Auton Agent Multi-Agent Syst (2008) 16:187–209
DOI 10.1007/s10458-007-9025-6

# Agents that argue and explain classifications

**Leila Amgoud · Mathieu Serrurier**

Published online: 28 December 2007

**Abstract** Argumentation is a promising approach used by autonomous agents for reasoning about inconsistent/incomplete/uncertain knowledge, based on the construction and the comparison of arguments. In this paper, we apply this approach to the classification problem, whose purpose is to construct from a set of training examples a model that assigns a class to any new example. We propose a formal argumentation-based model that constructs arguments in favor of each possible classification of an example, evaluates them, and determines among the conflicting arguments the acceptable ones. Finally, a "valid" classification of the example is suggested. Thus, not only the class of the example is given, but also the reasons behind that classification are provided to the user as well in a form that is easy to grasp. We show that such an argumentation-based approach for classification offers other advantages, like for instance classifying examples even when the set of training examples is inconsistent, and considering more general preference relations between hypotheses. In the particular case of concept learning, the results of version space theory developed by Mitchell are retrieved in an elegant way in our argumentation framework. Finally, we show that the model satisfies the rationality postulates identified in argumentation literature. This ensures that the model delivers sound results.

**Keywords** Argumentation · Classification

This article extends and revises results presented in preliminary form in the paper [9].

L. Amgoud (✉) · M. Serrurier
Institut de Recherche en Informatique de Toulouse, IRIT, Université Paul Sabatier,
118, route de Narbonne, 31062 Toulouse Cedex, France
e-mail: amgoud@irit.fr

M. Serrurier
e-mail: serrurier@irit.fr

## 1 Introduction

A rational agent can express claims and judgments, aiming at reaching a decision, a conclusion, or informing, convincing, negotiating with other agents. Pertinent information may be insufficient or on the contrary there may be too much relevant but partially incoherent information. In case of multi-agent interaction, conflicts of opinions between agents are inevitable. Thus, agents can be assisted by *argumentation*, a process based on the exchange and the valuation of interacting arguments which support opinions, claims, proposals, decisions.

Argumentation has become an Artificial Intelligence keyword for the last 15 years. It is also gaining increasing interest in multi-agent systems research community. Argumentation-based techniques are used to specify *autonomous agent reasoning*, such as belief revision, handling inconsistency in knowledge bases [3, 18, 32], decision making under uncertainty [8, 11, 17], merging information coming from different sources [5, 7], practical reasoning [1, 31], and goal generation [20]. Argumentation is also used for modeling *multi-agent interaction*. Indeed, since the seminal work by Walton and Krabbe [33] on the different categories of dialogue, different argumentation-based systems have been proposed for persuasion dialogues [6, 27], negotiation [2, 21, 22, 26], and inquiry dialogues [10].

*Classifying* objects or concepts is another important agent reasoning task. Indeed, an agent may want to classify a concept or an object by its own, or even with the help of other agents through dialogues. The basic idea behind a classification problem for objects in a particular domain is to separate these objects into smaller *classes*, and giving criteria for determining whether a particular object in the domain is in a particular class or not. For instance, one may want to classify animals; the classes here are birds, mammals, reptiles, fish, amphibians, arthropods, etc. Several classification systems have been proposed in the literature (for instance [12, 28, 30]). They rely on techniques in which a consistent collection of *training examples* is provided, as well as a number of classes, called also *clusters*. Each training example has the same structure, consisting for instance of a number of attribute/value pairs. One of these attributes represents the class of the example. The problem is to determine a model that predicts correctly the value of the class attribute of a new example. The model is intended to be sufficiently general in order to be reused on new examples. When the concept to learn is binary, i.e., examples of that concept can be either true or false, the problem is called *concept learning*.

In this paper, we propose to use argumentation techniques for modeling the above classification problem. That problem is thus reformulated as follows: given a set of training examples and a set of *hypotheses*, what should be the class of a new example? To answer this question, arguments are constructed in favor of all the possible classifications of that example. A classification can come either from a hypothesis, or from a training example. The obtained arguments may be conflicting since it may be the case that the same example is affected to different classes. Finally, a "valid" classification of the example is suggested. Thus, not only the class of the example is given, but also the reasons behind that classification are provided to the user as well in a form that is easy to grasp. We show that the results returned by the proposed argumentation framework are sound since the framework satisfies the rationality postulates defined in [13].

In addition to the explanatory power of argumentation, an argumentation-based approach for classification offers other advantages, like for instance classifying examples even when the set of training examples is inconsistent, and considering more general preference relations between hypotheses. Moreover, we show that in the particular case of concept learning, the results of the version space theory developed by Mitchell in [23] are retrieved in an elegant way in our argumentation framework. Indeed, the acceptability semantics defined in [14] allow us to identify and to characterize the *version space* as well as its lower and upper bounds. In summary, this paper proposes a "theoretical" framework for handling, analyzing and explaining the problem of classification. The model has the following features that make it original and flexible:

(1) it handles (i) the case of a consistent set of training examples; (ii) the case of an inconsistent set of training examples; and (iii) the case of an empty set of training examples. Note that the standard approach for classification handles only the case of consistent training examples.
(2) it allows one to reason directly on the set of hypotheses;
(3) examples are classified on the basis of the whole set of hypotheses rather than only one hypothesis as it is the case in standard classification models. Indeed, in the standard approach, a unique hypothesis is chosen, and all the new examples are classified on the basis of that hypothesis.
(4) it proposes new and intuitive decision criteria for choosing the class of an example.
(5) it computes in an elegant way the version space as well as its upper and lower bounds of the version space model of Mitchell.

The paper is organized as follows. Section 2 presents the classification problem. Section 3 introduces the basic argumentation framework of Dung. Section 4 introduces our argumentation-based model for classification as well as its properties. In Sect. 5, we show how the results of version space theory are retrieved in our model. Section 6 is devoted to some concluding remarks and perspectives. The proofs are given in an appendix at the end of the document.

## 2 Classification problem

In a classification problem, examples are described using a *feature space*, denoted by a set $\mathcal{X}$. Elements of $\mathcal{X}$ may be, for instance, pairs (attribute, value), first order facts, etc. The set $\mathcal{X}$ is equipped with an equivalence relation $\equiv$. The different classes are gathered in a set $\mathcal{C} = \{c_1, \dots, c_n\}$, called *concept space*. Elements of $\mathcal{C}$ are assumed to be distinct. Let us illustrate the above concepts through the following example where an agent tries to learn the concept 'sunny day'. This example is borrowed from [23].

*Example 1* (Learning the concept sunny day) In this example, the features space is defined on the basis of pairs (attribute, value). Three attributes are considered: pressure, temperature, and humidity. Each of them may take different values as described in table below.

| Attribute | Possible values |
|---|---|
| Pressure | Low, Medium, High |
| Temperature | Low, Medium, High |
| Humidity | Low, Medium, High |

The features space contains all the possible combinations of the three attributes. Examples of elements of $\mathcal{X}$ are:

- (Pressure, Low) $\wedge$ (Temperature, Low) $\wedge$ (Humidity, Low)
- (Pressure, Low) $\wedge$ (Temperature, Low) $\wedge$ (Humidity, Medium)
- (Pressure, Low) $\wedge$ (Temperature, Low) $\wedge$ (Humidity, High)
- (Pressure, Low) $\wedge$ (Temperature, Medium) $\wedge$ (Humidity, Low)
- ...

The concept to learn is binary, thus $\mathcal{C} = \{0, 1\}$ where 0 means that the day is not sunny, and 1 holds for a sunny day.

In what follows, we call an *example* any pair $(x, c)$ where $x \in \mathcal{X}$ and $c \in \mathcal{C}$. However, at some places we may refer only to $x$ for short. The meaning of the pair $(x, c)$ is that the example $x$ belongs to the class $c$.

A classification model takes as input a set $\mathcal{S}$ of *m training examples* defined as follows:

$$\mathcal{S} = \{(x_i, c_i) \text{ such that } x_i \in \mathcal{X} \text{ and } c_i \in \mathcal{C}, i = 1, \ldots, m\}$$

Information in $\mathcal{S}$ is supposed to be true, thus, new examples should be classified using $\mathcal{S}$ as a reference.

*Example 2* (Example 1 cont.) Let us assume that four training examples are given. They are summarized in table below. For instance (pressure, low) ∧ (temperature, medium) ∧ (humidity, high) is a negative example for the concept a sunny day, whereas (pressure, medium) ∧ (temperature, medium) ∧ (humidity, low) is a positive one.

| Pressure | Temperature | Humidity | Sunny |
|---|---|---|---|
| Low | Medium | High | 0 |
| Medium | Medium | Low | 1 |
| Low | Medium | Medium | 0 |
| Medium | High | Medium | 1 |

An important notion in classification is that of *consistency*. In fact, a set of examples is said to be consistent if it does not contain two logically equivalent examples with two different classes. Formally:

**Definition 1** (*Consistency*) Let $\mathcal{T} = \{(x_i, c_i)_{i=1,\ldots,n}$ such that $x_i \in \mathcal{X}$ and $c_i \in \mathcal{C}\}$ be a set of examples. $\mathcal{T}$ is consistent iff $\nexists (x_1, c_1), (x_2, c_2) \in \mathcal{T}$ such that $x_1 \equiv x_2$ and $c_1 \neq c_2$. Otherwise, $\mathcal{T}$ is said to be inconsistent.

Another important input of a classification model is a *hypotheses space* $\mathcal{H}$ which may be, for instance, decision trees, sets of rules, neural nets, etc. A *hypothesis* $h$ is a mapping from $\mathcal{X}$ to $\mathcal{C}$ (i.e., $h$: $\mathcal{X} \mapsto \mathcal{C}$). Thus, it classifies all the elements of the features space. Moreover, it does that in a coherent way, i.e. it puts each example in a unique class. Let us illustrate this notion of hypotheses space through the following examples.

