*Computational Models of Argument*
*P. Baroni et al. (Eds.)*


*doi:10.3233/978-1-61499-686-6-219*

# Argumentation for Machine Learning: A Survey

Oana COCARASCU [a,1], Francesca TONI [a]
[a] *Department of Computing, Imperial College London, UK*

**Abstract.** Existing approaches using argumentation to aid or improve machine learning differ in the type of machine learning technique they consider, in their use of argumentation and in their choice of argumentation framework and semantics. This paper presents a survey of this relatively young field highlighting, in particular, its achievements to date, the applications it has been used for as well as the benefits brought about by the use of argumentation, with an eye towards its future.

**Keywords.** Argumentation, Machine Learning

## 1. Introduction

Machine Learning (ML) [27] amounts to automatically learning from data and improving with experience. Nowadays, its use is becoming more and more important as much of the work on visual processing, language and speech recognition relies on it.

Argumentation (e.g. as overviewed in [37]) has proven successful in several domains, including multi-agent systems [6] and decision support in medicine [16] and engineering [2]. ML and argumentation are brought together in a number of settings, e.g. to support argument mining (e.g. see [26]) as well as to aid ML, in one sense or another. Also the integration of argumentation and applications of ML have been proven to be fruitful (e.g. as in [22]). In this paper we focus on the use of argumentation to aid ML and provide an overview of this relatively young field, with an eye to guide its future developments.

Existing approaches using argumentation for ML differ in the type of ML they consider and the specific method they use. Concretely:

- the Argumentation-Based Machine Learning (ABML) approach of [32] extends the CN2 rule induction algorithm [12] for *supervised learning*;
- the Argument-Based Inductive Logic Programming (ABILP) approach of [4] extends Inductive Logic Programming (ILP) for *supervised learning*;
- the hybrid approach of [21] uses as its starting point the Fuzzy Adaptive Resonance Theory (ART) model [7] for *unsupervised learning*;
- the fully argumentative concept learning method of [1] focuses on the version space learning framework [27] for *supervised learning*;

[1]Corresponding Author: Oana Cocarascu, Department of Computing, Imperial College London, United Kingdom; E-mail: oana.cocarascu11@imperial.ac.uk.

- the multi-agent inductive concept learning of [34] and its computational realisation [35] have concept learning [27] for *supervised learning* as their starting point;
- the Argumentation Accelerated Reinforcement Learning (AARL) of [17,18,19] extends SARSA [39] for *reinforcement learning*;
- the Classification enhanced with Argumentation (CleAr) method of [8,9] works with any *supervised learning* technique, and has been experimented with, in particular, Naïve Bayes classifiers [25], Support Vector Machines (SVMs) [13] and Random Forests [5].

Moreover, existing approaches differ in their use of argumentation and in their choice of argumentation framework/method. Finally, different approaches achieve different (desirable) outcomes, ranging from improving performances to rendering the ML process more transparent by improving its explanatory power.

The paper is organized as follows. In Section 2 we give an abstract re-interpretation of ML, in general and for supervised/unsupervised/reinforcement learning, to serve as a basis for a comparison amongst existing ML approaches using argumentation. In Section 3 we overview the different approaches using argumentation for ML, showing in particular how they use argumentation, and which kind thereof, to contribute to particular instantiations of the abstract model of Section 2. In Section 4 we provide a comparative analysis of the different approaches. In Section 5 we conclude, identifying in particular some challenges/open problems for an even more impactful use of argumentation in ML.

## 2. Machine Learning in the Abstract

In this section we give an abstract re-interpretation of ML, in general and for supervised/unsupervised/reinforcement learning, to serve as a basis for the comparison amongst different existing ML approaches using argumentation. Being tailored to providing an overview of existing approaches to ML *using argumentation*, this abstract interpretation has no pretence of being general or fully covering (e.g. it completely ignores the use of probabilistic information in ML).

In the abstract, a ML method can be characterised in terms of the following notions, that will be instantiated differently for the different ML methodologies (supervised/unsupervised/reinforcement learning) and for the different methods for the methodologies (e.g. CN2 for supervised learning, ART for unsupervised learning, and SARSA for reinforcement learning):

- $H$ is the *hypotheses space*, namely the set of all possible "reasoners" that a ML method may return;
- $\mathscr{S}$ is the *training* input, given to the ML method to trigger the learning process leading to generating a "reasoner" in $H$;[2]
- $X$ is the set of all possible descriptions of inputs for the ML method (the training input) and for the "reasoner" learnt by the ML method (the *unseen* input, e.g. used for testing);
- $\mathscr{L}$ is the set of all possible outputs that "reasoners" computed by a ML method may return, given the inputs.

