Reasoning about the Appropriateness of Proponents for Arguments

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

Abstract

Formal approaches to modelling argumentation provide ways to present arguments and counterarguments, and to evaluate which arguments are, in a formal sense, warranted. While these proposals allow for evaluating object-level arguments and counterarguments, they do not give sufficient consideration to evaluating the proponents of the arguments. Yet in everyday life we consider both the contents of an argument and its proponent. So if we do not trust a proponent, we may choose to not trust their arguments. Or if we are faced with an argument that we do not have the expertise to assess (for example when deciding whether to agree to having a particular surgical operation), we tend to agree to an argument by someone who is an expert. In general, we see that for each argument, we need to determine the appropriateness of the proponent for it. So for an argument about our health, our doctor is normally an appropriate proponent, but for an argument about our investments, our doctor is normally not an appropriate proponent. In this way, a celebrity is rarely an appropriate proponent for an argument, and a liar is not necessarily an inappropriate proponent for an argument. In this paper, we provide a logic-based framework for evaluating arguments in terms of the appropriateness of the proponents.

Introduction

When we consider an argument proposed by someone, we normally consider both the contents of the argument and the proponent of the argument. So, for example, if the premises do not imply its claim, then we may choose to reject the argument. Similarly, if the proponent of the argument has no particular expertise in the topic of the argument, then we may choose to reject the argument. Therefore reasoning about the appropriateness of a proponent is an important step in evaluating an argument.

Our proposal in this paper is to augment representation and reasoning with arguments at the object-level with a meta-level system for reasoning about the object-level arguments and their proponents. The meta-level system incorporates axioms for raising the object-level argumentation to the meta-level (in particular to capture when one argument is a counterargument to another argument), and meta-level axioms that specify when proponents are appropriate for arguments. The meta-level system is an argumentation system in that it supports the construction and comparison of meta-level arguments and counterarguments. This allows us to integrate the consideration of both object-level and meta-level aspects of assessing an argument and its counterarguments in the same argumentation system.

By formalizing appropriateness criteria as meta-level axioms, we can, or an agent on our behalf, perspicuously evaluate the appropriateness of the proponent for the argument. So in addition to, or instead of, evaluating those object-level arguments directly, we may consider the meta-level arguments pertaining to the appropriateness of the proponents. This also means that when we lack expertise in the area of the object-level arguments, we can use arguments from experts in a systematic way. But it also allows us to not automatically acquiesce to an expert's argument when we have grounds to doubt either the argument and/or the appropriateness of the expert. This also allows us to weigh up competing object-level arguments from different experts.

In this paper, we show that a logic-based argumentation system can be extended with meta-level reasoning about the appropriateness of proponents. We will build on the proposal of (Besnard and Hunter 2001), but the ideas are intended for use with any logic-based proposal (e.g. (Prakken and Vreeswijk 2002; Amgoud and Cayrol 2002; García and Simari 2004; Dung, Kowalski, and Toni 2006)).

Logical Argumentation

We assume that we can start with a large repository of information, represented by a set of formulae Δ, from which logical arguments can be constructed. There is no *a priori* restriction on the formulae in Δ, and so Δ can be inconsistent, and individual formulae in Δ can be inconsistent.

Given Δ, an argument is a set of formulae that can be used to prove some claim, together with that claim. Each claim is represented by a formula. We use a classical (propositional or first-order) language $\mathcal{L}$ with classical deduction denoted by $\vdash$. We use $\alpha, \beta, \gamma, \ldots$ to denote formulae and $\Delta, \Phi, \Psi, \ldots$ to denote sets of formulae.

Definition 1. *An* **argument** *is a pair $\langle \Phi, \alpha \rangle$ s.t.: (1) $\Phi \subseteq \Delta$; (2) $\Phi \not\vdash \perp$; (3) $\Phi \vdash \alpha$; and (4) there is no $\Phi' \subset \Phi$ s.t. $\Phi' \vdash \alpha$. 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. Also let* $\mathsf{Claim}(\langle \Phi, \alpha \rangle) = \alpha$ *and* $\mathsf{Support}(\langle \Phi, \alpha \rangle) = \Phi$.

Example 1. *Some arguments from the knowledgebase* $\Delta = \{p(a), \neg q(b), \neg\exists x, y.r(x, y), \forall x.(p(x) \rightarrow q(x))\}$ *are:*

$$\langle\{p(a), \forall x.(p(x) \rightarrow q(x))\}, q(a)\rangle$$
$$\langle\{\neg\exists x, y.r(x, y)\}, \forall x, y.\neg r(x, y)\rangle$$

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

Definition 2. *For an argument* $\langle\Phi, \alpha\rangle$*, an* **undercut** *is an argument* $\langle\Psi, \neg(\phi_1 \wedge \ldots \wedge \phi_n)\rangle$*, s.t.* $\{\phi_1, \ldots, \phi_n\} \subseteq \Phi$.

