Moving from Textual Relations to Ontologized Relations

Stephen Soderland and **Bhushan Mandhani**
Turing Center
Dept of Computer Science
University of Washington
Seattle, USA

Abstract

There has been recent research in open-ended information extraction from text that finds relational triples of the form (arg1, relation phrase, arg2), where the relation phrase is a text string that expresses a relation between two arbitrary noun phrases. While such a relational triple is a good first step, much further work is required to turn such a textual relation into a logical form that supports inferencing. The strings from arg1 and arg2 must be normalized, disambiguated, and mapped to a formal taxonomy. The relation phrase must likewise be normalized and mapped to a clearly defined logical relation. Some relation phrases can be mapped to a set of pre-defined relations such as Part-Of and Causes. We focus instead on arbitrary relation phrases that are discovered from text. For this, we need to automatically merge synonymous relations and discover meta-properties such as entailment. Ultimately, we want the coverage of a bottom-up approach together with the rich set of axioms associated with a top-down approach.

We have begun exploratory work in "ontologizing" the output of TextRunner, an open information extraction system that finds arbitrary relational triples from text. Our test domain is 2.5 million Web pages on health and nutrition, which yields relational triples such as (orange, contains, vitamin C) and (fruits, are rich in, antioxidants). We automatically disambiguate the strings arg1 and arg2, mapping them to WordNet synsets. We also learn entailments between normalized relation strings (e.g. "be rich in" entails "contain"). This enhanced ontology enables reasoning about relationships that are not seen in the corpus, but can be inferred by inheritance and entailment. Further, we define ontology-based relationships between the extracted triples themselves, and experimentally show that these can be used in significantly improving probability estimation for the triples.

1. Introduction

A dream from the early days of Artificial Intelligence is to build a system that can read autonomously with deep understanding. In the 1970's and 1980's natural language processing (NLP) research tackled this problem directly, but in a way that was too brittle to apply to unrestricted text. While we may dismiss this body of research as "toy systems" that required carefully selected input text, they genuinely performed deep reasoning about those texts.

Since the 1990's the mainstream of NLP research shifted to robust information extraction (IE) systems with shallow reasoning, often statistically based. Such IE systems can handle unseen text that the earlier NLP systems could not handle, but have the more modest goal of scanning for a limited number of predefined relations, and ignoring everything else. Current IE systems have hardly any mechanisms to make inferences beyond what is explicitly stated in the text.

However, the time may be ripe to combine the best of both eras, and build robust NLP systems that support deep reasoning. Consider textual relations with instances in the form (arg1, relation phrase, arg2) that have text strings for the arguments and the relation phrase. The argument strings must be normalized and mapped to a formal taxonomy to enable inheritance-based reasoning. Similarly, the relation string must be normalized and mapped to an axiomatized set of relations where possible. Learning entailments between relations can enable further inferencing.

The normalized instances of relations must be converted into probabilistic logical statements with clearly defined semantics. Suppose we have an instance (X, relation, Y) where X and Y are mapped to non-leaf nodes of a WordNet-like taxonomy. Does this mean that for all instances of X there exists an instance of Y where the relationship holds, or some other combination of universal and existential quantifiers? The ontologized instances should be in a format suitable for an inference engine that can make plausible inferences. We have taken some early steps in this research direction, but the bulk of these ideas still need to be tested and validated.

The remainder of this paper is organized as follows. Section 2 gives a brief overview of NLP research from the 1970's up to the present. Section 3 presents a case study of the steps needed to transform textual relations into ontologized relations. Section 4 gives some preliminary results where we use relationships between the ontologized relations for better confidence estimation of these relations. We conclude in Section 5 with some general discussion.

2. Trends in NLP Research

NLP research has seen extreme swings in goals and methods over the last thirty years. Research in the 1970's and 1980's lacked robust techniques, but had lofty goals of deep understanding. NLP systems by Roger Schank and his stu-

dents could read that John went into a restaurant, and infer that John was hungry and had a goal of eating, that he sat at a table, ordered food, ate it, paid, and left the restaurant. Good overviews of this work are given in (Schank & Abelson 1977; Dyer 1983; Lehnert 1988).