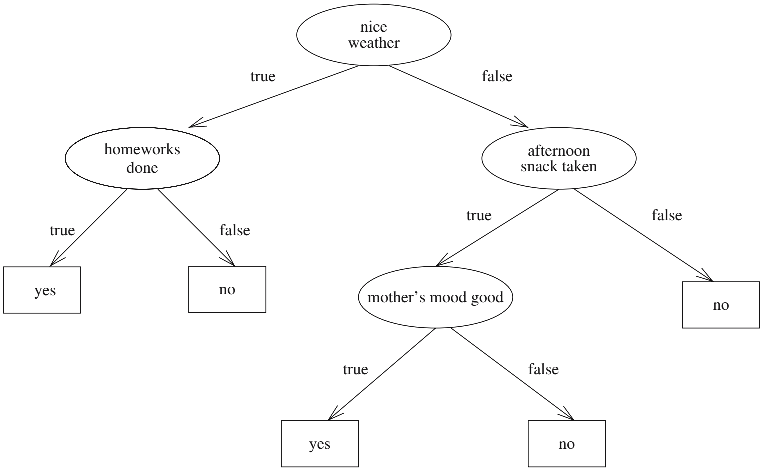
*Example 3* (Example 1 cont.) In this example we assume that the hypotheses space $\mathcal{H}$ is the space of *constraints* on the values of each attribute. Indeed, the constraints are conjunctions of accepted values of attributes. The special constraint Ø (resp. ?) means that no (resp. all) values of attributes are accepted. If a vector of values of attributes match all the constraints, then it is considered as a *positive* example, otherwise it is a *negative* one. The hypotheses ⟨Ø, Ø, Ø⟩ and ⟨?, ?, ?⟩ are respectively the lower and the upper bound of the hypothesis space $\mathcal{H}$.

Let us now consider another classification example in which hypotheses are encoded by decision trees.

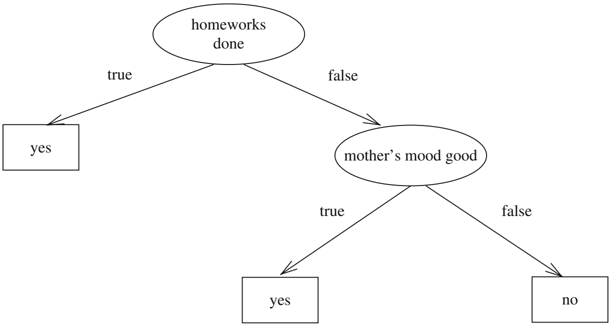
*Example 4* The concept to learn is whether it is possible for a child to play with his friend after school or not. The answer to this question is either yes or no, thus, $\mathcal{C} = \{\text{yes, no}\}$. The answer depends on four binary variables: Homework done, Mother's mood is good, nice weather and afternoon snack taken. Table below summarizes eight training examples.

| Homework done | Mother's mood good | Nice weather | Afternoon snack taken | Answer |
|---|---|---|---|---|
| True | False | True | False | Yes |
| False | True | False | True | Yes |
| True | True | True | False | Yes |
| True | False | True | True | Yes |
| False | True | True | True | No |
| False | True | False | False | No |
| True | False | False | True | No |
| True | True | False | False | No |

The hypotheses space in this example is the set of all possible decision trees that may be built from the features space. A decision tree is a tree whose nodes are the binary variables. Branches starting from a node correspond to possible values of the node. Finally, the leaves correspond to classes. Figure below depicts an example of a decision tree, thus a hypothesis $h$ of $\mathcal{H}$.



Note that the above hypothesis classifies correctly the eight training examples. Unfortunately, not all hypotheses do that. Figure below depicts a hypothesis that does not classify correctly the last training example.



Before defining the output of the framework, let us first introduce a key notion, that of *soundness*.

**Definition 2** (*Soundness*) Let $h \in \mathcal{H}$. A hypothesis $h$ is *sound* with respect to a training example $(x, c) \in \mathcal{S}$ iff $h(x) = c$. $h$ is said to be *sound* with $\mathcal{S}$ iff $\forall (x_i, c_i) \in \mathcal{S},\ h$ is sound w.r.t $(x_i, c_i)$.

The general task of classification is to identify a unique hypothesis $h \in \mathcal{H}$ that is sound with respect to the training examples. This hypothesis will be next used for classifying any new example. The main question is then "how this hypothesis is chosen among all elements of $\mathcal{H}$?" The most common approach for identifying this hypothesis is to use a greedy exploration of the hypotheses space, guided by a *preference* relation on hypotheses. An example of a preference relation is the one based on *utility functions*. Utility functions are generally based on the accuracy of the hypotheses (proportion of well classified examples) weighted by some complexity criteria (number of rules, etc.). Utility functions encode usually a total order. Another category of preference relations are the so-called *syntactic relations*. These may represent for instance entailment or subsumption in the logical case. In this case it encodes a partial preorder on $\mathcal{H}$.

## 3 Abstract argumentation framework

Argumentation is a promising approach for handling inconsistent knowledge, based on the justification of plausible conclusions by *arguments*. An argument is a reason for believing a claim, for doing an action, etc. Since knowledge may be inconsistent, arguments may be conflicting too. Thus, it is important to determine which arguments to keep among the conflicting ones, and finally to determine which conclusions to draw from the whole available knowledge. In summary, argumentation is a four steps process: (1) constructing *arguments* and counter-arguments, (2) defining the *strengths* of those arguments, (3) evaluating the *acceptability* of the different arguments, and (4) concluding or defining the *justified conclusions*. In [14], an argumentation system is defined as follows:

**Definition 3** (*Argumentation system*) An argumentation system is a pair $\mathsf{AS} = \langle \mathsf{Arg}, \mathcal{R} \rangle$ where $\mathsf{Arg}$ is a set of arguments and $\mathcal{R} \subseteq \mathsf{Arg} \times \mathsf{Arg}$ is an attack relation. An argument $A$ attacks an argument $B$ iff $(A, B) \in \mathcal{R}$ (or $A\mathcal{R}B$).

In the above definition arguments are abstract entities. Their origin and structure are left unknown. Note that with each argumentation system is associated a directed graph whose nodes are the different arguments, and the edges represent the attack relation between them. Let us illustrate the above concepts through the well-known example of Nixon Diamond.

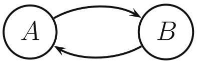
*Example 5* (Nixon Diamond) The scenario is described as follows:

- Usually, Quakers are pacifist
- Usually, Republicans are not pacifist
- Nixon is both a Quaker and a Republican

In this example, two arguments can be built. The first one is in favor of being pacifist and the second is against being pacifist.

- $A$: Nixon is a pacifist since he is a Quaker
- $B$: Nixon is not a pacifist since he is a Republican

It is clear that the two arguments are conflicting with each other. Thus, $\mathsf{Arg} = \{A, B\}$ and $\mathcal{R} = \{(A, B), (B, A)\}$. The graph associated with this argumentation system is depicted in figure below.



Among all the conflicting arguments, it is important to know which arguments to keep for inferring conclusions or for making decisions. In [14], different semantics for the notion of acceptability have been proposed. Let us recall them here.

**Definition 4** (Conflict-free, Defence) Let $\mathcal{B} \subseteq \mathsf{Arg}$.

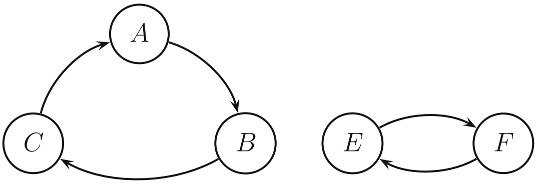
- $\mathcal{B}$ is conflict-free iff $\nexists A_i, A_j \in \mathcal{B}$ such that $A_i \mathcal{R} A_j$.
- $\mathcal{B}$ defends an argument $A_i$ iff for each argument $A_j \in \mathsf{Arg}$, if $A_j \mathcal{R} A_i$, then $\exists A_k \in \mathcal{B}$ such that $A_k \mathcal{R} A_j$.

*Example 6* (Example 5 Cont.) It is clear that the sets $\{A\}$ and $\{B\}$ are conflict-free. However, the set $\{A, B\}$ is not. Moreover, The set $\{A\}$ defends the argument $A$ since $A$ attacks its unique attacker $B$. Similarly, $\{B\}$ defends $B$ against $A$.

**Definition 5** (*Acceptability semantics*) Let $\mathcal{B}$ be a conflict-free set of arguments, and let $\mathcal{F} : 2^{\mathsf{Arg}} \mapsto 2^{\mathsf{Arg}}$ be a function such that $\mathcal{F}(\mathcal{B}) = \{A \mid \mathcal{B} \text{ defends } A\}$.

- $\mathcal{B}$ is a complete extension iff $\mathcal{B} = \mathcal{F}(\mathcal{B})$.
- $\mathcal{B}$ is a grounded extension iff it is the minimal (w.r.t. set-inclusion) complete extension.
- $\mathcal{B}$ is a preferred extension iff it is a maximal (w.r.t. set-inclusion) complete extension.
- $\mathcal{B}$ is a stable extension iff it is a preferred extension that attacks all arguments in $\mathsf{Arg} \backslash \mathcal{B}$.

*Example 7* Let us consider the following argumentation system.



There are three complete extensions: $\emptyset$, $\{E\}$ and $\{F\}$. The empty set is the grounded extension of this argumentation system. Note that, the system has two preferred extensions ($\{E\}$ and $\{F\}$), however, it has no stable extension. Indeed, the set $\{E\}$ (resp. $\{F\}$) does not attack the arguments $A$, $B$ and $C$.

Any argumentation system has only one grounded extension which may be empty as it is the case in the previous example. It contains all the arguments that are not attacked, and also the arguments which are defended directly or indirectly by non-attacked arguments. In [14], it has been shown that each argumentation system has at least one preferred extension. Moreover, each stable extension is a preferred one, but the reverse is not true (as can be checked in the above example). When the preferred and stable extensions of an argumentation system coincide, that system is said to be *coherent*. In [15], it has been proved that when the directed graph associated with an argumentation system has no odd length cycles, then that system is coherent.

**Proposition 1** (Coherence condition [15]) *If the graph associated with an argumentation system* $\mathsf{AS}$ *has no odd length cycles, then* $\mathsf{AS}$ *is coherent.*

Now that the acceptability semantics are defined, we are ready to define the status of any argument.

**Definition 6** (*Argument status*) Let $\langle \mathsf{Arg}, \mathcal{R} \rangle$ be an argumentation system, and $\mathcal{E}_1, \ldots, \mathcal{E}_n$ its extensions under a given semantics. Let $a \in \mathcal{A}$.

- $a$ is skeptically accepted iff $a \in \mathcal{E}_i, \forall \mathcal{E}_i$ with $i = 1, \ldots, n$.
- $a$ is credulously accepted iff $\exists \mathcal{E}_i$ such that $a \in \mathcal{E}_i$.
- $a$ is rejected iff $\nexists \mathcal{E}_i$ such that $a \in \mathcal{E}_i$.

It is clear from the above definition that if an argument skeptically accepted, then it is also credulously accepted. However, the converse is not true.

