

Implementing Weighted Abduction in Markov Logic

James Blythe USC ISI blythe@isi.edu	Jerry R. Hobbs USC ISI hobbs@isi.edu	Pedro Domingos University of Washington pedrod@cs.washington.edu
Rohit J. Kate University of Wisconsin-Milwaukee katerj@uwm.edu	Raymond J. Mooney University of Texas at Austin mooney@cs.utexas.edu	

Abstract

Abduction is a method for finding the best explanation for observations. Arguably the most advanced approach to abduction, especially for natural language processing, is weighted abduction, which uses logical formulas with costs to guide inference. But it has no clear probabilistic semantics. In this paper we propose an approach that implements weighted abduction in Markov logic, which uses weighted first-order formulas to represent probabilistic knowledge, pointing toward a sound probabilistic semantics for weighted abduction. Application to a series of challenge problems shows the power and coverage of our approach.

1 Introduction

Abduction is inference to the best explanation.¹ Typically, one uses it to find the best hypothesis explaining a set of observations, e.g., in diagnosis and plan recognition. In natural language processing the content of an utterance can be viewed as a set of observations, and the best explanation then constitutes the interpretation of the utterance. Hobbs et al. [7] described a variety of abduction called “weighted abduction” for interpreting natural language discourse. The key idea was that the best interpretation of a text is the best explanation or proof of the logical form of the text, allowing for assumptions. What counted as “best” was defined in terms of a cost function which favored proofs with the fewest number of assumptions and the most salient and plausible axioms, and in which the pervasive redundancy implicit in natural language discourse was exploited. It was argued in that paper that such interpretation problems as coreference and syntactic ambiguity resolution, determining the specific meanings of vague predicates and lexical ambiguity resolution, metonymy resolution, metaphor interpretation, and the recognition of discourse structure could be seen to “fall out” of the best abductive proof.

Specifically, weighted abduction has the following features:

1. In a goal expression consisting of an existentially quantified conjunction of positive literals, each literal is given a cost that represents the utility of proving that literal as opposed to assuming it. That is, a low cost on a literal will make it more likely for it to be assumed, whereas a high cost will result in a greater effort to find a proof.

¹We are indebted to Jesse Davis, Parag Singla and Marc Sumner for discussions about this work. This research was supported in part by the Defense Advanced Research Projects Agency (DARPA) Machine Reading Program under Air Force Research Laboratory (AFRL) prime contract no. FA8750-09-C-0172, in part by the Office of Naval Research under contract no. N00014-09-1-1029, and in part by the Army Research Office under grant W911NF-08-1-0242. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the view of the DARPA, AFRL, ONR, ARO, or the US government.

2. Costs are passed back across the implication in Horn clauses according to weights on the conjuncts in the antecedents. Specifically, if a consequent costs $\$c$ and the weight on a conjunct in the antecedent is v , then the cost on that conjunct will be $\$vc$. Note that if the weights add up to less than one, backchaining on the rule will be favored, as the cost of the antecedent will be less than the cost of the consequent. If the weights add up to more than one, backchaining will be disfavored unless a proof can be found for one or more of the conjuncts in the antecedent, thereby providing partial evidence for the consequent.
3. Two literals can be factored or unified, where the result is given the minimum cost of the two, providing no contradiction would result. This is a frequent mechanism for coreference resolution. In practice, only a shallow or heuristic check for contradiction is done.
4. The lowest-cost proof is the best interpretation, or the best abductive proof of the goal expression.

However, there are two significant problems with weighted abduction as it was originally presented. First, it required a large knowledge base of commonsense knowledge. This was not available when weighted abduction was first described, but since that time there have been substantial efforts to build up knowledge bases for various purposes, and at least two of these have been used with promising results in an abductive setting—Extended WordNet [6] for question-answering and FrameNet [11] for textual inference.

The second problem with weighted abduction was that the weights and costs did not have a probabilistic semantics. This, for example, hampers automatic learning of weights from data or existing resources. That is the issue we address in the present paper.

In the last decade and a half, a number of formalisms for adding uncertain reasoning to predicate logic have been developed that are well-founded in probability theory. Among the most widely investigated is Markov logic [14, 4]. In this paper we show how weighted abduction can be implemented in Markov logic. This demonstrates that Markov logic networks can be used as a powerful mechanism for interpreting natural language discourse, and at the same time provides weighted abduction with something like a probabilistic semantics.