The following definition of a canonical undercut is a "maximally conservative" undercut in the sense that the support and claim are the weakest possible for an undercut.

Definition 3. *A* **canonical undercut** *for an argument* $\langle\Phi, \alpha\rangle$ *is an argument of the following form* $\langle\Psi, \neg(\phi_1 \wedge ... \wedge \phi_n)\rangle$ *where* $\Phi = \{\phi_1, .., \phi_n\}$.

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

Definition 4. *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* $\Phi_1 \cup \cdots \cup \Phi_n$*; and (3) Each child node of a node* A *is a canonical undercut for* A *that obeys condition 2.*

The second condition in Definition 4 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).

In examples of argument trees, we abbreviate the claim of each canonical undercut by the $\Diamond$ symbol, since it is the negation of the conjunction of the support of its parent.

Example 2. *An argument tree for* $q(a)$ *is:*

$$\begin{array}{c}\langle\{p(a), \forall x.(p(x) \rightarrow q(x) \vee r(x)), \neg\exists x.r(x)\}, q(a)\rangle \\ \uparrow \\ \langle\{\neg\exists x.(p(x) \rightarrow q(x) \vee r(x))\}, \Diamond\rangle\end{array}$$

We now consider a widely used criterion in argumentation theory for determining whether the argument at the root of the argument tree is warranted (e.g. (García and Simari 2004)). For this, each node is marked as either U for **undefeated** or D for **defeated**.

Definition 5. *The* **judge function**, *denoted* Judge, *assigns either* $\mathsf{Warranted}$ *or* $\mathsf{Unwarranted}$ *to each argument tree* T *such that* $\mathsf{Judge}(T) = \mathsf{Warranted}$ *iff* $\mathsf{Mark}(A_r) = U$ *where* A_r *is the root node of* T*. For all nodes* A_i *in* 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. So for Example 2, we see that the root of the tree T is defeated, and hence $\mathsf{Judge}(T) = \mathsf{Unwarranted}$.

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 claim α that can be formed from Δ, and each of these trees has finite branching and a finite depth (Besnard and Hunter 2005).

Meta-level Argumentation

We now consider a meta-level formalization of argumentation that adapts and extends the proposal by (Wooldridge, McBurney, and Parsons 2006). We start by formalizing the meta-language as a classical first-order language in which formulae of the object language can be encoded. Each term in the object language is a term in the meta-language, and each formula in the object language is encoded as a term in the meta-language. In order to make the presentation clearer, we use teletype font for examples of predicate symbols in the meta-language. We assume that the meta-language is not typed, and therefore the language makes no distinction between terms that are object-level formulae, and the other terms: If x is an object-level variable, then x is a meta-level variable; If x is a meta-level variable, then x is a meta-level term; If s is an object-level logical, predicate or function symbol, then s is a meta-level function symbol; and If f is a meta-level function, and $t_1, . . , t_n$ are meta-level terms, then $f(t_1, .., t_n)$ is a meta-level term.

Using this set-up, each object-level argument can be represented as a meta-level term where $\langle\ldots\rangle$ is treated as a meta-level function symbol, and $\{\ldots\}$ is treated as a meta-level function symbol. For example, suppose $\langle\{b, b \rightarrow c\}, c\rangle$ is an argument using the object-level language, and `arg` is a monadic meta-level predicate symbol denoting an argument, then the following is a meta-level formula.

$$(F_1) \quad \mathtt{arg}(\langle\{b, b \rightarrow c\}, c\rangle)$$

We can use meta-level formulae to construct meta-level arguments. For example, the following is a simple argument at the meta-level.

$$\langle\{\mathtt{arg}(\langle\{b, b \rightarrow c\}, c\rangle)\}, \mathtt{arg}(\langle\{b, b \rightarrow c\}, c\rangle)\rangle$$

If we also assume equality, then we have the following slightly more complicated meta-level argument.

$$\langle\{\mathtt{arg}(\langle\{b\}, b \vee c\rangle), a_7 = \langle\{b\}, b \vee c\rangle\}, \mathtt{arg}(a_7)\rangle$$

The above example hints at how we can take argumentation using the object language and mirror it using the meta-language. To do this, we propose the following meta-level axioms ($M_1 - M_3$) that are intended to quantify over object-level arguments represented as meta-level terms. The first specifies that if x is an argument, then x is acceptable, the second specifies that if x is acceptable, then x is warranted, and the third specifies that if x_2 is acceptable and x_2 undercuts x_1, then x_1 is not acceptable. For the axioms, we assume the meta-level predicate symbols `acc` for acceptable argument, `wnt` for a warranted argument, and `ucut` for one argument undercutting another argument. Also, as we only give examples of universally quantified meta-level axioms with quantifiers outermost in this paper, we will omit the quantifiers to simplify the presentation, and we will use $w, ..., z$, perhaps with subscript index, as variable symbols.

