Arguments Extracted from Text in Argument Based Machine Learning: A Case Study

Martin Mozina[1], Claudio Giuliano[2], and Ivan Bratko[1]

[1] University of Ljubljana, Slovenia,
martin.mozina@fri.uni-lj.si
ivan.bratko@fri.uni-lj.si
[2] FBK-irst, Povo TN 38100, Italy,
giuliano@fbk.eu

Abstract. We introduce a novel approach to cross-media learning based on argument based machine learning (ABML). ABML is a recent method that combines argumentation and machine learning from examples, and its main idea is to provide expert's arguments for some of the learning examples. In this paper, we present an alternative approach, where arguments used in ABML are automatically extracted from text with a technique for relation extraction. We demonstrate and evaluate the approach through a case study of learning to classify animals by using arguments extracted from Wikipedia.

1 Introduction

Argument Based Machine Learning (ABML) [12] is a recently developed approach that combines the ideas of argumentation and machine learning. Argumentation [13] is a branch of logic that mimics human reasoning and discussion between humans in several ways. In ABML, the idea is to provide expert's arguments, or reasons, for some of the learning examples. We require that the theory induced from the examples explains the examples in terms of the given reasons. This makes the learning easier because it constrains the search space of candidate hypotheses.

In this paper, we will demonstrate a possible way of extracting arguments from text and using them in ABML. We will begin with a short introduction to ABML, then investigate how arguments can be extracted from text, and demonstrate it on an animal classification problem. We will conclude the paper with a summary of main findings and pointers for further work.

2 Argument Based Machine Learning

Argument based machine learning [12] is machine learning extended with some concepts from argumentation. Argumentation is a branch of artificial intelligence that analyzes reasoning where arguments for and against a certain claim are produced and evaluated. A typical example of such reasoning is a law dispute at court, where plaintiff and defendant give arguments for their opposing claims, and at the end of the process the party with better arguments wins the case.

Arguments are used in ABML to enhance learning examples. Each argument is attached to a single learning example only, while one example can have several arguments. There are two types of arguments: positive arguments are used to explain (or argue) why a certain learning example is in the class as given, and negative ones give reasons against the class as given.

In ABML, arguments are usually provided by domain experts who find it much easier to articulate their knowledge in this manner. While it is generally accepted that giving domain knowledge usually poses a problem, in ABML they need to focus on one specific case only at a time and provide knowledge that seems relevant for this case and does not need to be valid for the whole domain. In this paper, we suggest an approach where arguments are automatically extracted from text. This approach thus eliminates the reliance on an expert. The expected advantage of this idea is similar to that with experts; it should be much easier to extract from text specific relations in the form of arguments that concern concrete examples than extracting general theories from text.

An ABML method is required to induce a theory that uses given arguments to explain the examples. If an ABML method is used on standard examples only (without arguments), then it should work the same as a normal machine learning method. We will use method ABCN2 [12], an argument based extension of the well known method CN2, that learns a set of unordered probabilistic rules from argumented examples. There, the theory (a set of rules) is said to explain the examples using given arguments, when there exists at least one rule for each argumented example that contains at least one positive argument in the condition part.

When an expert provides arguments for ABML, it is crucial to present the expert with only critical examples, as experts are unlikely to be willing to provide arguments for the whole learning set. This would require too much time and effort. Critical examples are examples that the learning method can not explain sufficiently well [11]. In the case of this paper, arguments are automatically extracted and they are provided for all learning examples, therefore we do not have to deal with this constraint regarding human experts. In this study, we will assume that arguments are provided for all learning examples.

3 Extracting Arguments from Text

The extraction of arguments from text is based on relation extraction from text. Relation extraction is an important task in natural language processing, with many practical applications such as question answering, ontology population, and information retrieval. It requires the analysis of textual documents, with the aim of recognizing particular types of relations between named entities, nominals, and pronouns. Reliably extracting relations in natural-language documents is still a laborious and unsolved problem. Traditionally, relation extraction systems have been trained to recognize relations between names of people, organizations, locations, and proteins. In the last two decades, several evaluation campaigns such as MUC [2], ACE [1], SemEval [3] have helped to understand and properly formalize the problem, and provide comparative benchmarks.

In this paper, we are interested in finding semantic relations between class values and descriptive attributes (taken from data), and using them as arguments. For example, given the class value *reptile* and the attribute *eggs* we are interested in relations such as "Most reptiles lay eggs" and "Reptiles hatch eggs." Specifically, the relationships that exist between classes and attributes are extracted from the whole English Wikipedia,[3] an online encyclopedia written collaboratively by volunteers, that has grown to become one of the largest online repositories of encyclopedic knowledge, with millions of articles available for a large number of languages.