In Section 2 we briefly describe Markov logic and Markov logic networks. Section 3 then describes how weighted abduction can be implemented in Markov logic. In Section 4 we describe experiments in which fourteen published examples of the use of weighted abduction in natural language understanding are implemented in Markov logic networks, with good results. Section 5 on current and future directions briefly describes an ongoing experiment in which we are attempting to scale up to apply this procedure to the textual inference problem with a knowledge base derived from FrameNet with tens of thousands of axioms.

2 Markov Logic Networks and Related Work

Markov logic [14, 4] is a recently developed theoretically sound framework for combining first-order logic and probabilistic graphical models. A traditional first-order knowledge base can be seen as a set of hard constraints on the set of possible worlds: if a world violates even one formula, its probability is zero. In order to soften these constraints, Markov logic attaches a weight to each first-order logic formula in the knowledge base. Such a set of weighted first-order logic formulae is called a *Markov logic network* (MLN). A formula's weight reflects how strong a constraint it imposes on the set of possible worlds: the higher the weight, the lower the probability of a world that violates it; however, that probability need not be zero. An MLN with all infinite weights reduces to a traditional first-order knowledge base with only hard constraints.

Formally, an MLN L is a set of formula–weight pairs (F_i, w_i) . Given a set of constants, it defines a joint probability distribution over a set of boolean variables $X = (X_1, X_2, \dots)$ corresponding to the possible groundings (using the given constants) of the literals present in the first-order formulae:

$$P(X = x) = \frac{1}{Z} \exp(\sum_i w_i n_i(x))$$

where $n_i(x)$ is the number of true groundings of F_i in x and Z is a normalization term obtained by summing $P(X = x)$ over all values of X .

Semantically, an MLN can be viewed as a set of templates for constructing Markov networks [12], the undirected counterparts of Bayesian networks. An MLN and a set of constants produce a Markov network in which each ground literal is a node and every pair of ground literals that appear together in some grounding of some formula are connected by an edge. Different sets of constants produce different Markov networks; however, there are certain regularities in their structure and parameters. For example, all groundings of the same formula have the same weight.

Probabilistic inference for an MLN (such as finding the most probable truth assignment for a given set of ground literals, or finding the probability that a particular formula holds) can be performed by first producing the ground Markov network and then using well known inference techniques for Markov networks, like Gibbs sampling. Given a knowledge base as a set of first-order logic formulae, and a database of training examples each consisting of a set of true ground literals, it is also possible to learn appropriate weights for the MLN formulae which maximize the probability of the training data. An open-source software package for MLNs, called Alchemy², is also available with many built-in algorithms for performing inference and learning.

Much of the early work on abduction was done in a purely logical framework (e.g., [13, 3, 9, 10]). Typically the choice between alternative explanations is made on the basis of parsimony; the shortest proofs with the fewest assumptions are favored. However, a significant limitation of these purely logical approaches is that they are unable to reason under uncertainty or estimate the likelihood of alternative explanations. A probabilistic form of abduction is needed in order to account for uncertainty in the background knowledge and to handle noisy and incomplete observations.

In Bayesian networks [12] background knowledge with its uncertainties is encoded in a directed graph. Then, given a set of observations, probabilistic inference over the graph structure is done to compute the posterior probability of alternative explanations. However, Bayesian networks are based on propositional logic and cannot handle structured representations, hence preventing their use in situations, characteristic of natural language processing, that involve an unbounded number of entities with a variety of relations between them.

In recent years there have been a number of proposals attempting to combine the probabilistic nature of Bayesian networks with structured first-order representations. It is impossible here to review this literature here. A good review of much of it can be found in [5], and in [14] there are detailed comparisons of various models to MLNs.