## 4 An argumentation framework for Classification

The aim of this section is to propose an instantiation of the abstract framework of Dung that allows the classification of examples. Throughout this section, we will consider a features space $\mathcal{X}$, a concept space $\mathcal{C} = \{c_1, \ldots, c_n\}$, a (maybe *inconsistent*) set $\mathcal{S} = \{(x_i, c_i)$ such that $x_i \in \mathcal{X}$ and $c_i \in \mathcal{C}, i = 1, \ldots, m\}$ of $m$ training examples, a hypotheses space $\mathcal{H}$ that is equipped with a preference relation $\succeq$. Thus, $\succeq \subseteq \mathcal{H} \times \mathcal{H}$. This preference relation may be any one of those studied in the literature. For the purpose of generality, in this paper we don't restrict ourselves to particular relations. The only assumption we make is that the relation $\succeq$ is a *partial preorder* (i.e., reflexive and transitive). For two hypotheses $h_1$ and $h_2$, the notation $h_1 \succeq h_2$ means that $h_1$ is at least as good as $h_2$.

### 4.1 The classification model

In order to instantiate the abstract framework of Dung, one needs to define the set $\mathcal{A}$ of arguments as well as the attack relation between those arguments.

In our particular application, an agent argues about classifications, thus it builds arguments in favor of assigning particular classes from $\mathcal{C}$ to an example in $\mathcal{X}$. Indeed, an argument in favor of a pair $(x, c)$ represents the reason for assigning the class $c$ to the example $x$. Two reasons can be distinguished:

(1) $(x, c)$ is a training example in $\mathcal{S}$,
(2) there exists a hypothesis $h \in \mathcal{H}$ that classifies $x$ in $c$.

**Definition 7** (*Argument*) An argument is a triplet $A = \langle h, x, c \rangle$ such that:

(1) $h \in \mathcal{H}, x \in \mathcal{X}, c \in \mathcal{C}$
(2) If $h \neq \emptyset$, then $c = h(x)$
(3) If $h = \emptyset$, then $(x, c) \in \mathcal{S}$

$h$ is called the support of the argument, and $(x, c)$ its conclusion. Let $\texttt{Example}(A) = x$, and $\texttt{Class}(A) = c$.
We will call $\mathcal{A}$ the set of arguments built from $(\mathcal{H}, \mathcal{X}, \mathcal{C})$.

Note that from the above definition, for any training example $(x_i, c_i) \in \mathcal{S}, \exists \langle \emptyset, x_i, c_i \rangle \in \mathcal{A}$. Let $\mathcal{A}_\mathcal{S} = \{\langle \emptyset, x, c \rangle \in \mathcal{A}\}$, i.e., the set of arguments coming from the training examples. When the set of training examples is not empty, the set $\mathcal{A}_\mathcal{S}$ is not empty as well.

**Proposition 2** *Let $\mathcal{S}$ be a set of training examples.*

- $|\mathcal{S}| = |\mathcal{A}_\mathcal{S}|$[1]

[1] || denotes the cardinal of a given set.

- *If $\mathcal{S}$ is non-empty, then $\mathcal{A}_{\mathcal{S}} \neq \emptyset$*

It can also be checked that each example $x \in \mathcal{X}$ has exactly $|\mathcal{H}|$ arguments in its favor coming from hypotheses. Formally:

**Proposition 3** *Let $x \in \mathcal{X}$. $|\{\langle h_i, x, c_i\rangle \in \mathcal{A}$ such that $h_i \neq \emptyset\}| = |\mathcal{H}|$.*

Let us illustrate the notion of argument through Example 1.

*Example 8* In Example 1, there are exactly four arguments with an empty support, and they correspond to the training examples:

- $a_1 = \langle\emptyset$, (pressure, low) $\wedge$ (temperature, medium) $\wedge$ (humidity, high), 0$\rangle$
- $a_2 = \langle\emptyset$, (pressure, medium) $\wedge$ (temperature, medium) $\wedge$ (humidity, low), 1$\rangle$
- $a_3 = \langle\emptyset$, (pressure, low) $\wedge$ (temperature, medium) $\wedge$ (humidity, medium), 0$\rangle$
- $a_4 = \langle\emptyset$, (pressure, medium) $\wedge$ (temperature, high) $\wedge$ (humidity, medium), 1$\rangle$

There are also arguments with a non-empty support such as:

- $a_5 = \langle\langle$?, medium $\vee$ high, ?$\rangle$, (pressure, low) $\wedge$ (temperature, high) $\wedge$ (humidity, high), 1$\rangle$
- $a_6 = \langle\langle$medium $\vee$ high, ?, ?$\rangle$, (pressure, low) $\wedge$ (temperature, high) $\wedge$ (humidity, high), 0$\rangle$
- $a_7 = \langle\langle$medium, medium $\vee$ high, ?$\rangle$, (pressure, low) $\wedge$ (temperature, high) $\wedge$ (humidity, high), 0$\rangle$

In [3, 32], it has been argued that arguments may have different strengths depending on the quality of information used to construct them. In [32], for instance, arguments built from specific information are stronger than arguments built from more general ones. In our particular application, it is clear that arguments with an empty support are stronger than arguments with a non-empty one. This reflects the fact that classifications given by training examples take precedence over ones given by hypotheses in $\mathcal{H}$. It is also natural to consider that arguments based on most preferred hypotheses are stronger than arguments based on less preferred ones.

**Definition 8** (*Comparing arguments*) Let $\langle h, x, c\rangle$, $\langle h', x', c'\rangle$ be two arguments of $\mathcal{A}$. $\langle h, x, c\rangle$ is preferred to $\langle h', x', c'\rangle$, denoted by $\langle h, x, c\rangle \texttt{Pref} \langle h', x', c'\rangle$, iff:

1. $h = \emptyset$ and $h' \neq \emptyset$, or
2. $h \succeq h'$.

**Proposition 4** *The relation* Pref *is a partial preorder.*

In what follows, $\texttt{Pref}^{\succ}$ will denote the strict relation associated with Pref, i.e., for $A, B \in \mathcal{A}$, $A\texttt{Pref}^{\succ}B$ iff $A\texttt{Pref}B$ and not($B\texttt{Pref}A$).

Now that the set of arguments is defined, it is possible to define the attack relation $\mathcal{R}$ between arguments in $\mathcal{A}$. There are two ways in which an argument $A$ can attack another argument $B$: (1) by *rebutting* its *conclusion*, or (2) by *undercutting* its *support*.

In the case of rebutting, two arguments classify the same example in different classes. This relation is also used in argumentation literature [16], in particular for handling inconsistency in knowledge bases. The idea is that there is an argument in favor of a statement and another argument against it, i.e., in favor of its negation.

**Definition 9** (*Rebutting*) Let $\langle h, x, c\rangle$, $\langle h', x', c'\rangle$ be two arguments of $\mathcal{A}$. $\langle h, x, c\rangle$ rebuts $\langle h', x', c'\rangle$ iff:

- $x \equiv x'$
- $c \neq c'$

*Example 9* In example 8, we have for instance, the argument $a_5$ rebuts $a_6$, $a_5$ rebuts $a_7$, $a_6$ rebuts $a_5$, and $a_7$ rebuts $a_5$.

The idea behind the undercutting relation is to undermine a premise used in another argument [16]. In our case, an argument $A$ undercuts an argument $B$ when the support of $B$ classifies in a different way the example of the conclusion of $A$. This relation is only restricted to training examples. Indeed, only arguments built from training examples are allowed to undercut other arguments. Its role is to penalize hypotheses that do not classify correctly the given training examples. The idea behind this is that training examples are the only, in some sense, certain information that one has, and thus cannot be defeated by hypothesis. However, hypotheses have controversial status in the sense that their classifications of examples may be incorrect.

**Definition 10** (*Undercutting*) Let $\langle h, x, c\rangle$, $\langle h', x', c'\rangle$ be two arguments of $\mathcal{A}$. $\langle h, x, c\rangle$ undercuts $\langle h', x', c'\rangle$ iff:

- $h = \emptyset$
- $h'(x) \neq c$

*Example 10* In example 8, we have for instance, the argument $a_1$ undercuts $a_5$, and $a_3$ undercuts $a_5$.

Let us consider another example in order to illustrate more the importance of this relation.

*Example 11* Let us assume that $\mathcal{X} = \{x_1, x_2, x_3\}, \mathcal{C} = \{c_1, c_2\}, \mathcal{S} = \{(x_1, c_1), (x_2, c_2)\}$, and $\mathcal{H} = \{h_1, h_2\}$, where $h_1(x_1) = c_1$, $h_1(x_2) = c_2$, $h_1(x_3) = c_1$, $h_2(x_1) = c_2$, $h_2(x_2) = c_2$, $h_2(x_3) = c_2$. The arguments that may be built from these data are summarized in table below:

| | | |
|---|---|---|
| $a_1 = \langle\emptyset, x_1, c_1\rangle$ | $a_3 = \langle h_1, x_1, c_1\rangle$ | $a_6 = \langle h_2, x_1, c_2\rangle$ |
| $a_2 = \langle\emptyset, x_2, c_2\rangle$ | $a_4 = \langle h_1, x_2, c_2\rangle$ | $a_7 = \langle h_2, x_2, c_2\rangle$ |
| | $a_5 = \langle h_1, x_3, c_1\rangle$ | $a_8 = \langle h_2, x_3, c_2\rangle$ |

If we consider only the rebut relation, we will get two conflict-free extensions of arguments that contain arguments coming from the two training examples:

- $\mathcal{E}_1 = \{a_1, a_2, a_3, a_4, a_5, a_7\}$
- $\mathcal{E}_2 = \{a_1, a_2, a_3, a_4, a_7, a_8\}$

Indeed, since training examples are considered as correct classifications, their corresponding arguments appear in the extensions (of course in case the set $\mathcal{S}$ is consistent). In this example, there is a disagreement between $h_1$ and $h_2$ on the class of $x_3$. One would like to follow the classification given by $h_1$ since this latter satisfies all the training examples, whereas $h_2$ fails to classify the example $(x_1, c_1)$. Thus, by introducing the notion of undercut, the argument $a_1$ undercuts $a_8$. Thus, the only possible extension is $\mathcal{E}_1$, concluding that the class of $x_3$ is $c_1$.

It can be easily checked that if an argument $\langle h, x, c\rangle$ undercuts another argument $\langle h', x', c'\rangle$, then there exists a third argument $\langle h', x, c''\rangle$ such that $\langle h, x, c\rangle$ rebuts $\langle h', x, c''\rangle$. It is also easy to check that when the set $\mathcal{S}$ of training examples is consistent, then arguments of $\mathcal{A}_{\mathcal{S}}$ are not conflicting.