[2]Note that the training input excludes any *testing* input, to be deployed after learning has taken place to test the computed "reasoners".

### 2.1. Supervised learning

In this setting a "reasoner" in $H$ is a classifier, $\mathscr{L}$ is a set of alternative classifications, $X$ is a set of combinations of features that inputs may exhibit, and each element of the training input includes the correct classification for a given combination of features, whereas the unseen inputs only consist of features:

- $X$ is the *feature space*; for example, a feature may be an attribute/value pair;
- $\mathscr{L}$ is the set of possible *classifications*; for example, if the aim of the supervised learning method is to learn a concept, then $\mathscr{L} = \{0,1\}$;
- $\mathscr{S}$ is the set of *training instances*; a training instance is of the form $(x,l)$ for $x \in X$ and $l \in \mathscr{L}$; for example, if the aim of the supervised learning method is to learn a concept, then $(\{f_1,f_2\},1)$ indicates that the combination of features $f_1,f_2$ is an example of the concept, and $(\{f_1,f_3\},0)$ indicates that the combination of features $f_1,f_3$ is not;
- a generic member $h_s$ of $H$ can be abstractly seen as a mapping $h_s : X \mapsto \mathscr{L}$; at an abstract level, the goal of a supervised ML method is to determine a classifier $h_s$ such that (i) $h_s(x) = l$ for all (or for as many as possible) $(x,l) \in \mathscr{S}$ and (ii) $h_s$ generalises well by classifying instances not in $\mathscr{S}$ correctly (during testing).

### 2.2. Unsupervised learning

In this setting, a "reasoner" in $H$ is also a classifier, but the training instances in $\mathscr{S}$ are given in terms of their feature combinations only, as a correct classification for them is not available; the most popular unsupervised ML methods then compute *clusters* an input may belong to and determine the classifier/classification of the input using the clusters: here a cluster is a collection of instances which are "similar", while being "dissimilar" to instances in other clusters [27]. Thus:

- $X$ is the feature space, as in supervised learning, and $\mathscr{S} \subseteq X$; for example, inputs may be images of different fruits, and features may include pixels in these images;
- $\mathscr{L}$ is obtained from the "learnt" clusters; for example, one cluster may group together apples and another oranges;
- a generic member $h_u$ of $H$ can be seen as a mapping $h_u : X \mapsto \mathscr{L}$; abstractly, the goal of (cluster-based) unsupervised learning is to find a "good" way to assign inputs to clusters, as a basis for classification.

### 2.3. Reinforcement learning

In this setting, a "reasoner" is a *policy*, that, given inputs in the form of observations of *states*, returns outputs in the form of *actions*. Actions, during learning, are not known to be right or wrong, and thus classifications are not available. Instead, *rewards* are given for states reached by performing actions (these rewards are positive if the states are "desirable" and negative otherwise; negative rewards can be interpreted as punishments). Thus:

- $\mathscr{L}$ is the set of *actions* that can be performed by the learner; for example, if the learner is a robot, actions may include moves in several directions;

- $X$ is the set of all possible states; for example, a state may represent the physical environment a robot is situated in;
- $\mathscr{S}$ is the *reward function*, namely a mapping $\mathscr{S} : X \mapsto \mathbb{R}$; for example, a goal state, that the robot should aim at achieving, could be given a high reward;
- $H$ is the *policy space*, and a generic member $h_r$ of $H$ is a mapping $h_r : X \mapsto \mathscr{L}$; at an abstract level, the goal of a reinforcement learning method is to determine, with as little training as possible, a policy $h_r$ which is optimal, namely it gets the highest possible cumulative reward.

## 3. Approaches to Argumentation for Machine Learning

Here we show how existing approaches use argumentation, and which form thereof, for ML, in the context of suitable instantiations of the abstract model in Section 2.

### 3.1. Argumentation for Supervised Learning

*ABML [32].* Here arguments are associated with elements of $\mathscr{S}$ and are of the form:

$C$ because *Reasons* or

$C$ despite *Reasons*

where $C \in \mathscr{L}$ and $Reasons \subseteq X$. The first type of argument provides reasons (in terms of combinations of features) for why a certain training instance is classified as is, whereas the second type of argument indicates combinations of features that do not play a role in classifying the training instance the argument is associated with.

For example, let $\mathscr{S}$ represent credit applications and $\mathscr{L}$ represent whether the credit was approved or not. Then, a training instance $(x,l) \in \mathscr{S}$ with

$x = \{PaysRegularly = no, Rich = yes, HairColor = blond\}$

$l = CreditApproved$

may be associated with arguments

$$CreditApproved \text{ because } Rich = yes \tag{1}$$

$$CreditApproved \text{ despite } PaysRegularly = no \tag{2}$$

In ABML, arguments of these two forms are used to modify the CN2 rule induction algorithm for supervised ML so as to learn "better" hypotheses, while also reducing the size of $H$ and providing explanations for classifications. Here, hypotheses are rules of the form IF $F_1$ AND ... AND $F_n$ THEN $C$, for $n > 0$, $F_i \in X$, $C \in \mathscr{L}$.