Charniak and Shimony [2] define a variant of weighted abduction, called “cost-based abduction” in which weights are attached to terms rather than to rules or to antecedents in rules. Thus, the term P_i has the same cost whatever rule it is used in. The cost of an assignment to the variables in the domain is the sum of the costs of the variables that are true in the assignment. Charniak and Shimony provide a probabilistic semantics for their approach by showing how to construct a Bayesian network from a domain such that a most probable explanation solution to the Bayes net corresponds to a lowest-cost solution to the abduction problem. However, in natural language applications the utility of proving a proposition can vary by context; weighted abduction accomodates this, whereas cost-based abduction

²does not
<http://alchemy.cs.washington.edu>

3 Weighted Abduction and MLNs

Kate and Mooney [8] show how logical abduction can be implemented in Markov logic networks. They use forward inference in MLNs to perform abduction by adding clauses with reverse implications. Universally quantified variables from the left hand side of rules are converted to existentially quantified variables in the reversed clause. For example, suppose we have the following rule saying that mosquito bites transmit malaria:

$$\text{mosquito}(x) \wedge \text{infected}(x, \text{Malaria}) \wedge \text{bite}(x, y) \supset \text{infected}(y, \text{Malaria})$$

This would be translated into the soft rule

$$[w] \text{infected}(y, \text{Malaria}) \supset \exists x[\text{mosquito}(x) \wedge \text{infected}(x, \text{Malaria}) \wedge \text{bite}(x, y)]$$

Where there is more than one possible explanation, they include a closure axiom saying that one of the explanations must hold. Since blood transfusions also cause malaria, they have the hard rule

$$\begin{aligned} \text{infected}(y, \text{Malaria}) \supset \\ \exists x[\text{mosquito}(x) \wedge \text{infected}(x, \text{Malaria}) \wedge \text{bite}(x, y)] \\ \vee \exists x[\text{infected}(x, \text{Malaria}) \wedge \text{transfuse}(\text{Blood}, x, y)]. \end{aligned}$$

Kate and Mooney also add a soft mutual exclusivity clause that states that no more than one of the possible explanations is true.

In translating between weighted abduction and Markov logic, we need similarly to specify the axioms in Markov logic that correspond to a Horn clause axiom in weighted abduction. In addition, we need to describe the relation between the numbers in weighted abduction and the weights on the Markov logic axioms. Hobbs et al. [7] give only broad, informal guidelines about how the numbers correspond to probabilities. In this development, we elaborate on how the numbers can be defined more precisely within these guidelines in a way that links with the weights in Markov logic, thereby pointing to a probabilistic semantics for the weighted abduction numbers.

There are two sorts of numbers in weighted abduction—the weights on conjuncts in the antecedents of Horn clause axioms, and the costs on conjuncts in goal expressions, which are existentially quantified conjunctions of positive literals. We deal first with the weights, then with the costs.

The space of events over which probabilities are taken is the set of proof graphs constituting the best interpretations of a set of texts in a corpus. Thus, by the probability of $p(x)$ given $q(x)$, we mean the probability that $p(x)$ will occur in a proof graph in which $q(x)$ occurs.

The translation from weighted abduction axioms to Markov logic axioms can be broken into two steps. First we consider the “or” node case, determining the relative costs of axioms that have the same consequent. Then we look at the “and” node case, determining how the weights should be distributed across the conjuncts in the antecedent of a Horn clause, given the total weight for the antecedent.

Weights on Antecedents in Axioms. First consider a set of Horn clause axioms all with the same consequent, where we collapse the antecedent into a single literal, and for simplicity allow x to stand for all the universally quantified variables in the antecedent, and assume the consequent to have only those variables. That is, we convert all axioms of the form

$$p_1(x) \wedge \dots \supset q(x)$$

into axioms of the form

$$A_i(x) \supset q(x), \text{ where } p_1(x) \wedge \dots \equiv A_i(x)$$

To convert this into Markov logic, we first introduce the hard constraint

$$A_i(x) \supset q(x).$$

In addition, given a goal of proving $q(x)$, in weighted abduction we will want to backchain on at least (and usually at most) one of these axioms or we will want simply to assume $q(x)$. Thus, we can introduce another hard constraint with the disjunction of these antecedents as well as a literal $\text{Assume}Q(x)$ that means $q(x)$ is assumed rather than proved.

$$q(x) \supset A_1(x) \vee A_2(x) \vee \dots \vee A_n(x) \vee \text{Assume}Q(x).$$