**Proposition 5** *If $\mathcal{S}$ is consistent, then $\nexists A, B \in \mathcal{A}_{\mathcal{S}}$ such that A rebuts B, or A undercuts B.*

The two above conflict relations are brought together in a unique relation, called `Defeat`.

**Definition 11** (*Defeat*) Let $A = \langle h, x, c\rangle$, $B = \langle h', x', c'\rangle$ be two arguments of $\mathcal{A}$. $(A, B) \in$ `Defeat`, or $A$ defeats $B$ iff:

(1) $A$ rebuts (resp. undercuts) $B$, and
(2) not($B\texttt{Pref}^{\succ}A$)

*Example 12* With the argument defined in Example 8 we have for instance: $a_1$ defeats $a_5$ and $a_3$ defeats $a_5$.

From the above definition, it is easy to check that an argument with an empty-support cannot be defeated by an argument with a non-empty support.

**Proposition 6** $\forall A \in \mathcal{A}_{\mathcal{S}}$, $\nexists B \in \mathcal{A}\backslash\mathcal{A}_{\mathcal{S}}$ *such that B defeats A.*

The argumentation system for classification is then the following:

**Definition 12** (*Argumentation system*) An argumentation system for classification (ASC) is a pair $\langle \mathcal{A}, \texttt{Defeat}\rangle$, where $\mathcal{A}$ is the set of arguments defined in Definition 7 and `Defeat` is the relation defined in Definition 11.
Let $\mathcal{E}_1, \ldots, \mathcal{E}_n$ denote the different (preferred or stable) extensions of ASC.

The last step of an argumentation process consists of defining the *status* of conclusions, in our case, the classification of examples. In what follows we present different decision criteria for providing the class of each example. These criteria are presented from the cautious one to the adventurous one.

The basic idea behind the cautious criterion is that an example is affected to a given class if there exists an argument in favor of that classification that belongs to all the extensions of the argumentation system. Formally:

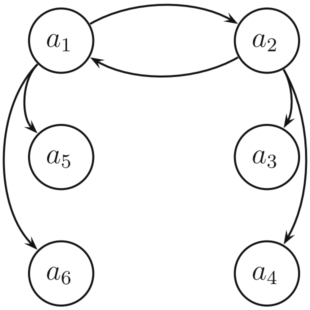
**Definition 13** (*Skeptical vote*) Let $\langle \mathcal{A}, \texttt{Defeat}\rangle$ be an ASC, and $\mathcal{E}_1, \ldots, \mathcal{E}_n$ its extensions under a given semantics. Let $x \in \mathcal{X}$ and $c \in \mathcal{C}$. $x$ is skeptically classified in $c$ iff $\exists\langle h, x, c\rangle$ such that $\langle h, x, c\rangle$ is skeptically accepted.
$CV$ denotes the set of all $(x, c)$ such that $x$ is cautiously classified in $c$.

The above criterion is very strong since it may be the case that an example is affected to the same class in each extension, but on the basis of different hypotheses. One would like to conclude that this is thus the right class of that example. To illustrate this idea, let us consider the following example.

*Example 13* Let $\mathcal{X} = \{x_1, x_2\}$, $\mathcal{C} = \{c_1, c_2, c_3, c_4\}$, $\mathcal{S} = \{(x_1, c_1), (x_1, c_2)\}$, and $\mathcal{H} = \{h_1, h_2\}$ with $h_1(x_1) = c_1$, $h_1(x_2) = c_1$, $h_2(x_1) = c_2$, and $h_2(x_2) = c_1$. Note that the set of training examples is inconsistent. The arguments that may be built from the three sets are summarized in table below:

| | | |
|---|---|---|
| $a_1 = \langle\emptyset, x_1, c_1\rangle$ | $a_3 = \langle h_1, x_1, c_1\rangle$ | $a_5 = \langle h_2, x_1, c_2\rangle$ |
| $a_2 = \langle\emptyset, x_1, c_2\rangle$ | $a_4 = \langle h_1, x_2, c_1\rangle$ | $a_6 = \langle h_2, x_2, c_1\rangle$ |

Figure below depicts the defeat relation among these arguments:



From the above graph it is clear that there are two preferred extensions:

- $\mathcal{E}_1 = \{a_1, a_3, a_4\}$
- $\mathcal{E}_2 = \{a_2, a_5, a_6\}$

The two extensions agree on the class of example $x_2$, which is $c_1$. However, the classification is not supported by the same argument in both extensions. Thus, $(x_2, c_1)$ is not a skeptical classification, but one would like to accept this classification.

In order to capture this idea, a new criterion, called here *universal vote*, is introduced. The universally classified examples are those that are supported by arguments in all the extensions. From a classification point of view, these correspond to examples classified by the most preferred hypotheses.

**Definition 14** (*Universal vote*) Let $\langle \mathcal{A}, \texttt{Defeat} \rangle$ be an ASC, and $\mathcal{E}_1, \ldots, \mathcal{E}_n$ its extensions under a given semantics. Let $x \in \mathcal{X}$ and $c \in \mathcal{C}$. $x$ is universally classified in $c$ iff $\forall \mathcal{E}_i$, $\exists \langle h, x, c \rangle \in \mathcal{E}_i$.
$UV$ denotes the set of all $(x, c)$ such that $x$ is universally classified in $c$.

*Example 14* (Example 13 cont.) In the previous example, $(x_2, c_1)$ is a universal classification since in both extensions $\mathcal{E}_1$ and $\mathcal{E}_2$ there is an argument in favor of this conclusion even if the argument is not the same.

The adventurous criterion consists of affecting an example to a given class as soon as there exists at least one extensions containing an argument in favor of that classification. This criterion is adventurous since it may be the case that it chooses more than one class for the same example. However, this criterion is not uninteresting since it gives a useful information on the possible classes of an example when it is not able to classify it in a certain way. Formally:

**Definition 15** (*Credulous vote*) Let $\langle \mathcal{A}, \texttt{Defeat} \rangle$ be an ASC, and $\mathcal{E}_1, \ldots, \mathcal{E}_n$ its extensions under a given semantics. Let $x \in \mathcal{X}$ and $c \in \mathcal{C}$. $x$ is credulously classified in $c$ iff $\exists \langle h, x, c \rangle$ that is credulously accepted.
$EV$ denotes the set of all $(x, c)$ such that $x$ is credulously classified in $c$.

The above criterion can be refined. The basic idea is that the conclusions are the ones that are supported by a majority of extensions. Formally:

**Definition 16** (*Majority vote*) Let $\langle \mathcal{A}, \texttt{Defeat} \rangle$ be an ASC, and $\mathcal{E}_1, \ldots, \mathcal{E}_n$ its extensions under a given semantics. Let $x \in \mathcal{X}$ and $c \in \mathcal{C}$. $x$ is classified by majority in $c$ iff $|\{\mathcal{E}_{i=1,\ldots,n} \text{ s.t } \exists \langle h, x, c \rangle \in \mathcal{E}_i\}| > |\{\mathcal{E}_{j=1,\ldots,n} \text{ s.t } \exists \langle h', x, c' \rangle \in \mathcal{E}_j\}|, \forall c' \neq c$.
$MV$ denotes the set of all $(x, c)$ such that $x$ is classified in $c$ by majority.

The following result shows the links between the four criteria.

**Proposition 7** *Let $\langle \mathcal{A}, defeat\rangle$ be a ASC, and $\mathcal{E}_1, \ldots, \mathcal{E}_n$ its extensions under a given semantics: $CV \subseteq UV \subseteq MV \subseteq EV$.*

## 4.2 The properties of the classification model

We will start by characterizing the acceptable arguments of the model presented in the previous section. It is clear that the arguments that are not defeated by other arguments in the sense of the relation Defeat will be acceptable. Let $\mathcal{U}$ denote that set of undefeated arguments, i.e., $\mathcal{U} = \{A \in \mathcal{A}$ such that $\nexists B \in \mathcal{A}$ and $B$ defeats $A\}$.

**Proposition 8** *If $\mathcal{S}$ is consistent, then $\mathcal{A}_{\mathcal{S}} \subseteq \mathcal{U}$.*

As said in Sect. 3, one of the acceptability semantics is the so-called 'grounded extension'. This extension can be defined using the characteristic function $\mathcal{F}$ given in Definition 5.

Due to the fact that $\mathcal{H}$ and $\mathcal{X}$ are not always finite, the system $\langle \mathcal{A}, \texttt{Defeat}\rangle$ is not always finite. By finite we mean that each argument is defeated by a finite number of arguments.

**Proposition 9** *If $\mathcal{H}$ and $\mathcal{X}$ are finite, then the system $\langle \mathcal{A}, \texttt{Defeat}\rangle$ is finite.*

When an argumentation system is finite, its characteristic function $\mathcal{F}$ is continuous. Consequently, the least fixed point of this function can be defined by an iterative application of $\mathcal{F}$ to the empty set.

**Proposition 10** *If the argumentation system $\langle \mathcal{A}, \texttt{Defeat}\rangle$ is finite, then the grounded extension $\mathcal{E}$ is:*

$$\mathcal{E} = \bigcup \mathcal{F}^{i \geq 0}(\emptyset) = \mathcal{U} \cup \left[ \bigcup_{i \geq 1} \mathcal{F}^{i}(\mathcal{U}) \right].$$

Such an extension is unique and maybe empty. However, we show that when the set $\mathcal{S}$ of training examples is non-empty and consistent, this grounded extension is not empty as well.

**Proposition 11** (Grounded extension) *If $\mathcal{S}$ is non-empty and consistent, then the argumentation system $\langle \mathcal{A}, \texttt{Defeat}\rangle$ has a non empty grounded extension $\mathcal{E}$.*

Let us now analyze the other acceptability semantics, namely preferred and stable ones. In general, the ASC has at least one preferred extension that may be empty. However, as for the case of grounded extension, we can show that in the particular case of a consistent set of training examples, the ASC has at least one non-empty preferred extension.

**Proposition 12** *If $\mathcal{S}$ is consistent, then the system $\mathsf{ASC} = \langle \mathcal{A}, \texttt{Defeat}\rangle$ has $n \geq 1$ non-empty preferred extensions.*

In general, the preferred extensions of an argumentation system are not stable. However, we can show that when the set $\mathcal{C}$ contains only two possible classes, this means that the concept to learn is binary, these extensions coincide. Thus, the argumentation system is coherent. This result is due to the fact that the directed graph associated with the above ASC has no odd length cycles in this case. However, it may contain even length ones.

