Identifying and Tracking Entity Mentions in a Maximum Entropy Framework

A. Ittycheriah, L. Lita*, N. Kambhatla, N. Nicolov, S. Roukos, M. Stys
I.B.M. T.J.Watson Research Center
P.O.Box 218, Route 134
Yorktown, NY 10598
{abei,nanda,nicolas,roukos,sml}@us.ibm.com, *llita@cs.cmu.edu

Abstract

We present a system for identifying and tracking named, nominal, and pronominal mentions of entities within a text document. Our maximum entropy model for mention detection combines two pre-existing named entity taggers (built to extract different entity categories) and other syntactic and morphological feature streams to achieve competitive performance. We developed a novel maximum entropy model for tracking all mentions of an entity within a document. We participated in the Automatic Content Extraction (ACE) evaluation and performed well. We describe our system and present results of the ACE evaluation.

1 Introduction

We present a system for identifying entities in text. Entities are groups of mentions where mentions are textual references to objects. Mentions have one of five types (person, organization, geo-political entity, location, facility) and can be named (as in standard Named Entity (NE) research), nominal and pronominal - the latter dimension is called the level of a mention. Additionally, mentions can be generic or specific. We break the original task into mention detection (finding all mentions in the text and their type, level and genericity) and mention tracking (combining mentions into groups of references to the same object in the document).

Our work is motivated by the requirements of a NIST-run evaluation on Automatic Content Extraction (ACE, 2002) where the goal is to build systems that detect entities (groups of mentions), relations among them and events in which they participate. Our team took part in the Entity detection track. The ACE task is inherently different and arguably harder than traditional named entity recognition, because of the complexity involved in extracting non-named mentions and chaining them together with named mentions.

We investigate maximum entropy models for both tasks. For mention detection we use a maximum entropy framework for learning semantic trees[1] (corresponding to the mentions) combining as features the output of two pre-existing statistical NE taggers. These taggers have been trained on different corpora using different categories (using 31 and 3 categories respectively).

In Section 2 we describe our mention detection component. Section 3 presents a novel approach for deciding when a mention will or will not be chained with previously created groups of mentions. Section 4 gives the results of our system from the last ACE evaluation.

2 Mention Detection

We use a maximum entropy semantic parser for detecting mentions. The labels of the tree nodes correspond to the combination of type, level and genericity, giving rise to $30 = 5 \times 3 \times 2$ categories for the learning framework.

We had two pre-existing statistical NE taggers (HMM and WINNOW) built with other applications in mind. Our strategy was to combine the hypotheses of the existing NE taggers (using their original models trained on different training data and with different labels) in a MaxEnt framework as well as use additional syntactic and semantic information.[2]

The underlying semantic parser (Ratnaparkhi, 1999) works in three stages: POS tagging, chunking and structure building. During chunking (similar to bottom up parsing) the next level of constutuent structure is discovered. During structure building the rest of the tree is built. All decisions are modeled using Maximum Entropy models. The nature of mention detection puts most burden on the chunking model. The chunking model features include: unigrams of current word (w_0), bigrams in w_{-1}, w_0, w_{+1}, trigrams in $w_{-2}, w_{-1}, w_0, w_{+1}, w_{+2}$, unigrams, bigrams, trigrams on combinations of words and their POS tags in

[1] We refer to this as a semantic parser.

[2] From an engineering perspective the particular way we take diverse information into acount is by using multiple synchronized streams as input to the MaxEnt semantic parser.

$[-1, 0, +1]$ window, the previous label, people and location suffixes.

As additional features we used unigrams, bigrams and trigrams on the output of two models. The first is from an HMM-based system implementing back-off strategies as in BBN's NYMBLE system (Bikel et al., 1999). It uses 31 categories and is trained on a large corpus of 1.5 million words. The system is developed as a component of a question answering system (Ittycheriah, 2001). The second system uses a generalized WINNOW approach (Zhang et al., 2002). It takes additional features: POS tags, lists of known locations, organizations, and person names. It is trained on MUC7 data and a subset of the above corpus for three common classes: person, location and organization.