Then we need to introduce soft constraints to indicate that each of these disjuncts is a possible explanation, or proof, of $q(x)$, with an associated probability, or weight.

$$\begin{aligned} [w_i] q(x) &\supset A_i(x), \dots \\ [w_0] q(x) &\supset \text{Assume}Q(x) \end{aligned}$$

The probability that $\text{Assume}Q(x)$ is true is the conditional probability P_0 that none of the antecedents is true given that $q(x)$ is true.

$$P_0 = P(\neg[A_1(x) \vee A_2(x) \vee \dots \vee A_n(x)] \mid q(x))$$

In weighted abduction, when the antecedent weight is greater than one, we prefer assuming the consequent to assuming the antecedent. When the antecedent weight is less than one we prefer to assume the antecedent. If the probability that an antecedent $A_i(x)$ is the explanation of $q(x)$ is greater than P_0 , it should be given a weighted abduction weight v_i less than 1, making it more likely to be chosen.³ Correspondingly, if it is less than P_0 , it should be given a weight v_i greater than 1, making it less likely to be chosen. In general, the weighted abduction weights should be in reverse order of the conditional probabilities P_i that $A_i(x)$ is the explanation of $q(x)$.

$$P_i = P(A_i(x) \mid q(x))$$

If we assign the weights v_i in weighted abduction to be

$$v_i = \frac{\log P_i}{\log P_0}$$

then this is consistent with informal guidelines in [7] on the meaning of these weights. We use the logs of the probabilities rather than the probabilities themselves to moderate the effect of one axiom being very much more probable than any of the others.

Kate and Mooney [8], in their translation of logical abduction into Markov logic, also include soft constraints stipulating that the different possible explanations $A_i(x)$ are normally mutually exclusive. We do not do that here, but we get a kind of soft mutual exclusivity constraint by virtue of the axioms below that levy a cost for any literal that is taken to be true. In general, more parsimonious explanations will be favored.

Nevertheless, in most cases a single explanation will suffice. When this is true, the probability of $A_i(x)$ holding when $q(x)$ holds is $\frac{e^{w_i}}{Z}$. Then a reasonable approximation for the relation between the weighted abduction weights v_i and the Markov logic weights w_i is

$$w_i = -v_i \log P_0$$

Weights on Conjuncts in Antecedents. Next consider how cost is spread across the conjuncts in the antecedent of a Horn clause in weighted abduction. Here we use u 's to represent the weighted abduction weights on the conjuncts.

$$p_1(x)^{u_1} \wedge p_2(x)^{u_2} \wedge \dots \equiv A(x)$$

The u 's should somehow represent the semantic contribution of each conjunct to the conclusion. That is, given that the conjunct is true, what is the probability that it is part of an explanation of the consequent? Conjuncts with a higher such probability should be given higher weights u ; they play a more significant role in explaining $A(x)$.

Let P_i be the conditional probability of the consequent given the i th conjunct in the antecedent.

$$P_i = P(A(x) \mid p_i(x))$$

and let Z be a normalization factor.

$$Z = \sum_{i=1}^n P_i$$

³We use v_i for these weighted abduction weights and w_i for Markov logic weights.

Let v be the weight of the entire antecedent as determined above.

Then it is consistent with the guidelines in [7] to define the weights on the conjuncts as follows:

$$u_i = \frac{vP_i}{Z}$$

The weights u_i will sum to v and each will correspond to the semantic contribution of its conjunct to the consequent.

In Markov logic, weights apply only to axioms as a whole, not parts of axioms. Thus, the single axiom above must be decomposed into one axiom for each conjunct and the dependencies must be written as

$$[w_i] p_i(x) \supset A(x), \dots$$

The relation between the weighted abduction weights u_i and the Markov logic weights w_i can be approximated by

$$u_i = \frac{ve^{-w_i}}{Z}$$

Costs on Goals. The other numbers in weighted abduction are the costs associated with the conjuncts in the goal expression. In weighted abduction these costs function as utilities. Some parts of the goal expression are more important to interpret correctly than others; we should try harder to prove these parts, rather than simply assuming them. In language it is important to recognize the referential anchor of an utterance in shared knowledge. Thus, those parts of a sentence most likely to provide this anchor have the highest utility. If we simply assume them, we lose their connection with what is already known. Those parts of a sentence most likely to be new information will have a lower cost, because we usually would not be able to prove them in any case.

