Montague Meets Markov: Deep Semantics with Probabilistic Logical Form

Islam Beltagy[§], **Cuong Chau**[§], **Gemma Boleda**[†], **Dan Garrette**[§], **Katrin Erk**[†], **Raymond Mooney**[§]
[§]Department of Computer Science
[†]Department of Linguistics
The University of Texas at Austin
Austin, Texas 78712
[§]{beltagy,ckcuong,dhg,mooney}@cs.utexas.edu
[†]gemma.boleda@utcompling.com,katrin.erk@mail.utexas.edu

Abstract

We combine logical and distributional representations of natural language meaning by transforming distributional similarity judgments into weighted inference rules using Markov Logic Networks (MLNs). We show that this framework supports both judging sentence similarity and recognizing textual entailment by appropriately adapting the MLN implementation of logical connectives. We also show that distributional phrase similarity, used as textual inference rules created on the fly, improves its performance.

1 Introduction

Tasks in natural language semantics are very diverse and pose different requirements on the underlying formalism for representing meaning. Some tasks require a detailed representation of the structure of complex sentences. Some tasks require the ability to recognize near-paraphrases or degrees of similarity between sentences. Some tasks require logical inference, either exact or approximate. Often it is necessary to handle ambiguity and vagueness in meaning. Finally, we frequently want to be able to learn relevant knowledge automatically from corpus data.

There is no single representation for natural language meaning at this time that fulfills all requirements. But there are representations that meet some of the criteria. Logic-based representations (Montague, 1970; Kamp and Reyle, 1993) provide an expressive and flexible formalism to express even complex propositions, and they come with standardized inference mechanisms. Distributional models (Turney and Pantel, 2010) use contextual similarity to predict semantic similarity of words and phrases (Landauer and Dumais, 1997; Mitchell and Lapata, 2010), and to model polysemy (Schütze, 1998; Erk and Padó, 2008; Thater et al., 2010). This suggests that distributional models and logic-based representations of natural language meaning are complementary in their strengths (Grefenstette and Sadrzadeh, 2011; Garrette et al., 2011), which encourages developing new techniques to combine them.

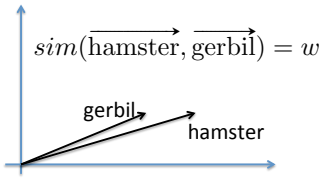

Figure 1: Turning distributional similarity into a weighted inference rule

Garrette et al. (2011; 2013) propose a framework for combining logic and distributional models in which logical form is the primary meaning representation. Distributional similarity between pairs of words is converted into weighted inference rules that are added to the logical form, as illustrated in Figure 1. Finally, Markov Logic Networks (Richardson and Domingos, 2006) (MLNs) are used to perform weighted inference on the resulting knowledge base. However, they only employed single-word distributional similarity rules, and only evaluated on a small

set of short, hand-crafted test sentences.

In this paper, we extend Garrette et al.'s approach and adapt it to handle two existing semantic tasks: recognizing textual entailment (RTE) and semantic textual similarity (STS). We show how this single semantic framework using probabilistic logical form in Markov logic can be adapted to support both of these important tasks. This is possible because MLNs constitute a flexible programming language based on probabilistic logic (Domingos and Lowd, 2009) that can be easily adapted to support multiple types of linguistically useful inference.

At the word and short phrase level, our approach model entailment through "distributional" similarity (Figure 1). If X and Y occur in similar contexts, we assume that they describe similar entities and thus there is some degree of entailment between them. At the sentence level, however, we hold that a stricter, logic-based view of entailment is beneficial, and we even model sentence similarity (in STS) as entailment.

There are two main innovations in the formalism that make it possible for us to work with naturally occurring corpus data. First, we use *more expressive distributional inference rules* based on the similarity of phrases rather than just individual words. In comparison to existing methods for creating textual inference rules (Lin and Pantel, 2001b; Szpektor and Dagan, 2008), these rules are computed on the fly as needed, rather than pre-compiled. Second, we use *more flexible probabilistic combinations of evidence* in order to compute degrees of sentence similarity for STS and to help compensate for parser errors. We replace deterministic conjunction by an average combiner, which encodes causal independence (Natarajan et al., 2010).

We show that our framework is able to handle both sentence similarity (STS) and textual entailment (RTE) by making some simple adaptations to the MLN when switching between tasks. The framework achieves reasonable results on both tasks. On STS, we obtain a correlation of $r = 0.66$ with full logic, $r = 0.73$ in a system with weakened variable binding, and $r = 0.85$ in an ensemble model. On RTE-1 we obtain an accuracy of 0.57. We show that the distributional inference rules benefit both tasks and that more flexible probabilistic combinations of evidence are crucial for STS. Although other approaches could be adapted to handle both RTE and STS, we do not know of any other methods that have been explicitly tested on both problems.