Additional streams we used were: flags, gazetteers, chunk, left corner and WordNet: flags specify capitalization patterns (Bikel et al., 1999; Borthwick et al., 1998; Zhou & Su, 2002); the gazetteer stream indicates presence of a word in lists; chunk states the label of the mother node of each preterminal in the parse tree[3]; left corner specifies whether the current word is inside an NP and the identity of the leftmost leaf if it has the tag DT (determiner); WordNet specifies whether triggers have fired for the five mention types. Here is an example sentence with its corresponding streams:

Sent	*The*	*senator*	*visited*	*Rome*
Left corner	The	The	x	NP
HMM	x	x	x	LOC-unary
WINNOW	x	x	x	B-LOC

3 Mention Tracking

Mention tracking is the process of recognizing mentions as belonging to an entity. We used a statistical approach for tracking mentions of an entity in a document. Mentions are scored pairwise by a relevancy score and then greedily clustered together into a chain representing a single entity. Resolving pronoun mentions to their antecedents is a classic NLP problem (Hobbs, 1976; Ge, 2000; Mitkov, 2002). A method similar to ours for merging templates in the MUC-6 task has been described by (Kehler, 1997).

This work differs from the previous research in reference resolution in three respects: (1) instead of a restrictive search of antecedents of a given mention, we apply a greedy methodology of symmetric pairwise comparison of all link probabilities (2) we track nominal, pronominal and named mentions of different semantic types, (3) a large corpus of mentions has enabled us to produce a trainable system for mention tracking.

Our approach is based on two elements (1) the relevancy model introduced in (Ittycheriah, 2001) for question answering and (2) a greedy pairwise linking strategy. In the current application of the model, we seek to link the current mention to an entity, $\hat{e}$, which satisfies,

$$\hat{e} = \arg\max_{e_j} p(l|m_i, e_j)|_{l=\text{linked}}$$

where the binary-valued l is either 'linked' or '¬linked'. The algorithm operates on the mentions in *document order* and from the view of each mention there are:

- partially formed clusters to the left, $\mathcal{L}$
- free, unlabeled mentions to the right, $\mathcal{R}$

The algorithm[4] for linking the current mention, m_c is as follows:

Greedy-Chain($\mathcal{L}$, $\mathcal{R}$, m_c)

for m_i **in** $\mathcal{L}$
  if $p(l = \text{linked}|m_i, \textbf{null}) < \text{thresh}_{\text{SingleMention}}$
    Add(m_i, $\bar{\mathcal{L}}$)
for m_i **in** $\bar{\mathcal{L}}$
  Rank(m_i, $\bar{\mathcal{L}}'$)
for m_i **in** $\bar{\mathcal{L}}'$
  if $p(l = \text{linked}|m_c, m_i) > \text{thresh[type]}$
    return Merge(m_c, m_i)
for m_i in $\mathcal{R}$
  if $p(l = \text{linked}|m_c, m_i) > \text{thresh[type]}$
    return DiscourseNew(m_c)
return SingleMention(m_c)

Separate thresholds were established for name, nominal, and pronoun merging, as well as the number of entities considered on the left and the number of mentions to the right.

The model is built on binary-valued features, which are defined as functions of the form $f(\text{link decision}, m_i, m_j)$. The features in our model can be grouped into proxies relying on *similarity* (such as exact and partial matches, overlapping word tokens between mention heads), *distance* measures (in terms of the word and sentence number between the two mentions, and string edit distance), *text location* (quantized sentence number containing a mention), *length* (e.g. number of words within a mention head), *frequency* counts (number of times a mention head occurred within a given document) as well as *syntactic* (e.g. appositive) and *semantic* features (WordNet, semantic entity type, definiteness proxies). A detailed description of the algorithm along with incremental results with different features are presented in (Ittycheriah-Stys, 2003).

4 Results

In this section we present results of our participation in the September 2002 NIST Automatic Content Extraction

[3] We use a statistical parser trained on the Penn Tree Bank.

[4] $\bar{\mathcal{L}}$ is a set of clusters to the left where individual mentions are potential link candidates, $\bar{\mathcal{L}}'$ is a set of clusters to the left where mentions have been ranked by their type, and thresh[type] are thresholds specific to the mention type.

Experiment	Mention Detection F-measure	ACE metric % value
All streams	74.0%	60.1%
w/o HMM	65.3%	50.9%
w/o WINNOW	66.3%	45.6%
w/o LC	66.4%	55.0%
w/o HMM-WINNOW-LC	50.1%	34.7%

Table 1: Mention detection F-measure and entity detection ACE value for different models on the non-degraded text sources of the ACE September 2002 evaluation data set.

evaluation (ACE, 2002). The evaluation measured the performance of systems on entity and relation extraction from newspaper and news wire articles, and broadcast news segments.

