

Combining Lexical, Syntactic, and Semantic Features with Maximum Entropy Models for Extracting Relations

Nanda Kambhatla

IBM T. J. Watson Research Center
1101 Kitchawan Road Route 134
Yorktown Heights, NY 10598
nanda@us.ibm.com

Abstract

Extracting semantic relationships between entities is challenging because of a paucity of annotated data and the errors induced by entity detection modules. We employ Maximum Entropy models to combine diverse lexical, syntactic and semantic features derived from the text. Our system obtained competitive results in the Automatic Content Extraction (ACE) evaluation. Here we present our general approach and describe our ACE results.

1 Introduction

Extraction of semantic relationships between entities can be very useful for applications such as biography extraction and question answering, e.g. to answer queries such as “Where is the Taj Mahal?”. Several prior approaches to relation extraction have focused on using syntactic parse trees. For the Template Relations task of MUC-7, BBN researchers (Miller et al., 2000) augmented syntactic parse trees with semantic information corresponding to entities and relations and built generative models for the augmented trees. More recently, (Zelenko et al., 2003) have proposed extracting relations by computing kernel functions between parse trees and (Culotta and Sorensen, 2004) have extended this work to estimate kernel functions between augmented dependency trees.

We build Maximum Entropy models for extracting relations that combine diverse lexical, syntactic and semantic features. Our results indicate that using a variety of information sources can result in improved recall and overall F measure. Our approach can easily scale to include more features from a multitude of sources—e.g. WordNet, gazetteers, output of other semantic taggers etc.—that can be brought to bear on this task. In this paper, we present our general approach, describe the features we currently use and show the results of our participation in the ACE evaluation.

Automatic Content Extraction (ACE, 2004) is an evaluation conducted by NIST to measure Entity

Detection and Tracking (EDT) and relation detection and characterization (RDC). The EDT task entails the detection of mentions of entities and chaining them together by identifying their coreference. In ACE vocabulary, *entities* are objects, *mentions* are references to them, and *relations* are explicitly or implicitly stated relationships among entities. Entities can be of five types: persons, organizations, locations, facilities, and geo-political entities (geographically defined regions that define a political boundary, e.g. countries, cities, etc.). Mentions have *levels*: they can be names, nominal expressions or pronouns.

The RDC task detects implicit and explicit relations¹ between entities identified by the EDT task. Here is an example:

The American Medical Association voted yesterday to install the heir apparent as its president-elect, rejecting a strong, upstart challenge by a District doctor who argued that the nation's largest physicians' group needs stronger ethics and new leadership.

In electing Thomas R. Reardon, an Oregon general practitioner who had been the chairman of its board, ...

In this fragment, all the underlined phrases are mentions referring to the American Medical Association, or to Thomas R. Reardon or the board (an organization) of the American Medical Association. Moreover, there is an explicit *management* relation between **chairman** and **board**, which are references to Thomas R. Reardon and the board of the American Medical Association respectively. Relation extraction is hard, since successful extraction implies correctly detecting both the argument mentions, correctly chaining these mentions to their re-

¹Explicit relations occur in text with explicit evidence suggesting the relationship. Implicit relations need not have explicit supporting evidence in text, though they should be evident from a reading of the document.

Type	Subtype	Count
AT	<i>based-In</i>	496
	<i>located</i>	2879
	<i>residence</i>	395
NEAR	<i>relative-location</i>	288
PART	<i>other</i>	6
	<i>part-Of</i>	1178
ROLE	<i>subsidiary</i>	366
	<i>affiliate-partner</i>	219
	<i>citizen-Of</i>	450
	<i>client</i>	159
	<i>founder</i>	37
	<i>general-staff</i>	1507
	<i>management</i>	1559
	<i>member</i>	1404
	<i>other</i>	174
	<i>owner</i>	274
SOCIAL	<i>associate</i>	119
	<i>grandparent</i>	10
	<i>other-personal</i>	108
	<i>other-professional</i>	415
	<i>other-relative</i>	86
	<i>parent</i>	149
	<i>sibling</i>	23
	<i>spouse</i>	89

Table 1: The list of relation types and subtypes used in the ACE 2003 evaluation.

spective entities, and correctly determining the type of relation that holds between them.

This paper focuses on the relation extraction component of our ACE system. The reader is referred to (Florian et al., 2004; Ittycheriah et al., 2003; Luo et al., 2004) for more details of our mention detection and mention chaining modules. In the next section, we describe our extraction system. We present results in section 3, and we conclude after making some general observations in section 4.

2 Maximum Entropy models for extracting relations

We built Maximum Entropy models for predicting the type of relation (if any) between every pair of mentions within each sentence. We only model *explicit* relations, because of poor inter-annotator agreement in the annotation of implicit relations.

Table 1 lists the types and subtypes of relations for the ACE RDC task, along with their frequency of occurrence in the ACE training data². Note that only 6 of these 24 relation types are symmetric:

²The reader is referred to (Strassel et al., 2003) or LDC’s web site for more details of the data.