2 Related work

Distributional semantics Distributional models define the semantic relatedness of words as the similarity of vectors representing the contexts in which they occur (Landauer and Dumais, 1997; Lund and Burgess, 1996). Recently, such models have also been used to represent the meaning of larger phrases. The simplest models compute a phrase vector by adding the vectors for the individual words (Landauer and Dumais, 1997) or by a component-wise product of word vectors (Mitchell and Lapata, 2008; Mitchell and Lapata, 2010). Other approaches, in the emerging area of distributional compositional semantics, use more complex methods that compute phrase vectors from word vectors and tensors (Baroni and Zamparelli, 2010; Grefenstette and Sadrzadeh, 2011).

Wide-coverage logic-based semantics Boxer (Bos, 2008) is a software package for wide-coverage semantic analysis that produces logical forms using Discourse Representation Structures (Kamp and Reyle, 1993). It builds on the C&C CCG parser (Clark and Curran, 2004).

Markov Logic In order to combine logical and probabilistic information, we draw on existing work in Statistical Relational AI (Getoor and Taskar, 2007). Specifically, we utilize Markov Logic Networks (MLNs) (Domingos and Lowd, 2009), which employ weighted formulas in first-order logic to compactly encode complex undirected probabilistic graphical models. MLNs are well suited for our approach since they provide an elegant framework for assigning weights to first-order logical rules, combining a diverse set of inference rules and performing sound probabilistic inference.

An MLN consists of a set of weighted first-order clauses. It provides a way of softening first-order logic by allowing situations in which not all clauses are satisfied. More specifically, they provide a well-founded probability distribution across possible worlds by specifying that the probability of a

world increases exponentially with the total weight of the logical clauses that it satisfies. While methods exist for learning MLN weights directly from training data, since the appropriate training data is lacking, our approach uses weights computed using distributional semantics. We use the open-source software package Alchemy (Kok et al., 2005) for MLN inference, which allows computing the probability of a query literal given a set of weighted clauses as background knowledge and evidence.

Tasks: RTE and STS Recognizing Textual Entailment (RTE) is the task of determining whether one natural language text, the *premise*, implies another, the *hypothesis*. Consider (1) below.

(1) *p:* Oracle had fought to keep the forms from being released
h: Oracle released a confidential document

Here, *h* is not entailed. RTE directly tests whether a system can construct semantic representations that allow it to draw correct inferences. Of existing RTE approaches, the closest to ours is by Bos and Markert (2005), who employ a purely logical approach that uses Boxer to convert both the premise and hypothesis into first-order logic and then checks for entailment using a theorem prover. By contrast, our approach uses Markov logic with probabilistic inference.

Semantic Textual Similarity (STS) is the task of judging the similarity of two sentences on a scale from 0 to 5 (Agirre et al., 2012). Gold standard scores are averaged over multiple human annotations. The best performer in 2012's competition was by Bär et al. (2012), an ensemble system that integrates many techniques including string similarity, n-gram overlap, WordNet similarity, vector space similarity and MT evaluation metrics.

Weighted inference, and combined structural-distributional representations One approach to weighted inference in NLP is that of Hobbs et al. (1993), who proposed viewing natural language interpretation as abductive inference. In this framework, problems like reference resolution and syntactic ambiguity resolution become inferences to best explanations that are associated with costs. However, this leaves open the question of how costs are assigned. Raina et al. (2005) use this framework for RTE, deriving inference costs from WordNet similarity and properties of the syntactic parse.

Garrette et al. (2011; 2013) proposed an approach to RTE that uses MLNs to combine traditional logical representations with distributional information in order to support probabilistic textual inference. This approach can be viewed as a bridge between Bos and Markert (2005)'s purely logical approach, which relies purely on hard logical rules and theorem proving, and distributional approaches, which support graded similarity between concepts but have no notion of logical operators or entailment.

There are also other methods that combine distributional and structured representations. Stern et al. (2011) conceptualize textual entailment as tree rewriting of syntactic graphs, where some rewriting rules are distributional inference rules. Socher et al. (2011) recognize paraphrases using a "tree of vectors," a phrase structure tree in which each constituent is associated with a vector, and overall sentence similarity is computed by a classifier that integrates all pairwise similarities. (This is in contrast to approaches like Baroni and Zamparelli (2010) and Grefenstette and Sadrzadeh (2011), who do not offer a proposal for using vectors at multiple levels in a syntactic tree simultaneously.)

3 MLN system

Our system extends that of Garrette et al. (2011; 2013) to support larger-scale evaluation on standard benchmarks for both RTE and STS. We conceptualize both tasks as probabilistic entailment in Markov logic, where STS is judged as the average degree of mutual entailment, i.e. we compute the probability of both $S_1 \models S_2$ and $S_2 \models S_1$ and average the results. Below are some sentence pairs that we use as examples in the discussion below:

(2) S_1: A man is slicing a cucumber.
S_2: A man is slicing a zucchini.

