work reported here differs in the adoption of a probabilistic setting which caters for collections of statement labellings and the search of interesting argumentation graphs.

The work reported may be also related to case-based reasoning, and such reasoning was investigated in (Čyras, Satoh, and Toni, 2016) using abstract argumentation. This work investigates case-based reasoning problems where cases are represented by abstract factors and outcomes, and given a new case represented by abstract factors, an outcome needs to be established. However, (Čyras et al., 2016) does not aim at constructing a synthetic account of the cases.

Some argumentation frameworks go beyond abstract argumentation or semi-abstract argumentation. For instance, an early investigation (Amgoud and Serrurier, 2008) put forward an argumentation-based model that constructs arguments for/against possible classification of an example. On the relationship between inductive reasoning and non-monotonic reasoning akin to argument-based reasoning, Ontañón, Dellunde, Godo, and Plaza (2012) gave an analysis of logical induction with a focus for hypothesis selection and an illustration in rule-based argumentation. This work, however, has no probabilistic setting, and has little or no consideration in the inductive process for the different reasoning levels characterising argumentation frameworks (such as the labelling of arguments and the labelling of statements) along with the fine granularity of possible labellings.

Instead of tackling the synthesis of predictive models, the proposed investigation is about the generation of explanatory models which are meant to complement predictive approaches by explaining their predictions. For example, if a neural network as in (Bench-Capon, 1993) takes as input a case K where the person is not a spouse of an in-patient in distance and all other conditions are fulfilled, and predicts that this person is not qualified, then the case can be explained by the minomit $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling $\langle\{\mathsf{A}^-, \mathsf{B}^+, \mathsf{C}^+, \mathsf{E}^+, \mathsf{F}^+, \mathsf{G}^+, \mathsf{H}^+\}, \{\mathsf{G}^-, \mathsf{H}^-\}, \emptyset, \{\mathsf{A}_1^+, \mathsf{A}_2^+, \mathsf{D}^+\}\rangle$ (amongst others) of the induced argumentation graph (drawn in Figure 32) consistent with the input. The minomit $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling is illustrated in Figure 34.

From this labelling, one can understand that the person has not been qualified because this person is not a spouse of the patient. In any explanatory labellings, any arguments supporting that the person has been qualified cannot be even reasonably advanced (and then rejected) because one of their subarguments cannot be advanced (the argument D^+ is labelled OFF). Hence, the proposal allows the generation and ordering of explanatory argumentation models which are complementary to powerful and off-the-shelf predictive techniques.

Eventually, it would be interesting to 'plug' the proposed work with neuro-symbolic settings for argumentation as in (Riveret et al., 2015a,b), which are using the same labelling framework to probabilistic argumentation. By doing so, induced argumentation graphs could guide (or constrain) samplings of argument labellings for predictive purposes.

The proposed framework is relatively abstract, and thus it may be adapted to fit particular domains. In the domain of philosophy of science for example, Šešelja and Straßer (2013) propose to enhance Dung's abstract argumentation graphs/frameworks with explanatory capabilities, and it is argued that resulting 'explanatory argumentation frameworks' are useful tool for modelling scientific debates (this work does not investigate the automated construction of synthetic accounts to explain a collection of cases). In the legal domain, (case-based) reasoning models are captured by a variety of works, see e.g. (Aleven, 1997; Ashley, 1991; Bench-Capon and Sartor, 2003; Chorley and Bench-Capon, 2005; McCarty, 1995; Prakken, Wyner, Bench-Capon, and Atkinson, 2015; Rissland and Skalak, 1991; Rissland, Valcarce, and Ashley, 1984; Verheij, 2017b). These various works suggest that the proposed framework could/should be adapted to represent and reason upon cases in the context of specific domains.

11. Conclusion

This paper addresses the problem of finding interesting explanations for a given collection of cases. Any case is a statement bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling, and an explanation for a case takes the form of a semi-abstract argumentation graph along with its labellings that are consistent with the case. Accordingly, the problem of finding interesting explanations for a dataset of cases was specified as the problem of finding, within a computational budget, a most interesting order of argumentation graphs wrt the dataset.

To tackle the problem, a probabilistic argumentation setting (Riveret et al., 2018) has been adopted along with particular complete $\{\mathsf{IN}, \mathsf{OUT}, \mathsf{UND}, \mathsf{OFF}\}$-labelling semantics for arguments and a bivalent $\{\mathsf{y}, \mathsf{n}\}$-labelling semantics for statements. Then, some criteria of interestingness have been canvassed to reduce the search space. On this basis, and bearing in mind the idea of mimicking scientific enquiries, a local iterated search with the local search as an iterative deepening depth-first search has been proposed. Some heuristics based on the relative consistency of argumentation graphs and their subgraphs was also proposed to orient the local iterated search. An implementation of the search was put forward and evaluated. Results witnessed the goodness of elicited interestingness criteria.