(M_1) $\texttt{arg}(x) \rightarrow \texttt{acc}(x)$
(M_2) $\texttt{acc}(x) \rightarrow \texttt{wnt}(x)$
(M_3) $\texttt{acc}(y) \wedge \texttt{ucut}(y, x) \rightarrow \neg\texttt{acc}(x)$

From now on, the support of a meta-level argument will be represented by the labels of the formulae rather than the formulae themselves. For example, using formulae F_1 and M_1, we have the following argument.

$$\langle\{F_1, M_1\}, \texttt{acc}(\langle\{b, b \rightarrow c\}, c\rangle)\rangle$$

Using the meta-level axioms, together with some atoms that delineate the arguments, and the undercut relationship between them, we can present an argument tree that uses the object-level language in an isomorphic argument tree that uses the meta-language, as is illustrated next.

Example 3. *Consider arguments a_1, a_2, and a_3, and the tree T where a_1 is the root, a_2 is the child of a_1, and a_3 is the child of a_2.*

(a_1) $\langle\{\forall x.p(x) \rightarrow \forall x.q(x), \neg\exists x.\neg p(x)\}, \forall x.q(x)\rangle$
(a_2) $\langle\{\exists x.(\neg p(x) \wedge r(x))\}, \diamond\rangle$
(a_3) $\langle\{\forall x.\neg r(x)\}, \diamond\rangle$

This can be represented by the meta-level atoms: (F_2) $\texttt{arg}(a_1)$*;* (F_3) $\texttt{arg}(a_2)$*;* (F_4) $\texttt{arg}(a_3)$*;* (F_5) $\texttt{ucut}(a_2, a_1)$*; and* (F_6) $\texttt{ucut}(a_3, a_2)$*. Hence, we get the following argument tree using the meta-language which is isomorphic with the argument tree T.*

$$\begin{array}{c}\langle\{F_2, M_1, M_2\}, \texttt{wnt}(a_1)\rangle \\ \uparrow \\ \langle\{F_3, F_5, M_1, M_3\}, \diamond\rangle \\ \uparrow \\ \langle\{F_4, F_6, M_1, M_3\}, \diamond\rangle\end{array}$$

In general, we have the following result that shows for any argument tree using the object language there is an isomorphic argument tree that uses axioms $M_1 - M_3$.

Proposition 1. *Let T be an argument tree where the support and claim of each argument in T uses the object language and the root of the tree is a_1. If Δ' is a knowledgebase that contains $M_1 - M_3$ and exactly the atoms specified as follows*

- $\texttt{arg}(a_i) \in \Delta'$ *iff a_i is a node in T*
- $\texttt{ucut}(a_j, a_i) \in \Delta'$ *iff a_j is a child of a_i in T*

then there is an isomorphic argument tree T' for $\texttt{wnt}(a_1)$ where the support of each argument in T' is a subset of Δ' and $\mathsf{Judge}(T) = \mathsf{Judge}(T')$ *.*

Now we can explore the advantage of considering argumentation using meta-level axioms. Essentially, we can introduce extra criteria into the definition of warrant and/or the definition of acceptable, and we can then use further meta-level knowledge to reason with these axioms. In the rest of this paper, we will consider alternative definitions for acceptable with a particular focus on formalizing the appropriateness of proponents for arguments.

Logical Formalization of Appropriateness

In order to reason about the appropriateness of proponents for arguments, we reason about the proponents and their arguments at the meta-level. If proponent p has asserted argument a (where a is of the form $\langle\Phi, \alpha\rangle$), the meta-level atom $\texttt{ast}(p, a)$ represents this. To reason with these `ast` facts, we assume a knowledgebase, called a **metabase**, that is a set of meta-level formulae. Our approach is based on the following simple axiom M_4 in our metabase that says if proponent x has asserted argument y and x is an appropriate proponent for y, then y is acceptable.

$$(M_4) \quad \texttt{ast}(x, y) \wedge \texttt{app}(x, y) \rightarrow \texttt{acc}(y)$$

In the rest of this paper, we will use M_4 for our definition of acceptable instead of M_1 given earlier. To use the axiom M_4, we assume that for a particular p and a, if $\texttt{ast}(p, a)$ holds, then we have $\texttt{ast}(p, a)$ explicitly as a fact in our metabase. So the flexibility comes in how we define the `app` predicate. For this we assume further axioms in our metabase. Different applications call for different axioms. But once the core axioms have been fixed, then any argument a by any proponent in that domain can be consider as to whether it is an acceptable argument (i.e. whether $\texttt{acc}(a)$ is undefeated).