Consider the two sentences

The smart man is tall.

The tall man is smart.

The logical form for each of them will be

$$(\exists x)[smart(x) \wedge tall(x) \wedge man(x)]$$

In weighted abduction, an interpretation of the sentence is a proof of the logical form, allowing assumptions. In the first sentence we want to prove $smart(x)$ to anchor the sentence referentially. Then $tall(x)$ is new information; it will have to be assumed. We will want to have a high cost on $smart(x)$ to force the proof procedure to find this referential anchor. The cost on $tall(x)$ will be low, to allow it to be assumed without expending too much effort in trying to locate that fact in shared knowledge.

In the second sentence, the case is the reverse.

Let's focus on the first sentence and assume we know that educated people are smart and big people are tall, and furthermore that John is educated and Bill is big.

$$\begin{aligned} educated(x)^{1,2} &\supset smart(x) \\ big(x)^{1,2} &\supset tall(x) \\ educated(J), big(B) & \end{aligned}$$

In weighted abduction, the best interpretation will be that the smart man is John, because he is educated, and we pay the cost for assuming he is tall. The interpretation we want to avoid is one that says x is Bill; he is tall because he is big, and we pay the cost of assuming he is smart. Weighted abduction with its differential costs on conjuncts in the goal expression favors the first and disfavors the second.

In weighted abduction, only assumptions cost; literals that are proved cost nothing. When the above axioms are translated into Markov logic, it would be natural to capture the differential costs by attaching a negative weight to $smart(x)$ to model the cost associated with assuming it. However, this weight would apply to any assignment in which $smart(J)$ is true, regardless of whether it was assumed, derived from

an assumed fact, or derived from a known fact. A potential solution might be to attach the negative weight to $AssumeSmart(x)$. But the first axiom above allows us to bypass the negative weight on $smart(x)$. We can hypothesize that x is Bill, pay a low cost on $AssumeEducated(B)$, derive $smart(B)$, and get the wrong assignment. Thus it is not enough to attach a negative weight to high-cost conjuncts in the goal expression. This negative weight would have to be passed back through the whole knowledge base, making the complexity of setting the weights at problem time in the MLN knowledge base equal to the complexity of running the inference problem.

An alternative solution, which avoids this problem when the forward inferences are exact, is to use a set of predicates that express knowing a fact without any assumptions. In the current example, we would add $Ksmart(x)$ for knowing that an entity is smart. The facts asserted in the data base are now $KEducated(J)$ and $Kbig(B)$. For each hard axiom involving non- K predicates, we have a corresponding axiom that expresses the relation between the K -predicates, and we have a soft axiom allowing us to cross the border between the K predicates and their non- K counterparts.

$$\begin{aligned} KEducated(x) \supset Ksmart(x), \dots \\ [w] Ksmart(x) \supset smart(x), \dots \end{aligned}$$

Here the positive weight w attached is chosen to counteract the negative weight we would attach to $smart(x)$ to reflect the high cost of assuming it.

This removes the weight associated with assuming $smart(x)$ regardless of the inference path that leads to knowing $smart(x)$ ($KSmart(x)$). Further, this translation takes linear time in the size of the goal expression to compute, since we do not need to know the equivalent weighted abduction cost assigned to the possible antecedents of $smart(x)$.

If the initial facts do not include $KEducated(B)$ and instead $educated(B)$ must be assumed, then the negative weight associated with $smart(B)$ is still present. In this solution, there is no danger that the inference process can by-pass the cost of assuming $smart(B)$, since it is attached to the required predicate and can only be removed by inferring $KSmart(B)$.

Finally, there is a tendency in Markov logic networks for assignments of high probability for propositions for which there is no evidence one way or the other. To suppress this, we associate a small negative weight with every predicate. In practice, it has turned out that a weight of -1 effectively suppresses this behavior.

4 Experimental Results

We have tested our approach on a set of fourteen challenge problems from [7] and subsequent papers, designed to exercise the principal features of weighted abduction and show its utility for solving natural language interpretation problems. The knowledge bases used for each of these problems are sparse, consisting of only the axioms required for solving the problems plus a few distractors.