(3) S_1: A boy is riding a bicycle.
S_2: A little boy is riding a bike.

(4) S_1: A man is driving.
S_2: A man is driving a car.

System overview. To compute the probability of an entailment $S_1 \models S_2$, the system first constructs logical forms for each sentence using Boxer and then translates them into MLN clauses. In example (2) above, the logical form for S_1:

$$\exists x_0, e_1, x_2 \big(man(x_0) \wedge slice(e_1) \wedge Agent(e_1, x_0) \wedge cucumber(x_2) \wedge Patient(e_1, x_2) \big)$$

is used as evidence, and the logical form for S_2 is turned into the following formula (by default, variables are assumed to be universally quantified):

$$man(x) \wedge slice(e) \wedge Agent(e, x) \wedge zucchini(y) \wedge Patient(e, y) \rightarrow result()$$

where $result()$ is the query for which we have Alchemy compute the probability.

However, S_2 is not strictly entailed by S_1 because of the mismatch between "cucumber" and "zucchini", so with just the strict logical-form translations of S_1 and S_2, the probability of $result()$ will be zero. This is where we introduce distributional similarity, in this case the similarity of "cucumber" and "zucchini", $\cos(\overrightarrow{\text{cucumber}}, \overrightarrow{\text{zucchini}})$. We create inference rules from such similarities as a form of background knowledge. We then treat similarity as degree of entailment, a move that has a long tradition (e.g., (Lin and Pantel, 2001b; Raina et al., 2005; Szpektor and Dagan, 2008)). In general, given two words a and b, we transform their cosine similarity into an inference-rule weight $wt(a, b)$ using:

$$wt(a, b) = log(\frac{\cos(\vec{a}, \vec{b})}{1 - \cos(\vec{a}, \vec{b})}) - prior \qquad (5)$$

Where $prior$ is a negative weight used to initialize all predicates, so that by default facts are assumed to have very low probability. In our experiments, we use $prior = -3$. In the case of sentence pair (2), we generate the inference rule:

$$cucumber(x) \rightarrow zucchini(x) \mid wt(\text{cuc., zuc.})$$

Such inference rules are generated for all pairs of words (w_1, w_2) where $w_1 \in S_1$ and $w_2 \in S_2$.[1]

[1]We omit inference rules for words (a, b) where $\cos(a, b) < \theta$ for a threshold θ set to maximize performance on the training data. Low-similarity pairs usually indicate dissimilar words. This removes a sizeable number of rules for STS, while for RTE the tuned threshold was near zero.

The distributional model we use contains all lemmas occurring at least 50 times in the Gigaword corpus (Graff et al., 2007) except a list of stop words. The dimensions are the 2,000 most frequent of these words, and cell values are weighted with point-wise mutual information. [2]

Phrase-based inference rules. Garrette et al. only considered distributional inference rules for pairs of individual words. We extend their approach to distributional inference rules for pairs of phrases in order to handle cases like (3). To properly estimate the similarity between S_1 and S_2 in (3), we not only need an inference rule linking "bike" to "bicycle", but also a rule estimating how similar "boy" is to "little boy". To do so, we make use of existing approaches that compute distributional representations for phrases. In particular, we compute the vector for a phrase from the vectors of the words in that phrase, using either vector addition (Landauer and Dumais, 1997) or component-wise multiplication (Mitchell and Lapata, 2008; Mitchell and Lapata, 2010). The inference-rule weight, $wt(p_1, p_2)$, for two phrases p_1 and p_2 is then determined using Eq. (5) in the same way as for words.

Existing approaches that derive phrasal inference rules from distributional similarity (Lin and Pantel, 2001a; Szpektor and Dagan, 2008; Berant et al., 2011) precompile large lists of inference rules. By comparison, distributional phrase similarity can be seen as a generator of inference rules "on the fly", as it is possible to compute distributional phrase vectors for arbitrary phrases on demand as they are needed for particular examples.

Inference rules are generated for all pairs of constituents (c_1, c_2) where $c_1 \in S_1$ and $c_2 \in S_2$, a constituent is a single word or a phrase. The logical form provides a handy way to extract phrases, as they are generally mapped to one of two logical constructs. Either we have multiple single-variable predicates operating on the same variable. For example the phrase "a little boy" has the logical form $boy(x) \wedge little(x)$. Or we have two unary predicates connected with a relation. For example, "pizza slice" and "slice of pizza" are both mapped to the

[2]It is customary to transform raw counts in a way that captures association between target words and dimensions, for example through point-wise mutual information (Lowe, 2001).

logical form, $slice(x_0) \wedge of(x_0, x_1) \wedge pizza(x_1)$. We consider all binary predicates as relations.