For example, from the credit approval example above, CN2 alone may obtain the rule IF $HairColor = blond$ THEN $CreditApproved$, whereas using argument (1), ABML may obtain the "better" rule IF $HairColor = blond$ AND $Rich = yes$ THEN $CreditApproved$.

ABML does not make use of any particular argumentation framework in the literature, but uses ad hoc arguments of the forms given earlier, suitable for the specific ML setting considered. Moreover, ABML does not make use of any argumentation semantics or methodology for assessing the acceptability/strength of arguments (these are taken at face value instead).

*ABILP [4].* Here arguments are as in ABML but they are integrated within ILP.

*Concept Learning as Argumentation (CLA) [1].* This method reinterprets concept learning in argumentation terms. Here arguments are obtained from $\mathscr{S}$ and $H$ and are of the form $\langle h,x,l\rangle$ for $h \in H \cup \{\emptyset\}$, $x \in X$ and $l \in \mathscr{L}$ such that

if $h = \emptyset$ then $(x,l) \in \mathscr{S}$, and

if $h \neq \emptyset$ then $h(x) = l$,

namely each training instance in $\mathscr{S}$ and each hypothesis in $H$ gives an argument. Moreover, an argument *a attacks* an argument *b* by *rebutting* if the two arguments give different classifications for the same features, or by *undercutting* if *a* is drawn from an example and *b* is drawn from a hypothesis which disagrees with the example.

This method then uses standard semantics of extensions [14] applied to abstract argumentation frameworks with arguments obtained from $\mathscr{S}$ and $H$ as above, and a relation of *defeat* between arguments such that *a* defeats *b* iff *a* attacks *b* by rebutting or undercutting and *b* is not *preferred* to *a*, where given a preference relation over $H$, standardly used in concept learning:

- arguments obtained from $\mathscr{S}$ are stronger than arguments obtained from $H$;
- arguments obtained from most preferred hypotheses are stronger than arguments obtained from less preferred hypotheses.

For example, consider $X = \{x_1,x_2\}$, $\mathscr{S} = \{(x_1,c_1),(x_1,c_2)\}$[3], $\mathscr{L} = \{c_1,c_2,c_3,c_4\}$ and $H = \{h_1,h_2\}$ with $h_1(x_1) = c_1$, $h_1(x_2) = c_1$, $h_2(x_1) = c_2$, and $h_2(x_2) = c_1$. The corresponding abstract argumentation framework has arguments $a_1 = \langle\emptyset,x_1,c_1\rangle$, $a_2 = \langle\emptyset,x_1,c_2\rangle$, $a_3 = \langle h_1,x_1,c_1\rangle$, $a_4 = \langle h_1,x_2,c_1\rangle$, $a_5 = \langle h_2,x_1,c_2\rangle$ and $a_6 = \langle h_2,x_2,c_1\rangle$. Also, assuming that the two hypotheses are equally preferred, the defeat relation is such that $a_1$ defeats $a_2$, $a_1$ defeats $a_5$, $a_1$ defeats $a_6$, $a_2$ defeats $a_1$, $a_2$ defeats $a_3$ and $a_2$ defeats $a_4$. The resulting abstract argumentation framework has an empty grounded extension and two preferred/stable extensions $\varepsilon_1 = \{a_1,a_3,a_4\}$ and $\varepsilon_2 = \{a_2,a_5,a_6\}$, both classifying $x_2$ as $c_1$.

The grounded extension of the abstract argumentation framework corresponding to a given concept learning setting corresponds to the output of the version space method for concept learning when the latter is applicable, namely when the given $\mathscr{S}$ is not inconsistent. Moreover, if $\mathscr{S}$ is inconsistent (as in our earlier illustration), argumentation can still return an output, e.g. $c_1$ or $c_2$ for $x_1$.

*Argumentation for Multi-Agent Inductive Concept Learning (MAICL) [34,35].* In this approach, $\mathscr{L} = \{0,1\}$ and $\mathscr{S}$ is assumed to be consistent as well as distributed amongst agents, so that each agent is only aware of some subset of $\mathscr{S}$. Arguments are hypotheses induced by individual agents from training instances they are aware of. These hypotheses/arguments are rules. For uniformity of presentation, we assume here that these rules/hypotheses/arguments are in the same form as the rules learnt by CN2, presented earlier. Then an argument IF $F_1$ AND ... AND $F_n$ THEN $C$ *attacks* an argument IF $F'_1$ AND ... AND $F'_m$ THEN $C'$ iff $C \neq C'$ and $\{F_1,\ldots,F_n\} \supseteq \{F'_1,\ldots,F'_m\}$.