These deep understanding systems parsed the text into unambiguous meaning, represented by a network of semantic primitives (Schank & Reisbeck 1982) and a rich knowledge base used to make plausible inferences from this semantic representation. Unfortunately, neither the parsing nor the reasoning could scale to the variablility of naturally occurring text.

NLP research since the early 1990's has focused primarily on robustness, at the expense of deep reasoning. The Message Understanding Conferences (MUC) (Sundheim 1991; Chinchor 1998) spurred the NLP community to create systems that could selectively extract relevant information from previously unseen texts. However, hardly any inferencing is done beyond normalizing the information that goes into an output template. Statistical and machine learning systems have almost completely replaced the intricate hand-crafted knowledge of earlier language understanding systems.

Some recent IE systems have extracted domain-independent relations, but still narrowly focused on a few relations. Pennachiotti and Pantel developed a system that extracts the relations such as *is-a*, *part-of*, and *succession* from the Trec-9 corpus and *is-a*, *part-of*, *production*, and *reaction* from a chemistry corpus (Pennacchiotti & Pantel 2006). Snow, Jurafsky, and Ng used a hyponym classifier to extract *is-a* relations from a large collection of newswire and encyclopedia text (Snow, Jurafsky, & Ng 2006).

Recently, a few systems have been created that do "open IE". This is general purpose information extraction that operates bottom-up from a text corpus, without a predefined notion of what information is relevant. Open IE gathers all relational triples it finds, in the form (arg1, relation phrase, arg2), where arg1 and arg2 are noun phrases and the relation phrase expresses a logical relation of some kind between the two arguments.

The most advanced open IE system is TextRunner from Etzioni's research group (Banko *et al.* 2007), which extracts relational triples from text corpora using domain-independent, self-supervised learning. TextRunner merges duplicate triples and assigns a redundancy-based probability of correctness to each triple (Downey, Etzioni, & Soderland 2005). Examples of triples found from a corpus of Web pages on nutrition are (tomatoes, are high in, lycopene) and (antioxidant, can help prevent, many forms of cancer).

Recently some NLP systems are moving toward deep reasoning. An impressive example is the HALO project (Friedland *et al.* 2004), which combines a common sense knowledge base with knowledge gleaned from chemistry textbooks. The chemistry knowledge is input by a combination of automatic NLP and manual knowledge engineering. This general purpose and domain-specific knowledge enables an NLP system to read and answer questions from a standardized Advanced Placement test in Chemistry, and perform at a level close to that of typical high school students who take that test.

One of the three research teams for HALO used the University of Texas KM[1] component library (Barker, Porter, & Clark 2001) as their knowledge base. This knowledge base contains hundreds of axiomatized concepts, such as the concept *Be-Contained*, which is defined as a relation between a *Tangible-Entity* as the object (passive participant in an event) and a *Tangible-Entity* as the base (relatively fixed thing).

The KM system can reason about how *Be-Contained* is related to *Spatial-Entities* as origin and destination and how *Be-Contained* is related to the concepts *Encloses*, *Has-Region*, *Is-Inside*, *Is-Region-Of*, *Move-Into*, and *Move-Out-Of*. Such fully axiomatized concepts, along with the KM reasoner, provide the capability of deep understanding equal to that of Schank's semantic primitives.

3. Towards Deep Information Extraction

In this section, we will discuss the individual steps involved in ontologizing textual relations, and getting them ready for use in an automated reasoning system. We will use as our running example, textual relations from TextRunner for a corpus of nutrition Web pages. We enumerate the steps below. We will discuss many of these in greater detail in later sections.

1. **Normalize the argument phrases.** This involves mapping the argument strings to the "objects" of some logical domain. In this paper, we consider WordNet noun synsets as our objects domain. Normalizing arguments thus becomes a problem of word sense disambiguation. We need to be able to enhance this domain and add new synsets, if we want to handle unrestricted text. We don't yet cover this case in our implementation. Also note that normalization solves the important problem of identity resolution – do two strings s_i and s_j refer to the same real world entity or not?

2. **Normalize the relation phrase**. This involves normalizing to drop unimportant differences between phrases, such as adverbs or verb morphological form. It may also involve mapping predicate phrases to a pre-defined set of concepts.

3. **Formalize the semantics**. Ontologized relations must be given precise logical semantics before automated reasoning can be applied. Does (x, rel, y) mean that for all hyponyms of x the relation holds for all hyponyms of y, or some other combination of the two quantifiers?