Apart from the above axiom M_4, plus M_2 and M_3 given earlier, we are making no further constraints on what first-order formulae can appear in a metabase. There is a myriad of possible axioms we may consider. For a simple example, we may consider that if x is a liar, then for any argument y, x is not an appropriate proponent for y as follows.

$$(M_5) \quad \texttt{liar}(x) \rightarrow \neg\texttt{app}(x, y)$$

As another example, we may think a celebrity is appropriate for any argument they assert as follows.

$$(M_6) \quad \texttt{celebrity}(x) \rightarrow \texttt{app}(x, y)$$

So supposing we also have the facts (F_7) $\texttt{ast}(c, a)$ and (F_8) $\texttt{celebrity}(c)$, then we get the following argument.

$$\langle\{F_7, F_8, M_2, M_4, M_6\}, \texttt{wnt}(a)\rangle$$

For many arguments, just being a celebrity is not sufficient for being an appropriate proponent for an argument. We may therefore prefer to have the following more restricted axiom (instead of M_6) that restricts the appropriate arguments to being about "showbiz". Here, we let $\texttt{topic}(a, showbiz)$ denote that the topic of argument a is $showbiz$.

$$\texttt{celebrity}(x) \wedge \texttt{topic}(y, showbiz) \rightarrow \texttt{app}(x, y)$$

While we impose few specific constraints on what is in a metabase, the aim is to just have formulae in the metabase that are relevant to either deriving arguments using M_4 as a premise, and $\texttt{wnt}(a)$ as a claim for object-level arguments a, or to derive rebutting or undercutting arguments to those arguments, and by recursion undercuts to undercuts.

Example 4. *Consider the metabase Δ with M_2, M_3, M_4 and M_5 plus the following meta-level atoms:* (F_9) $\texttt{ast}(p_1, a_1)$*;* (F_{10}) $\texttt{ast}(p_2, a_2)$*;* (F_{11}) $\texttt{app}(p_1, a_1)$*;* (F_{12})

$\texttt{app}(p_2, a_2)$; (F_{13}) $\texttt{ucut}(a_2, a_1)$; *and* (F_{14}) $\texttt{liar}(p_2)$. *From* Δ*, we get the following argument tree.*

$$\begin{array}{c}\langle\{F_9, F_{11}, M_2, M_4\}, \texttt{wnt}(a_1)\rangle \\ \uparrow \\ \langle\{F_{10}, F_{12}, F_{13}, M_3, M_4\}, \diamond\rangle \\ \uparrow \\ \langle\{F_{14}, M_5\}, \diamond\rangle\end{array}$$

We cannot guarantee that for every argument tree using the object language, there is an isomorphic argument tree that uses M_2, M_3 and M_4. Consider Ex. 4, where there are only object-level arguments a_1 and a_2, a_2 undercuts a_1, and a_1 is defeated, but at the meta-level there is $\langle\{F_{14}, M_5\}, \diamond\rangle$ which allows $\texttt{wnt}(a_1)$ to be the claim of the undefeated root argument. However, we do have the following which identifies an isomorphic subtree obtainable from the metabase.

Proposition 2. *Let T be an argument tree where the support and claim of each argument in T uses the object language and the root of the tree is a_1. If Δ' is a knowledgebase that contains only M_2, M_3 and M_4 and exactly the atoms specified as follows where p_i, a_i and a_j are constant symbols.*

- $\texttt{ast}(p_i, a_i) \in \Delta'$ *iff a_i is a node in T*
- $\texttt{app}(p_i, a_i) \in \Delta'$ *iff a_i is a node in T*
- $\texttt{ucut}(a_j, a_i) \in \Delta'$ *iff a_j is a child of a_i in T*

then there is an isomorphic argument tree T' for $\texttt{wnt}(a_1)$ *where the support of each argument in T' is a subset of Δ' and* $\mathsf{Judge}(T) = \mathsf{Judge}(T')$.