For example, let $\mathscr{S} = \{e_1,e_2,e_3\}$ represent a mammal dataset where

$e_1 = (\{hair,milk,backbone\},1)$

$e_2 = (\{toothed,backbone,twolegged\},1)$

[3]Note that this set is *inconsistent*, as it classifies differently the same features.

$e_3 = (\{toothed, backbone\}, 0)$

and 1 stands for mammal, 0 stands for non-mammal. Consider two agents, $ag_1$ and $ag_2$, aware of $\{e_1, e_2\}$ and $\{e_3\}$, respectively, and let

IF *backbone* THEN 1

IF *backbone* AND *toothed* AND *twolegged* THEN 1

be the rules learnt by $ag_1$ and

IF *backbone* AND *toothed* THEN 0

be the rule learnt by $ag_2$. Then, $ag_1$'s second rule/argument attacks $ag_2$'s rule/argument.

In this approach, agents communicate arguments and attacks to construct dialectical trees as defined in [11,38] and determine which arguments are defeated/undefeated. For example, in the earlier illustration, $ag_2$'s rule/argument is defeated. Here, argumentation helps building hypotheses in a distributed manner when examples are not held centrally. Also, this method is supported by a computational realisation [35].

*CleAr [8,9].* In this approach, arguments and relations amongst them are drawn from a given set of templates (an *Argument base*) for a given *testing* instance that has already been classified by means of a "reasoner" (classifier) learnt by any standard supervised learning methods. The relations amongst arguments are of *attack* or *support* and thus the resulting argumentation frameworks, associated with training instances, are *bipolar* [10]. In addition, a *base score* is associated with arguments, as in QuAD frameworks [2,36]. Arguments are either elements of $\mathscr{L}$ or express domain knowledge of the learning task at hand and, in this latter case, are of the form

*Premise* $\Rightarrow$ *Conclusion*

where *Premise* may represent any information, including, but not limited to, combinations of elements of $X$, and *Conclusion* is either an element of $\mathscr{L}$ or it represents a statement agreeing or disagreeing with the *Premise* of some other argument.

For example, consider the task of determining sentiment polarity in tweets. Then $\mathscr{L} = \{positive, negative\}$ and $X$ are (syntactic or semantic) features extracted from tweets. Suppose that some existing classifier $h$ assigns positive polarity to the tweet:

*'more depressed than you could ever imagine that I wont be going to Vegas.*

*I hate having to be financially responsible'*

The resulting argumentation framework, for this testing instance, may include arguments *positive* and *negative* (the elements of $\mathscr{L}$) as well as arguments

'hate' occurs in the tweet $\Rightarrow$ *negative*

a negation ('wont') occurs in the tweet $\Rightarrow$ *negative*

and, in addition, that the arguments attack *positive* and/or support *negative*.

In this approach, base scores for the arguments are derived from the output or the performances of $h$ (the given classifier) or are drawn from the given Argument Base.

The dialectical strength of each classification in $\mathscr{L}$ is then computed using a quantitative semantics (e.g. as in [15,2,36]) and the classification with maximal strength is assigned as the final classification for the testing instance. In our earlier illustration, assuming that *positive* and *negative* have a base score of 0.6 and 0.4 respectively and the other two arguments above are supporters of *negative* and have a base score of 0.4, the computed strength may be 0.75 for *negative* and 0.6 for *positive*. Hence, the use of argumentation, in this case, would change the classification to *negative*. In general, in this approach, argumentation contributes a (possibly revised) classification and a justification thereof.

### 3.2. Argumentation for Unsupervised Learning

*Argumentation for ART (A-ART) [21].* In this approach, arguments, attacks and semantics are as in DeLP [20,11,38], but instantiated so as to reason with the output of a fuzzy ART network, when this assigns a training instance to different clusters. In this case, the classification choice for the given instance by the $h_u$ being learnt is, conventionally, that of a randomly chosen cluster. By arguing, instead, this choice can be "reasoned" upon.

As an example, consider a fuzzy ART network which identifies three clusters $c_1^+, c_2^-, c_3^- \in \mathscr{L}$ for an instance $e$, such that $c_1^+$ subsumes $c_3^-$. Suppose also that, from the given DeLP program, DeLP arguments can be constructed with the following informal reading:

$+$ because $e$ belongs to $c_1^+$

$-$ because $e$ belongs to $c_3^-$

with the second argument attacking the first but not vice versa, as $c_1^+$ subsumes $c_3^-$. Then, the dialectical analysis of [11,38] gives classification $-$, drawn from membership of $e$ in $c_3^-$.