To extract such relations from textual documents, we have to deal with two major problems. The first concerns the lack of information on the relation type we are seeking. In relation extraction, we usually know in advance the type of the relations to be extracted, here we only know class values and attributes, namely the arguments of a possible relation. Thus, the task is restricted to discover whether or not a relation exists between the two arguments.

The second problem is related with the lexicalization of the class values and attribute descriptions. The names of attributes and classes should be meaningful, or, in other words, should be similar to those used in texts. Using their background knowledge, humans can naturally interpret the concepts expressed by class values and attributes, however, due to the variability of natural language, it can be very difficult to find occurrences of the lexicalizations of such concepts in the same sentence and, consequently, to determine whether or not a relation exists. To address the first problem, we do not try to find specific assertions of relations in text, but rather we exploit the simple idea that if many different sentences reference both the class value and attribute, then the class value and attribute are likely to be related. On the other hand, to deal with the variability of natural language, we generated alternative lexical variants using a WordNet [7], a lexical database containing semantic relations among words. Specifically, we generated variants for all class values and attributes using the following semantic relations in WordNet: synonyms (e.g., breathe → respire) and morphological derivations (e.g., predator → predators).

As most relation extraction systems [10, 5, 9], we identify relations mentioned in text documents considering only those pairs that are mentioned in the same sentence. Let $c_1, \ldots, c_k$ be class values in data and $a_1, \ldots, a_n$ attributes. Then, the relation $\#r(c_i, a_j)$ is defined as the number of sentences across the whole English Wikipedia, where the class c_i and the attribute a_j co-occur.

We shall now define the construction of an argument given the number of relations between class and attribute values. An argument is a conjunction of a set of reasons, where each reason is related to a single attribute in the domain. To determine whether and attribute a_j is a possible reason for class c_i, we first evaluate whether $\#r(c_i, a_j)$ is statistically different from the expected value

[3] http://en.wikipedia.org

$E(\#r(c_i, a_j))$, namely is the number could be obtained purely by chance. A possible method for this task is the standard χ^2 test for 2×2 matrices.

When $\#r(c_i, a_j)$ is statistically different from $E(\#r(c_i, a_j))$, it can be either higher or lower. If $\#r(c_i, a_j) > E(\#r(c_i, a_j))$, then we say that a_j is a positive reason for c_i. Such a positive argument can be given to an example if it is from class c_i and the value of a_j is "positive". The positiveness of attribute values must be defined prior to learning and it intends to distinguish between values that should occur more frequently in the class-attribute relations in text than it is expected. Although, it is impossible to say which of the values will have this property, we believe that a good heuristics to select positive attribute values is to select those that ascribe the presence of a property described by a_j to the example. For instance, if an animal has the value of attribute *breathes* 1, this value states that then the animal is breathing (presence of this property), and the value of attribute is positive. If the number of found relations is less than expected, i.e. $\#r(c_i, a_j) < E(\#r(c_i, a_j))$, then we can use such reason only if the example has negative value of a_j.

An argument for a certain example is thus constructed from all positive and negative reasons consistent with the values of this example. Sometimes, such an argument will be overly specific. To alleviate this problem all arguments are pruned with REP (reduced error pruning) [8] principle before they are appended to the example.

4 Case study: Animal Classification

The approach will be illustrated and evaluated on a learning domain named ZOO taken from the UCI repository [4]. It contains descriptions of 101 animals (instances) with 17 attributes: *hair, feathers, eggs, milk, predator, toothed, domestic, backbone, fins, legs, tail, catsize, airborne, aquatic, breathes, venomous,* and *type*, which is the class attribute. Type has seven possible values: *mammal, bird, reptile, fish, amphibian, insect,* and *other*.

We began the experiment by learning rules with ABCN2 without any arguments extracted from text. Induced rules were:

- IF milk=yes THEN type=mammal
- IF feathers=yes THEN type=bird
- IF eggs=yes AND fins=yes THEN type=fish
- IF aquatic=no AND legs=6 THEN type=insect
- IF feathers=no AND eggs=yes AND backbone=yes AND aquatic=no THEN type=r
- IF milk=no AND domestic=no AND hair=no AND tail=yes AND fins=no AND legs=0 THEN type=reptile
- IF toothed=yes AND legs=4 AND eggs=yes AND aquatic=yes THEN type=amphib
- IF feathers=no AND hair=no AND airborne=no AND backbone=no AND predator=yes THEN type=other
- IF fins=no AND backbone=no AND legs=no THEN type=other

In the following step, we sought through Wikipedia for relations between class values (e.g., mammal) and positive attribute values (e.g., milk=yes). Table 1 shows the alternative lexical variants for some attributes generated using

Table 1. The alternative lexical variants for attributes generated using WordNet and morphological derivations.