By addressing the problem with the proposed combination of argument and statement labellings, along with an iterated local search, we have learnt that multiple interesting explanatory argumentation graphs can be found, even for non-deterministic datasets. For deterministic datasets, the search can actually find argumentation graphs which can be used for prediction.

When generated explanatory argumentation graphs cannot be used for prediction, they may nevertheless complement other powerful techniques for prediction. For example, while artificial neural networks have been successful in multiple application domains, they have attracted less enthusiasm in the legal domain. A major reason is that a neural network often remains an inscrutable structure of interconnected neural units unable to provide any sort of intelligible explanations that are essential to back legal outcomes. As briefly illustrated when discussing related work, the proposed labelling semantics along with the argumentation graphs induced from the training dataset of neural networks may be used to explain outcomes of trained networks.

Future developments can be multiple, in particular to get more practical applications. For example, the reported work assumes Boolean cases: the search can be developed to deal with scalar attributes. The neighbourhood relation implies pretty flat structures: a richer neighbourhood may allow more depth via 'chains' of supporting arguments with intermediary concepts. Different search strategies and alternative orders of argumentation graphs may also be investigated. The consistency requirement in the problem definition may be relaxed to better deal with other sorts of noise and avoid overfitting, and the problem of eliciting a particular graph amongst the interesting graphs can be addressed with some other criteria. Finally, another future work could regard the induction of defeasible theories from the graphs.

References

R. Agrawal, T. Imieliński, and A. Swami. Mining association rules between sets of items in large databases. *SIGMOD Rec.*, 22(2):207–216, 1993.

V. Aleven. Teaching case-based argumentation through a model and examples, Phd Thesis, Univ. of Pittsburg, 1997.

L. Amgoud and M. Serrurier. Agents that argue and explain classifications. *Autonomous Agents and Multi-Agent Systems*, 16(2):187209, 2008.

K. D. Ashley. *Modeling Legal Arguments: Reasoning with Cases and Hypotheticals.* MIT Press, 1991.

K. Atkinson, P. Baroni, M. Giacomin, A. Hunter, H. Prakken, C. Reed, G. R. Simari, M. Thimm, and S. Villata. Towards artificial argumentation. *AI Magazine*, 2017.

P. Baroni and R. Riveret. Enhancing statement evaluation in argumentation via multi-labelling systems. *J. Artif. Intell. Res.*, 66:793–860, 2019.

P. Baroni, M. Caminada, and M. Giacomin. An introduction to argumentation semantics. *Knowledge Engineering Review*, 26(4):365–410, 2011.

P. Baroni, M. Giacomin, and B. Liao. Dealing with generic contrariness in structured argumentation. In *Proc. of the 24th Int. Joint Conf. on Artificial Intelligence*, pages 2727–2733. AAAI Press, 2015.

P. Baroni, G. Governatori, and R. Riveret. On labelling statements in multi-labelling argumentation. In *Proc. of the 22nd Eur. Conf. on Artif. Intell.*, pages 489–497. IOS Press, 2016.

P. Baroni, M. Giacomin, and B. Liao. A general semi-structured formalism for computational argumentation: Definition, properties, and examples of application. *Artificial Intelligence*, 257:158 – 207, 2018.

R. Baumann. What does it take to enforce an argument? minimal change in abstract argumentation. In *Proc. of the 20th Eur. Conf. on Artif. Intell.*, pages 127–132. IOS Press, 2012.

R. Baumann and G. Brewka. Expanding argumentation frameworks: Enforcing and monotonicity results. In *Proc. of the 3rd Int. Conf. on Computational Models of Argument*, pages 75–86. IOS Press, 2010.

W. Bechtel and A. Abrahamsen. Explanation: A mechanist alternative. *Studies in History and Philosophy of Biol and Biomed Sci*, 36(2):421–441, 2005.

T. Bench-Capon. Neural networks and open texture. In *Proc. of the 4th Int. Conf. on Artif. Intell. and Law*, pages 292–297. ACM, 1993.

T. Bench-Capon. Representation of case law as an argumentation framework. In *Proc. of the 15th Conf. on Legal Knowledge and Information Systems*, pages 103–112. IOS Press, 2002.