### 3.3. Argumentation for Reinforcement Learning

*AARL [17,18,19].* In this approach, arguments represent recommendations of actions to individual agents in a multi-agent system and are of the form:

*Conclusion* IF *Premise*

where *Conclusion* is an action (in $\mathscr{L}$) to be performed by an agent and *Premise* describes conditions under which the argument is applicable and may, for example, amount to properties of the state (in $X$) of the environment where the agent is situated. Then an argument attacks another argument iff

- the arguments support the same action but for different agents, or
- the arguments support different actions by the same agent.

For example, in a given state of the environment in which a RoboCup agent is situated, the applicable arguments may be

agent $a_1$ should tackle the ball IF $a_1$ is closest to the ball keeper

agent $a_1$ should mark agent $a_2$ IF $a_1$ is closest to $a_2$

with the two arguments attacking one another. At each iteration of learning one such abstract argumentation framework is generated, by instantiating a set of argument templates given up-front, representing domain knolwedge.

AARL then uses preferences over arguments and adapts value-based argumentation [3] to choose actions (supported by arguments in some extension, e.g. the grounded extension) and shape rewards, thus modifying the reward function $\mathscr{S}$. For example, if tackling is more preferred than marking, for our earlier illustration, then the attack from the second to the first argument is deleted, as in value-based argumentation, and tackling gets extra reward at the current iteration of learning.

## 4. A Comparative Analysis of Argumentation for Machine Learning

In this section we provide a comparative analysis of the different approaches we have overviewed in Section 3. First, we note that existing approaches differ considerably in their choice of argumentation framework/semantics:

- ABML and ABILP use ad hoc arguments and no argumentation framework or semantics;
- CLA and AARL instantiate abstract argumentation, with arguments equipped with preferences, and deploy standard semantics of extensions;
- MAICL uses abstract argumentation, but deploys the dialectical trees of [11,38] as a semantics, rather than extensions;
- A-ART uses the DeLP argumentation framework and again the dialectical trees of [11,38] as a semantics;
- CleAr uses bipolar abstract argumentation extended with base scores or, equivalently, QuAD frameworks, and quantitative semantics.

Moreover, some approaches (i.e. ABML and AARL) use argumentation *during* learning, some (i.e. MAICL, CleAr and A-ART) use argumentation *after* learning, to process the output of standard ML techniques, and some (i.e. CLA) use argumentation *instead of* learning, to re-interpret the learning process. Furthermore, some approaches (i.e. MAICL and AARL) are developed to coordinate agents in multi-agent systems. Finally, different approaches are used for different applications and have different advantages over standard ML techniques, ranging from improving performances to rendering the ML process more transparent by improving its explanatory power or using argumentation to better elicit domain knowledge, of benefit to the learning process, from users. In the remainder of this section we analyse how the approaches overviewed in Section 3 have been applied and evaluated as well as their advantages.

*ABML [32].* Compared with standard CN2, ABML has the advantage of *reducing the size of the hypotheses space H*, in that it forces the rules to be learnt to take into account the arguments associated with the examples, and thus allowing fewer rules to be legitimate hypotheses.

ABML was tested on several domains (notably law [31], medicine [40] and zoology [28]), and was shown to *improve classification accuracy* across the board. For example, by including arguments, the accuracy was improved on a zoo dataset from 94.51% to 96.75% [28]. Also, on a dataset related to severe bacterial infections [40], ABML achieved similar accuracy to CN2 and a further ML technique, C4.5 (88%), whilst Naïve Bayes (NB) and Logistic Regression performed worse (with accuracy under 86.5%). Further, using AUC (Area Under the Curve, an alternative measure to standard accuracy), ABML outperformed all other classifiers, the improvement varying between 0.03% and 0.2%. ABML was also tested on chess, improving the initial accuracy of 72% to 95% when learning the concept of bad bishop [29] of 84% to 91% when learning the concept of an attack on the castled king [30].

ABML was shown to be *robust*, in the presence of noise in the examples as well as random arguments. Indeed, ABML performed better in the presence of *noise*, compared to CN2, on a welfare benefit dataset [31]: the class of each example was randomly replaced with a value from $\mathscr{L}$ with probability $p\%$ (for $p \in \{0,2,5,10,20,40\}$) with dis-

tribution (0.5, 0.5), and the average accuracy of ABML was better than CN2 by 0.3% at 0% noise, by 3.3% at 20% noise and by 1.7% at 40% noise. Moreover, ABML was tested in the presence of *random* arguments, and shown to still outperform or perform similarly to the original CN2 [32]: here, random arguments were given for $k$ randomly selected examples ($k \in \{2,5,10,20\}$), each example could have up to five random arguments and each argument could have up to five random reasons. Thus, ABML is robust in that it is not negatively affected by "bad" domain knowledge.