attribute	lexical variants	attribute	lexical variants
hair	hairs, fur, furs	backbone	spine
feather	feathers, plumage	breathe	breathes, respire, respires
egg	eggs, spawn, spawns	venomous	poisonous
milk	milking	fin	fins
airborne	winged	leg	legs, limb, limbs
aquatic	aquatics, marine	tail	tails
predator	predators	domestic	domesticated, pet
toothed	tooth, teeth, fang, fangs, fanged	catsize	-

WordNet and morphological derivations. In this search we omitted to use class other, since it does not represent any actual animal class. Table 2 shows number of all relations found for the ZOO domain. For example, we found a strong correlation between the class bird and the attribute feather, merely expanding the attribute with the lexical variants *feathers* and *plumage*. On the other hand, despite the fact that it is intuitive for humans to answer the question if a reptile has approximately the same size of a cat, it is almost impossible to find occurrences of the class reptile and the attribute "catsize" in the same text, due to the erroneous lexicalization of this attribute introduced for comparing animals by size. The last row of Table 2 show that the attribute catsize gets scores of zero for all classes.

The absolute values $\#r(c_i, a_j)$ are not strongly related to the correlation between class c_i and attribute a_j. For instance, it seems that *aquatic* is the most important feature of *amphibians*. But, is it really, as being aquatic is common for all classes? On the other hand, the text extraction tool found only 6 relations between *amphibians* and *breathing*. However, there is still a strong positive relation between them, due to a much lesser presence of the concept breathing and animal type amphibian in text when compared to other attributes and animals.

For this reason, we applied the standard χ^2 ($sig = 0.05$) test to determine whether $\#r(c_i, a_j)$ is statistically different from $E(\#r(c_i, a_j))$ (any appropriate statistical test could be applied here). Table 3 shows results of χ^2 test, where:

value 0 means that the relation is not significant;
value 1 denotes positive reasons, and
value -1 denotes negative reasons.

Table 3 shows which of the attributes can be used as arguments for every class. For example, in the case of amphibians, we can use attributes *aquatic*, *breathes*, and *legs* as reasons in the argument, if the values of the corresponding case are positive ("yes" or > 0 for legs). Similarly, attributes *hair*, *feathers*, *predator* can be used as reasons when attribute values are negative ("no" or $= 0$ for legs).

After augmenting all examples with arguments, the rules for *mammals*, *birds*, *fishes*, *insects*, and *other* stayed the same. The rule for amphibians changed to:

Table 2. Number of positive relations $\#r(c_i, a_j)$ between animal classes and attributes found in Wikipedia.

	amphibian	reptile	insect	mammal	bird	fish
hair	1	16	70	187	106	87
feathers	0	14	17	15	894	31
eggs	34	117	339	174	894	645
milk	2	2	10	67	25	120
airborne	2	15	117	15	196	21
aquatic	81	184	271	1008	284	1072
predator	5	16	123	95	285	217
toothed	10	59	54	150	102	250
backbone	0	1	3	11	7	51
breathes	6	3	8	11	16	45
venomous	6	43	39	44	41	47
fins	4	4	3	17	21	529
legs	27	42	206	51	364	111
tail	16	26	53	57	504	279
domestic	9	40	31	81	317	166
catsize	0	0	0	0	0	0

– IF legs=4 AND breathes=yes AND aquatic=yes AND hair=no THEN type=a

The original and the new rule are actually very alike. In the latter attributes *toothed* and *eggs* were replaced with *breathes* and *hair*. From a point of an expert, the second rule is better, since it is not entirely true that *amphibians* do not have *teeth*. Most amphibian larvae have tiny teeth. Nevertheless, although most adult amphibians retain their teeth, teeth can be reduced in size or not present at all.

The rules for reptiles also changed with the use of arguments:

– IF toothed=yes AND milk=no AND fins=no AND legs=0 THEN type=reptile
– IF feathers=no AND milk=no AND fins=no AND breathes=yes AND backbone=yes AND aquatic=no THEN type=reptile

The rules are again similar to the ones above with some differences. Specifically, the second rule in the original set mentions *domestic=no* as a condition for a reptile, although there are many reptiles used as pets (e.g., turtles, snakes, etc.)

We evaluated the method with 10-times repeated 10-fold cross-validation to avoid effects of randomness on one split only. In each iteration, all examples in the learn set were argumented, then a model was built from these and evaluated on the test set. Using ABCN2 without arguments resulted in 94.51% classification accuracy, while ABCN2 with arguments scored, on average, 96.75% classification accuracy. As a comparison, some standard machine learning methods (as implemented in Orange [6]) scored 90% (SVM), 92.57% (C4.5) and 92.6% (naïve Bayes).