Using the State function on a meta-level argument A, we get a set of ground literals as follows: $\mathsf{State}(A) = \{\phi \mid \mathsf{Support}(A) \vdash \phi$ and ϕ is a ground meta-level literal$\}$.

Example 5. *Consider T given in Example 4. Let the root be b_1, the child of b_1 be b_2, and the child of b_2 be b_3. Applying the state function, to each of these meta-level arguments, we get the following atoms.* $\mathsf{State}(b_1)$ = { $\texttt{ast}(p_1, a_1)$, $\texttt{app}(p_1, a_1)$, $\texttt{acc}(a_1)$, $\texttt{wnt}(a_1)$ }, $\mathsf{State}(b_2)$ = { $\texttt{ast}(p_2, a_2)$, $\texttt{app}(p_2, a_2)$, $\texttt{acc}(a_2)$, $\texttt{ucut}(a_2, a_1)$, $\neg\texttt{acc}(a_1)$ }, *and* $\mathsf{State}(b_3)$ = { $\texttt{liar}(p_2)$, $\neg\texttt{app}(p_2, a_2)$ }.

The state of each argument provides a summary of the essential points arising in the meta-level argumentation.

Properties for Expert Argumentation

In this section, we consider four key criteria, adapted from (Walton 2006), for delineating good expert argumentation.

Qualified proponent The expert is suitably qualified in the field of the argument being proposed.

Confident proponent The expert offers sufficient confidence in the argument being proposed.

Best argument The argument by the expert is better than any competing argument by any expert.

Safe argument No counterargument to the expert's argument has been overlooked.

In order to formalize these criteria as general properties for any argumentation system, we need to introduce the notions of a constellation and a filter. A **constellation**, denoted E, is a general notion that is intended to represent a set of arguments that can arise either in abstract argumentation frameworks or in logic-based argumentation. For instance, in Example 4, let E_1 be the set of arguments in the tree, and let E_2 be the set of undefeated arguments (i.e. the root and leaf). As another instance, consider abstract argumentation given by a graph (Dung 1995), and so there is a constellation E that contains all the nodes in the graph, and then there are constellations E_i that are each defined using one of the notions of extension such as preferred or stable.

Definition 6. *Let $\mathcal{C}$ be a set of constellations. A* **filter** *on $\mathcal{C}$ is a tuple (Q, C, B, S) such that Q, C, B, and S are functions, and for each $E \in \mathcal{C}$, $Q(E)$, $C(E)$, $B(E)$, and $S(E)$ are each a subset of E.*

A filter is a very general definition, but we see in the following definition how a filter can be defined so that $Q(E)$ is the set of arguments in E that have a qualified proponent, $C(E)$ is the set of arguments in E that the proponent is confident about, $B(E)$ is the set of counterarguments in E for putative best arguments in E, and $S(E)$ is the set of counterarguments in E for putative safe arguments in E.

Definition 7. *Let $\mathcal{C}$ be a set of constellations, and let $\mathcal{F} = (Q, C, B, S)$ be a filter on $\mathcal{C}$. For $E \in \mathcal{C}$ and $A \in E$,*

- *A has a* **qualified proponent** *in E w.r.t. $\mathcal{F}$ iff $A \in Q(E)$*
- *A has a* **confident proponent** *in E w.r.t. $\mathcal{F}$ iff $A \in C(E)$*
- *A is a* **best argument** *in E w.r.t. $\mathcal{F}$ iff $B(E) = \emptyset$*
- *A is a* **safe argument** *in E w.r.t. $\mathcal{F}$ iff $S(E) = \emptyset$*

We say A is a **good expert argument** *in E w.r.t. $\mathcal{F}$ iff A has a qualified proponent, has a confident proponent, is a best argument, and is a safe argument in E w.r.t. $\mathcal{F}$.*

In general, we can define a filter according to a given constellation. So for example, if we are given an abstract argument graph (Dung 1995), or an extension of it, we may be able to specify a filter for it so that the above properties hold (as illustrated next).