ABML has been shown to support *knowledge elicitation* well [23,24,41] by identifying *critical examples* (namely instances that the learnt hypotheses, using ABML, do not classify well) and eliciting arguments for them and retraining, using ABML again. On a medical dataset, this knowledge elicitation-enriched ABML increased the performance from 60% to 80% for CN2 [23] and, on a larger medical dataset, from 52% to 82% [24]. Knowledge elicitation was also employed during an interactive learning session using python code [41] to distinguish between classifications in $\mathscr{L} = \{basic, advanced\}$ programming style achieving 87.1% accuracy when using ABML compared to 86.7% manual student classification.

*ABILP [4].* This approach is in the same spirit as ABML. The advantages of this approach are potentially the same as for ABML.

*CLA [1].* The advantages of this approach are theoretical, rather than of an experimental nature. Indeed, CLA can handle inconsistent sets $\mathscr{S}$ of training instances, whereas standard concept learning cannot. Thus, the method is *robust*. In addition, by using argumentation, CLA supports in principle the generation of *explanations* for classifications.

*MAICL [34,35].* At a theoretical level, MAICL allows agents to agree classifications even when they hold *partial information*, in the form of subsets of the set $\mathscr{S}$ of training instances. A-MAIL [35], an implementation of a generalisation of MAICL, not restricted to $\mathscr{L} = \{0,1\}$, uses four datasets [33] to test experimentally whether this method can work in practice and, in particular, whether the method can cope with a large number of agents and several forms of data distribution. The experiments showed, in particular, that the use of A-MAIL can lead to a *recall increase*, which is higher for five agents, each having the same portion of $\mathscr{S}$, than with two agents. In the case of more agents (10 or 20), more examples need to be exchanged by communication, as expected, but recall increases can still be observed (e.g., with 20 agents, from 0.35% to 0.88%). In the case of unbalanced distributions of training instances between two agents when $ag_1$ receives only $p$% of $\mathscr{S}$ ($p \in \{50,30,10,0\}$), using A-MAIL results in an improvement in recall for $ag_1$ at the cost of arguments exchanged as $ag_1$ has more information to obtain from $ag_2$. Overall, the experiments show that A-MAIL can improve performances at a relatively reasonable cost in terms of number of messages being exchanged.

*CleAr [8,9].* CleAr has been applied to two problems within the computational linguistic setting: cross-domain sentiment polarity classification [8,9], with $\mathscr{L} = \{Positive, Negative\}$, and relation-based argument mining to determine relations between pieces of text [9], with $\mathscr{L} = \{Attack, Support, Neither\}$. In these two settings, CleAr has been instantiated with three types of supervised ML methods (i.e. NB, Support Vector Machines (SVM) and Random Forests (RF)) with suitably defined Argument Bases. Deploying CleAr with these Argument Bases gives an *increase in accuracy* of up to 14% for Sentiment Polarity Classification, from 50% to 64%, and *performance im-*

*provements* varying between 0.006% and 0.022% on various datasets for relation-based argument mining, with respect to using the standard ML methods alone.

*A-ART [21].* The advantages of this approach, as presented in [21], are theoretical, rather than of an experimental nature. Here, argumentation is used to *resolve inconsistency* amongst classifications of clusters to which an instance is assigned as well as to *explain* the final classification dialectically.

*AARL [17,18,19].* AARL has been deployed in RoboCup, and in particular for Keep-Away and TakeAway games, as well as other standard RL benchmarks. Experimentally, AARL, combined with a distance-oriented reward system, performs better overall when compared with SARSA or hand-coded strategies in terms of *stability*, *average convergence time* and *average optimal performance*. Moreover, this method is *robust* to errors in arguments.

| **Method** | **ML method** | **AF** | **Semantics** | **D/A ML** | **Multi agent** | **Advantages** | **Apps.** |
|---|---|---|---|---|---|---|---|
| ABML | CN2 | ✗ | ✗ | D | | experimental (accuracy, robustness); elicitation | law; medicine; zoology; chess; coding |
| ABILP | ILP | ✗ | ✗ | D | | | |
| CLA | concept learning | AA with prefs. | extensions | ✗ | | theoretical (inconsistency tolerance); explanation | |
| MAICL | concept learning | AA | dialectical trees | A | ✓ | experimental (recall); partial info | |
| CleAr | Random Forests; NB; SVM | Bipolar AA/ QuAD | quantitative | A | | experimental (accuracy) | Sentiment Analysis; Argument Mining |
| A-ART | Fuzzy ART | DeLP | dialectical trees | A | | explanation; inconsistency resolution | |
| AARL | SARSA | Value-based AA | extensions | D | ✓ | experimental (stability; convergence time; optimal performance) | RoboCup; Wumpus |