We report the F-measure of mention detection and a NIST-defined value metric for entity detection (ACE, 2002) which computes a weighted cost of the misses, false alarms and errors. The cost is normalized and subtracted from 1 to arrive at a normalized "value", with 0 corresponding to no output and 1 corresponding to perfect entity detection. We present results only of our site's participation as per NIST guidelines for the evaluation.

Our training set (provided by NIST) comprised of 417 documents, 191,501 words, 30,492 mentions and 12,630 entities and the evaluation set contained 186 documents, 104,877 words, 10,665 mentions and 4396 entities including ASR and OCR versions of broadcast news and newswire documents. We report results only on the original (not degraded) text documents.

Table 1 shows F-measure and ACE value for our submission system ("All streams"). We also show results with four other mention detection models trained without the HMM stream ("w/o HMM"), the Winnow stream ("w/o WINNOW"), the left corner of NPs ("w/o LC"), and without the HMM, WINNOW and LC streams ("w/o HMM-WINNOW-LC"). For all experiments, we used the same mention tracking model described in Section 3.

We achieved competitive scores (both F-measure and ACE value) for this task. As indicated by the results in Table 1, we were able to obtain a higher overall performance by using all streams. The Winnow NE tagger is very good at detecting person names, which the ACE value metric weights highly. This may account for the relatively sharp decrease in ACE value for the model without the Winnow stream compared to the drop in F-measure, which does not assign weights to categories. Our results suggest that our model was able to use the complementary information provided by the different streams. In particular, the Named Entity extractions of the two pre-existing NE taggers were complimentary and helped the overall system.

5 Conclusions

We have presented a system for identifying named, nominal, and pronominal mentions of entities in text and tracking them within documents. We participated in the NIST Automatic Content Extraction evaluation and performed well.

For mention detection, we pulled together two existing named entity taggers trained with different categories and combined them with other syntactic and lexical sources of information using a maximum entropy framework for building semantic trees. Combining the complementary information provided by the pre-existing taggers helped us rapidly achieve a high F-measure.

For mention tracking, we proposed a novel statistical technique for tracking named, nominal and pronominal mentions of an entity within a document. Using a unified trainable approach helped us perfom well in the evaluation.

Ongoing work includes improving the mention detection and mention tracking by adding morphological, syntactic (derived from parse trees) and semantic (e.g. WordNet) information streams, and extracting relations between the detected entities using statistical models.

References

ACE. 2002. The ACE Evaluation Plan. `www.nist.gov/speech/tests/ace/index.htm`

D. Bikel, R. Schwartz & R.M. Weischedel. 1999. An Algorithm that Learns What's in a Name. *Machine Learning*, 34(1-3).

A. Borthwick, J. Sterling, E. Agichtein & R. Grishman 1998. Exploiting Diverse Knowledge Sources via Max-Entropy in NE Recognition *6th Workshop on Very Large Corpora*, Montreal.

N. Ge. 2000. *An Approach to Anaphora Resolution.* PhD Thesis, Dept. of Computer Science, Brown University.

J. Hobbs. 1976. Pronoun resolution. *Computer Science Dept., City College, CUNY, Technical Report TR76-1.*

A. Ittycheriah. 2001. *Trainable Question Answering Systems.* PhD Thesis, Dept. of Electrical and Computer Engineering, Rutgers - The State Univ. of New Jersey.

A. Ittycheriah & M. Stys. 2003. *A Greedy Algorithm for Mention Tracking.* Submitted to ACL'2003.

A. Kehler. 1997. Probabilistic Coreference in Information Extraction. *EMNLP-2*, 163-173.

R. Mitkov. 2002. *Anaphora Resolution.* Pearson, London.

A. Ratnaparkhi. 1999. Learning to Parse Natural Language with Maximum Entropy. *Machine Learning*, 34(1-3).

T. Zhang, F. Damerau & D.E. Johnson. 2002. Text Chunking based on a Generalization of Winnow. *Journal of Machine Learning Research*, 2:615–637.

G. Zhou & J. Su. 2002. Named Entity Recognition using HMM Chunk Tagger. *ACL'02*, 473-480. Philadelphia.