Example 6. *Consider (d_1) "The neighbour says that it is a mole because it has looked like that for years", and (d_2) "The oncologist says that it is a melanoma because it has features not seen in moles". For the argumentation framework with d_1 and d_2 such that d_1 attacks d_2 and d_2 attacks d_1, consider $\mathcal{C} = \{E_1, E_2, E_3\}$ where $E_1 = \{d_1\}$, $E_2 = \{d_2\}$, and $E_3 = \{d_1, d_2\}$. By reading the text of the arguments we may agree to the filter $\mathcal{F}$ in the following table: For E_1, d_1 is confident but not qualified, and there are no counterarguments for d_1 in E_1; For E_2, d_2 is confident and qualified, and there are no counterarguments for d_2 in E_2; And for E_3, d_1 is confident but not qualified, d_2 is qualified and confident, d_2 is a counterargument for d_1 being a best argument (but not vice versa), and d_1 is a counterargument for d_2 being a safe argument (and vice versa). So d_2 is a good expert argument in E_2 w.r.t. $\mathcal{F}$. There are no other constellations in $\mathcal{C}$ with a good expert argument.*

E_i	$Q(E_i)$	$C(E_i)$	$B(E_i)$	$S(E_i)$
$E_1 = \{d_1\}$	$\emptyset$	$\{d_1\}$	$\emptyset$	$\emptyset$
$E_2 = \{d_2\}$	$\{d_2\}$	$\{d_2\}$	$\emptyset$	$\emptyset$
$E_3 = \{d_1, d_2\}$	$\{d_2\}$	$\{d_1, d_2\}$	$\{d_2\}$	$\{d_1, d_2\}$

The above example illustrates that for abstract argumentation, when arguments are atomic, it is difficult to define filters systematically. When the arguments are atomic, there is no structure to the notation of them that describes the expertise of the proponent, the confidence the proponent has in the argument, etc., and therefore a filter is an extra piece of knowledge that has to be provided by the user. What we really want is to be able to define a filter systematically for a general class of constellations and then ascertain whether the properties for good expert argumentation hold.

We also need to consider the inter-play between the choice of constellation and filter, and the notion of extension, in finding good expert arguments. For instance, we may want to find the arguments with appropriate proponents before taking the counterarguments into account (so, in the example, if we select E_2, we reject the argument d_1 by the unqualified neighbour before considering the remainder). However, we do need to also ensure that the appropriateness of those proponents is undefeated.

To address both the need to be systematic in defining filters and the need to consider counterarguments (for the content and proponent of each argument) in this process, we use our meta-level axiomatization of appropriateness.

Formalization for Expert Argumentation

We now consider two axioms that allow for reasoning about the appropriateness of experts for an argument. The axiom M_7 specifies that a proponent for an argument is appropriate if the topic of the argument is in the scope of the role of the proponent. For this, we use the meta-level predicates `role` (for proponent x has role y) and `scope` (for the scope of role y covers topic w).

$$(M_7) \quad \texttt{topic}(z, w) \wedge \texttt{role}(x, y) \wedge \texttt{scope}(y, w) \rightarrow \texttt{app}(x, z)$$

Example 7. *From the following meta-level atoms (where gp denotes "general/family practitioner")* (H_1) $\texttt{ast}(p_1, a_1)$*;* (H_2) $\texttt{topic}(a_1, \mathit{infarction})$*;* (H_3) $\texttt{role}(p_1, gp)$*; and* (H_4) $\texttt{scope}(gp, \mathit{infarction})$*; we get the following argument.*

$$(n_1) \quad \langle \{H_1, H_2, H_3, H_4, M_2, M_4, M_7\}, \texttt{wnt}(a_1)\rangle$$

For our next axiom M_8, we assume that in some situations it is possible to have two arguments, where one is not undercutting nor rebutting the other but accepting one requires rejecting the other. For example, consider the following two informal arguments. Here we may choose to define arguments concerning the prescription for a patient to be competing if the prescription is different and so here these arguments are competing with each other. For this, we use the meta-level predicate `competing` for when arguments y_1 and y_2 in some sense conflict.

- The patient has high blood pressure therefore we should prescribe betablockers
- The patient has high blood pressure therefore we should prescribe diuretics

The axiom M_8 states that an argument is not acceptable when there is a competing argument by a proponent who is more qualified. For this, we assume that proponents can be ordered according to their qualifications. For example, a cardiologist is more qualified than a gp for any arguments concerning cardiology, and a gp is more qualified than a layperson with respect to any arguments concerning medicine. For this, we use the meta-level predicate `outrank` (for proponent x_2 being higher ranked than proponent x_1).

$$(M_8) \quad \texttt{ast}(x_2, y_2) \wedge \texttt{acc}(y_2) \wedge \texttt{competing}(y_1, y_2) \wedge \texttt{outrank}(x_2, x_1) \rightarrow \neg\texttt{app}(x_1, y_1)$$

Example 8. *Continuing Example 7, we add the following meta-level atoms to* Δ*:* (H_5) $\texttt{ast}(p_2, a_2)$*;* (H_6) $\texttt{topic}(a_2, \mathit{infarction})$*;* (H_7) $\texttt{role}(p_2, cardiologist)$*;* (H_8) $\texttt{scope}(cardiologist, infarction)$*;* (H_9) $\texttt{competing}(a_1, a_2)$*; and* (H_{10}) $\texttt{outrank}(p_2, p_1)$*. From this, we get the following undercut for* n_1*.*

$$(n_2) \quad \langle \{H_5, H_6, H_7, H_8, H_9, H_{10}, M_4, M_7, M_8\}, \diamond\rangle$$

Hence, the tree T_2 *with* n_1 *as root and* n_2 *as child is such that* $\mathsf{Judge}(T_2) = \mathsf{Unwarranted}$.