**Table 1.** Overview of approaches using argumentation to aid ML (D=During, A=After, Apps. = Applications).

## 5. Conclusion

We have surveyed existing approaches using argumentation to aid ML, focusing on the type of ML method they augment, the form of arguments and argumentation frameworks and semantics they deploy, as well as their advantages, ranging from improving performances to rendering the ML process more transparent by improving its explanatory power. Table 1 summarises our analysis.

The existing approaches show promise for further future developments and substantial potential impact in ML, to improve performances and allow the incorporation of domain knowledge by users as well as user-friendly explanations and transparency of the output of ML.

## References

[1] Amgoud, L., Serrurier, M.: Agents that argue and explain classifications. Autonomous Agents and Multi-Agent Systems 16(2), 187–209 (2007)
[2] Baroni, P., Romano, M., Toni, F., Aurisicchio, M., Bertanza, G.: Automatic evaluation of design alternatives with quantitative argumentation. Argument and Computation (2015)
[3] Bench-Capon, T.J.M., Atkinson, K.: Abstract argumentation and values. In: Rahwan, L., Simari, G. (eds.) Argumentation in Artificial Intelligence. Springer (2009)
[4] Bratko, I., Žabkar, J., Možina, M.: Argument-based machine learning. In: Simari, G., Rahwan, I. (eds.) Argumentation in Artificial Intelligence, pp. 463–482. Springer, 1st edn. (2009)
[5] Breiman, L.: Random forests. Machine Learning 45(1), 5–32 (Oct 2001)
[6] Bromuri, S., Urovi, V., Morge, M., Stathis, K., Toni, F.: A multi-agent system for service discovery, selection and negotiation. In: Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 2. pp. 1395–1396. AAMAS '09 (2009)
[7] Carpenter, G.A., Grossberg, S., Rosen, D.B.: Fuzzy art: Fast stable learning and categorization of analog patterns by an adaptive resonance system. Neural Networks 4(6), 759–771 (Nov 1991)
[8] Carstens, L., Toni, F.: Improving out-of-domain sentiment polarity classification using argumentation. In: IEEE International Conference on Data Mining Workshop, ICDMW. pp. 1294–1301 (2015)
[9] Carstens, L., Toni, F.: Using Argumentation to improve classification in Natural Language problems. Ph.D. thesis, Imperial College London (2016)
[10] Cayrol, C., Lagasquie-Schiex, M.C.: Symbolic and Quantitative Approaches to Reasoning with Uncertainty: 8th European Conference, ECSQARU 2005, chap. On the Acceptability of Arguments in Bipolar Argumentation Frameworks, pp. 378–389. Springer (2005)
[11] Chesñevar, C.I., Simari, G.R.: A lattice-based approach to computing warranted beliefs in skeptical argumentation frameworks. In: Proceedings of the 20th International Joint Conference on Artifical Intelligence. pp. 280–285. IJCAI'07 (2007)
[12] Clark, P., Niblett, T.: The CN2 induction algorithm. Machine Learning 3(4), 261–283 (1989)
[13] Cortes, C., Vapnik, V.: Support-vector networks. Machine Learning 20(3), 273–297 (Sep 1995)
[14] Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artificial Intelligence 77(2), 321 – 357 (1995)
[15] Evripidou, V., Toni, F.: Argumentation and voting for an intelligent user empowering business directory on the web. In: Web Reasoning and Rule Systems - 6th International Conference, RR. pp. 209–212 (2012)
[16] Fox, J., Glasspool, D., Grecu, D., Modgil, S., South, M., Patkar, V.: Argumentation-based inference and decision making–a medical perspective. IEEE Intelligent Systems 22(6), 34–41 (2007)
[17] Gao, Y., Toni, F.: Argumentation accelerated reinforcement learning for robocup keepaway-takeaway. In: Theory and Applications of Formal Argumentation - Second International Workshop, TAFA. vol. 8306, pp. 79–94 (2013)
[18] Gao, Y., Toni, F.: Argumentation accelerated reinforcement learning for cooperative multi-agent systems. In: ECAI 2014 - 21st European Conference on Artificial Intelligence. pp. 333–338 (2014)