“relative-location”, “associate”, “other-relative”, “other-professional”, “sibling”, and “spouse”. We only model the relation subtypes, after making them unique by concatenating the type where appropriate (e.g. “OTHER” became “OTHER-PART” and “OTHER-ROLE”). We explicitly model the argument order of mentions. Thus, when comparing mentions p_1 and p_2 , we distinguish between the case where p_1 -citizen-Of- p_2 and p_2 -citizen-Of- p_1 . We thus model the extraction as a classification problem with 49 classes, two for each relation subtype and a “NONE” class for the case where the two mentions are not related.

For each pair of mentions, we compute several *feature streams* shown below. All the syntactic features are derived from the syntactic parse tree and the dependency tree that we compute using a statistical parser trained on the PennTree Bank using the Maximum Entropy framework (Ratnaparkhi, 1999). The feature streams are:

Words The words of both the mentions and all the words in between.

Entity Type The entity type (one of PERSON, ORGANIZATION, LOCATION, FACILITY, Geo-Political Entity or GPE) of both the mentions.

Mention Level The mention level (one of NAME, NOMINAL, PRONOUN) of both the mentions.

Overlap The number of words (if any) separating the two mentions, the number of other mentions in between, flags indicating whether the two mentions are in the same noun phrase, verb phrase or prepositional phrase.

Dependency The words and part-of-speech and chunk labels of the words on which the mentions are dependent in the dependency tree derived from the syntactic parse tree.

Parse Tree The path of non-terminals (removing duplicates) connecting the two mentions in the parse tree, and the path annotated with head words.

Here is an example. For the sentence fragment,

*been the **chairman** of *its* **board** ...*

the corresponding syntactic parse tree is shown in Figure 1 and the dependency tree is shown in Figure 2. For the pair of mentions **chairman** and **board**, the feature streams are shown below.

Words $chairman_{m11}$, of_{b1} , its_{b2} , $board_{m21}$.

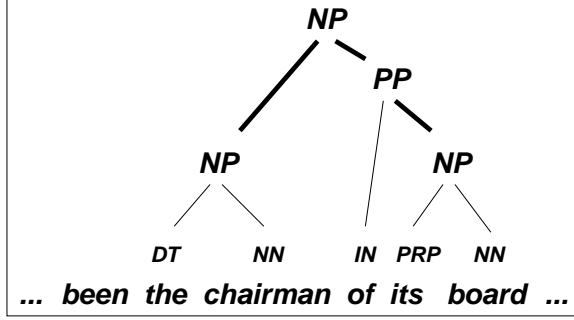


Figure 1: The syntactic parse tree for the fragment “chairman of its board”.

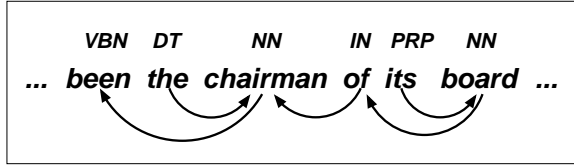


Figure 2: The dependency tree for the fragment “chairman of its board”.

Entity Type $PERSON_{m1}$ (for “chairman”),
 $ORGANIZATION_{m2}$ (for “board”).

Mention Level $NOMINAL_{m1}$,
 $NOMINAL_{m2}$.

Overlap *one-mention-in-between* (the word “its”),
two-words-apart, in-same-noun-phrase.

Dependency $been_{m1-dep}$ (word on which $m1$ is dependent), VBN_{m1-dep} (POS of word on which $m1$ is dependent), VP_{m1-dep} (chunk label of word on which $m1$ is dependent), of_{m2-dep} , IN_{m2-dep} , PP_{m2-dep} , $m1-m2-dependent-in-second-level$ (number of links traversed in dependency tree to go from one mention to another in Figure 2).

Parse Tree $PERSON-NP-PP-ORGANIZATION$,
 $PERSON-NP-PP:of-ORGANIZATION$ (both derived from the path shown in **bold** in Figure 1).

We trained Maximum Entropy models using features derived from the feature streams described above.

3 Experimental results

We divided the ACE training data provided by LDC into separate training and development sets. The training set contained around 300K words, and 9752 instances of relations and the development set contained around 46K words, and 1679 instances of relations.

Features	P	R	F	Value
Words	81.9	17.4	28.6	8.0
+ Entity Type	71.1	27.5	39.6	19.3
+ Mention Level	71.6	28.6	40.9	20.2
+ Overlap	61.4	38.8	47.6	34.7
+ Dependency	63.4	44.3	52.1	40.2
+ Parse Tree	63.5	45.2	52.8	40.9

Table 2: The Precision, Recall, F-measure and the ACE Value on the development set with **true** mentions and entities.

We report results in two ways. To isolate the performance of relation extraction, we measure the performance of relation extraction models on “true” mentions with “true” chaining (i.e. as annotated by LDC annotators). We also measured performance of models run on the deficient output of mention detection and mention chaining modules.

We report both the F-measure³ and the **ACE value** of relation extraction. The ACE value is a NIST metric that assigns 0% value for a system which produces no output and 100% value for a system that extracts all the relations and produces no false alarms. We count the *misses*; the true relations not extracted by the system, and the *false alarms*; the spurious relations extracted by the system, and obtain the ACE value by subtracting from 1.0, the normalized weighted cost of the misses and false alarms. The ACE value counts each relation only once, even if it was expressed many times in a document in different ways. The reader is referred to the ACE web site (ACE, 2004) for more details.