T. Bench-Capon and P. Dunne. Argumentation in artificial intelligence. *Artif. Intell.*, 171(10):619 – 641, 2007.

T. Bench-Capon and G. Sartor. A model of legal reasoning with cases incorporating theories and values. *Artif. Intell.*, 150(1-2):97–143, 2003.

T. Bench-Capon, F. Coenen, and P. Leng. An experiment in discovering association rules in the legal domain. In *Proc. of the 11th Int. Workshop on Database and Expert Systems Applications*, pages 1056–. IEEE Computer Society, 2000.

P. Besnard and A. Hunter. Constructing argument graphs with deductive arguments: a tutorial. *Argument & Computation*, 5(1):5–30, 2014.

P. Besnard, A. Garcia, A. Hunter, S. Modgil, H. Prakken, G. Simari, and F. Toni. Introduction to structured argumentation. *Argument & Computation*, 5(1):1–4, 2014.

S. Bistarelli and T. Mantadelis. A possible world view and a normal form for the constellation semantics. In *Proc. of tje 16th Eur. Conf. on Logics in Artificial Intelligence*, pages 58–68. Springer, 2019.

C. Cayrol and M. Lagasquie-Schiex. Bipolarity in argumentation graphs: Towards a better understanding. *Int. J. Approx. Reasoning*, 54(7):876–899, 2013.

F. Cerutti, S. A. Gaggl, M. Thimm, and J. P. Wallner. Foundations of implementations for formal argumentation. *IfCoLog J. of Logics and their Applications*, 4(8):2623–2706, October 2017.

A. Chorley and T. Bench-Capon. Agatha: Using heuristic search to automate the construction of case law theories. *Artif. Intell. and Law*, 13(1):9–51, 2005.

P. Clark and T. Niblett. The CN2 induction algorithm. *Mach. Learn.*, 3(4):261–283, 1989.

A. Cohen, S. Gottifredi, A. J. García, and G. R. Simari. A survey of different approaches to support in argumentation systems. *Knowledge Eng. Review*, 29(5):513–550, 2014.

S. Coste-Marquis, S. Konieczny, J. Mailly, and P. Marquis. A translation-based approach for revision of argumentation frameworks. In *Proc. of 14th Eur. Conf. on Logics in Artificial Intelligence*, pages 397–411. Springer, 2014.

S. Coste-Marquis, S. Konieczny, J.-G. Mailly, and P. Marquis. On the revision of argumentation systems: Minimal change of arguments statuses. In *Proc. of the 14th Int. Conf. on Principles of Knowledge Representation and Reasoning*, pages 72–81. AAAI Press, 2014.

S. Coste-Marquis, S. Konieczny, J. Mailly, and P. Marquis. Extension enforcement in abstract argumentation as an optimization problem. In *Proc. of the 24th Int. Joint Conf. on Artificial Intelligence*, pages 2876–2882, 2015.

D. Doran, S. Schulz, and T. R. Besold. What does explainable AI really mean? A new conceptualization of perspectives. *CoRR*, abs/1710.00794, 2017.

F. Doshi-Velez and B. Kim. Towards a rigorous science of interpretable machine learning. *CoRR*, abs/1702.08608, 2017.

P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.*, 77(2):321–358, 1995.

P. M. Dung and P. M. Thang. Towards (probabilistic) argumentation for jury-based dispute resolution. In *Proceedings of the 3rd Conference on Computational Models of Argument*, pages 171–182. IOS Press, 2010.

P. M. Dung and P. M. Thang. Closure and consistency in logic-associated argumentation. *Artif. Intell. Research.*, 49(1):79–109, 2014.

P. E. Dunne, W. Dvorák, T. Linsbichler, and S. Woltran. Characteristics of multiple viewpoints in abstract argumentation. *Artif. Intell.*, 228:153–178, 2015.

S. K. Dyrkolbotn. How to argue for anything: Enforcing arbitrary sets of labellings using afs. In *Proc. of the 14th Int. Conf. on Principles of Knowledge Representation and Reasoning*. AAAI Press, 2014.

B. Fazzinga, S. Flesca, and F. Furfaro. Computing extensions' probabilities in probabilistic abstract argumentation: Beyond independence. In *Proc. of the 22nd Eur. Conf. on Artificial Intelligence*, volume 285, pages 1588–1589. IOS Press, 2016.

B. Fazzinga, S. Flesca, and F. Furfaro. Probabilistic bipolar abstract argumentation frameworks: complexity results. In *Proc. of the 27th Int. Joint Conf. on Artificial Intelligence*, pages 1803–1809. ijcai.org, 2018.