4. **Learn meta-properties of relation phrases**. Are two relation phrases synonymous, or does relation r_1 existing between x and y imply that relation r_2 also holds between them (entailment)? Is a relation transitive with respect to another relation?

5. **Assign probability of correctness**. Automatically extracted relations will inevitably include errors. Can we use relationships between the ontologized relations to better estimate their probabilities. We later define such relationships, using the hyponymy and siblinghood relationships that we have between objects (available from Word-

---

[1]http://www.cs.utexas.edu/users/mfkb/RKF/clib.html

Net), and the entailment relationships that we have derived between relation phrases.

6. **Inference engine.** The inference engine should be able to combine the relations that were derived from text with other relations and axioms in a knowledge base.

3.1 Case Study: Fruit and Antioxidants

We first present some examples of TextRunner relations to better motivate the issues involved in executing the steps we listed above. Figure 1 shows three such relations. These illustrate textual relations that we want to ontologize. Some normalization has been done to the original source text phrases: modifiers and adverbs have been dropped, and nouns and verbs are transformed into their morphological base. This normalization, however, is not sufficient to enable machine reasoning about the relations.

---

1. (orange, contain, vitamin C)
   "One orange contains all the vitamin C your body needs for the day!"

2. (fruit, be rich in, antioxidant)
   "Whole grains, fruits, and vegetables are also rich in antioxidants, such as vitamins C and E, and beta-carotene."

3. (green bean, should be alert to, problem)
   "Growers who rotate with soybeans or green beans should be alert to potential problems with fields infested with soybean cyst nematodes."

Figure 1: Examples of textual relations extracted from a corpus of Web pages on nutrition, along with a sentence from which the relation was extracted.

While the first two of these textual relations seem perfectly reasonable, they have limited utility for an automated reasoner. The first two relations have no terms in common and, thus, appear to be totally unrelated strings. A language *understanding* system should be able to reason that oranges are a type of fruit, that vitamin C is a type of antioxidant, and that being "rich in" x can imply containing x. None of this reasoning is possible at the level of text phrases.

While the first two relations have interrelated meaning, and thus corroborate each other, the third relation can be immediately labeled as false by a human reader. Green beans or any other vegetable having cognition runs counter to any other true assertions in a human reader's knowledge, and thus can be assigned low belief.

3.2 Normalizing Argument Noun Phrases

As mentioned above, we normalize the argument noun phrases of a relation by mapping them to WordNet noun synsets (Harabagiu, Miller, & Moldovan 1999). Thus, we are dealing with a word sense disambiguation (WSD) problem, since most terms in WordNet occur in multiple synsets.

For example "orange" could be a fruit, a color, a tree, a pigment, or a river in South Africa. The fruit sense is what we want in the context of example 1 in Figure 1.

We now describe the technique we use for WSD. To disambiguate x in (x, rel, y), we consider all the sentences $S_1, \ldots, S_k$ from which this relation was extracted. Using these, and a window size of 4 words on each side of x, we create a set C of context words for x. Words with similar meaning tend to be distributionally similar, and have similar PMI (pointwise mutual information) values with other words. Thus these occurrences of x will on average have greater similarity with words similar to the correct sense compared to those similar to incorrect senses. For each sense, we compute the average similarity of x with all the WordNet synonyms, siblings and direct hyponyms of that sense. We measure similarity according to the metric proposed in (Dagan, Marcus, & Markovitch 1993). It computes similarity between corresponding vectors of PMI values. The set of words forming the PMI vector space is just the set C and not all the words in the corpus (this is important). We assign the sense for which this average similarity is highest. In the experiments section, we report the results we got with this method.

A formal taxonomy enables a system to make inferences that cannot be made otherwise. If (orange, contain, vitamin C), we can infer from inheritance that (orange, contain, vitamin) and that (orange, contain, antioxidant), since vitamin C is a subclass of vitamin and also a subclass of antioxidant. Similarly, (orange, contain, vitamin C) is a specialization of (citrus fruit, contain, vitamin C).