We built several models to compare the relative utility of the feature streams described in the previous section. Table 2 shows the results we obtained when running on “truth” for the development set and Table 3 shows the results we obtained when running on the output of mention detection and mention chaining modules. Note that a model trained with only words as features obtains a very high precision and a very low recall. For example, for the mention pair **his** and **wife** with no words in between, the lexical features together with the fact that there are no words in between is sufficient (though not necessary) to extract the relationship between the two entities. The addition of entity types, mention levels and especially, the word proximity features (“overlap”) boosts the recall at the expense of the very

³The F-measure is the harmonic mean of the *precision*, defined as the percentage of extracted relations that are valid, and the *recall*, defined as the percentage of valid relations that are extracted.

Features	P	R	F	Value
Words	58.4	11.1	18.6	5.9
+ Entity Type	43.6	14.0	21.1	12.5
+ Mention Level	43.6	14.5	21.7	13.4
+ Overlap	35.6	17.6	23.5	21.0
+ Dependency	35.0	19.1	24.7	24.6
+ Parse Tree	35.5	19.8	25.4	25.2

Table 3: The Precision, Recall, F-measure, and ACE Value on the development set with **system output** mentions and entities.

Eval Set	Value (T)	F (T)	Value (S)	F (S)
Feb'02	31.3	52.4	17.3	24.9
Sept'03	39.4	55.2	18.3	23.6

Table 4: The F-measure and ACE Value for the test sets with true (T) and system output (S) mentions and entities.

high precision. Adding the parse tree and dependency tree based features gives us our best result by exploiting the consistent syntactic patterns exhibited between mentions for some relations. Note that the trends of contributions from different feature streams is consistent for the “truth” and system output runs. As expected, the numbers are significantly lower for the system output runs due to errors made by the mention detection and mention chaining modules.

We ran the best model on the official ACE Feb'2002 and ACE Sept'2003 evaluation sets. We obtained competitive results shown in Table 4. The rules of the ACE evaluation prohibit us from disclosing our final ranking and the results of other participants.

4 Discussion

We have presented a statistical approach for extracting relations where we combine diverse lexical, syntactic, and semantic features. We obtained competitive results on the ACE RDC task.

Several previous relation extraction systems have focused almost exclusively on syntactic parse trees. We believe our approach of combining many kinds of evidence can potentially scale better to problems (like ACE), where we have a lot of relation types with relatively small amounts of annotated data. Our system certainly benefits from features derived from parse trees, but it is not inextricably linked to them. Even using very simple lexical features, we obtained high precision extractors that can poten-

tially be used to annotate large amounts of unlabeled data for semi-supervised or unsupervised learning, without having to parse the entire data. We obtained our best results when we combined a variety of features.

Acknowledgements

We thank Salim Roukos for several invaluable suggestions and the entire ACE team at IBM for help with various components, feature suggestions and guidance.

References

- ACE. 2004. The nist ace evaluation website. <http://www.nist.gov/speech/tests/ace/>.
- Aron Culotta and Jeffrey Sorensen. 2004. Dependency tree kernels for relation extraction. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, Barcelona, Spain, July 21–July 26.
- Radu Florian, Hany Hassan, Hongyan Jing, Nanda Kambhatla, Xiaqiang Luo, Nicolas Nicolov, and Salim Roukos. 2004. A statistical model for multilingual entity detection and tracking. In *Proceedings of the Human Language Technologies Conference (HLT-NAACL'04)*, Boston, Mass., May 27 – June 1.
- Abraham Ittycheriah, Lucian Lita, Nanda Kambhatla, Nicolas Nicolov, Salim Roukos, and Margo Stys. 2003. Identifying and tracking entity mentions in a maximum entropy framework. In *Proceedings of the Human Language Technologies Conference (HLT-NAACL'03)*, pages 40–42, Edmonton, Canada, May 27 – June 1.
- Xiaoqiang Luo, Abraham Ittycheriah, Hongyan Jing, Nanda Kambhatla, and Salim Roukos. 2004. A mention-synchronous coreference resolution algorithm based on the bell tree. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, Barcelona, Spain, July 21–July 26.
- Scott Miller, Heidi Fox, Lance Ramshaw, and Ralph Weischedel. 2000. A novel use of statistical parsing to extract information from text. In *1st Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 226–233, Seattle, Washington, April 29–May 4.
- Adwait Ratnaparkhi. 1999. Learning to parse natural language with maximum entropy. *Machine Learning (Special Issue on Natural Language Learning)*, 34(1-3):151–176.
- Stephanie Strassel, Alexis Mitchell, and Shudong Huang. 2003. Multilingual resources for entity detection. In *Proceedings of the ACL 2003 Workshop on Multilingual Resources for Entity Detection*.
- Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. 2003. Kernel methods for relation extraction. *Journal of Machine Learning Research*, 3:1083–1106.