Table 3. Positive (1) and Negative (-1) reasons. Relations with value 0 are not significantly different from the expected value and cannot be used as reasons in arguments.

	amphibian	reptile	insect	mammal	bird	fish
hair	-1	0	1	1	-1	-1
feathers	-1	-1	-1	-1	1	-1
eggs	0	0	1	-1	1	-1
milk	0	-1	-1	1	-1	1
airborne	0	0	1	-1	1	-1
aquatic	1	1	-1	1	-1	1
predator	-1	-1	1	-1	1	0
toothed	0	1	-1	1	-1	1
backbone	0	0	0	0	-1	1
breathes	1	0	0	0	-1	1
venomous	0	1	1	0	-1	-1
fins	0	-1	-1	-1	-1	1
legs	1	0	1	-1	1	-1
tail	0	-1	-1	-1	1	0
domestic	0	0	-1	-1	1	-1
catsize	0	0	0	0	0	0

5 Conclusions and Future Work

We developed and implemented a method for combining raw data and text through the use of argument based machine learning and automatic extraction of arguments from text. The method was demonstrated and evaluated on the animal classification domain. Despite the fact that we used the simplest method for finding relations in a text, namely counting relations, we still obtained very promising results. However, assuming that if a pair of class value/attribute occurs very frequently, then this is evidence that there exists a relationship is not correct since high frequency can be accidental. In the current paper, we used a simple statistical test for validating the "true" relationship. For future work, we want to use an information-theoretically motivated measure (e.g., the pointwise mutual information) for discovering interesting relations, which should result in even better arguments.

The described combination of data and text could be used also in other domains. A possible example of such a domain is medicine, where we try to provide a diagnosis for patients based on clinical values, and the arguments could be extracted from several scientific papers and other published material on the particular issue. However, the described approach in this paper is not the only possible way of extracting arguments from text. Sometimes learning examples have already attached commentaries given by domain experts that are written in natural language. In medicine, for instance, doctors usually provide their explanation of laboratory results. Another example of such a domain are technical experiments (e.g., efficiency of jet engines), where experts explain obtained results. We believe that in all domains of this type, sifting through several documents to find relations between class and attributes is not the best option, but a careful analysis of the arguments already provided would provide better results.

Finally, X-Media[4] is a European project that addresses the issue of cross-media knowledge management in complex distributed environments. A part of X-Media is concerned with development of principles where different types of data (raw data, text, images) are used together to enable learning more accurate models. We believe that ABML is a promising way of combining raw data with textual data, whenever we can extract arguments from text.

Acknowledgments

The authors are partially supported by the X-Media project (http://www.x-media-project.org), sponsored by the European Commission as part of the Information Society Technologies (IST) program under EC grant number IST-FP6-026978.

References

1. ACE – Automatic Content Extraction, http://www.nist.gov/speech/tests/ace.
2. MUC - Message Understanding Conferences, http://cs.nyu.edu/faculty/grishman/m
3. Semeval-2007, http://nlp.cs.swarthmore.edu/semeval/.
4. A. Asuncion and D.J. Newman. UCI machine learning repository, 2007.
5. Razvan Bunescu and Raymond J. Mooney. Subsequence kernels for relation extraction. In *Proceedings of the 19th Conference on Neural Information Processing Systems*, Vancouver, British Columbia, 2005.
6. J. Demšar and B. Zupan. Orange: From experimental machine learning to interactive data mining. White Paper [http://www.ailab.si/orange], Faculty of Computer and Information Science, University of Ljubljana, 2004.
7. Christiane Fellbaum. *WordNet. An Electronic Lexical Database*. MIT Press, 1998.
8. Johannes Furnkranz. Pruning algorithms for rule learning. *Machine Learning*, 27(2):139–171, 1997.
9. Claudio Giuliano, Alberto Lavelli, and Lorenza Romano. Relation extraction and the influence of automatic named entity recognition. *ACM Transactions on Speech and Language Processing*, 5(1), 2007.
10. Takaaki Hasegawa, Satoshi Sekine, and Ralph Grishman. Discoverying relations among named entities from large corpora. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004)*, Barcelona, Spain, 2004. Association for Computational Linguistics.
11. Martin Možina, Matej Guid, Jana Krivec, Aleksander Sadikov, and Ivan Bratko. Fighting knowledge acquisition bottleneck with argument based machine learning. In *Proceedings of 18th European Conference on Artificial Intelligence (ECAI 2008)*, Patras, Greece, 2008. IOS Press.
12. Martin Možina, Jure Žabkar, and Ivan Bratko. Argument based machine learning. *Artificial Intelligence*, 171(10/15):922–937, 2007.
13. Henry Prakken and Gerard Vreeswijk. *Handbook of Philosophical Logic, second edition*, volume 4, chapter Logics for Defeasible Argumentation, pages 218–319. Kluwer Academic Publishers, Dordrecht etc, 2002.

[4] http://www.x-media-project.org/