Real Arguments are Approximate Arguments

Anthony Hunter
Department of Computer Science
University College London
Gower Street, London WC1E 6BT, UK

Abstract

There are a number of frameworks for modelling argumentation in logic. They incorporate a formal representation of individual arguments and techniques for comparing conflicting arguments. A common assumption for logic-based argumentation is that an argument is a pair $\langle \Phi, \alpha \rangle$ where Φ is minimal subset of the knowledgebase such that Φ is consistent and Φ entails the claim α. However, real arguments (i.e. arguments presented by humans) usually do not have enough explicitly presented premises for the entailment of the claim. This is because there is some common knowledge that can be assumed by a proponent of an argument and the recipient of it. This allows the proponent of an argument to encode an argument into a real argument by ignoring the common knowledge, and it allows a recipient of a real argument to decode it into an argument by drawing on the common knowledge. If both the proponent and recipient use the same common knowledge, then this process is straightforward. Unfortunately, this is not always the case, and raises the need for an approximation of the notion of an argument for the recipient to cope with the disparities between the different views on what constitutes common knowledge.

Introduction

Argumentation is a vital aspect of intelligent behaviour by humans. Consider diverse professionals such as politicians, journalists, clinicians, scientists, and administrators, who all need to collate and analyse information looking for pros and cons for consequences of importance when attempting to understand problems and make decisions.

There are a number of proposals for logic-based formalisations of argumentation (for reviews see (Chesnevar, Maguitman, & Loui 2000; Prakken & Vreeswijk 2002)). These proposals allow for the representation of arguments for and against some claim, and for attack relationships between arguments. In a number of key examples of argumentation systems, an argument is a pair where the first item in the pair is a minimal consistent set of formulae that proves the second item which is a formula. Furthermore, in these approaches, a key form of counterargument is an undercut: One argument undercuts another argument when the claim of the first argument negates the premises of the second argument.

Unfortunately, real arguments do not normally fit this mould. Real arguments (i.e. those presented by people in general) are normally enthymemes (Walton 1989). An enthymeme only explicitly represents some of the premises for entailing its claim. So if Γ is the set of premises explicitly given for an enthymeme, and α is the claim, then Γ does not entail α, but there are some implicitly assumable premises Γ' such that $\Gamma \cup \Gamma'$ is a minimal consistent set of formulae that entails α.

For example, for a claim that *you need an umbrella today*, a husband may give his wife the premise *the weather report predicts rain*. Clearly, the premise does not entail the claim, but it is easy for the wife to identify the common knowledge used by the husband in order to reconstruct the intended argument correctly.

Whilst humans are constantly handling examples like this, the logical formalization that characterizes the process remains underdeveloped. Therefore, we need to investigate enthymemes because of their ubiquity in the real world, and because of the difficulties they raise for formalizing and automating argumentation. If we want to build agents that can understand real arguments coming from humans, they need to identify the missing premises with some reliability. And if we want to build agents that can generate real arguments for humans, they need to identify the premises that can be missed without causing undue confusion.

In this paper, we present real arguments as approximate arguments. For this, we start with an existing framework for argumentation based on classical logic, and extend it into a framework for approximate arguments. We will represent each real argument as an approximate argument. Then by using common knowledge we show how real arguments can be encoded by a proponent for consignment to a recipient, and how they can be decoded by a recipient. For this, a proponent of a real argument can miss premises from the intended argument that it preceives to be common knowledge, and a recipient of a real argument can aim to identify the missing premises for the intended argument from what it perceives to be common knowledge.

Logical Argumentation

In this section, we review an existing proposal for logic-based argumentation (Besnard & Hunter 2001). We consider a classical propositional language $\mathcal{L}$ with classical deduction denoted by the symbol $\vdash$. We use $\alpha, \beta, \gamma, \ldots$ to denote formulae and $\Delta, \Phi, \Psi, \ldots$ to denote sets of formulae.

For the following definitions, we first assume a knowledgebase Δ (a finite set of formulae) and use this Δ throughout. We further assume that every subset of Δ is given an enumeration $\langle \alpha_1, \ldots, \alpha_n \rangle$ of its elements, which we call its canonical enumeration. This really is not a demanding constraint: In particular, the constraint is satisfied whenever we impose an arbitrary total ordering over Δ. Importantly, the order has no meaning and is not meant to represent any respective importance of formulae in Δ. It is only a convenient way to indicate the order in which we assume the formulae in any subset of Δ are conjoined to make a formula logically equivalent to that subset.

The paradigm for the approach is a large repository of information, represented by Δ, from which arguments can be constructed for and against arbitrary claims. Apart from information being understood as declarative statements, there is no *a priori* restriction on the contents, and the pieces of information in the repository can be arbitrarily complex. Therefore, Δ is not expected to be consistent. It need not even be the case that every single formula in Δ is consistent.

The framework adopts a very common intuitive notion of an argument. Essentially, an argument is a set of relevant formulae that can be used to classically prove some claim, together with that claim. Each claim is represented by a formula.

Definition 1. *An* **argument** *is a pair $\langle \Phi, \alpha \rangle$ such that: (1) $\Phi \subseteq \Delta$; (2) $\Phi \not\vdash \bot$; (3) $\Phi \vdash \alpha$; and (4) there is no $\Phi' \subset \Phi$ such that $\Phi' \vdash \alpha$. We say that $\langle \Phi, \alpha \rangle$ is an argument for α. We call α the* **claim** *of the argument and Φ the* **support** *of the argument (we also say that Φ is a support for α).*

Example 1. *Let $\Delta = \{\alpha, \alpha \to \beta, \gamma \to \neg\beta, \gamma, \delta, \delta \to \beta, \neg\alpha, \neg\gamma\}$. Some arguments are:*

$$\langle \{\alpha, \alpha \to \beta\}, \beta \rangle$$
$$\langle \{\neg\alpha\}, \neg\alpha \rangle$$
$$\langle \{\alpha \to \beta\}, \neg\alpha \vee \beta \rangle$$
$$\langle \{\neg\gamma\}, \delta \to \neg\gamma \rangle$$

Arguments are not independent. In a sense, some encompass others (possibly up to some form of equivalence). To clarify this requires a few definitions as follows.