An example of a relatively simple problem is #5 in the table below, resolving “he” in the text

I saw my doctor last week. He told me to get more exercise.

where we are given a knowledge base that says a doctor is a person and a male person is a “he”. Solving the problem requires assuming the doctor is male.

$$\begin{aligned} (\forall x)[doctor(x)^{1.2} \supset person(x)] \\ (\forall x)[male(x)^{.6} \wedge person(x)^{.6} \supset he(x)] \end{aligned}$$

The logical form fragment to prove is $(\exists x)he(x)$, where we know $doctor(D)$.

A problem of intermediate difficulty (#7) is resolving the three lexical ambiguities in the sentence

The plane taxied to the terminal.

where we are given a knowledge base saying that airplanes and wood smoothers are planes, planes moving on the ground and people taking taxis are both described as “taxiing”, and computer terminals and airport terminals are both terminals.

An example of a difficult problem is #12, finding the coherence relation, thereby resolving the pronoun “they”, between the sentences

The police prohibited the women from demonstrating. They feared violence.

The axioms specify relations between fearing, not wanting, and prohibiting, as well as the defeasible transitivity of causality and the fact that a causal relation between sentences makes the discourse coherent.

The weights in the axioms were mostly distributed evenly across the conjuncts in the antecedents and summed to 1.2.

For each of these problems, we compare the performance of the method described here with a manually constructed gold standard and also with a method based on Kate and Mooney’s (KM) approach. In this method, weights were assigned to the reversed clauses based on the negative log of the sum of weights in the original clause. This approach does not capture different weights for different antecedents of the same rule, and so has less fidelity to weighted abduction than our approach. In each case, we used *Alchemy*’s probabilistic inference to determine the most probable explanation (MPE) [12].

In some of the problems the system should make more than one assumption, so there are 22 assumptions in total over all 14 problems in the gold standard. Using our method, 18 of the assumptions were found, while 15 were found using the KM method. Table 1 shows the number of correct assumptions found and the running time for the two approaches for each problem. Our method in particular provides good coverage, with a recall of over 80% of the assumptions made in the gold standard. It has a shorter running time overall, approximately 5.3 seconds versus 8.7 seconds for the reversal method. This is largely due to one problem in the test set, problem #9, where the running time for the KM method is relatively high because the technique finds a less sparse network, leading to larger cliques. There were two problems in the test set that neither approach could solve. One of these contains predicates that have a large number of arguments, leading to large clique sizes.

5 Current and Future Directions

In other work [11] we are experimenting with using weighted abduction with a knowledge base with tens of thousands of axioms derived from *FrameNet* for solving problems in recognizing textual entailment (RTE2) from the *Pascal* dataset [1]. For a direct comparison between standard weighted abduction and the Markov logic approach described here, we are also experimenting with using the latter on the same task with the same knowledge base.

For each text-hypothesis pair, the sentences are parsed and a logical form is produced. The output for the first sentence forms the specific knowledge the system has while the output for the second sentence is used as the target to be explained. If the cost of the best explanation is below a threshold we take the target sentence to be true given the initial information.

It is a major challenge to scale our approach to handle all the problems from the RTE2 development and test sets. We are not yet able to address the most complex of these using inference in Markov logic networks. However, we have devised a number of pre-processing steps to reduce the complexity of the resultant network, which significantly increase the number of problems that are tractable.

The *FrameNet* knowledge base contains a large number of axioms with general coverage. For any individual entailment problem, most of them are irrelevant and can be removed after a simple graphical analysis. We are able to remove more irrelevant axioms and predicates with an iterative approach that in

Problem	Our Method		KM Method		Gold standard
	score	seconds	score	seconds	
1	3	300	3	16	3
2	1	250	1	265	1
3	1	234	1	266	1
4	2	234	2	203	2
5	1	218	1	218	1
6	1	218	0	265	1
7	3	300	3	218	3
8	1	200	1	250	1
9	2	421	0	5000	2
10	1	2500	1	1500	3
11	0		0		1
12	0		0		1
13	1	250	1	250	1
14	1	219	1	219	1
Total	18	5344	15	8670	22

Table 1: Performance on each problem in our test set, comparing two encodings of weighted abduction into Markov logic networks and a gold standard.

each iteration both drops axioms that are shown to be irrelevant and simplifies remaining axioms in such a way as not to change the probability of entailment.