Even though an argument has been proposed by an appropriate expert, there may be object-level arguments that undercut it (as illustrated next).

Example 9. *Continuing Example 8, we add the following meta-level atoms:* (H_{11}) $\texttt{ast}(p_1, a_3)$*;* (H_{12}) $\texttt{topic}(a_3, \mathit{infarction})$*; and* (H_{13}) $\texttt{ucut}(a_3, a_2)$*. Hence, we get the following.*

$$(n_3) \quad \langle \{H_3, H_4, H_{11}, H_{12}, H_{13}, M_3, M_4, M_7\}, \diamond\rangle$$

Hence, the tree T_3 *with* n_1 *as root and* n_2 *as child to* n_1 *and* n_3 *as child to* n_2 *is such that* $\mathsf{Judge}(T_3) = \mathsf{Warranted}$.

So by using axioms M_2, M_3, M_4, M_7, and M_8, in our metabase, we can analyse object-level arguments and their object-level counterarguments, as well as analyse the appropriateness as experts of the proponents of those arguments.

Now we provide a systematic way of defining a filter for when we have a metabase that includes M_2, M_3, and M_4.

Definition 8. *For an argument tree* T *constructed from a metabase including* M_2*,* M_3*, and* M_4*, let* E *be the set of arguments occurring in* T*, and let* $\mathcal{F} = (Q, C, B, S)$ *be the filter obtained as follows.*

$$\begin{aligned}
Q(E) &= \{A \in E \mid \mathsf{Claim}(A) \vdash \exists x, y.\texttt{app}(x, y)\}\\
C(E) &= \{A \in E \mid \mathsf{Claim}(A) \vdash \exists x, y.\texttt{ast}(x, y)\}\\
B(E) &= \{A \in E \mid \mathsf{Claim}(A) \vdash \exists x, y.\neg\texttt{app}(x, y)\}\\
S(E) &= \{A \in E \mid \mathsf{Claim}(A) \vdash \exists y.\neg\texttt{acc}(y)\}
\end{aligned}$$

This filter is motivated as follows: A meta-level argument A is in $Q(E)$ when there is an object-level argument a and a proponent p s.t. $\texttt{app}(p, a) \in \mathsf{State}(A)$, and hence A is an argument for p being a sufficiently qualified expert for proposing a; A meta-level argument A is in $C(E)$ when there is an object-level argument a and a proponent p s.t. $\texttt{ast}(p, a) \in \mathsf{State}(A)$ and hence A is an argument for p being confident in proposing a; A meta-level argument A is in $B(E)$ when there is an object-level argument a and a proponent p s.t. $\neg\texttt{app}(p, a) \in \mathsf{State}(A)$ and hence A is a counterargument for p being a sufficiently qualified expert for proposing a; And a meta-level argument A is in $S(E)$ when there is an object-level argument a s.t. $\neg\texttt{acc}(a) \in \mathsf{State}(A)$ and hence A is a counterargument for a being acceptable.

Example 10. *Continuing Ex. 9, with the filter given in Definition 8, consider E_1 containing all the arguments (i.e. E_1 = $\{n_1, n_2, n_3\}$), and E_2 containing all the undefeated arguments with a_1 as a term (i.e. E_2 = $\{n_1\}$). Since $Q(E_1)$ = $C(E_1)$ = $\{n_1, n_2, n_3\}$, $B(E_1)$ = $\{n_2\}$, and $S(E_1)$ = $\{n_3\}$, there is no good expert argument in E_1 w.r.t. $\mathcal{F}$. In contrast, $Q(E_2)$ = $C(E_2)$ = $\{n_1\}$, and $B(E_2)$ = $S(E_2)$ = $\emptyset$, and so n_1 is a good expert argument in E_2 w.r.t. $\mathcal{F}$.*

This example illustrates how undefeated arguments can be good expert arguments, and the following formalizes how warranted arguments are good expert arguments.