Definition 2. *An argument $\langle \Phi, \alpha \rangle$ is* **more conservative** *than an argument $\langle \Psi, \beta \rangle$ iff $\Phi \subseteq \Psi$ and $\beta \vdash \alpha$.*

Example 2. *$\langle \{\alpha\}, \alpha \vee \beta \rangle$ is more conservative than $\langle \{\alpha, \alpha \to \beta\}, \beta \rangle$.*

Some arguments directly oppose the support of others, which amounts to the notion of an undercut.

Definition 3. *An* **undercut** *for an argument $\langle \Phi, \alpha \rangle$ is an argument $\langle \Psi, \neg(\phi_1 \wedge \ldots \wedge \phi_n) \rangle$ where $\{\phi_1, \ldots, \phi_n\} \subseteq \Phi$.*

Example 3. *Let $\Delta = \{\alpha, \alpha \to \beta, \gamma, \gamma \to \neg\alpha\}$. Then, $\langle \{\gamma, \gamma \to \neg\alpha\}, \neg(\alpha \wedge (\alpha \to \beta)) \rangle$ is an undercut for $\langle \{\alpha, \alpha \to \beta\}, \beta \rangle$. A less conservative undercut for $\langle \{\alpha, \alpha \to \beta\}, \beta \rangle$ is $\langle \{\gamma, \gamma \to \neg\alpha\}, \neg\alpha \rangle$.*

Definition 4. *$\langle \Psi, \beta \rangle$ is a* **maximally conservative undercut** *of $\langle \Phi, \alpha \rangle$ iff $\langle \Psi, \beta \rangle$ is an undercut of $\langle \Phi, \alpha \rangle$ such that no undercuts of $\langle \Phi, \alpha \rangle$ are strictly more conservative than $\langle \Psi, \beta \rangle$ (that is, for all undercuts $\langle \Psi', \beta' \rangle$ of $\langle \Phi, \alpha \rangle$, if $\Psi' \subseteq \Psi$ and $\beta \vdash \beta'$ then $\Psi \subseteq \Psi'$ and $\beta' \vdash \beta$).*

The value of the following definition of canonical undercut is that we only need to take the canonical undercuts into account. This means we can justifiably ignore the potentially very large number of non-canonical undercuts.

Definition 5. *An argument $\langle \Psi, \neg(\phi_1 \wedge \ldots \wedge \phi_n) \rangle$ is a* **canonical undercut** *for $\langle \Phi, \alpha \rangle$ iff it is a maximally conservative undercut for $\langle \Phi, \alpha \rangle$ and $\langle \phi_1, \ldots, \phi_n \rangle$ is the canonical enumeration of Φ.*

An argument tree describes the various ways an argument can be challenged, as well as how the counter-arguments to the initial argument can themselves be challenged, and so on recursively.

Definition 6. *An* **argument tree** *for α is a tree where the nodes are arguments such that*

1. *The root is an argument for α.*
2. *For no node $\langle \Phi, \beta \rangle$ with ancestor nodes $\langle \Phi_1, \beta_1 \rangle, \ldots, \langle \Phi_n, \beta_n \rangle$ is Φ a subset of $\Phi_1 \cup \cdots \cup \Phi_n$.*
3. *Each child node of a node N is a canonical undercut for N that obeys 2.*

The second condition in Definition 6 ensures that each argument on a branch has to introduce at least one formula in its support that has not already been used by ancestor arguments. This is meant to avoid making explicit undercuts that simply repeat over and over the same reasoning pattern except for switching the role of some formulae (e.g. in mutual exclusion, stating that α together with $\neg\alpha \vee \neg\beta$ entails $\neg\beta$ is exactly the same reasoning as expressing that β together with $\neg\alpha \vee \neg\beta$ entail $\neg\alpha$, because in both cases, what is meant is that α and β exclude each other).

Example 4. *Let $\Delta = \{\alpha \vee \beta, \alpha \to \gamma, \neg\gamma, \neg\beta, \delta \leftrightarrow \beta\}$. For this, an argument tree for the consequent $\alpha \vee \neg\delta$ is given.*

$$\begin{array}{c} \langle \{\alpha \vee \beta, \neg\beta\}, \alpha \vee \neg\delta \rangle \\ \uparrow \\ \langle \{\alpha \to \gamma, \neg\gamma\}, \neg((\alpha \vee \beta) \wedge \neg\beta) \rangle \end{array}$$

A complete argument tree (i.e. an argument tree with all the canonical undercuts for each node as children of that node) provides an efficient representation of the arguments and counterarguments. Furthermore, if Δ is finite, there is a finite number of argument trees with the root being an argument with consequent α that can be formed from Δ, and each of these trees has finite branching and a finite depth (the finite tree property). Note, also the definitions presented in this section can be used directly with first-order classical logic, so Δ and α are from the first-order classical language. Interestingly, the finite tree property also holds for the first-order case (Besnard & Hunter 2005).

Approximate Arguments

An **approximate argument** is a pair $\langle \Phi, \alpha \rangle$ where $\Phi \subseteq \mathcal{L}$ and $\alpha \in \mathcal{L}$. This is a very general definition. It does not assume that Φ is consistent, or that it even entails α.