A general-purpose taxonomy like WordNet is likely to have gaps for some domain concepts – a taxonomy may have "fat" but not its hyponym "unsaturated fat". (Snow, Jurafsky, & Ng 2006) have shown that adding terms to a taxonomy can be done with high precision, given a good classifier for expressions that denote *is-a* relations. Augmenting WordNet on the fly while normalizing relations is in our plan for future work. Proper names are another case where a taxonomy is unlikely to help. There has been considerable research recently in named entity recognition (Collins & Singer 1999; McCallum & Li 2003) and in identity resolution across alternate forms of the same name (Dong, Halevey, & Madhavan 2005; Kanani, McCallum, & Pal 2007). We would also like to leverage these in future work.

3.3 Normalizing Relation Phrases

We first normalize the relation phrase by dropping adverbs and reducing verbs to a morphological base form. Thus, "are also rich in" is normalized to "be rich in". We also learn entailments between relation phrases – (X, be rich in, Y) implies that (X, contains, Y) is also true with high probability, as described in Section 3.5. This allows inheritance between relations that goes beyond what would be provided by a verb taxonomy.

Our future plans are to tie the normalized relation phrases to formal concepts such as *Be-Contained* from the KM component library that was described in Section 2.3. Each of these formal concepts is fully axiomatized to enable a rich set of inferences. KM currently has a few hundred of such

general purpose concepts, and ultimately may need several thousand axiomatized concepts for adequate coverage of free text.

A scalable approach to mapping relation phrases to a set of formal concepts is to associate each formal concept with a small set of the most common terms used to refer to the concept. This is already done for each of the manually engineered KM concepts. The set of relations that map to a given concept is then expanded by learning entailments. If (X, contain, Y) is manually mapped to *Be-Contained*, and (X, be rich in, Y) entails (X, contain, Y), then "be rich in" is implicitly mapped to *Be-Contained* as well.

3.4 Quantification

Human language is typically underspecified about scoping and quantification. Automated reasoning systems that are based on first order logic require a clear resolution to scoping and quantification.

Does example 2 in Figure 1 mean that all instances of fruit are rich in all instances of antioxidants (interpretation A). Or does it mean that at all instances of fruit are rich in at least one instance of antioxidant (B)? Or does it mean that at least one instance of fruit is rich in all antioxidants (C) or that at least one fruit is rich in at least one antioxidant(D)?

A.	$\forall x \in$ fruit $\forall y \in$ antioxidant, rich_in(x, y)
B.	$\forall x \in$ fruit $\exists y \in$ antioxidant, rich_in(x, y)
C.	$\exists x \in$ fruit $\forall y \in$ antioxidant, rich_in(x, y)
D.	$\exists x \in$ fruit $\exists y \in$ antioxidant, rich_in(x, y)

Figure 2: Possible semantics of the textual relation (fruit, rich in, antioxidant).

The most likely interpretation is a probabilistic version of B: most fruits contain at least one antioxidant. Interpretations A and C contradict domain knowledge about nutrition, and interpretation D is a trivially weak version of B. Example 1 with its generic use of "the orange", is an even knottier problem for determining quantification automatically. Even with domain knowledge, determining correct quantification from natural language is a difficult problem.

We choose option B as the default quantification. This enables inference of relations where arg1 is replaced with a hyponym (specialization). If the system believes that (citrus fruit, contain, vitamin C), it can infer with equal certainty that (grapefruit, contain, vitamin C) and (pink grapefruit, contain, vitamin C). Arg2 may similarly be replaced with a hypernym (generalized class). If (citrus fruit, contain, vitamin C) then (citrus fruit, contain, antioxidant) and, somewhat vacuously, (citrus fruit, contain, substance).

3.5 Learning Meta-properties of Relation Phrases

Our goal is to automatically extract relational triples spanning a large set of relation phrases, which precludes manually engineering axioms for these relations. Instead, we can learn *meta-properties* of relation phrases, such as *entailment*, *transitivity*, and *transitivity-through*.