Average Combiner to determine similarity in the presence of missing phrases. The logical forms for the sentences in (4): are

S_1: $\exists x_0, e_1 \big(man(x_0) \wedge agent(x_0, e_1) \wedge drive(e_1)\big)$

S_2: $\exists x_0, e_1, x_2 \big(man(x_0) \wedge agent(x_0, e_1) \wedge drive(e_1) \wedge patient(e_1, x_2) \wedge car(x_2)\big)$

If we try to prove $S_1 \models S_2$, the probability of the $result()$ will be zero: There is no evidence for a car, and the hypothesis predicates are conjoined using a deterministic AND. For RTE, this makes sense: If one of the hypothesis predicates is False, the probability of entailment should be zero. For the STS task, this should in principle be the same, at least if the omitted facts are vital, but it seems that annotators rated the data points in this task more for overall similarity than for degrees of entailment. So in STS, we want the similarity to be a function of the number of elements in the hypothesis that are inferable. Therefore, we need to replace the deterministic AND with a different way of combining evidence. We chose to use the average evidence combiner for MLNs introduced by Natarajan et al. (2010). To use the average combiner, the full logical form is divided into smaller clauses (which we call mini-clauses), then the combiner averages their probabilities. In case the formula is a list of conjuncted predicates, a mini-clause is a conjunction of a single-variable predicate with a relation predicate(as in the example below). In case the logical form contains a negated sub-formula, the negated sub-formula is also a mini-clause. The hypothesis above after dividing clauses for the average combiner looks like this:

$$man(x_0) \wedge agent(x_0, e_1) \rightarrow result(x_0, e_1, x_2) \mid w$$
$$drive(e_1) \wedge agent(x_0, e_1) \rightarrow result(x_0, e_1, x_2) \mid w$$
$$drive(e_1) \wedge patient(e_1, x_2) \rightarrow result(x_0, e_1, x_2) \mid w$$
$$car(x_2) \wedge patient(e_1, x_2) \rightarrow result(x_0, e_1, x_2) \mid w$$

where $result$ is again the query predicate. Here, $result$ has all of the variables in the clause as arguments in order to maintain the binding of variables across all of the mini-clauses. The weights w are the following function of n, the number of mini-clauses (4 in the above example):

$$w = \frac{1}{n} \times (log(\frac{p}{1-p}) - prior) \qquad (6)$$

where p is a value close to 1 that is set to maximize performance on the training data, and $prior$ is the same negative weight as before. Setting w this way produces a probability of p for the $result()$ in cases that satisfy the antecedents of *all* mini-clauses. For the example above, the antecedents of the first two mini-clauses are satisfied, while the antecedents of the last two are not since the premise provides no evidence for an object of the verb $drive$. The similarity is then computed to be the maximum probability of any grounding of the $result$ predicate, which in this case is around $\frac{p}{2}$.[3]

An interesting variation of the average combiner is to omit variable bindings between the mini-clauses. In this case, the hypothesis clauses look like this for our example:

$$man(x) \wedge agent(x, e) \rightarrow result() \mid w$$
$$drive(e) \wedge agent(x, e) \rightarrow result() \mid w$$
$$drive(e) \wedge patient(e, x) \rightarrow result() \mid w$$
$$car(x) \wedge patient(e, x) \rightarrow result() \mid w$$

This implementation loses a lot of information, for example it does not differentiate between "A man is walking and a woman is driving" and "A man is driving and a woman is walking". In fact, logical form without variable binding degrades to a representation similar to a set of independent syntactic dependencies, [4] while the average combiner with variable binding retains all of the information in the original logical form. Still, omitting variable binding turns out to be useful for the STS task.

It is also worth commenting on the efficiency of the inference algorithm when run on the three different approaches to combining evidence. The average combiner without variable binding is the fastest and has the least memory requirements because all cliques in the ground network are of limited size (just 3 or 4 nodes). Deterministic AND is much slower than the average combiner without variable binding, because the maximum clique size depends on the sentence. The average combiner *with* variable binding is the most memory intensive since the

[3] One could also give mini-clauses different weights depending on their importance, but we have not experimented with this so far.

[4] However, it is not completely the same since we do not divide up formulas under negation into mini-clauses.

number of arguments of the $result()$ predicate can become large (there is an argument for each individual and event in the sentence). Consequently, the inference algorithm needs to consider a combinatorial number of possible groundings of the $result()$ predicate, making inference very slow.

Adaptation of the logical form. As discussed by Garrette et al. (2011), Boxer's output is mapped to logical form and augmented with additional information to handle a variety of semantic phenomena. However, we do not use their additional rules for handling implicatives and factives, as we wanted to test the system without background knowledge beyond that supplied by the vector space.

Unfortunately, current MLN inference algorithms are not able to efficiently handle complex formulas with nested quantifiers. For that reason, we replaced universal quantifiers in Boxer's output with existentials since they caused serious problems for Alchemy. Although this is a radical change to the semantics of the logical form, due to the nature of the STS and RTE data, it only effects about 5% of the sentences, and we found that most of the universal quantifiers in these cases were actually due to parsing errors. We are currently exploring more effective ways of dealing with this issue.