For an approximate argument $\langle \Phi, \alpha \rangle$, let $\mathsf{Support}(\langle \Phi, \alpha \rangle)$ be Φ, and let $\mathsf{Claim}(\langle \Phi, \alpha \rangle)$ be α. For a set of approximate arguments Λ, $\mathsf{Args}(\Lambda) = \{A \in \Lambda \mid A$ is an argument$\}$.

In this paper, we restrict consideration to particular kinds of approximate arguments that relax the definition of an argument: If $\Phi \vdash \alpha$, then $\langle \Phi, \alpha \rangle$ is **valid**; If $\Phi \not\vdash \bot$, then $\langle \Phi, \alpha \rangle$ is **consistent**; If $\Phi \vdash \alpha$, and there is no $\Phi' \subset \Phi$ such that $\Phi' \vdash \alpha$, then $\langle \Phi, \alpha \rangle$ is **minimal**; And if $\Phi \vdash \alpha$, and $\Phi \not\vdash \bot$, then $\langle \Phi, \alpha \rangle$ is **expansive** (i.e. it is valid and consistent, but it may have unnecessary premises).

In addition, we require a further kind of approximate argument that has the potential to be transformed into an argument: If $\Phi \not\vdash \alpha$, and $\Phi \not\vdash \neg\alpha$, then $\langle \Phi, \alpha \rangle$ is a **precursor** (i.e. it is a precursor for an argument). Therefore, if $\langle \Phi, \alpha \rangle$ is a precursor, then there exists some $\Psi \subset \mathcal{L}$ such that $\Phi \cup \Psi \vdash \alpha$ and $\Phi \cup \Psi \not\vdash \bot$, and hence $\langle \Phi \cup \Psi, \alpha \rangle$ is expansive.

Example 5. *Let $\Delta = \{\alpha, \neg\alpha \vee \beta, \gamma, \neg\beta, \beta, \neg\gamma, \neg\beta \vee \gamma\}$. Some approximate arguments from Δ that are valid include $\{A_1, A_2, A_3, A_4, A_5\}$ of which $\{A_1, A_3, A_5\}$ are expansive, $\{A_2, A_5\}$ are minimal, and A_5 is an argument. Also, some approximate arguments that are not valid include $\{A_6, A_7\}$ of which A_6 is a precursor.*

$$\begin{array}{l} A_1 = \langle \{\alpha, \neg\alpha \vee \beta, \gamma, \beta\}, \beta \rangle \\ A_2 = \langle \{\gamma, \neg\gamma\}, \beta \rangle \\ A_3 = \langle \{\alpha, \neg\alpha \vee \beta, \gamma\}, \beta \rangle \\ A_4 = \langle \{\alpha, \neg\alpha \vee \beta, \gamma, \neg\gamma\}, \beta \rangle \\ A_5 = \langle \{\alpha, \neg\alpha \vee \beta\}, \beta \rangle \\ A_6 = \langle \{\neg\alpha \vee \beta\}, \beta \rangle \\ A_7 = \langle \{\neg\alpha \vee \beta, \neg\beta \vee \gamma, \neg\gamma\}, \beta \rangle \end{array}$$

Some observations that we can make concerning approximate arguments include: (1) If $\langle \Gamma, \alpha \rangle$ is expansive, then there is a $\Phi \subseteq \Gamma$ such that $\langle \Phi, \alpha \rangle$ is an argument; (2) If $\langle \Phi, \alpha \rangle$ is minimal, and $\langle \Phi, \alpha \rangle$ is expansive, then $\langle \Phi, \alpha \rangle$ is an argument; (3) If $\langle \Phi, \alpha \rangle$ is an argument, and $\Psi \subset \Phi$, then $\langle \Psi, \alpha \rangle$ is a precursor; and (4) If $\langle \Gamma, \alpha \rangle$ is a precursor, then $\langle \Gamma, \alpha \rangle$ is consistent.

Framework for Real Arguments

Given an argument, an enthymeme is simply a precursor that can be generated from it.

Definition 7. *Let $\langle \Phi, \alpha \rangle$ be a precursor and $\langle \Psi, \alpha \rangle$ be an argument. $\langle \Phi, \alpha \rangle$ is an* **enthymeme** *for $\langle \Psi, \alpha \rangle$ iff $\Phi \subset \Psi$.*

So if a proponent has an argument that it wishes a recipient to be aware of, (we refer to this argument as the **intended argument**), then the proponent may send an enthymeme instead of the intended argument to the recipient. We refer to whatever the proponent sends to the recipient (whether the intended argument or an enthymeme for that intended argument) as the **real argument**.

Example 6. *Let α be "you need an umbrella today", and β be "the weather report predicts rain". So for an intended argument $\langle \{\beta, \beta \rightarrow \alpha\}, \alpha \rangle$, the real argument sent by the proponent to the recipient may be $\langle \{\beta\}, \alpha \rangle$*

We can see the use of enthymemes both in monological argumentation, for example by a politician giving a lecture (as illustrated next) or a journalist writing an article, and in dialogical argumentation, for example lawyers arguing in court, or academics debating in a seminar.