Relation *r* entails another relation *r'* if an instance *(x, r, y)* implies that *(x, r', y)* is also true. For example, in the domain of nutrition, *(x, be rich in, y)* entails *(x, contain, y)*. Transitivity holds for a relation *r* if *(a, r, b)* and *(b, r, c)* imply that *(a, r, c)* is also true. Another meta-property that can be learned is transitive-through. Relation *r'* is transitive through relation *r* if whenever *(a, r, b)* and *(b, r', c)* are true, then so is *(a, r', c)*. For example, if *(tomato, contain, lycopene)* and *(lycopene, help prevent, cancer)*, then *(tomato, help prevent, cancer)*

We now describe how we compute entailments in our implementation. For any relation r, we say the ordered pair (x, y) is an instance of r if (x, r, y) has been extracted from the corpus. Given relations r_1, r_2, let their respective instance sets be I_1, I_2. If $r_1 \Rightarrow r_2$ holds, then we should have $I_1 \subseteq I_2$. Due to data sparseness, we don't expect this containment to actually hold. However, we would expect a measure like $|I_1 \cap I_2|/|I_1|$ to be a good score to give to this entailment. In our experiments, this did not turn out to work well. The distribution we had for instance set sizes for relations was very skewed. It varied from 6 to more than $10,000$. If I_2 is orders of magnitude larger than I_1 then this score can be high just by chance overlap. We need to consider $|I_2|$ too in assigning a score. We experimented with several metrics, and the one that was found to work best was $|I_1 \cap I_2|/|I_1| \log|I_2|$. We chose a threshold 0.01 for this score, and chose all entailments with score greater than this value.

3.6 Assigning Probabilities

A set of relations that have been extracted automatically from text are bound to contain some errors. Probability estimation for extracted relations is an important component of information extraction systems. We consider three types of evidence for such probability estimates.

1. *Language Based Features:* These are features derived from the output of POS taggers and deep parsers on the corpus text. These include the POS tags themselves, their bigrams and trigrams, the number of words separating the argument noun phrases, features derived from paths connecting them in the parse tree, and so on (Kambhatla 2004). State of the art relation extraction systems are mostly based on these features.

2. *Redundancy:* The more frequently an instance has been extracted from a large corpus, the more likely it is to be correct (Downey, Etzioni, & Soderland 2005). When redundancy is present, this has been shown to be a very effective source of evidence. However, an open IE system like TextRunner is bound to give a lot of good extractions with low counts, in addition to good extractions with high counts.

3. *Ontological Corroboration:* This is a new source of evidence that we introduce. It becomes available from our ontologizing the relations. It is derived from ontology-based relationships between the extracted relations themselves. We describe these next.

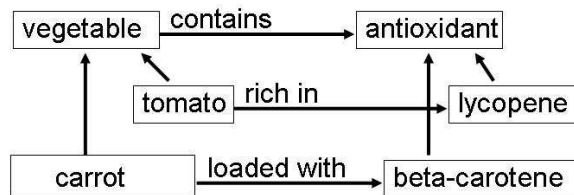We define the following relationships between ontologized relations.

Figure 3: Illustration of relationships between ontologized triples

	Acc	Precision	Recall	F
Naive Bayes – BFS	69%	0.57	0.61	0.59
Naive Bayes – OFS	74%	0.65	0.61	0.63
Log Reg – BFS	72%	0.67	0.46	0.54
Log Reg – OFS	75%	0.70	0.54	0.61
SVM – BFS	73%	0.70	0.43	0.53
SVM – OFS	75%	0.71	0.50	0.59

Table 1: Classifier performance comparison between the two feature sets

1. *Hyponymy:* We define a relation (x', r', y') to be a hyponym of (x, r, y) if x' is a hyponym or synonym of x, y' is a hyponym or synonym of y, and relation phrase r' entails r. For example, in Figure 3, (tomato, rich in, lycopene) can be seen to be a hyponym of (vegetable, contains, antioxidant).

2. *Hypernymy:* We define hypernymy to just be the inverse relationship of hyponymy.

3. *Siblinghood:* We define a relation (x', r', y') to be a sibling of (x, r, y) if x' is a sibling or synonym of x, y' is a sibling or synonym of y, and r and r' are related by an entailment in either direction. For example, in Figure 3, (tomato, rich in, lycopene) can be seen to be a sibling of (carrot, loaded with, beta-carotene).

It is easy to see why, if two relations are related by hyponymy, this provides additional evidence that they are both true. The hyponym relation can be thought of as an "instance" of the hypernym relation. We discussed this earlier when we talked about the semantics of relations between non-leaf nodes in the ontology. We claim that two relations related by siblinghood present a similar scenario. Relation phrases, like predicates in logic, tend to have fixed argument classes. Siblinghood between relations is evidence for particular argument classes for the involved relation phrases.