**Proposition 13** *If $\mathcal{C} = \{c_1, c_2\}$ with $c_1 \neq c_2$, then:*

- *The graph associated with the system* $\langle \mathcal{A}, \texttt{Defeat} \rangle$ *has no odd length cycles.*
- *The system* $\langle \mathcal{A}, \texttt{Defeat} \rangle$ *is coherent.*

In [13], Amgoud and Caminada have defined rationality postulates that need to be satisfied by any argumentation system. These postulates ensure that the system returns safe conclusions. In what follows, we will show that the argumentation system proposed in this paper for classifying examples satisfies these postulates.

**Proposition 14** (Consistency) *Let* $\mathsf{ASC} = \langle \mathcal{A}, \texttt{Defeat} \rangle$, $\mathcal{E}_1, \ldots, \mathcal{E}_n$ *its extensions under a given semantics, and* $UV$, $MV$ *its sets of conclusions.*

- $\forall \mathcal{E}_i$, *the set* $\{(x, c) | \exists \langle h, x, c \rangle \in \mathcal{E}_i\}$ *is consistent.*
- *The sets* $UV$ *and* $MV$ *are consistent.*

Let us now consider the case where $\mathcal{S}$ is inconsistent, with $\mathcal{S}$ can be divided into $\mathcal{S}_1, \ldots, \mathcal{S}_n$, such that each $\mathcal{S}_i$ is a maximal (for set inclusion) consistent subset of $\mathcal{S}$. This means that some training examples are classified in different classes. However, all the elements of $\mathcal{S}$ are supposed to be equally preferred. In this case, two arguments supporting such conflicting training examples rebut each other, thus can defeat each other as well.

Let $\mathcal{A}_{\mathcal{S}_1}, \ldots, \mathcal{A}_{\mathcal{S}_n}$ be the sets of arguments with an empty support and whose conclusions are 'respectively' in the subsets of training examples $\mathcal{S}_1, \ldots, \mathcal{S}_n$. It is clear that each set $\mathcal{A}_{\mathcal{S}_i}$ is conflict-free, however, it is defeated by arguments in $\mathcal{A}_{\mathcal{S}_j}$ with $i \neq j$. Arguments with an empty support are preferred to arguments built from hypothesis. Note that each preferred/stable extension contains one of the sets $\mathcal{A}_{\mathcal{S}_1}, \ldots, \mathcal{A}_{\mathcal{S}_n}$. Moreover, the same set $\mathcal{A}_{\mathcal{S}_i}$ may belong to several extensions at the same time. It can be shown that all the hypothesis that are used to build arguments in a given extension are sound with the subset of training examples of that extension. Indeed, for each consistent subset of $\mathcal{S}$, we get the extensions of the consistent case previously studied.

Note that, the grounded extension can be empty in this particular case of inconsistent training examples. However, this does not mean that it is not possible to classify examples. Let us re-consider Example 13.

*Example 15* (Example 13 cont.) In Example 13, the grounded extension is empty since there is no undefeated arguments. However, there are two preferred/stable extensions:

- $\mathcal{E}_1 = \{a_1, a_3, a_4\}$
- $\mathcal{E}_2 = \{a_2, a_5, a_6\}$

It is clear that $(x_2, c_1)$ is a universal classification. However, $(x_1, c_1)$ and $(x_1, c_2)$ are credulous classifications.

Note that an argumentation-based approach for classification provides more results than classical approaches that do not classify any example when training examples are inconsistent. Another feature of an argumentation-based approach is that even when it cannot provide a class for an example in an unquestionable way as it is the case for $x_1$, it may provide the range of possible classes. In the previous example, there are four possible classes as indicated by the set $\mathcal{C} = \{c_1, c_2, c_3, c_4\}$. However, from our argumentation system, $x_1$ may be either $c_1$ or $c_2$. The two remaining classes are impossible.

Another interesting case is when the set of training examples is empty. In this case, the classification problem consists of classifying examples only on the basis of a set $\mathcal{H}$ of hypothesis. This is, indeed, a particular case of the previous case where $\mathcal{S}$ is inconsistent. The corresponding argumentation system constructs then arguments only on the basis of hypothesis, thus there is no argument with an empty support.

## 5 Retrieving version space theory

As said before, *concept learning* is a particular case of classification, where the concept to learn is binary. In [23], Mitchell has proposed the famous general and abstract framework, called *version space learning*, for concept learning. That framework takes as input a *consistent* set of *training examples* on the concept to learn. $\mathcal{C}$ contains only two classes, denoted respectively by 0 and 1. Thus, $\mathcal{C} = \{0, 1\}$. The set $\mathcal{H}$ is equipped with a "particular" *partial preorder* $\succeq$ that reflects the idea that some hypotheses are more general than others in the sense that they classify positively more examples. This preorder defines a lattice on the hypotheses space. Formally:

**Definition 17** (*Generality order on hypotheses*) Let $h_1, h_2 \in \mathcal{H}$. $h_1$ is more general than $h_2$, denoted by $h_1 \succeq h_2$, iff $\{x \in \mathcal{X} | h_1(x) = 1\} \supseteq \{x \in \mathcal{X} | h_2(x) = 1\}$.

The framework identifies the *version space*, which is the set $\mathcal{V}$ of all the hypotheses of $\mathcal{H}$ that are sound with $\mathcal{S}$. The idea is that a "good" hypothesis should at least classify the training examples correctly.

**Definition 18** (*Version space.*) $\mathcal{V} = \{h \in \mathcal{H} |$ h is sound with $\mathcal{S}\}$.

Version space learning aims at identifying the *upper* and the *lower* bounds of this version space $\mathcal{V}$. The upper bound will contain the most general hypotheses, i.e., the ones that classify more examples, whereas the lower bound will contain the most specific ones, i.e., the hypotheses that classify less examples.

**Definition 19** (*General/specific hypotheses*)

- The set of general hypotheses is $\mathcal{V}_G = \{h \in \mathcal{H} | h$ is sound with $\mathcal{S}$ and $\nexists h' \in \mathcal{H}$ with $h'$ sound with $\mathcal{S}$, and $h' \succeq h\}$.
- The set of specific hypotheses is $\mathcal{V}_S = \{h \in \mathcal{H} | h$ is sound with $\mathcal{S}$ and $\nexists h' \in \mathcal{H}$ with $h'$ sound with $\mathcal{S}$, and $h \succeq h'\}$.

From the above definition, we have the following simple property characterizing elements of $\mathcal{V}$.

**Proposition 15** [23]. $\mathcal{V} = \{h \in \mathcal{H} | \exists h_1 \in \mathcal{V}_S, \exists h_2 \in \mathcal{V}_G, h_2 \succeq h \succeq h_1\}$.

In [23], an algorithm that computes the version space $\mathcal{V}$ by identifying its upper and lower bounds $\mathcal{V}_S$ and $\mathcal{V}_G$ has been proposed.

The above framework has some limits. First, finding the version space is not sufficient for classifying examples out of the training set. This is due to possible conflicts between hypotheses. Second, it has been shown that the complexity of the algorithm that identifies $\mathcal{V}_S$ and $\mathcal{V}_G$ is very high. In order to palliate that limit, learning algorithms try in general to reach only one hypothesis in the version space by using heuristical exploration of $\mathcal{H}$ (from general to specific exploration, for instance FOIL [29], or from specific to general exploration, for instance PROGOL [25]). That hypothesis is then used for classifying new objects. Moreover, it is obvious that this framework does not support inconsistent set of examples:

**Proposition 16** [23] *If the set $\mathcal{S}$ of training examples is inconsistent, then the version space $\mathcal{V} = \emptyset$.*

A consequence of the above result is that no concept can be learned. This problem may appear in the case of noisy training data set. Let us now show how the ASC proposed in the previous section can retrieve the results of the version space learning, namely the version space and its lower and upper bounds. Before doing that, we start first by introducing some useful notations. Let Hyp be a function that returns for a given set of arguments, their non empty supports. In other words, this function returns all the hypotheses used to build arguments:

**Definition 20** Let $T \subseteq \mathcal{A}$. $\texttt{Hyp}(T) = \{h | \exists \langle h, x, u\rangle \in T \text{ and } h \neq \emptyset\}$

Now we will show that the argumentation-based model for concept learning computes in an elegant way the version space $\mathcal{V}$.

**Proposition 17** *Let $\langle \mathcal{A}, \texttt{Defeat}\rangle$ be an* ASC. *Let $\mathcal{E}$ be its grounded extension, and $\mathcal{E}_1, \ldots, \mathcal{E}_n$ its preferred (stable) extensions. If the set $\mathcal{S}$ is consistent then:*

$$\texttt{Hyp}(\mathcal{E}) = \texttt{Hyp}(\mathcal{E}_1) = \cdots = \texttt{Hyp}(\mathcal{E}_n) = \mathcal{V}$$

*where $\mathcal{V}$ is the version space.*

The above result is of great importance. It shows that to get the version space, one only needs to compute the grounded extension. We can also show that if a given argument is in an extension $\mathcal{E}_i$, then any argument based on a hypothesis from the version space that supports the same conclusion is in that extension. Formally:

**Proposition 18** *Let $\mathcal{E}_1, \ldots, \mathcal{E}_n$ be the extensions under a given semantics of $\langle \mathcal{A}, \texttt{Defeat}\rangle$. If $\langle h, x, u\rangle \in \mathcal{E}_i$, then $\forall h' \in \mathcal{V}$ s.t. $h' \neq h$ if $h'(x) = u$ then $\langle h', x, u\rangle \in \mathcal{E}_i$.*

Using the grounded extension, one can characterize the upper and the lower bounds of the version space as follows.

**Proposition 19** *Let $\langle \mathcal{A}, defeat\rangle$ be an* ASC, *and $\mathcal{E}$ its grounded extension.*

- $\mathcal{V}_G = \{h | \exists \langle h, x, u\rangle \in \mathcal{E}\ s.t \forall \langle h', x', u'\rangle \in \mathcal{E}, \langle h, x, u\rangle \texttt{Pref} \langle h', x', u'\rangle\}$.
- $\mathcal{V}_S = \{h | \exists \langle h, x, u\rangle \in \mathcal{E}\ s.t\ \forall \langle h', x', u'\rangle \in \mathcal{E}, \langle h', x', u'\rangle \texttt{Pref}^{\succ} \langle h, x, u\rangle\}$.

The upper bound corresponds to the set of hypotheses that are involved in most preferred arguments (w.r.t Pref) of the grounded extension, whereas the lower bound corresponds to the set of hypotheses involved in less preferred arguments of that extension.