Example 7. *Consider a politician who says "The government will support the expansion of JFK airport with new legislation because it will be good for the local and national economy. And we will address the disturbance to local people with tighter regulations on night time flights and on older more polluting aircraft". This short speech can be analysed as follows: Let α be "The government will support the expansion of JFK airport with new legislation", let β be "the expansion of JFK airport will be good for everyone", let γ be "expansion will improve the local and national economy", let δ be "the local environment will suffer pollution", let ϕ be "there will be tighter regulations on night time flights", and let ψ be "there will be tighter regulations on older more polluting aircraft". So in the first sentence of the speech, the politician effectively gives the enthymeme $\langle \{\gamma\}, \alpha \rangle$, and then in the second sentence, the politician gives the enthymemes $\langle \{\delta \rightarrow \neg\beta\}, \neg\beta \rangle$, and $\langle \{\phi, \psi\}, \neg\delta \rangle$. The intended arguments for each of these enthymemes are as follows.*

$$\begin{array}{l} A_1 = \langle \{\gamma, \gamma \rightarrow \beta, \beta \rightarrow \alpha\}, \alpha \rangle \\ A_2 = \langle \{\delta, \delta \rightarrow \neg\beta\}, \neg\beta \rangle \\ A_3 = \langle \{\phi, \psi, \phi \wedge \psi \rightarrow \neg\delta\}, \neg\delta \rangle \end{array}$$

In general, since there can be more than one real argument that can be generated from an intended argument, a proponent i needs to choose which to send to a recipient j. To facilitate this selection, the proponent consults what it believes is common knowledge for i and j. We assume that each agent i has a knowledgebase Δ_i of knowledge, called a **perbase**, that is its personal knowledgebase, and so if i is a proponent, the support of the intended argument comes from Δ_i. In addition, agent i has a function $\mu_{i,j} : \mathcal{L} \mapsto [0, 1]$, called a **cobase**, that represents what an agent i believes is common knowledge for i and j. For $\alpha \in \mathcal{L}$, the higher the value of $\mu_{i,j}(\alpha)$, the more that i regards α as being common knowledge for i and j. So if $\mu_{i,j}(\alpha) = 0$, then i believes that α is not knowledge in common for i and j, whereas if if $\mu_{i,j}(\alpha) = 1$, then i believes that there is no knowledge as being more in common for i and j than α.

Example 8. *In Ex. 6, with $\beta, \beta \rightarrow \alpha \in \Delta_i$, proponent i could have the cobase $\mu_{i,j}$ where $\mu_{i,j}(\beta \rightarrow \alpha) = 1$ representing that the premise $\beta \rightarrow \alpha$ is superfluous in any real argument consigned by proponent i to recipient j.*

Note, $\mu_{i,j}$ reflects the perception i has of the common knowledge between i and j, and $\mu_{j,i}$ reflects the perception j has of the common knowledge between i and j, and so it is not necessarily the case that $\mu_{i,j} = \mu_{j,i}$. Furthermore, it is not necessarily the case that i regards the common knowledge between i and j as being consistent, and so it is possible, for some α, that $\mu_{i,j}(\alpha) > 0$ and $\mu_{i,j}(\neg\alpha) > 0$.

Now consider an agent i who has an intended argument $\langle \Phi, \alpha \rangle$ that it wants agent j to be aware of. So Φ is a subset of Δ_i, and i is the proponent of the argument and j is the recipient of the argument. By reference to its representation of the common knowledge $\mu_{i,j}$, agent i will remove premises ϕ

from Φ for which $\mu_{i,j}(\phi)$ is greater than a particular threshold τ. The result of this encodation process is either the intended argument or an enthymeme for that argument.

Definition 8. *For an argument $\langle\Phi,\alpha\rangle$, the* **encodation** *of $\langle\Phi,\alpha\rangle$ from a proponent i for a recipient j, denoted $C(\langle\Phi,\alpha\rangle,\mu_{i,j},\tau)$, is the approximate argument $\langle\Psi,\alpha\rangle$, where $\Psi = \{\phi \in \Phi \mid \mu_{i,j}(\phi) \leq \tau\}$ for threshold $\tau \in [0,1]$.*

Example 9. *In Ex. 8, when $\mu_{i,j}(\beta \rightarrow \alpha) = 1$, and $\mu_{i,j}(\beta) = 0.5$, and $\tau = 0.7$, $C(\langle\{\beta,\beta \rightarrow \alpha\},\alpha\rangle,\mu_{i,j},\tau)$ is $\langle\{\beta\},\alpha\rangle$.*

So given a cobase $\mu_{i,j}$, it is simple for a proponent i to obtain an encodation for a recipient j. Note, for an intended argument A, it is possible that $C(A,\mu_{i,j},\tau) = B$ where $\mathsf{Support}(B) = \emptyset$. This raises the question of whether a proponent would want to send a real argument with empty support to another agent since it is in effect "stating the obvious". Nevertheless, there may be a rhetorical or pragmatic motivation for such a real argument. For example when a husband issues a reminder like *don't forget your umbrella* to his wife when the common knowledge includes the facts that the month is April, the city is London, and London has many showers in April. Hence, *don't forget your umbrella* is the claim, and the support for this real argument is empty.

Generalizing Argument Trees

An **annotated tree** is a tree where each node is an approximate argument. This is a generalization of an argument tree. So, if T is an argument tree, then T is an annotated tree. Furthermore, given an argument tree and a cobase, we can obtain an annotated tree composed of real arguments. For this, let $\mathsf{Nodes}(T)$ be the set of nodes in a tree T.