4. Preliminary Results

Our dataset consisted of $272,960$ relations extracted from a corpus of web pages on health and nutrition. These spanned around $27,000$ WordNet synsets, and $7,000$ distinct relation phrases. Our entailment threshold of 0.01 gave us $20,550$ entailments. Based on a sample of 60 drawn from these, we obtained a precision of 0.67. For word sense disambiguation, we tagged a sample of 125 relations. 70% of these were found to have correct senses assigned to both their arguments. The baseline WSD that always selected the first WordNet synset had accuracy 55%. The baseline was pretty good, presumably because WordNet numbers senses according to decreasing frequency of usage.

We now turn to probability estimation for these relations. We trained classifiers to predict labels (true or false) for them, and assign them truth probabilities. The Baseline Feature Set (BFS) consists of POS tags, bigrams of POS tags, size of the relation phrase, and the extraction frequency of the relation. We used feature selection methods to choose eight features from the large candidate set of POS tags and their bigrams. The Ontology Feature Set (OFS) uses all the features in BFS. In addition, it uses three additional features derived from the kind of ontological evidence that we discussed earlier. For each relation, we use the counts for the number of hyponyms, hypernyms, and siblings that the relation has in the entire set of $272,960$ relations, as features. We expect non-zero values for these features to provide additional evidence that the relation is true.

We first evaluate the accuracy and F-measure performance. We tried three classifiers – Naive Bayes, Logistic Regression, and SVMs with a RBF kernel using (Frank *et al.* 2005). We used a set of 355 tagged relations, and performed 5-fold cross-validation. The set had 127 correct relations, giving an overall precision of 0.36. The results we obtained are shown in Table 1.

Evaluating the quality of the assigned probabilities is of greater interest to us. We will assess this by using the probabilities as a ranking function on the relations, and determining the precision – recall tradeoff that occurs as we descend this ranked list. We determine precision and recall values for the top k relations for a range of k values. We determine precision by tagging a random sample of size 50. For recall, we had a common, fixed set of 100 correct relations, randomly chosen from all relations. The fraction of this set contained in the top k relations gave the corresponding recall value. Figure 4 compares the precision–recall curves given by Logistic Regression and Naive Bayes when using the two feature sets. We see significant improvements when using the OFS feature set compared to BFS. The results seems to indicate that while these ontological features are not active for all correct relations, when they are active they are very accurate evidence for the relation being true.

5. Discussion

We have outlined the basic steps needed to "ontologize" instances of relations of the form (arg1, relation phrase, arg2). This maps the argument strings to a formal taxonomy, adding new concepts to the taxonomy as required, normalizes the relation phrases, and learns meta-relations that form semantic links between relations. This enables a reasoner to draw plausible inferences beyond what is stated directly in the original text.

We have presented experimental results for several of these steps. We have shown that current technology is sufficient to disambiguate noun phrases, to normalize relation phrases, and to learn entailments between relations. We also
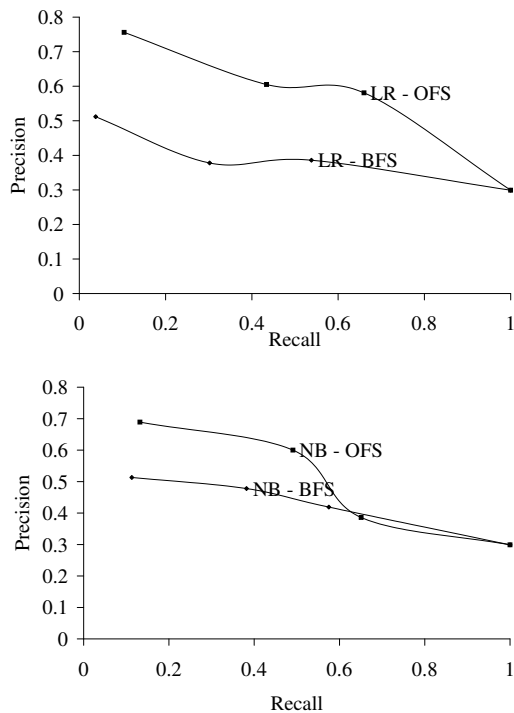
Figure 4: Using ontological feature set (OFS) for a Logistic Regression (LR) or Naive Bayes (NB) classifier gives a large boost to the precision-recall compared with a baseline feature set (BFS).

demonstrated that the resulting ontology can assists in information extraction by validating extractions that are corroborated by the ontology. In preliminary experiments the recall-precision curve was raised considerably when we incorporated ontology-based features in estimating relation probabilities. The high precision end of the curve was raised from precision 0.51 to precision 0.76, and area under the curve was increased by 32%.