## 6 Conclusion

Recently, some researchers have tried to use argumentation techniques in machine learning [24, 34]. On the contrary to the framework presented in this paper which makes a real bridge between argumentation and machine learning, in [24, 34] Bratko et al. consider argumentation as a tool for improving machine learning algorithms. In their approach, arguments are viewed as a bias for the hypotheses search through the hypothesis space. Arguments are provided by the user in order to describe some kind of preference on the syntax of hypotheses (sets of decision rules). Thus, arguments allows them to influence the choice of a hypothesis in the set of acceptable ones (which can be for instance the version space). But there is no guarantee that the use of arguments will increase the quality of the model found in terms of classification performance.

Another interesting work where argumentation is used for classifying concepts/examples is that done by Gomez and Chesnevar [19]. The authors started by noticing that existing classification models based on neural networks may classify the same example in different classes. In such a case, a random choice is made for choosing the class to keep. The authors have then suggested a hybrid approach that applies first the neural network-based model. In case of conflicts, i.e., an example is classified in different classes, an argumentation system is used to make the final choice in a rational way.

This paper has proposed, to the best of our knowledge, the first framework for classification that is completely argumentation-based, and that uses Dung's semantics. This framework considers the classification problem as a process that follows four main steps: it first constructs arguments in favor of classifications of examples from a set of training examples, and a set of hypotheses. Conflicts between arguments may appear when two arguments classify the same example in different classes. Arguments are then compared on the basis of their strengths. The idea is that arguments coming from training examples are stronger than arguments built from hypotheses. Similarly, arguments based on most preferred hypotheses are stronger than arguments built from less preferred hypotheses. We have shown that the extensions of acceptable arguments of the ASC retrieve and even characterize the version space and its upper and lower bounds. Thus, the argumentation-based approach gives another interpretation of the version space as well as its two bounds in terms of arguments. We have also shown that when the set of training examples is inconsistent, it is still possible to classify examples. Indeed, in this particular case, the version space is empty as it is the case in the version space learning framework. A last and not least feature of our model consists of defining the class of each example on the basis of all the hypotheses and not only one, and also to suggest two intuitive decision criteria for that purpose.

A first urgent extension of this framework would be to test the tractability of our approach. For that purpose, we will explore the proof theories in argumentation that test directly whether a given argument is in the grounded extension without computing this last. This means that one may know the class of an example without exploring the whole hypothesis space. We will start by experimenting the proof procedure proposed by Amgoud and Cayrol in [4], and compare its results to existing algorithms. Another interesting extension would be to study persuasion dialogues between different autonomous agents that try together to find the class of a given example.

**Acknowledgements** The authors would like to thank the three anonymous referees for their very useful comments that helped us improving the paper.

## Appendix

**Proposition 2** *Let $\mathcal{S}$ be a set of training examples.*

- $|\mathcal{S}| = |\mathcal{A}_{\mathcal{S}}|$[2].
- *If $\mathcal{S}$ is not empty, then $\mathcal{A}_{\mathcal{S}} \neq \emptyset$.*

*Proof* The first item follows from the above definition, and from the fact that an hypothesis $h$ cannot be empty. The second point follows directly from the first property, i.e. $|\mathcal{S}| = |\mathcal{A}_{\mathcal{S}}|$, and the assumption that $\mathcal{S} \neq \emptyset$. □

**Proposition 3** *Let $x \in \mathcal{X}$. $|\{\langle h_i, x, c_i\rangle \in \mathcal{A}$ such that $h_i \neq \emptyset\}| = |\mathcal{H}|$.*

*Proof* This follows directly from the fact that:

[2] || denotes the cardinal of a given set.

(1) Each hypothesis classifies all examples of $\mathcal{X}$, thus $\forall h_i \in \mathcal{H}, \exists c_i \in \mathcal{C}$ such that $h_i(x) = c_i$.
(2) Each hypothesis assigns a unique class to each example since the classifications of each hypothesis are consistent. □

**Proposition 4** *The relation Pref is a partial preorder.*

*Proof* This is due to the fact that the relation $\succeq$ is a partial preorder. □

**Proposition 5** *If $\mathcal{S}$ is consistent, then $\nexists A, B \in \mathcal{A}_{\mathcal{S}}$ such that $A$ rebuts $B$, or $A$ undercuts $B$.*

*Proof* Let $A = \langle \emptyset, x, u\rangle$, $B = \langle \emptyset, x', u'\rangle \in \mathcal{S}$ such that $A$ rebuts $B$. According to Definition 9, $x \equiv x'$ and $u \neq u'$. This contradicts the fact that $\mathcal{S}$ is consistent (in the sense of Definition 1). □

**Proposition 6** *$\forall A \in \mathcal{A}_{\mathcal{S}}$, $\nexists B \in \mathcal{A} \backslash \mathcal{A}_{\mathcal{S}}$ s.t $B$ defeats $A$.*

*Proof* Let $A \in \mathcal{A}_{\mathcal{S}}$ and $B \in \mathcal{A} \backslash \mathcal{A}_{\mathcal{S}}$ such that $B$ defeats $A$. This means that $B$ rebuts $A$ (because according to Definition 10, an argument with a non-empty support cannot undercut an argument with an empty one. Moreover, according to Definition 11, we have not($B\,\mathtt{Pref}^{\succ}$ $A$). This is impossible because according to Definition 8, arguments in $\mathcal{A}_{\mathcal{S}}$ are always preferred to arguments with a non-empty support. □

**Proposition 7** *Let $\langle \mathcal{A}, \mathtt{Defeat}\rangle$ be a* ASC*, and $\mathcal{E}_1, \ldots, \mathcal{E}_n$ its extensions under a given semantics: $UV \subseteq MV$.*

*Proof* Let $x \in \mathcal{X}$. It is clear from the definition of universally classified examples that if $(x, c)$ is universally classified, then $\sum_{\exists \langle h,x,c\rangle \in \mathcal{E}_i} \mathcal{E}_i = n$. Moreover, since according to Proposition 14, the classifications of each extension are consistent, then $\sum_{\exists \langle h',x,c'\rangle \in \mathcal{E}_j} \mathcal{E}_j = 0$. Thus, $(x, c)$ is also classified by the majority of extensions. □

**Proposition 8** *If $\mathcal{S}$ is consistent, then $\mathcal{A}_{\mathcal{S}} \subseteq \mathcal{U}$.*

*Proof* Let $A \in \mathcal{A}_{\mathcal{S}}$. Let us assume that $\exists B \in \mathcal{A}$ such that $B$ defeats $A$. According to Proposition 6, $B \notin \mathcal{A} \backslash \mathcal{A}_{\mathcal{S}}$. Thus, $B \in \mathcal{A}_{\mathcal{S}}$. Moreover, $B$ defeats $A$ means that $B$ rebuts $A$ since an argument coming from a training example is not allowed to undercut another argument coming from a training example. This means then that $A$ classifies a training example in $u$, and $B$ classifies an equivalent example in $u' \neq u$. This contradicts the fact that the set $\mathcal{S}$ is consistent. □

**Proposition 9** *If $\mathcal{H}$ and $\mathcal{X}$ are finite, then the system $\langle \mathcal{A}, \mathtt{Defeat}\rangle$ is finite.*

*Proof* This follows directly from the fact that the set $\mathcal{A}$ of arguments is built from three sets: $\mathcal{S}$, $\mathcal{H}$ and $\mathcal{X}$. We assumed in our framework that the set $\mathcal{S}$ is finite. When the two sets $\mathcal{H}$ and $\mathcal{X}$ are also assumed finite, then the set of arguments $\mathcal{A}$ is finite. Consequently, the system $\langle \mathcal{A}, \mathtt{Defeat}\rangle$ is finite. □

**Proposition 10** *If the argumentation system $\langle \mathcal{A}, \mathtt{Defeat}\rangle$ is finite, then the grounded extension $\mathcal{E}$ is:*

$$\mathcal{E} = \bigcup \mathcal{F}^{i\geq 0}(\emptyset) = \mathcal{U} \cup \left[\bigcup_{i\geq 1} \mathcal{F}^{i}(\mathcal{U})\right].$$

*Proof* When the argumentation system $\langle \mathcal{A}, \texttt{Defeat} \rangle$ is finite, then the characteristic function $\mathcal{F}$ given in Definition 5 is continuous. Consequently, its least fixpoint (thus, the grounded extension) is $\mathcal{E} = \bigcup \mathcal{F}^{i \geq 0}(\emptyset)$. It is also clear that $\mathcal{F}(\emptyset) = \mathcal{U}$. Thus, $\mathcal{E} = \bigcup \mathcal{F}^{i \geq 0}(\emptyset) = \mathcal{U} \cup [\bigcup_{i \geq 1} \mathcal{F}^i(\mathcal{U})]$. □

**Proposition 11** *If $\mathcal{S}$ is consistent, then the argumentation system $\langle \mathcal{A}, \texttt{Defeat} \rangle$ has a non empty grounded extension $\mathcal{E}$.*

*Proof* According to Proposition 2, the set $\mathcal{A}_\mathcal{S} \neq \emptyset$. Moreover, according to Proposition 8, it has been shown that when $\mathcal{S}$ is consistent, then $\mathcal{A}_\mathcal{S} \subseteq \mathcal{U}$. Thus, $\mathcal{U} \neq \emptyset$ in this case. Finally, according to Proposition 10, the grounded extension is $\mathcal{E} = \bigcup \mathcal{F}^{i \geq 0}(\emptyset) = \mathcal{U} \cup [\bigcup_{i \geq 1} \mathcal{F}^i(\mathcal{U})]$. Thus, $\mathcal{U} \subseteq \mathcal{E}$. Consequently, $\mathcal{E}$ is not empty. □

**Proposition 12** *If $\mathcal{S}$ is consistent, then the system $\mathsf{ASC} = \langle \mathcal{A}, \texttt{Defeat} \rangle$ has $n \geq 1$ non-empty preferred extensions.*

*Proof* In [14], it has been shown that the grounded extension is included in every preferred extension. Since the grounded extension is not empty (according to Proposition 11), then there exists at least one non-empty preferred extension. □

**Proposition 13** *If $\mathcal{C} = \{c_1, c_2\}$ with $c_1 \neq c_2$, then:*

- *The graph associated with the system $\langle \mathcal{A}, \texttt{Defeat} \rangle$ has no odd length cycles.*
- *The system $\langle \mathcal{A}, \texttt{Defeat} \rangle$ is coherent.*

*Proof* Part 1: Let $A, B, C$ be three arguments such that $A$ defeats $B$, $B$ defeats $C$, and $C$ defeats $A$.

*Case 1*: Let us suppose that $A \in \mathcal{A}_\mathcal{S}$.
According to Property 5, $B \in \mathcal{A} \backslash \mathcal{A}_\mathcal{S}$. According to Proposition 6, $C$ should be in $\mathcal{A} \backslash \mathcal{A}_\mathcal{S}$. Contradiction because according to Proposition 6, $C$ cannot defeat $A$, which is in $\mathcal{A}_\mathcal{S}$.