[19] Gao, Y., Toni, F.: Argumentation accelerated reinforcement learning. Ph.D. thesis, Imperial College London (2015)
[20] García, A.J., Simari, G.R.: Defeasible logic programming: An argumentative approach. Theory and Practice of Logic Programming 4(1-2), 95–138 (2004)
[21] Gómez, S.A., Chesñevar, C.I.: A hybrid approach to pattern classification using neural networks and defeasible argumentation. In: Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference, Miami Beach, Florida, USA. pp. 393–398 (2004)
[22] Grosse, K., González, M.P., Chesñevar, C.I., Maguitman, A.G.: Integrating argumentation and sentiment analysis for mining opinions from twitter. AI Communications 28(3), 387–401 (2015)
[23] Groznik, V., Guid, M., Sadikov, A., Možina, M., Georgiev, D., Kragelj, V., Ribaric, S., Pirtosek, Z., Bratko, I.: Elicitation of neurological knowledge with ABML. In: Artificial Intelligence in Medicine - 13th Conference on Artificial Intelligence in Medicine, AIME. vol. 6747, pp. 14–23 (2011)
[24] Guid, M., Možina, M., Groznik, V., Georgiev, D., Sadikov, A., Pirtosek, Z., Bratko, I.: ABML knowledge refinement loop: A case study. In: Foundations of Intelligent Systems - 20th International Symposium, ISMIS. vol. 7661, pp. 41–50 (2012)
[25] John, G.H., Langley, P.: Estimating continuous distributions in bayesian classifiers. In: Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence. pp. 338–345. UAI'95 (1995)
[26] Lippi, M., Torroni, P.: Argumentation mining: State of the art and emerging trends. ACM Transactions on Internet Technology 16(2), 10:1–10:25 (Mar 2016)
[27] Mitchell, T.M.: Machine Learning. McGraw-Hill, Inc., 1 edn. (1997)
[28] Možina, M., Giuliano, C., Bratko, I.: Argument based machine learning from examples and text. In: First Asian Conference on Intelligent Information and Database Systems, ACIIDS. pp. 18–23 (2009)
[29] Možina, M., Guid, M., Krivec, J., Sadikov, A., Bratko, I.: Fighting knowledge acquisition bottleneck with argument based machine learning. In: Proceedings of the 2008 Conference on ECAI 2008: 18th European Conference on Artificial Intelligence. pp. 234–238 (2008)
[30] Možina, M., Guid, M., Krivec, J., Sadikov, A., Bratko, I.: Learning to explain with ABML. In: Explanation-aware Computing, Papers from the 2010 ECAI Workshop. pp. 37–48 (2010)
[31] Možina, M., Zabkar, J., Bench-Capon, T.J.M., Bratko, I.: Argument based machine learning applied to law. Artificial Intelligence and Law 13(1), 53–73 (2005)
[32] Možina, M., Zabkar, J., Bratko, I.: Argument based machine learning. Artificial Intelligence 171(10-15), 922–937 (2007)
[33] Murphy, P., Aha, D.: UCI Repository of machine learning databases. Tech. rep., University of California, Department of Information and Computer Science, Irvine, CA, US. (1994)
[34] Ontañón, S., Dellunde, P., Godo, L., Plaza, E.: A defeasible reasoning model of inductive concept learning from examples and communication. Artificial Intelligence 193, 129–148 (2012)
[35] Ontañón, S., Plaza, E.: Coordinated inductive learning using argumentation-based communication. Autonomous Agents and Multi-Agent Systems 29(2), 266–304 (2014)
[36] Rago, A., Toni, F., Aurisicchio, M., Baroni, P.: Discontinuity-free decision support with quantitative argumentation debates. In: Principles of Knowledge Representation and Reasoning: Proceedings of the Fifteenth International Conference. pp. 63–73 (2016)
[37] Rahwan, I., Simari, G.R.: Argumentation in Artificial Intelligence. Springer, 1st edn. (2009)
[38] Rotstein, N.D., Moguillansky, M.O., Simari, G.R.: Dialectical abstract argumentation: A characterization of the marking criterion. In: Proceedings of the 21st International Jont Conference on Artifical Intelligence. pp. 898–903. IJCAI'09 (2009)
[39] Rummery, G.A., Niranjan, M.: On-line Q-learning using connectionist systems. Tech. rep., Cambridge University Engineering Department (1994)
[40] Zabkar, J., Možina, M., Videcnik, J., Bratko, I.: Argument based machine learning in a medical domain. In: Computational Models of Argument: Proceedings of COMMA. Frontiers in Artificial Intelligence and Applications, vol. 144, pp. 59–70 (2006)
[41] Zapušek, M., Možina, M., Bratko, I., Rugelj, J., Guid, M.: Intelligent Tutoring Systems: 12th International Conference, chap. Designing an Interactive Teaching Tool with ABML Knowledge Refinement Loop, pp. 575–582 (2014)