Much work remains to move from extraction of textual relations to formal ontologies that support automatic reasoning. While deep reasoning at a human level is beyond our reach in the near future, a first pass at each of the steps outlined in this paper can be accomplished with current techniques in natural language processing and knowledge representation.

References

Banko, M.; Cafarella, M.; Soderland, S.; Broadhead, M.; and Etzioni, O. 2007. Open Information Extraction from the Web. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*.

Barker, K.; Porter, B.; and Clark, P. 2001. A library of generic concepts for composing knowledge bases. In *Proceedings of First International Conference on Knowledge Capture (K-CAP 2001)*.

Chinchor, N. 1998. Overview of MUC-7.

Collins, M., and Singer, Y. 1999. Unsupervised Models for Named Entity Classification. In *Procs. of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 100–111.

Dagan, I.; Marcus, S.; and Markovitch, S. 1993. Contextual word similarity and estimation from sparse data. In *Proceedings of the 31st ACL*, 164–171.

Dong, X.; Halevey, A. Y.; and Madhavan, J. 2005. Reference reconciliation in complex information spaces. In *Procs. of SIGMOD Conference*.

Downey, D.; Etzioni, O.; and Soderland, S. 2005. A Probabilistic Model of Redundancy in Information Extraction. In *Procs. of IJCAI*.

Dyer, M. 1983. *In-Depth Understanding*. MIT Press.

Frank, E.; Hall, M. A.; Holmes, G.; Kirkby, R.; and Pfahringer, B. 2005. Weka - a machine learning workbench for data mining. In *The Data Mining and Knowledge Discovery Handbook*. 1305–1314.

Friedland, N. S.; Allen, P. G.; Matthews, G.; Witbrock, M.; Baxter, D.; Curtis, J.; Shepard, B.; P, M.; Angele, J.; Moench, E.; Oppermann, H.; Wenke, D.; Israel, D.; Chaudhri, V.; Porter, B.; Barker, K.; Fan, J.; Chaw, S. Y.; Yeh, P.; Tecuci, D.; and Clark, P. 2004. Project Halo: towards a digital aristotle. *AI Magazine*.

Harabagiu, S.; Miller, G.; and Moldovan, D. 1999. Wordnet 2 - a morphologically and semantically enhanced resource.

Kambhatla, N. 2004. Combining lexical, syntactic, and semantic features with maximum entropy models for information extraction. In *The Companion Volume to the Proceedings of 42st ACL*, 178–181.

Kanani, P.; McCallum, A.; and Pal, C. 2007. Improving author coreference by resource-bounded information gathering from the Wdb. In *Procs. of IJCAI*.

Lehnert, W. 1988. Knowledge-based natural language understanding. In *Exploring Artificial Intelligence*. Morgan Kaufmann. 83–131.

McCallum, A., and Li, W. 2003. Early results for named entity recognition with conditional random fields. In *Proc. of CoNLL 2003*.

Pennacchiotti, M., and Pantel, P. 2006. A Bootstrapping Algorithm for automatically harvesting semantic relations. In *Proceedings of the 2006 International Conference on Organisational Semiotics*.

Schank, R., and Abelson, R. 1977. *Scripts, Plans, Goals and Understanding*. Lawrence Erlbaum.

Schank, R., and Reisbeck, C. 1982. *Inside Computer Understanding*. Lawrence Erlbaum.

Snow, R.; Jurafsky, D.; and Ng, A. Y. 2006. Semantic taxonomy induction from heterogenous evidence. In *Proceedings of COLING/ACL 2006*.

Sundheim, B. 1991. Overview of the third message understanding evaluation and conference. In *Proceedings of the 3rd Message Understanding Conference*.