*Case 2*: Let us suppose that $A, B, C \in \mathcal{A} \backslash \mathcal{A}_\mathcal{S}$. This means that $A$ rebuts $B$, $B$ rebuts $C$, and $C$ rebuts $A$ (according to Definition 10). Consequently, $\texttt{Example}(A) \equiv \texttt{Example}(B) \equiv \texttt{Example}(C)$, and $\texttt{Class}(A) \neq \texttt{Class}(B)$, $\texttt{Class}(B) \neq \texttt{Class}(C)$. Due to the fact that $\mathcal{U} = \{0, 1\}$, we have $\texttt{Class}(A) = \texttt{Class}(C)$. This contradicts the assumption that $C$ rebuts $A$.

Part 2: This is a consequence of the fact that there is no odd cycles in the system (see Part 1). □

**Proposition 14** *Let $\mathsf{ASC} = \langle \mathcal{A}, \texttt{Defeat} \rangle$, $\mathcal{E}_1, \ldots, \mathcal{E}_n$ its extensions under a given semantics, and $UV$, $MV$ its sets of conclusions.*

- *$\forall \mathcal{E}_i$, the set $\{(x, c) | \exists \langle h, x, c \rangle \in \mathcal{E}_i\}$ is consistent.*
- *The sets $UV$ and $MV$ are consistent.*

*Proof* Let $\mathsf{ASC} = \langle \mathcal{A}, \texttt{Defeat} \rangle$, $\mathcal{E}_1, \ldots, \mathcal{E}_n$ its extensions under a given semantics, and $UV$, $MV$ its sets of conclusions.

(1) Let $\mathcal{E}$ be a given preferred extension. Let us assume that the set $\{(x, c) | \exists \langle h, x, c \rangle \in \mathcal{E}\}$ is inconsistent. This means that $\exists \langle h_1, x, c_1 \rangle, \langle h_2, x, c_2 \rangle \in \mathcal{E}$ with $c_1 \neq c_2$. Thus $\langle h_1, x, c_1 \rangle$ rebuts $\langle h_2, x, c_2 \rangle$. There are three cases:

*Case 1* $h_1 = \emptyset$ and $h_2 = \emptyset$: Since training examples have the same importance, then $\langle h_1, x, c_1\rangle$ defeats $\langle h_2, x, c_2\rangle$ and $\langle h_2, x, c_2\rangle$ defeats $\langle h_1, x, c_1\rangle$. This means that the set $\mathcal{E}$ is not conflict-free. This contradicts the fact that $\mathcal{E}$ is a preferred extension.

*Case 2* $h_1 = \emptyset$ and $h_2 \neq \emptyset$ (or $h_1 \neq \emptyset$ and $h_2 = \emptyset$): Since $h_1 = \emptyset$ then $\langle h_1, x, c_1\rangle$ defeats $\langle h_2, x, c_2\rangle$. This means that the set $\mathcal{E}$ is not conflict-free. This contradicts the fact that $\mathcal{E}$ is a preferred extension.

*Case 3* $h_1 \neq \emptyset$ and $h_2 \neq \emptyset$: Since $\langle h_1, x, c_1\rangle, \langle h_2, x, c_2\rangle \in \mathcal{E}$ and $\mathcal{E}$ is conflict-free this means that $\langle h_1, x, c_1\rangle$ does not defeat $\langle h_2, x, c_2\rangle$ and $\langle h_2, x, c_2\rangle$ does not defeat $\langle h_1, x, c_1\rangle$. This means also that $\langle h_1, x, c_1\rangle \mathtt{Pref}^{\succ} \langle h_2, x, c_2\rangle$ and $\langle h_2, x, c_2\rangle \mathtt{Pref}^{\succ} \langle h_1, x, c_1\rangle$. This is impossible.

(2) Let us assume that the set $UV$ is inconsistent. This means that $\exists \mathcal{E}_i, \mathcal{E}_j$ such that $i \neq j$ and $\exists \langle h_i, x, c_i\rangle \in \mathcal{E}_i$ and $\exists \langle h_j, x, c_j\rangle \in \mathcal{E}_j$ with $c_i \neq c_j$. However, according to the definition of $UV$, $\exists \langle h'_j, x, c_i\rangle \in \mathcal{E}_j$ since $(x, c)$ is universally classified. This means that the set of conclusions of $\mathcal{E}_j$ is inconsistent. This contradict the result of Part 1 above.

Let us now assume that the set $MV$ is inconsistent. This means that $(x, c)$ is classified by the majority of extensions, say by $y$ extensions, and that $(x, c')$ is classified by the majority of extensions, say by $Z$ extensions. According to the definition of $MV$, we have both $y > z$ and $z > y$. This is impossible. □

**Proposition 17** *Let* $\langle \mathcal{A}, \mathtt{Defeat}\rangle$ *be an* ASC. *Let* $\mathcal{E}$ *be its grounded extension, and* $\mathcal{E}_1, \ldots, \mathcal{E}_n$ *its preferred (stable) extensions. If the set* $\mathcal{S}$ *is consistent then:*

$$\mathtt{Hyp}(\mathcal{E}) = \mathtt{Hyp}(\mathcal{E}_1) = \cdots = \mathtt{Hyp}(\mathcal{E}_n) = \mathcal{V}$$

*where* $\mathcal{V}$ *is the version space.*

*Proof* Let $\mathcal{E}_i$ be an extension under a given semantics.

$\mathtt{Hyp}(\mathcal{E}_i) \subseteq \mathcal{V}$: Let $h \in \mathtt{Hyp}(\mathcal{E}_i)$, then $\exists \langle h, x, u\rangle \in \mathcal{E}_i$.
Let us assume that $\exists (x_i, u_i) \in \mathcal{S}$ such that $h(x_i) \neq u_i$. This means $\langle \emptyset, x_i, u_i\rangle$ undercuts $\langle h, x, u\rangle$ (according to Definition 10). Consequently, $\langle \emptyset, x_i, u_i\rangle$ defeats $\langle h, x, u\rangle$. However, according to Property 2, $\langle \emptyset, x_i, u_i\rangle \in \mathcal{A}_{\mathcal{S}}$, thus $\langle \emptyset, x_i, u_i\rangle \in \mathcal{E}_i$. Contradiction because $\mathcal{E}_i$ is an extension, thus by definition it is conflict-free.

$\mathcal{V} \subseteq \mathtt{Hyp}(\mathcal{E}_i)$: Let $h \in \mathcal{V}$, and let us assume that $h \notin \mathtt{Hyp}(\mathcal{E}_i)$. Since $h \in \mathcal{V}$, then $\forall (x_i, u_i) \in \mathcal{S}, h(x_i) = u_i$ (1)

Let $(x, u) \in \mathcal{S}$, thus $h(x) = u$ and consequently $\langle h, x, u\rangle \in \mathcal{A}$. Moreover, since $h \notin \mathtt{Hyp}(\mathcal{E})$, then $\langle h, x, u\rangle \notin E$. Thus, $\exists \langle h', x', u'\rangle$ that defeats $\langle h, x, u\rangle$.

- *Case 1*: $h' = \emptyset$. This means that $\langle \emptyset, x', u'\rangle$ undercuts $\langle h, x, u\rangle$ and $h(x') \neq u'$ Contradiction with (1).
- *Case 2*: $h' \neq \emptyset$. This means that $\langle h', x', u'\rangle$ rebuts $\langle h, x, u\rangle$. Consequently, $x \equiv x'$ and $u \neq u'$. However, since $h \in \mathcal{V}$, then $h$ is sound with $\mathcal{S}$. Thus, $\langle \emptyset, x, u\rangle$ defeats $\langle h', x', u'\rangle$, then $\langle \emptyset, x, u\rangle$ defeats $\langle h, x, u\rangle$. Since $\langle \emptyset, x, u\rangle \in \mathcal{S}$, then $\langle h, x, u\rangle \in \mathcal{F}(\mathcal{C})$ and consequently, $\langle h, x, u\rangle \in \mathcal{E}_i$. □

**Proposition 18** *Let* $\mathcal{E}_1, \ldots, \mathcal{E}_n$ *be the extensions under a given semantics of* $\langle \mathcal{A}, \mathtt{Defeat}\rangle$. *If* $\langle h, x, u\rangle \in \mathcal{E}_i$, *then* $\forall h' \in \mathcal{V}$ *s.t.* $h' \neq h$ *if* $h'(x) = u$ *then* $\langle h', x, u\rangle \in \mathcal{E}_i$.

*Proof* Let $\mathcal{E}_i$ be a given extension, and let $\langle h, x, u\rangle \in \mathcal{E}_i$. Let $h' \in \mathcal{V}$ such that $h'(x) = u$. Let us assume that $\langle h', x, u\rangle \notin \mathcal{E}_i$.

*Case 1*: $\mathcal{E}_i \cup \{\langle h', x, u\rangle\}$ is not conflict-free. This means that $\exists\langle h'', x'', u''\rangle \in \mathcal{E}_i$ such that $\langle h'', x'', u''\rangle$ defeats $\langle h', x, u\rangle$. Consequently, $\langle h'', x'', u''\rangle$ undercuts $\langle h', x, u\rangle$ if $h'' = \emptyset$, or $\langle h'', x'', u''\rangle$ rebuts $\langle h', x, u\rangle$ if $h'' \neq \emptyset$.
If $h'' = \emptyset$, then $h'(x'') \neq u''$, this contradicts the fact that $h' \in \mathcal{V}$.
If $h'' \neq \emptyset$, then $x'' \equiv x$ and $u'' \neq u$ and either $h'' \succeq h'$, or $h', h''$ are not comparable. Thus, $\langle h'', x'', u''\rangle$ rebuts $\langle h, x, u\rangle$. Since $\langle h, x, u\rangle$, $\langle h'', x'', u''\rangle \in \mathcal{E}_i$, then $h$ and $h''$ are not comparable. But, this means that $\langle h, x, u\rangle$ defeats $\langle h'', x'', u''\rangle$, and $\langle h'', x'', u''\rangle$ defeats $\langle h, x, u\rangle$. Consequently, $\mathcal{E}_i$ is not conflict-free. Contradiction because $\mathcal{E}_i$ is an extension.

Case 2: $\mathcal{E}_i$ does not defend $\langle h', x, u\rangle$. This means that $\exists\langle h'', x'', u''\rangle$ defeats $\langle h', x, u\rangle$.