We also simplify predications by removing unnecessary arguments. The most natural way to convert FrameNet frames to axioms is to treat a frame as a predicate whose arguments are the frame elements for all of its roles. After converting to Markov logic, this results in rules having large numbers of existentially quantified variables in the consequent. This can lead to a combinatorial explosion in the number of possible ground rules. Many of the variables in the frame predicate are for general use and can be pruned in the particular entailment. Our approach essentially creates abstractions of the original predicates that preserve all the information that is relevant to the current problem but greatly reduces the number of ground instances to consider.

Before implementing these pre-processing steps, only two or three problems could be run to completion on a Macbook Pro with 8 gigabytes of RAM. After making them, 28 of the initial 100 problems could be run to completion.

Work on this effort continues.

6 Summary

Weighted abduction is a logical reasoning framework that has been successfully applied to solve a number of interesting and important problems in computational natural-language semantics ranging from word sense disambiguation to coreference resolution. However, its method for representing and combining assumption costs to determine the most preferred explanation is ad hoc and without a firm theoretical foundation. Markov Logic is a recently developed formalism for combining first-order logic with probabilistic graphical models that has a well-defined formal semantics in terms of specifying a probability distribution over possible worlds. This paper has presented a method for mapping weighted abduction

to Markov logic, thereby providing a sound probabilistic semantics for the approach and also allowing it to exploit the growing toolbox of inference and learning algorithms available for Markov logic. Complementarily, it has also demonstrated how Markov logic can thereby be applied to help solve important problems in computational semantics.

References

- [1] Roy Bar-Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. The second pascal recognising textual entailment challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, Venice, Italy, 2006.
- [2] Eugene Charniak and Solomon E. Shimony. Cost-based abduction and map explanation. *Artificial Intelligence Journal*, 66(2):345–374, 1994.
- [3] P. T. Cox and T. Pietrzykowski. Causes for events: Their computation and applications. In J. Siekmann, editor, *8th International Conference on Automated Deduction (CADE-8)*, Berlin, 1986. Springer-Verlag.
- [4] P. Domingos and D. Lowd. *Markov Logic: An Interface Layer for Artificial Intelligence*. Morgan & Claypool, San Rafael, CA, 2009.
- [5] L. Getoor and B. Taskar, editors. *Introduction to Statistical Relational Learning*. MIT Press, Cambridge, MA, 2007.
- [6] S. Harabagiu and D.I. Moldovan. Lcc’s question answering system. In *11th Text Retrieval Conference, TREC-11*, Gaithersburg, MD., 2002.
- [7] Jerry R. Hobbs, Mark E. Stickel, Douglas E. Appelt, and Paul A. Martin. Interpretation as abduction. *Artificial Intelligence*, 63(1-2):69–142, 1993.
- [8] Rohit Kate and Ray Mooney. Probabilistic abduction using markov logic networks. In *IJCAI 09 Workshop on Plan, Activity and Intent Recognition*, 2009.
- [9] Hector J. Levesque. A knowledge-level account of abduction. In *Eleventh International Joint Conference on Artificial Intelligence*, volume 2, pages 1061–1067, Detroit, Michigan, 1989.
- [10] Hwee Tou Ng and Raymond J. Mooney. The role of coherence in constructing and evaluating abductive explanations. In P. O’Rorke, editor, *Working Notes, AAAI Spring Symposium on Automated Abduction*, Stanford, California, March 1990.
- [11] E. Ovchinnikova, N. Montazeri, T. Alexandrov, J. Hobbs, M. McCord, and R. Mulkar-Mehta. Abductive reasoning with a large knowledge base for discourse processing. In *Proceedings of the 9th International Conference on Computational Semantics*, Oxford, United Kingdom, 2011.
- [12] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Francisco, CA, 1988.
- [13] Harry E. Pople. On the mechanization of abductive logic. In *Third International Joint Conference on Artificial Intelligence*, pages 147–152, Stanford, California, August 1973.
- [14] M. Richardson and P. Domingos. Markov logic networks. *Machine Learning*, 62:107–136, 2006.