4 Task 1: Recognizing Textual Entailment

4.1 Dataset

In order to compare directly to the logic-based system of Bos and Markert (2005), we focus on the RTE-1 dataset (Dagan et al., 2005), which includes 567 Text-Hypothesis (T-H) pairs in the development set and 800 pairs in the test set. The data covers a wide range of issues in entailment, including lexical, syntactic, logical, world knowledge, and combinations of these at different levels of difficulty. In both development and test sets, 50% of sentence pairs are true entailments and 50% are not.

4.2 Method

We run our system for different configurations of inference rules and evidence combiners. For distributional inference rules (DIR), three different levels are tested: without inference rules (**no DIR**), inference rules for individual words (**word DIR**), and inference rules for words and phrases (**phrase DIR**). Phrase vectors were built using vector addition, as point-wise multiplication performed slightly worse. To combine evidence for the $result()$ query, three different options are available: without average combiner which is just using Deterministic AND (**Deterministic AND**), average combiner with variable binding (**AvgComb**) and average combiner without variable binding (**AvgComb w/o VarBind**). Different combinations of configurations are tested according to its suitability for the task; RTE and STS.

We also tested several "distributional only" systems. The first such system builds a vector representation for each sentence by adding its word vectors, then computes the cosine similarity between the sentence vectors for S_1 and S_2 (**VS-Add**). The second uses point-wise multiplication instead of vector addition (**VS-Mul**). The third scales pairwise words similarities to the sentence level using weighted average where weights are inverse document frequencies idf as suggested by Mihalcea et al. (2006) (**VS-Pairwise**).

For the RTE task, systems were evaluated using both *accuracy* and *confidence-weighted score* (*cws*) as used by Bos and Markert (2005) and the RTE-1 challenge (Dagan et al., 2005). In order to map a probability of entailment to a strict prediction of True or False, we determined a threshold that optimizes performance on the development set. The cws score rewards a system's ability to assign higher confidence scores to correct predictions than incorrect ones. For cws, a system's predictions are sorted in decreasing order of confidence and the score is computed as:

$$cws = \frac{1}{n}\sum_{i=1}^{n}\frac{\#\text{correct-up-to-rank-i}}{i}$$

where *n* is the number of the items in the test set, and *i* ranges over the sorted items. In our systems, we defined the confidence value for a T-H pair as the distance between the computed probability for the $result()$ predicate and the threshold.

4.3 Results

The results are shown in Table 1. They show that the distributional only baselines perform very poorly. In particular, they perform worse than strict

Method	acc	cws
Chance	0.50	0.50
Bos & Markert, strict	0.52	0.55
Best system in RTE-1 challenge (Bayer et al., 2005)	0.59	0.62
VS-Add	0.49	0.53
VS-Mul	0.51	0.52
VS-Pairwise	0.50	0.50
AvgComb w/o VarBind + phrase DIR	0.52	0.53
Deterministic AND + phrase DIR	0.57	0.57

Table 1: Results on the RTE-1 Test Set.

entailment from Bos and Markert (2005), a system that uses only logic. This illustrates the important role of logic-based representations for the entailment task. Due to intractable memory demands of Alchemy inference, our current system with deterministic AND fails to execute on 118 of the 800 test pairs, so, by default, the system classifies these cases as False (non-entailing) with very low confidence. Comparing the two configurations of our system, using deterministic AND vs. the average combiner without variable binding (last two lines in Table 1), we see that for RTE, it is essential to retain the full logical form.

Our system with deterministic AND obtains both an accuracy and cws of 0.57. The best result in the RTE-1 challenge by Bayer et al. (2005) obtained an accuracy of 0.59 and cws of 0.62. [5] In terms of both accuracy and cws, our system outperforms both "distributional only" systems and strict logical entailment, showing again that integrating both logical form and distributional inference rules using MLNs is beneficial. Interestingly, the strict entailment system of Bos and Markert incorporated generic knowledge, lexical knowledge (from WordNet) and geographical knowledge that we do not utilize. This demonstrates the advantage of using a model that operationalizes entailment between words and phrases as distributional similarity.

[5] On other RTE datasets there are higher previous results. Hickl (2008) achieves 0.89 accuracy and 0.88 cws on the combined RTE-2 and RTE-3 dataset.