Definition 9. *Let $\mu_{i,j}$ be a cobase, let T be an argument tree, and let T' be an annotated tree. T' is a $\mu_{i,j}$* **abstraction** *of T iff there is a bijection $f : \mathsf{Nodes}(T) \mapsto \mathsf{Nodes}(T')$ s.t. $\forall A \in \mathsf{Nodes}(T)$, $f(A)$ is $C(A,\mu_{i,j},\tau)$.*

Example 10. *Consider the following argument tree T formed from $\Delta_i = \{\beta,\gamma,\beta\wedge\gamma \rightarrow \alpha,\delta,\neg\delta\vee\neg\beta,\neg\beta\vee\neg\gamma,\epsilon,\epsilon\rightarrow\neg\delta\}$, and where ϕ denotes $\beta\wedge\gamma\wedge(\beta\wedge\gamma\rightarrow\alpha)$ and ψ denotes $\delta\wedge(\neg\delta\vee\neg\beta)$.*

$$
\begin{array}{ccc}
 & \langle\{\beta,\gamma,\beta\wedge\gamma\rightarrow\alpha\},\alpha\rangle & \\
 \nearrow & & \nwarrow \\
\langle\{\delta,\neg\delta\vee\neg\beta\},\neg\phi\rangle & & \langle\{\neg\beta\vee\neg\gamma\},\neg\phi\rangle \\
\uparrow & & \\
\langle\{\epsilon,\epsilon\rightarrow\neg\delta\},\neg\psi\rangle & &
\end{array}
$$

Let $\mu_{i,j}$ be such that $\mu_{i,j}(\beta\wedge\gamma\rightarrow\alpha) = 1$, $\mu_{i,j}(\neg\delta\vee\neg\beta) = 1$, $\mu_{i,j}(\epsilon) = 1$, $\mu_{i,j}(\theta) = 0.5$ for all other θ. So we get the following $\mu_{i,j}$ abstraction with $\tau = 0.8$.

$$
\begin{array}{ccc}
 & \langle\{\beta,\gamma\},\alpha\rangle & \\
 \nearrow & & \nwarrow \\
\langle\{\delta\},\neg\phi\rangle & & \langle\{\neg\beta\vee\neg\gamma\},\neg\phi\rangle \\
\uparrow & & \\
\langle\{\epsilon\rightarrow\neg\delta\},\neg\psi\rangle & &
\end{array}
$$

We now consider a widely used criterion in argumentation theory for determing whether the argument (intended or real) at the root of the annotated tree is warranted (adapted from (García & Simari 2004)). For this, each node is marked as either U for **undefeated** or D for **defeated**.

Definition 10. *The* **judge function**, *denoted* Judge, *from the set of annotated trees to $\{\mathsf{Warranted},\mathsf{Unwarranted}\}$ such that $\mathsf{Judge}(T) = \mathsf{Warranted}$ iff $\mathsf{Mark}(A_r) = U$ where A_r is the root node of T. For all $A_i \in \mathsf{Nodes}(T)$, if there is child A_j of A_i such that $\mathsf{Mark}(A_j) = U$, then $\mathsf{Mark}(A_i) = D$, otherwise $\mathsf{Mark}(A_i) = U$.*

So the root is undefeated iff all its children are defeated.

Example 11. *For T in Ex. 10, $\mathsf{Judge}(T) = \mathsf{Unwarranted}$.*

A direct consequence of the definition of abstraction is the following which indicates that by abstraction the overall judgement of the tree remains the same.

Proposition 1. *Let $\mu_{i,j}$ be a cobase, let T be an argument tree, and let T' be an annotated tree. If T' is a $\mu_{i,j}$ abstraction of T, then $\mathsf{Judge}(T) = \mathsf{Judge}(T')$.*

Whilst an argument tree, with the judge function, is useful for analysing arguments, as we discuss next, abstraction is not always an ideal way of analysing real arguments.

Sequences of Real Arguments

Real arguments do not occur in isolation. Normally, there is some sequence of them (including counterarguments) presented by a proponent. Consider for example a speech by a politician or an article by a journalist. Furthermore, as we see in the next example, such a sequence is not necessarily an abstraction of an argument tree.

Example 12. *In Ex 7, the sequence of real arguments $A_4 = \langle\{\gamma\},\alpha\rangle$, $A_5 = \langle\{\delta\rightarrow\neg\beta\},\neg\beta\rangle$ and $A_6 = \langle\{\phi,\psi\},\neg\delta\rangle$, can be composed into an annotated tree T' below.*

$$
\begin{array}{c}
\langle\{\gamma\},\alpha\rangle \\
\uparrow \\
\langle\{\delta\rightarrow\neg\beta\},\neg\beta\rangle \\
\uparrow \\
\langle\{\phi,\psi\},\neg\delta\rangle
\end{array}
$$

However, this annotated tree involving $A_4,..,A_6$ is not an abstraction of an argument tree because the claim of each child is not of the right form: It is not the negation of the support of its parent. In other words, there is no argument tree T and no $\mu_{i,j}$ such that T' is a $\mu_{i,j}$ abstraction of T. Furthermore, the intended arguments $A_1,..,A_3$ (as given in Ex 7), are such that none is a canonical undercut of any other, and so cannot be used together in an argument tree.

So we need a way for a proponent to send a sequence of real arguments, without them being constrained to be an abstraction of an argument tree. To faciliate this, we use the following notion of a realization.

Definition 11. *Let $\mu_{i,j}$ be a cobase, let T be an argument tree, and let T' be an annotated tree. T' is a $\mu_{i,j}$* **realization** *of T iff there is a bijection $f : \mathsf{Nodes}(T) \mapsto \mathsf{Nodes}(T')$ s.t. $\forall A \in \mathsf{Nodes}(T)$, $\mathsf{Support}(f(A))$ is $\mathsf{Support}(C(A,\mu_{i,j},\tau))$ and $\mathsf{Claim}(f(A)) \vdash \mathsf{Claim}(A)$.*

If T' is an abstraction of T, then T' is a realization of T.