B. Fazzinga, S. Flesca, and F. Furfaro. Complexity of fundamental problems in probabilistic abstract argumentation: Beyond independence. *Artif. Intell.*, 268:1–29, 2019.

A. A. Freitas. Comprehensible classification models: A position paper. *SIGKDD Explor. Newsl.*, 15(1): 1–10, 2014.

J. Fürnkranz, D. Gamberger, and N. Lavrač. *Foundations of Rule Learning*. Springer, 2014.

M. Gendreau and J.-Y. Potvin. *Handbook of Metaheuristics*. Springer, 2nd edition, 2010.

L. Getoor and B. Taskar. *Introduction to Statistical Relational Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2007.

L. Getoor, N. Friedman, D. Koller, A. Pfeffer, and B. Taskar. Probabilistic relational models. In *An Introduction to Statistical Relational Learning*. MIT Press, 2007.

C. G. Hempel and P. Oppenheim. Studies in the logic of explanation. *Philosophy of Science*, 15(2): 135–175, 1948.

A. Hunter. A probabilistic approach to modelling uncertain logical arguments. *Int. J. of Approximate*

Reasoning, 54(1):47 – 81, 2013.

A. Hunter and M. Thimm. Probabilistic reasoning with abstract argumentation frameworks. *J. Artif. Intell. Res.*, 59:565–611, 2017.

B. Johnston and G. Governatori. Induction of Defeasible Logic theories in the legal domain. In *Proc. of the 9th Int. Conf. on Artificial Intelligence and Law*, pages 204–213. ACM, 2003.

F. Keil. Explanation and understanding. *Annual review of psychology*, 57:227–254, 2005.

D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. The MIT Press, 2009.

R. E. Korf. Depth-first iterative-deepening: An optimal admissible tree search. *Artif. Intell.*, 27(1): 97–109, 1985.

H. Lam, G. Governatori, and R. Riveret. On ASPIC^{+} and Defeasible Logic. In *Proc. of the 6th Conf. on Computational Models of Argument*, pages 359–370. IOS Press, 2016.

J. Lawrence and C. Reed. Argument mining: A survey. *Computational Linguistics*, 45(4):765–818, 2020.

H. Li, N. Oren, and T. J. Norman. Probabilistic argumentation frameworks. In *Proc. of the 1st Int. Conf. on Theory and Applications of Formal Argumentation*, pages 1–16. Springer-Verlag, 2012.

T. Linsbichler, J. Pührer, and H. Strass. A uniform account of realizability in abstract argumentation. In *Proc. of the 22nd Eur. Conf. on Artificial Intelligence*, pages 252–260. IOS Press, 2016.

M. Lippi and P. Torroni. Argumentation mining: State of the art and emerging trends. *ACM Trans. Internet Technol.*, 16(2):10:1–10:25, 2016.

P. Lipton. *What Good is an Explanation?*, pages 43–59. Springer, 2001.

Z. C. Lipton. The mythos of model interpretability. *CoRR*, abs/1606.03490, 2016.

T. Lombrozol. The structure and function of explanations. *Trends in Cognitive Sciences*, 10(10): 464–470, 2006.

M. J. G. Lucero, C. I. Chesñevar, and G. R. Simari. On the accrual of arguments in defeasible logic programming. In *Proc. of the 21st Int. Joint Conf. on Artificial Intelligence*, pages 804–809, 2009.

L. T. McCarty. An implementation of Eisner v. Macomber. In *Proc. of the 5th Int. Conf. on Artif. Intell. and Law*, pages 276–286. ACM, 1995.

S. Modgil and H. Prakken. The *ASPIC*$^{+}$ framework for structured argumentation: a tutorial. *Argument & Computation*, 5(1):31–62, 2014.

M. Možina, J. Žabkar, T. Bench-Capon, and I. Bratko. Argument based machine learning applied to law. *Artif. Intell. and Law*, 13(1):53–73, 2005.

M. Mozina, J. Zabkar, and I. Bratko. Argument based machine learning. *Artif. Intell.*, 171(10-15): 922–937, 2007.

A. Niskanen, J. P. Wallner, and M. Jarvisalo. Synthesizing argumentation frameworks from examples. In *Proc. of the 22nd Eur. Conf. on Artif. Intell.*, pages 551–559. IOS Press, 2016.

A. Niskanen, J. P. Wallner, and M. Jarvisalo. Synthesizing argumentation frameworks from examples. *J. of Artificial Intelligence Research*, pages 503 – 554, 2019.