- *Case 1*: $h'' = \emptyset$. This means that $h'(x'') \neq u''$. Contradiction because $h' \in \mathcal{V}$.
- *Case 2*: $h'' \neq \emptyset$. This means that $x \equiv x''$, $u \neq u''$, and $h'' \succeq h'$. Thus, $\langle h'', x'', u''\rangle$ rebuts $\langle h, x, u\rangle$.
If $h \succeq h''$, then $\langle h, x, u\rangle$ defeats $\langle h'', x'', u''\rangle$, thus $\langle h, x, u\rangle$ defends $\langle h', x, u\rangle$. If $h'' \succeq h$, then $\langle h'', x'', u''\rangle$ defeats $\langle h, x, u\rangle$. However, since $\langle h, x, u\rangle \in \mathcal{E}$, then $\mathcal{E}$ defends $\langle h, x, u\rangle$ against $\langle h'', x'', u''\rangle$. Thus, $\mathcal{E}$ defends $\langle h', x, u\rangle$. Contradiction □

**Proposition 19** *Let $\langle \mathcal{A}, \texttt{Defeat}\rangle$ be an* ASC*, and $\mathcal{E}$ its grounded extension.*

- $\mathcal{V}_G = \{h | \exists\langle h, x, u\rangle \in \mathcal{E}\ s.t\ \forall\ \langle h', x', u'\rangle \in \mathcal{E}, \langle h, x, u\rangle\ \texttt{Pref}\langle h', x', u'\rangle\}$.
- $\mathcal{V}_S = \{h | \exists\langle h, x, u\rangle \in \mathcal{E}\ s.t\ \forall\ \langle h', x', u'\rangle \in \mathcal{E}, \langle h', x', u'\rangle\ \texttt{Pref}^{\succ}\langle h, x, u\rangle\}$.

*Proof*

$\mathcal{V}_G = \{h | \exists\langle h, x, u\rangle \in \mathcal{E}\ s.t\ \forall\ \langle h', x', u'\rangle \in \mathcal{E},\ not(\langle h', x', u'\rangle\ \texttt{Pref}\langle h, x, u\rangle)\}$.

- Let $h \in \mathcal{V}_G$, thus $h \in \mathcal{V}$, and $\forall h' \in \mathcal{V}$, $h \succeq h'$. Since $h \in \mathcal{V}$, thus, $h \in \texttt{Hyp}(\mathcal{E})$, with $\mathcal{E}$ an extension. Then, $\exists\langle h, x, u\rangle \in \mathcal{E}$. Since $h \succeq h'$ for any $h' \in \mathcal{V}$, then $h \succeq h'$ for any $h' \in \texttt{Hyp}(\mathcal{E})$. Thus, $\langle h, x, u\rangle\texttt{Pref}\langle h', x', u'\rangle$, $\forall\langle h', x', u'\rangle \in \mathcal{E}$.
- Let $\langle h, x, u\rangle \in \mathcal{E}$ such that $\forall\langle h', x', u'\rangle \in \mathcal{E}$, and not($\langle h', x', u'\rangle\texttt{Pref}\langle h, x, u\rangle$). Thus, $h \in \texttt{Hyp}(\mathcal{E})$, and $\forall h' \in \texttt{Hyp}(\mathcal{E})$, not($h' \succeq h$), thus $h \in \mathcal{V}_G$.

$\mathcal{V}_S = \{h | \exists\langle h, x, u\rangle \in \mathcal{E}\ s.t\ \forall\ \langle h', x', u'\rangle \in \mathcal{E},\ not(\langle h, x, u\rangle\ \texttt{Pref}\langle h', x', u'\rangle)\}$.

- Let $h \in \mathcal{V}_S$, thus $\nexists h' \in \mathcal{V}$ such that $h \succeq h'$. Since $h \in \mathcal{V}_S$, then $h \in \mathcal{V}$ and consequently, $h \in \texttt{Hyp}(\mathcal{E})$. This means that $\exists\langle h, x, u\rangle \in \mathcal{E}$. Let us assume that $\exists\langle h', x', u'\rangle \in \mathcal{E}$ such that $\langle h, x, u\rangle\texttt{Pref}\langle h', x', u'\rangle$, thus $h \succeq h'$. Contradiction with the fact that $h \in \mathcal{V}_S$.
- Let $\langle h, x, u\rangle \in \mathcal{E}$ such that $\forall\langle h', x', u'\rangle \in \mathcal{E}$, and not($\langle h, x, u\rangle\texttt{Pref}\langle h', x', u'\rangle$), thus not($h \succeq h'$). Since $h \in \mathcal{V}$, and $\forall h' \in \mathcal{V}$, not($h \succeq h'$), then $h \in \mathcal{V}_S$. □

## References

1. Amgoud, L. (2003). A formal framework for handling conflicting desires. In *7th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty* (pp. 552–563). LNAI 2711.
2. Amgoud, L., Belabbes, S., & Prade, H. (2005). Towards a formal framework for the search of a consensus between autonomous agents. In *4th International Joint Conference on Autonomous Agents and Multi-Agent Systems* (pp. 537–543).

3. Amgoud, L., & Cayrol, C. (2002). Inferring from inconsistency in preference-based argumentation frameworks. *International Journal of Automated Reasoning, 29*(2), 125–169.
4. Amgoud, L., & Cayrol, C. (2002). A reasoning model based on the production of acceptable arguments. *Annals of Mathematics and Artificial Intelligence, 34*, 197–216.
5. Amgoud, L., & Kaci, S. (2005). An argumentation framework for merging conflicting knowledge bases: The prioritized case. In *8th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*.
6. Amgoud, L., Maudet, N., & Parsons, S. (2000). Modelling dialogues using argumentation. In *4th International Conference on MultiAgent Systems*, ICMAS 2000. Boston, USA: IEEE Press.
7. Amgoud, L., & Parsons, S. (2002). An argumentation framework for merging conflicting knowledge bases. In *8th European Conference on Logics in Artificial Intelligence* (pp. 27–37). LNCS 2424.
8. Amgoud, L., & Prade, H. (2006). Explaining qualitative decision under uncertainty by argumentation. In *National Conference on Artificial Intelligence* (pp. 219–224). AAAI Press.
9. Amgoud, L., Serrurier, M. (2007). Arguing and explaining classifications. In O. Sheory & M. Huhns (Eds.), *International Joint Conference on Autonomous Agents and Multiagent Systems* (AAMAS 2007) (pp. 979-985). ACM Press.
10. Black, E., & Hunter, A. (2007). A generative inquiry dialogue system. In *6th International Joint Conference on Autonomous Agents and Multi-Agents systems*.
11. Bonet, B., & Geffner, H. (1996). Arguing for decisions: A qualitative model of decision making. In *12th Conference on Uncertainty in Artificial Intelligence* (pp. 98–105).
12. Breiman, O. S. (1984). *Friedman. Classification and decision trees*. Wadsworth Press.
13. Caminada, M., & Amgoud, L. (2007). On the evaluation of argumentation formalisms. *Artificial Intelligence Journal*, *171*(5–6), 286–310.
14. Dung, P. M. (1995). On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and $n$-person games. *Artificial Intelligence Journal, 77*, 321–357.
15. Dunne, P., Capon, T. B. (2002). Coherence in finite argument systems. *Artificial Intelligence journal, 141*(1–2), 187–203.
16. Elvang-Gransson, M., Krause, P., & Fox, J. (1993). Acceptability of arguments as 'logical uncertainty' In *Proceedings of the European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty* (pp. 85–90).
17. Fox, J., & Parsons, S. (1997). On using arguments for reasoning about actions and values. In *Proceedings of the AAAI Spring Symposium on Qualitative Preferences in Deliberation and Practical Reasoning, Stanford*.
18. Gómez, S. A., & Chesñevar, C. I. (2003). Integrating defeasible argumentation with fuzzy art neural networks for pattern classification. In *Proceedings of the ECML'03*, Dubrovnik, September 2003.
19. Gomez, S. A., & Chesnevar, C. I. (2004). A hybrid approach to pattern classification using neural networks and defeasible argumentation. In *17th International FLAIRS 2004 Conference* (pp. 393–398). AAAI Press.
20. Hulstijn, J., & van der Torre, L. (2004). Combining goal generation and planning in an argumentation framework. In J. Delgrande, & T. Schaub (Eds.), *10th Workshop on Non-Monotonic Reasoning*.
21. Kakas, A., Moraitis, P. (2006). Adaptive agent negotiation via argumentation. In *Proceedings of the 5th International Joint Conference on Autonomous Agents and Multi-Agents systems* (pp. 384–391).
22. Kraus, S., Sycara, K., & Evenchik, A. (1998). *Reaching agreements through argumentation: A logical model and implementation*. *Journal of Artificial Intelligence, 104*(1–2), 1–69.
23. Mitchell, T. (1982). Generalization as search. *Artificial intelligence, 18*, 203–226.
24. Mozina, M., Zabkar, J., & Bratko, I. (2006). Argument based rule learning. In *17th European Conference on Artificial Intelligence* (pp. 504–508).
25. Muggleton, S. (1995). Inverse entailment and Progol. *New Generation Computing, 13*, 245–286.
26. Parsons, S., & Jennings, N. R. (1996). Negotiation through argumentation—a preliminary report. In *Proceedings of the 2nd International Conference on Multi Agent Systems* (pp. 267–274).
27. Prakken, H. (2006). Formal systems for persuasion dialogue. *Knowledge Engineering Review, 21*, 163–188.
28. Quinlan, J. R. (1987). Simplifying decision trees. *International Journal of Man-Machine Studies, 27*, 221–234.
29. Quinlan, J. R. (1990). Learning logical definitions from relations. *Machine Learning, 5*, 239–266.
30. Quinlan, J. R. (1993). A decision science perspective on decision trees. In *Programs for Machine Learning*. Morgan Kauffman.
31. Rahwan, I., & Amgoud, L. (2006). An argumentation-based approach for practical reasoning. In *International Joint Conference on Autonomous Agents and Multiagent Systems*.

32. Simari, G. R., & Loui, R. P. (1992). A mathematical treatment of defeasible reasoning and its implementation. *Artificial Intelligence and Law, 53*, 125–157.
33. Walton, D. N., & Krabbe, E. C. W. (1995). *Commitment in dialogue: Basic concepts of interpersonal reasoning*. SUNY Series in Logic and Language. Albany: State University of New York Press.
34. Zabkar, J., Mozina, M., Videcnik, J., & Bratko, I. (2006). Argument based machine learning in a medical domain. In I. Press (Ed.), *Proceedings of the 1st International Conference on Computational Models of Argument* (pp. 59–70).