Example 13. *Consider the following arguments which give an argument tree T, where A_7 is the root, A_8 is the child of A_7, and A_9 is the child of A_8.*

$$A_7 = \langle \{\gamma, \gamma \to \beta, \beta \to \alpha\}, \alpha \rangle$$
$$A_8 = \langle \{\delta, \delta \to \neg\beta\}, \neg(\gamma \wedge (\gamma \to \beta) \wedge (\beta \to \alpha)) \rangle$$
$$A_9 = \langle \{\phi, \psi, \phi \wedge \psi \to \neg\delta\}, \neg(\delta \wedge (\delta \to \neg\beta)) \rangle$$

So for the real arguments given in Ex 7, $A_4 = \langle \{\gamma\}, \alpha \rangle$), $A_5 = \langle \{\delta \to \neg\beta\}, \neg\beta \rangle$), and $A_6 = \langle \{\phi, \psi\}, \neg\delta \rangle$), let $f(A_7) = A_4$, $f(A_8) = A_5$, and $f(A_9) = A_6$. So T' as given in Example 12 is a realization of T.

So we envisage that given a perbase Δ_i, a proponent may construct an argument tree T, and then from this construct an annotated tree of real arguments T' such that T' is a realization of T. Indeed, implicity, it is a two step process: From the argument tree T, each argument is turned into an intended argument (by strengthening the claim perhaps by negating just one of the formulae in the support of the argument being undercut), and then each intended argument is turned into a real argument (by simplifying the support). The tree structure may then be sent by the proponent with the real arguments to the recipient. So if we consider Ex. 13, from the argument tree T involving A_7, A_8, and A_9, we form an isomorphic annotated tree with the intended arguments A_1, A_2, and A_3 (of Ex. 7), and then from this we form the isomorphic tree T' (as in Ex. 12) with the real arguments A_4, A_5, and A_6.

Starting with a complete argument tree gives the discipline for ensuring that all the canonical undercuts are considered. Then, using real arguments in a realization offers an intuitive simplification of the argument tree, with the advantage of the evaluation via the judge function remaining the same.

Proposition 2. *Let $\mu_{i,j}$ be a cobase, let T be an argument tree, and let T' be an annotated tree. If T' is a $\mu_{i,j}$ realization of T, then $\mathsf{Judge}(T) = \mathsf{Judge}(T')$.*

Finally, as a by-product of sending a set of real arguments, the agents can augment their common knowledge. For instance, if agent i sends the real argument $\langle \Phi, \alpha \rangle$ to agent j, then Φ can be used to update $\mu_{i,j}$ and $\mu_{j,i}$. Indeed, we could refine the above framework to allow the common knowledge to grow with each real argument sent.

Decoding Enthymemes

When $\langle \Psi, \alpha \rangle$ is an encodation of $\langle \Phi, \alpha \rangle$, it is either the intended argument or an enthymeme for the intended argument. If it is an enthymeme, then the recipient has to decode it using the common knowledge $\mu_{j,i}$ (i.e. the knowledge that j believes is common knowledge between i and j) by adding formulae Ψ' to the support of the enthymeme, creating $\langle \Psi \cup \Psi', \alpha \rangle$, which will be expansive but not necessarily minimal. It would be desirable for $\langle \Psi \cup \Psi', \alpha \rangle$ to be the intended argument, but this cannot be guaranteed. It may be that the wrong formulae from $\mu_{j,i}$ are used, or it could be that common knowledge as viewed by agent i is not the same as that viewed by agent j (i.e. $\mu_{i,j} \neq \mu_{j,i}$). Nevertheless, using the ranking information in a cobase, we can aim for a reasonable decoding of an enthymeme. For this, we use the following ordering over $\wp(\mathcal{L})$ (adapted from (Cayrol, Royer, & Saurel 1993)) which is just one of a number of possible definitions for ranking $\wp(\mathcal{L})$ given a cobase.

Definition 12. *Let Φ and Ψ be two non-empty subsets of $\mathcal{L}$. Φ is preferred to Ψ, denoted $\Phi >_{i,j} \Psi$ iff for all $\phi \in \Phi \setminus \Psi$, there is a $\psi \in \Psi \setminus \Phi$ s.t. $\mu_{i,j}(\phi) > \mu_{i,j}(\psi)$. For all non-empty subsets Φ of $\mathcal{L}$, $\emptyset >_{i,j} \Phi$.*

Definition 13. *For an encodation $\langle \Psi, \alpha \rangle$ from a proponent i for a recipient j, a* **decodation** *is of the form $\langle \Psi \cup \Psi', \alpha \rangle$, where $\Psi' \subseteq \mathcal{L}$, and $\langle \Psi \cup \Psi', \alpha \rangle$ is expansive, and there is no Ψ'' such that $\Psi'' >_{j,i} \Psi'$ and $\langle \Psi \cup \Psi'', \alpha \rangle$ is expansive. Let $D(\langle \Psi, \alpha \rangle, \mu_{j,i})$ denote the set of decodations of $\langle \Psi, \alpha \rangle$.*

Example 14. *If $\langle \{\alpha, \alpha \to \beta\}, \beta \rangle$ is an intended argument from proponent i to recipient j, where $\mu_{i,j}(\alpha) = 0$, $\mu_{i,j}(\alpha \to \beta) = 1$, and $\tau = 0.9$, then the encodation is $\langle \{\alpha\}, \beta \rangle$. Now suppose, $\mu_{j,i}(\alpha \to \beta) = 1$, $\mu_{j,i}(\alpha \to \epsilon) = 1$, and $\mu_{j,i}(\epsilon \to \beta) = 1$, and for all other ϕ, $\mu_{j,i}(\phi) = 0$. So for $\langle \{\alpha\}, \beta \rangle$, the decodations are $\langle \{\alpha, \alpha \to \beta\}, \beta \rangle$ and $\langle \{\alpha, \alpha \to \epsilon, \epsilon \to \beta\}, \beta \rangle$. If we change the cobase so that $\mu_{j,i}(\alpha \to \beta) = 0.5$, then we get the second decodation as the unique decodation.*