5 Task 2: Semantic Textual Similarity

5.1 Dataset

The dataset we use in our experiments is the MSR Video Paraphrase Corpus (MSR-Vid) subset of the STS 2012 task, consisting of 1,500 sentence pairs. The corpus itself was built by asking annotators from Amazon Mechanical Turk to describe very short video fragments (Chen and Dolan, 2011). The organizers of the STS 2012 task (Agirre et al., 2012) sampled video descriptions and asked Turkers to assign similarity scores (ranging from 0 to 5) to pairs of sentences, without access to the video. The gold standard score is the average of the Turkers' annotations. In addition to the MSR Video Paraphrase Corpus subset, the STS 2012 task involved data from machine translation and sense descriptions. We do not use these because they do not consist of full grammatical sentences, which the parser does not handle well. In addition, the STS 2012 data included sentences from the MSR Paraphrase Corpus, which we also do not currently use because some sentences are long and create intractable MLN inference problems. This issue is discussed further in section 6. Following STS standards, our evaluation compares a system's similarity judgments to the gold standard scores using Pearson's correlation coefficient r.

5.2 Method

Our system can be tested for different configuration of inference rules and evidence combiners which are explained in section 4.2. The tested systems on the STS task are listed in table 2. Out experiments showed that using average combiner (**AvgComb**) is very memory intensive and MLN inference for 28 of the 1,500 pairs either ran out of memory or did not finish in reasonable time. In such cases, we back off to **AvgComb w/o VarBind**.

We compare to several baselines; our MLN system without distributional inference rules (**AvgComb + no DIR**), and distributional-only systems (**VS-Add**, **VS-Mul**, **VS-Pairwise**). These are the natural baselines for our system, since they use only one of the two types of information that we combine (i.e. logical form and distributional representations).

Finally, we built an ensemble that combines the output of multiple systems using regression trained

Method	r
AvgComb + no DIR	0.58
AvgComb + word DIR	0.60
AvgComb + phrase DIR	0.66
AvgComb w/o VarBind + no DIR	0.58
AvgComb w/o VarBind + word DIR	0.60
AvgComb w/o VarBind + phrase DIR	0.73
VS-Add	0.78
VS-Mul	0.58
VS-Pairwise	0.77
Ensemble (VS-Add + VS-Mul + VS-Pairwise)	0.83
Ensemble ([AvgComb + phrase DIR] + VS-Add + VS-Mul + VS-Pairwise)	0.85
Best MSR-Vid score in STS 2012 (Bär et al., 2012)	0.87

Table 2: Results on the STS video dataset.

on the training data. We then compare the performance of an ensemble with and without our system. This is the same technique used by Bär et al. (2012) except we used additive regression (Friedman, 2002) instead of linear regression since it gave better results.

5.3 Results

Table 2 summarizes the results of our experiments. They show that adding distributional information improves results, as expected, and also that adding phrase rules gives further improvement: Using only word distributional inference rules improves results from 0.58 to 0.6, and adding phrase inference rules further improves them to 0.66. As for variable binding, note that although it provides more precise information, the STS scores actually improve when it is dropped, from 0.66 to 0.73. We offer two explanations for this result: First, this information is very sensitive to parsing errors, and the C&C parser, on which Boxer is based, produces many errors on this dataset, even for simple sentences. When the C&C CCG parse is wrong, the resulting logical form is wrong, and the resulting similarity score is greatly affected. Dropping variable binding makes the systems more robust to parsing errors. Second, in contrast to RTE, the STS dataset does not really test the important role of syntax and logical form in determining meaning. This also explains why the "distributional only" baselines are actually doing better than the MLN systems.

Although the MLN system on its own does not perform better than the distributional compositional models, it does provide complementary information, as shown by the fact that ensembling it with the rest of the models improves performance (0.85 with the MLN system, compared to 0.83 without it). The performance of this ensemble is close to the current best result for this dataset (0.87).

6 Future Work

The approach presented in this paper constitutes a step towards achieving the challenging goal of effectively combining logical representations with distributional information automatically acquired from text. In this section, we discuss some of limitations of the current work and directions for future research.

As noted before, parse errors are currently a significant problem. We use Boxer to obtain a logical representation for a sentence, which in turn relies on the C&C parser. Unfortunately, C&C misparses many sentences, which leads to inaccurate logical forms. To reduce the impact of misparsing, we plan to use a version of C&C that can produce the top-n parses together with parse re-ranking (Ng and Curran, 2012). As an alternative to re-ranking, one could obtain logical forms for each of the top-n parses, and create an MLN that integrates all of them (together with their certainty) as an underspecified meaning representation that could then be used to directly support inferences such as STS and RTE.

We also plan to exploit a greater variety of distributional inference rules. First, we intend to incorporate logical form translations of existing distributional inference rule collections (e.g., (Berant et al., 2011; Chan et al., 2011)). Another issue is obtaining improved rule weights based on distributional phrase vectors. We plan to experiment with more sophisticated approaches to computing phrase vectors such as those recently presented by Baroni and Zamparelli (2010) and Grefenstette and Sadrzadeh (2011). Furthermore, we are currently deriving symmetric similarity ratings between word pairs or phrase pairs, when really what we need is di-

rectional similarity. We plan to incorporate directed similarity measures such as those of Kotlerman et al. (2010) and Clarke (2012).