Proposition 3. *Let Δ be the knowledgebase containing only axioms M_2, M_3, M_4, M_7, and M_8 and ground atoms. Let T be an argument tree constructed from Δ with the claim of the root being $\texttt{wnt}(a)$ for some object-level argument a. Let $E = \{A \text{ is a node in } T \mid \mathsf{Mark}(A) = U \text{ and } a \text{ is a constant symbol in } \mathsf{Claim}(A)\}$ and $\mathcal{F}$ is given by Definition 8. So $\mathsf{Judge}(T) = \mathsf{Warranted}$ iff $\texttt{wnt}(a)$ is a good expert argument in E w.r.t. $\mathcal{F}$.*

This result ensures that for an object-level argument, $\texttt{wnt}(a)$ is not a good expert argument if there is an undefeated object-level counterargument to a or an undefeated counterargument to the proponent of a being appropriate.

We can generalize the result by adding further axioms to the metabase. For instance, the following axioms give further criteria for when a proponent for an argument is not appropriate, where $\texttt{vestedint}(x, z)$ denotes proponent x has vested interests in topic z, and $\texttt{inexp}(x, z)$ denotes proponent x is inexperienced in topic z.

$$(M_{11}) \quad \texttt{topic}(y, z) \land \texttt{vestedint}(x, z) \rightarrow \neg\texttt{app}(x, y)$$
$$(M_{12}) \quad \texttt{topic}(y, z) \land \texttt{inexp}(x, z) \rightarrow \neg\texttt{app}(x, y)$$

Also we can generalize the above result by adding axioms to Δ to allow inferring some of the meta-level atoms. For example, assuming a partial ordering over roles, and then by transitivity, and the `role` atoms, infer the `outrank` atoms. Similarly, the `topic`, `role`, and `scope` predicates can involve subsidiary axioms (e.g. we may axiomatize so that if argument a is on the topic of $cancer$, and a is on the topic of $surgery$, then a is on the topic of $cancer\ surgery$).

Discussion

In this paper, we have made the following contributions: (1) A framework for meta-reasoning about object-level arguments that allows for the presentation of richer criteria for determining whether an object-level argument is warranted and these criteria can draw on meta-information about the arguments including the proponents and provenance of the arguments; and (2) An axiomatization using this framework for reasoning about the appropriateness of expert proponents for arguments, and shown how it can conform to some proposed properties for good expert argumentation.

Our proposal also offers a potentially interesting logic-based representation and reasoning framework for capturing the argument schemes of (Walton 2006). Furthermore, by representing them as meta-level axioms, competing schemes may be used by constructing and comparing arguments that incorporate the axioms representing the schemes.

Because we want to draw out the object-level and meta-level features of argumentation, in particular to infer perspicuous reasons to accept or reject an argument in terms of the content of the (object-level) argument and of the proponent of the argument, we have not proposed a graph-based approach with atomic arguments. Nonetheless, proposals such valued-based argumentation frameworks (Bench-Capon 2003), potentially offer illuminating semantics for axiomatizations and/or filters in our framework.

Whilst, our proposal has been presented using a particular existing proposal for argumentation for both the object-level and the meta-level, it would be straightforward to use alternative proposals for argumentation (e.g. (García and Simari 2004; Amgoud and Cayrol 2002; Dung, Kowalski, and Toni 2006)), perhaps harnessing encodations from (Wooldridge, McBurney, and Parsons 2006; Wyner and Bench-Capon 2007). Note, also that the object-level arguments do not necessarily have to confirm to the same argumentation system as the meta-level. For example, we could obtain the object-level arguments using defeasible logic, and then reason with them at the meta-level using classical logic (or vice versa).

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.

Bench-Capon, T. 2003. Persuasion in practical argument using value based argumentation frameworks. *Journal of Logic and Computation* 13(3):429–448.

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.

Dung, P. 1995. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming, and n-person games. *Artificial Intelligence* 77:321–357.

Dung, P.; Kowalski, R.; and Toni, F. 2006. Dialectical proof procedures for assumption-based admissible argumentation. *Artificial Intelligence* 170:114–159.

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.

Walton, D. 2006. *Fundamentals of Critical Argumentation*. Cambridge University Press.

Wooldridge, M.; McBurney, P.; and Parsons, S. 2006. On the meta-logic of arguments. In *Argumentation in Multi-agent Systems*, volume 4049 of *LNCS*, 42–56. Springer.

Wyner, A., and Bench-Capon, T. 2007. Towards an extensible argumentation system. In *Symbolic and Quanititative Approaches to Reasoning with Uncertiainty*, volume 4724 of *LNCS*, 283–294. Springer.