Example 15. *If $\langle \{\beta, \gamma, \beta \wedge \gamma \to \alpha\}, \alpha \rangle$ is an intended argument from proponent i to recipient j, where $\mu_{i,j}(\beta) = \mu_{i,j}(\gamma) = 0$, $\mu_{i,j}(\beta \wedge \gamma \to \alpha) = 1$, and $\tau = 0.9$, then the encodation is $\langle \{\beta, \gamma\}, \alpha \rangle$. Now suppose, $\mu_{j,i}(\beta \wedge \gamma \to \alpha) = 0.5$, and $\mu_{j,i}(\beta \to \alpha) = 0.9$, and for all other ϕ, $\mu_{j,i}(\phi) = 0$. So for $\langle \{\beta\}, \alpha \rangle$, the decodation is $\langle \{\beta, \gamma, \beta \to \alpha\}, \alpha \rangle$.*

Proposition 3. *If $\langle \Psi, \alpha \rangle$ is a real argument, then $|D(\langle \Psi, \alpha \rangle, \mu_{j,i})| \geq 1$ and $|\mathsf{Args}(D(\langle \Psi, \alpha \rangle, \mu_{j,i}))| \geq 0$.*

So when a recipient decodes an enthymeme, it does not know for certain what the intended argument is, and it is not guaranteed to find it even if $\mathsf{Args}(D(\langle \Psi, \alpha \rangle, \mu_{j,i})) = 1$. However, if the proponent and recipient have identical common knowledge, then the intended argument is one of the decodations.

Proposition 4. *Let $\mu_{i,j} = \mu_{j,i}$, and for all $\phi \in \mathcal{L}$, $\mu_{i,j}(\phi) = 1$ or $\mu_{i,j}(\phi) = 0$. For $\langle \Phi, \alpha \rangle$, if $C(\langle \Phi, \alpha \rangle, \mu_{i,j}, \tau) = \langle \Psi, \alpha \rangle$, then $\langle \Phi, \alpha \rangle \in D(\langle \Psi, \alpha \rangle, \mu_{j,i})$.*

If there is a unique decodation that is an argument, and a high confidence that $\mu_{i,j} = \mu_{j,i}$, then the recipient may have high confidence that the decodation is the same as the intended argument. Furthermore, if the real argument is an argument, then the decodation is unique and correct.

Proposition 5. *For any $\langle \Phi, \alpha \rangle$, if $\langle \Phi, \alpha \rangle$ is an argument, then $D(\langle \Phi, \alpha \rangle, \mu_{j,i}) = \{\langle \Phi, \alpha \rangle\}$.*

No decodation has a support that is a subset of any other.

Proposition 6. *If $\langle \Phi_1, \alpha \rangle \in D(\langle \Psi, \alpha \rangle, \mu_{j,i})$, and $\langle \Phi_2, \alpha \rangle \in D(\langle \Psi, \alpha \rangle, \mu_{j,i})$, then $\Phi_1 \not\subset \Phi_2$.*

If a sequence of real arguments are given with tree structure (i.e. a $\mu_{i,j}$ realization), the constraint that the intended argument for a child has to undercut the intended argument for its parent, can substantially simplify the choice of which decodations to use.

Quality of Enthymemes

When a recipient gets a real argument, it seeks a decodation of it. The decodation may be a **mutation** of the intended argument, by which we mean some of the support for the decodation may be different from the intended argument. So as part of the process of encodation and decodation, some of the premises may have changed.

To quantify mutation, we can compare the supports of an intended argument and its decodation. Since, we are comparing a pair of theories, we can harness a model-theoretic way to compare them. Each interpretation is represented by a subset of the atoms of the language. For a set of formulae V, let $\mathsf{M}(V)$ be the set of models of V.

Example 16. *For the atoms* $\{\alpha, \beta, \gamma\}$*, let* $X = \{\alpha\}$ *and let* $Y = \{\alpha \wedge \beta \wedge \gamma\}$*. So* $\mathsf{M}(X) = \{ \{\alpha, \beta, \gamma\}, \{\alpha, \beta\}, \{\alpha, \gamma\}, \{\alpha\} \}$ *and* $\mathsf{M}(Y) = \{ \{\alpha, \beta, \gamma\} \}$*.*

The degree of entailment of X for Y is the number of models in common divided by the number of models for X.

Definition 14. *Let X and Y be sets of classical propositional formulae each of which is consistent (i.e. $X \not\vdash \bot$ and $Y \not\vdash \bot$). The* **degree of entailment** *of X for Y, denoted $\mathsf{E}(X, Y)$, is defined as follows:*

$$\mathsf{E}(X, Y) = \frac{|\mathsf{M}(X \cup Y)|}{|\mathsf{M}(X)|}$$

To simplify the example, the brackets have been dropped.

Example 17. $\mathsf{E}(\alpha, \alpha \wedge \beta) = 1/2$, $\mathsf{E}(\alpha, \alpha \wedge \beta \wedge \gamma) = 1/4$, $\mathsf{E}(\alpha, \alpha \wedge \beta \wedge \gamma \wedge \delta) = 1/8$, $\mathsf{E}(\alpha \wedge \beta, \alpha \vee \beta) = 1$, $\mathsf{E}(\alpha \wedge \beta, \alpha \wedge \epsilon) = 1/2$, $\mathsf{E}(\alpha \wedge \beta \wedge \gamma, \alpha \wedge \epsilon) = 1/2$, $\mathsf{E}(\alpha \wedge \beta \wedge \gamma \wedge \delta, \alpha \wedge \epsilon) = 1/2$, $\mathsf{E}(\alpha \wedge \epsilon, \alpha \wedge \beta \wedge \gamma \wedge \delta) = 1/8$, $\mathsf{E}(\alpha \wedge \beta, \alpha \wedge \neg\beta) = 0$.