A primary problem for our approach is the limitations of existing MLN inference algorithms, which do not effectively scale to large and complex MLNs. We plan to explore "coarser" logical representations such as Minimal Recursion Semantics (MRS) (Copestake et al., 2005). Another potential approach to this problem is to trade expressivity for efficiency. Domingos and Webb (2012) introduced a tractable subset of Markov Logic (TML) for which a future software release is planned. Formulating the inference problem in TML could potentially allow us to run our system on longer and more complex sentences.

7 Conclusion

In this paper we have used an approach that combines logic-based and distributional representations for natural language meaning. It uses logic as the primary representation, transforms distributional similarity judgments to weighted inference rules, and uses Markov Logic Networks to perform inferences over the weighted clauses. This approach views textual entailment and sentence similarity as degrees of "logical" entailment, while at the same time using distributional similarity as an indicator of entailment at the word and short phrase level. We have evaluated the framework on two different tasks, RTE and STS, finding that it is able to handle both tasks given that we adapt the way evidence is combined in the MLN. Even though other entailment models could be applied to STS, given that similarity can obviously be operationalized as a degree of mutual entailment, this has not been done before to our best knowledge. Our framework achieves reasonable results on both tasks. On RTE-1 we obtain an accuracy of 0.57. On STS, we obtain a correlation of $r = 0.66$ with full logic, $r = 0.73$ in a system with weakened variable binding, and $r = 0.85$ in an ensemble model. We find that distributional word and phrase similarity, used as textual inference rules on the fly, leads to sizeable improvements on both tasks. We also find that using more flexible probabilistic combinations of evidence is crucial for STS.

Acknowledgements

This research was supported in part by the NSF CAREER grant IIS 0845925, by the DARPA DEFT program under AFRL grant FA8750-13-2-0026, by MURI ARO grant W911NF-08-1-0242 and by an NDSEG grant. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the view of DARPA, AFRL, ARO, DoD or the US government.

Some of our experiments were run on the Mastodon Cluster supported by NSF Grant EIA-0303609.

References

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In *Proceedings of SemEval*.

Daniel Bär, Chris Biemann, Iryna Gurevych, and Torsten Zesch. 2012. UKP: Computing semantic textual similarity by combining multiple content similarity measures. In *SemEval-2012*.

Marco Baroni and Roberto Zamparelli. 2010. Nouns are vectors, adjectives are matrices: Representing adjective-noun constructions in semantic space. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1183–1193, Cambridge, MA, October. Association for Computational Linguistics.

Samuel Bayer, John Burger, Lisa Ferro, John Henderson, and Alexander Yeh. 2005. MITREs Submissions to the EU Pascal RTE Challenge. In *In Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment*, pages 41–44.

Jonathan Berant, Ido Dagan, and Jacob Goldberger. 2011. Global learning of typed entailment rules. In *Proceedings of ACL*, Portland, OR.

Johan Bos and Katja Markert. 2005. Recognising textual entailment with logical inference. In *Proceedings of EMNLP 2005*, pages 628–635, Vancouver, B.C., Canada.

Johan Bos. 2008. Wide-coverage semantic analysis with boxer. In Johan Bos and Rodolfo Delmonte, editors, *Semantics in Text Processing. STEP 2008 Conference Proceedings*, Research in Computational Semantics, pages 277–286. College Publications.

Tsz Ping Chan, Chris Callison-Burch, and Benjamin Van Durme. 2011. Reranking bilingually extracted

paraphrases using monolingual distributional similarity. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, pages 33–42, Edinburgh, UK.

David L. Chen and William B. Dolan. 2011. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 190–200, Portland, Oregon, USA, June.

Stephen Clark and James R. Curran. 2004. Parsing the WSJ using CCG and log-linear models. In *Proceedings of ACL 2004*, pages 104–111, Barcelona, Spain.

Daoud Clarke. 2012. A context-theoretic framework for compositionality in distributional semantics. *Computational Linguistics*, 38(1).

Ann Copestake, Dan Flickinger, Carl Pollard, and Ivan A Sag. 2005. Minimal recursion semantics: An introduction. *Research on Language and Computation*, 3(2-3):281–332.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The PASCAL Recognising Textual Entailment Challenge. In *In Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment*, pages 1–8.

Pedro Domingos and Daniel Lowd. 2009. *Markov Logic: An Interface Layer for Artificial Intelligence*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers.

Pedro Domingos and W Austin Webb. 2012. A tractable first-order probabilistic logic. In *Proceedings of the Twenty-Sixth National Conference on Artificial Intelligence*.

Katrin Erk and Sebastian Padó. 2008. A structured vector space model for word meaning in context. In *Proceedings of EMNLP 2008*, pages 897–906, Honolulu, HI.

Jerome H Friedman. 2002. Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38(4):367–378.

Dan Garrette, Katrin Erk, and Raymond Mooney. 2011. Integrating logical representations with probabilistic information using markov logic. In *Proceedings of IWCS*, Oxford, UK.