S. Ontañón, P. Dellunde, L. Godo, and E. Plaza. A defeasible reasoning model of inductive concept learning from examples and communication. *Artif. Intell.*, 193:129–148, Dec. 2012.

S. Polberg and A. Hunter. Empirical evaluation of abstract argumentation: Supporting the need for bipolar and probabilistic approaches. *Int. J. of Approximate Reasoning*, 93:487 – 543, 2018.

H. Prakken. A study of accrual of arguments, with applications to evidential reasoning. In *Proc. of the 10th Int. Conf. on Artificial Intelligence and Law*, pages 85–94. ACM, 2005.

H. Prakken. On support relations in abstract argumentation as abstractions of inferential relations. In *Proc, of the 21st Eur Conf. on Artif. Intell.*, pages 735–740. IOS Press, 2014.

H. Prakken, A. Wyner, T. Bench-Capon, and K. Atkinson. A formalization of argumentation schemes for legal case-based reasoning in ASPIC+. *Logic and Computation*, 25(5):1141–1166, 2015.

J. Pührer. Realizability of three-valued semantics for abstract dialectical frameworks. In *Proc. of the 24th Int. Conf. on Artificial Intelligence*, pages 3171–3177. AAAI Press, 2015.

Régis, A. Rotolo, and G. Sartor. Probabilistic rule-based argumentation for norm-governed learning agents. *Artif. Intell. Law*, 20(4):383–420, 2012.

M. Richardson and P. Domingos. Markov logic networks. *Machine Learning*, 62(1-2):107–136, 2006.

E. L. Rissland and D. B. Skalak. Cabaret: Rule interpretation in a hybrid architecture. *Man-Mach.*

Stud., 34(6):839–887, June 1991. ISSN 0020-7373.

E. L. Rissland, E. M. Valcarce, and K. D. Ashley. Explaining and arguing with examples. In *Proc. of the 4th AAAI Conf. on Artif. Intell.*, pages 288–294. AAAI Press, 1984.

R. Riveret. On learning abstract argumentation graphs from bivalent statement labellings. In *Proc. of the 17th Int. Conf. on Tools with Artif. Intell.*, pages 190–195. IEEE Computer Society, 2016.

R. Riveret and G. Governatori. On learning attacks in probabilistic abstract argumentation. In *Proc. of the 15th Int. Conf. on Autonomous Agents & Multiagent Systems*, pages 653–661. IFAAMAS, 2016.

R. Riveret, A. Rotolo, G. Sartor, H. Prakken, and B. Roth. Success chances in argument games: a probabilistic approach to legal disputes. In *Proc. of the 20th Conf. on Legal Knowledge and Information Systems*, pages 99–108. IOS Press, 2007.

R. Riveret, H. Prakken, A. Rotolo, and G. Sartor. Heuristics in argumentation: A game theory investigation. In *Proc. of 2nd Conf. on Computational Models of Argument*, pages 324–335. IOS Press, 2008.

R. Riveret, D. Korkinof, M. Draief, and J. Pitt. Probabilistic abstract argumentation: an investigation with boltzmann machines. *Argument & Computation*, 6(2):178–218, 2015a.

R. Riveret, J. Pitt, D. Korkinof, and M. Draief. Neuro-symbolic agents: Boltzmann machine and Probabilistic Abstract Argumentation with Sub-arguments. In *Proc. of the 14th Int. Joint Conf. on Autonomous Agents & Multiagent Systems*, pages 1481–1489. IFAAMAS, 2015b.

R. Riveret, P. Baroni, Y. Gao, G. Governatori, A. Rotolo, and G. Sartor. A labelling framework for probabilistic argumentation. *Annals of Mathematics and Artificial Intelligence*, 2018.

F. Toni. A tutorial on assumption-based argumentation. *Argument & Computation*, 5(1):89–117, 2014.

K. Čyras, K. Satoh, and F. Toni. Abstract argumentation for case-based reasoning. In *Proc. of the 15th Int. Conf. on Principles of Knowledge Representation and Reasoning*, pages 549–552. AAAI Press, 2016.

B. Verheij. Proof with and without probabilities - correct evidential reasoning with presumptive arguments, coherent hypotheses and degrees of uncertainty. *Artif. Intell. Law*, 25(1):127–154, 2017a.

B. Verheij. Formalizing arguments, rules and cases. In *Proc. of the 16th Int. Conf. on Artificial Intelligence and Law*, pages 199–208. ACM, 2017b.

D. Šešelja and C. Straßer. Abstract argumentation and explanation applied to scientific debates. *Synthese*, 190(12):2195–2217, 2013.