Proposition 7. *Let X, Y, and Z be sets of classical propositional formulae: (1) $0 \leq \mathsf{E}(X, Y) \leq 1$; (2) $X \vdash \wedge Y$ iff $\mathsf{E}(X, Y) = 1$; (3) $X \vdash \neg \wedge Y$ iff $\mathsf{E}(X, Y) = 0$; (4) If $\mathsf{E}(X, Y) = 1$ then $0 < \mathsf{E}(Y, X)$; and (5) $\mathsf{E}(X, Y) = 0$ iff $\mathsf{E}(Y, X) = 0$.*

We quantify mutation with the following measures.

Definition 15. *For an intended argument $\langle \Phi, \alpha \rangle$, and a decodation $\langle \Psi, \alpha \rangle$, the* **efficiency** *of $\langle \Psi, \alpha \rangle$ for $\langle \Phi, \alpha \rangle$ is $\mathsf{E}(\Phi, \Psi)$ and the* **adequacy** *of $\langle \Psi, \alpha \rangle$ for $\langle \Phi, \alpha \rangle$ is $\mathsf{E}(\Psi, \Phi)$.*

Example 18. *Let $A = \langle \{\beta, \beta \rightarrow \alpha\}, \alpha \rangle$ be an intended argument, and let $B = \langle \{\beta, \beta \rightarrow \gamma, \gamma \rightarrow \alpha\}, \alpha \rangle$ be a decodation. So the efficiency of B for A is $\mathsf{E}(\{\beta, \beta \rightarrow \alpha\}, \{\beta, \beta \rightarrow \gamma, \gamma \rightarrow \alpha\}) = 1/2$ and the adequacy of B for A is $\mathsf{E}(\{\beta, \beta \rightarrow \gamma, \gamma \rightarrow \alpha\}, \{\beta, \beta \rightarrow \alpha\}) = 1$.*

Efficiency of less than 1 means that some premises of the decodation do not follow from the intended argument. So the decodation is inefficient as the extra premises are not required for the intended argument. The closer efficiency is to 1, the less is this inefficiency. Adequacy of less than 1 means that some premises of the intended argument do not follow from the decodation. So the decodation is inadequate as the missing premises are required for the intended argument. The closer adequacy is to 1, the less is this inadequacy.

Measuring mutation in terms of efficiency and adequacy is an external way of evaluating the quality of argumentation undertaken by a pair of agents. However, unless the proponent also sends its intended argument to the recipient, or the recipient sends its decodation to the proponent, the agents cannot measure mutation. Nonetheless, evalulating efficiency and adequacy is a useful way that the "owner" of some intelligent software agents can measure their success, and update their cobases accordingly.

Discussion

Argumentation is an important cognitive activity that needs to be better understood if we are to build intelligent systems better able to deal with conflicts arising in information and between agents. Enthymemes are a ubiquitous phenomenon in the real-world, and so if we are to build intelligent systems that generate arguments (e.g. to justify their actions, to persuade other agents, etc), and process arguments from other agents, then we need to build the capacity into these systems to generate and process enthymemes. We believe this proposal could be adapted for a variety of other argumentation systems (e.g. (García & Simari 2004; Amgoud & Cayrol 2002)), and there are diverse ways that the notion of common knowledge could be refined (e.g. (Sperber & Wilson 1995)). Finally, decodation is a form of abduction, and so techniques and algorithms developed for abduction could be harnessed for improving the quality of decodation (e.g. (Eiter, Gottlob, & Leone 1997)).

References

Amgoud, L., and Cayrol, C. 2002. A model of reasoning based on the production of acceptable arguments. *Annals of Mathematics and Artificial Intelligence* 34:197–216.

Besnard, P., and Hunter, A. 2001. A logic-based theory of deductive arguments. *Artificial Intelligence* 128:203–235.

Besnard, P., and Hunter, A. 2005. Practical first-order argumentation. In *Proc. of the 20th National Conference on Artificial Intelligence (AAAI'05)*, 590–595. MIT Press.

Cayrol, C.; Royer, V.; and Saurel, C. 1993. Management of preferences in assumption based reasoning. In *Information Processing and the Management of Uncertainty in Knowledge based Systems (IPMU'92)*, volume 682 of *Lecture Notes in Computer Science*. Springer.

Chesnevar, C.; Maguitman, A.; and Loui, R. 2000. Logical models of argument. *ACM Comp. Surveys* 32:337–383.

Eiter, T.; Gottlob, G.; and Leone, N. 1997. Semantics and complexity of abduction from default theories. *Artificial Intelligence* 90:177–223.

García, A., and Simari, G. 2004. Defeasible logic programming: An argumentative approach. *Theory and Practice of Logic Programming* 4(1):95–138.

Prakken, H., and Vreeswijk, G. 2002. Logical systems for defeasible argumentation. In Gabbay, D., and Guenthner, F., eds., *Handbook of Philosophical Logic*, volume 4. Kluwer. 219–318.

Sperber, D., and Wilson, D. 1995. *Relevance: Communication and Cognition*. Blackwells.

Walton, D. 1989. *Informal Logic: A Handbook for Critical Argumentation*. Cambridge University Press.