Dan Garrette, Katrin Erk, and Raymond Mooney. 2013. A formal approach to linking logical form and vector-space lexical semantics. In Harry Bunt, Johan Bos, and Stephen Pulman, editors, *Computing Meaning, Vol. 4*.

Lise Getoor and Ben Taskar, editors. 2007. *Introduction to Statistical Relational Learning*. MIT Press, Cambridge, MA.

David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2007. English Gigaword Third Edition. `http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2007T07`.

Edward Grefenstette and Mehrnoosh Sadrzadeh. 2011. Experimental support for a categorical compositional distributional model of meaning. In *Proceedings of EMNLP*, Edinburgh, Scotland, UK.

Andrew Hickl. 2008. Using Discourse Commitments to Recognize Textual Entailment. In *Proceedings of COLING 2008*, pages 337–344.

Jerry R. Hobbs, Mark Stickel, Douglas Appelt, and Paul Martin. 1993. Interpretation as abduction. *Artificial Intelligence*, 63(1–2):69–142.

Hans Kamp and Uwe Reyle. 1993. *From Discourse to Logic; An Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and DRT*. Kluwer, Dordrecht.

Stanley Kok, Parag Singla, Matthew Richardson, and Pedro Domingos. 2005. The Alchemy system for statistical relational AI. Technical report, Department of Computer Science and Engineering, University of Washington. `http://www.cs.washington.edu/ai/alchemy`.

Lili Kotlerman, Ido Dagan, Idan Szpektor, and Maayan Zhitomirsky-Geffet. 2010. Directional distributional similarity for lexical inference. *Natural Language Engineering*, 16(04):359–389.

Thomas Landauer and Susan Dumais. 1997. A solution to Platos problem: the latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, 104(2):211–240.

Dekang Lin and Patrick Pantel. 2001a. DIRT - discovery of inference rules from text. In *In Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 323–328.

Dekang Lin and Patrick Pantel. 2001b. Discovery of inference rules for question answering. *Natural Language Engineering*, 7(4):343–360.

Will Lowe. 2001. Towards a theory of semantic space. In *Proceedings of the Cognitive Science Society*, pages 576–581.

Kevin Lund and Curt Burgess. 1996. Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments, and Computers*, 28:203–208.

Rada Mihalcea, Courtney Corley, and Carlo Strapparava. 2006. Corpus-based and knowledge-based measures of text semantic similarity. In *Proceedings of the national conference on artificial intelligence*, volume 21, page 775. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999.

Jeff Mitchell and Mirella Lapata. 2008. Vector-based models of semantic composition. In *Proceedings of ACL*, pages 236–244.

Jeff Mitchell and Mirella Lapata. 2010. Composition in distributional models of semantics. *Cognitive Science*, 34(8):1388–1429.

Richard Montague. 1970. Universal grammar. *Theoria*, 36:373–398. Reprinted in Thomason (1974), pp 7-27.

Sriraam Natarajan, Tushar Khot, Daniel Lowd, Prasad Tadepalli, Kristian Kersting, and Jude Shavlik. 2010. Exploiting causal independence in markov logic networks: Combining undirected and directed models. In *Proceedings of European Conference in Machine Learning (ECML)*, Barcelona, Spain.

Dominick Ng and James R Curran. 2012. Dependency hashing for n-best ccg parsing. In *In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*.

Rajat Raina, Andrew Y. Ng, and Christopher D. Manning. 2005. Robust textual inference via learning and abductive reasoning. In *Proceedings of AAAI*.

Matthew Richardson and Pedro Domingos. 2006. Markov logic networks. *Machine Learning*, 62:107–136.

Hinrich Schütze. 1998. Automatic word sense discrimination. *Computational Linguistics*, 24(1).

Richard Socher, Eric Huang, Jeffrey Pennin, Andrew Ng, and Christopher Manning. 2011. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. In J. Shawe-Taylor, R.S. Zemel, P. Bartlett, F.C.N. Pereira, and K.Q. Weinberger, editors, *Proceedings of NIPS*.

Asher Stern, Amnon Lotan, Shachar Mirkin, Eyal Shnarch, Lili Kotlerman, Jonathan Berant, and Ido Dagan. 2011. Knowledge and tree-edits in learnable entailment proofs. In *TAC*, Gathersburg, MD.

Idan Szpektor and Ido Dagan. 2008. Learning entailment rules for unary templates. In *Proceedings of COLING*.

Stefan Thater, Hagen Fürstenau, and Manfred Pinkal. 2010. Contextualizing semantic representations using syntactically enriched vector models. In *Proceedings of ACL 2010*, pages 948–957, Uppsala, Sweden.

Richmond H. Thomason, editor. 1974. *Formal Philosophy. Selected Papers of Richard Montague*. Yale University Press, New Haven.

Peter Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37:141–188.