RC22998 (W0311-125) November 14, 2003
Computer Science

IBM Research Report

Tracking Mentions to Put Them in Chains

Abraham Ittycheriah, Malgorzata Stys
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598

Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

Tracking Mentions To Put Them In Chains

Abraham Ittycheriah, Malgorzata Stys
1101 Kitchawan Road, Route 134
Yorktown Heights, NY 10598
{abei,margostys}@us.ibm.com

Abstract

We present a novel statistical approach for tracking mentions of an entity in a document. Mentions are scored pairwise by a relevance score and then clustered into chains representing single entities. Our approach handles all mentions of proper names, nominals and pronouns, but in this work we restrict our attention to the five mention types concerning ACE (Automatic Content Extraction). Our results show that this method achieves an ACE value score of 88.8% on true mentions.

1 Introduction

A mention is an expression in language of an entity; the names of the authors above are mentions. Mention tracking is the process of recognizing mentions as belonging to an entity, which can be represented by either *named*, *nominal* or *pronoun* mentions (e.g. chaining the co-referring "George Bush", "the president" and "he" in a hypothetical document). Resolving pronoun mentions to their antecedents is a classic NLP problem Hobbs (1976), Ge (2000), and the state of the art in anaphora resolution is reviewed in Mitkov (2002). In the last decade, we have seen empirical methods utilized to develop robust practical NLP solutions. In the field of reference resolution, researchers have mostly focused on restricted cases such as pronominal resolution Ge (2000) or resolution of definite descriptions Poesio and Vieira (1998). However, these cases constitute only a fraction of all referring expressions or *mentions* in ACE's terminology. A method similar to ours has been described by Kehler (1997) in merging templates in the MUC-6 domain. While template merging is similar to our task, the features incorporated in our model and the clustering algorithm presented below are specific to mention tracking. Also W. Soon and Lim (2001) have developed a machine learning approach for coreference resolution and evaluated on MUC-6 and MUC-7.

This work differs from the previous research in reference resolution in the following respects:

- In order to link two given mentions, we use their pairwise link probability instead of searching for antecedents of the current mention. This addresses the coreference reciprocity of named and nominal mentions, where we do not make apriori assumptions about the position of the best mention to which we link the current candidate.
- We track nominal, pronominal and named mentions of different semantic types simultaneously.
- A large corpus of mentions has enabled us to automatically induce rules and derive weights which allows a trainable system for mention tracking.
- We have developed specific features to address single mention entities.

The data used in this paper was provided by NIST (National Institute of Standards and Technology) for the ACE program participants ACE (2002). The ACE mention types are limited to the following five semantic categories: PERSON, ORGANIZATION, LOCATION, FACILITY and GPE (Geo-political entity). A brief definition of these semantic categories can be found in ACE (2002) and the complete annotation guidelines in LDC (2002). The ACE 2002 Evaluation task was bipartite: (1) detect all mentions and their types (2) track all mentions referring to a particular entity. In this paper, we focus our efforts on the second part of the task.

The algorithm developed here obtained an ACE value of 88.8% (maximum value in the ACE metric is 100) on true mentions (i.e. human annotated) which is 35.6% better than a rule based model which merges only named

Type	Level	Generic	ID	Text
PERSON	NAME	FALSE	E2	Goldwater
PERSON	NOMINAL	TRUE	E59	president
ORG	NAME	FALSE	E3	Republican
PERSON	NOMINAL	FALSE	E2	nominee
PERSON	PRONOUN	FALSE	E2	he
PERSON	NOMINAL	TRUE	E69	extremist

Table 1: Entity Information for the example text

mentions. The algorithm was utilized in (organization removed for blind review) submissions for the September 2002 and the English ACE 2003 evaluation.

2 Mention Tracking

As an example of mention tracking in ACE consider the following text,

> *And yet, when Goldwater ran for president as the Republican nominee in nineteen sixty four, he was regarded as an extremist.*

The entities to be detected are shown in Table 1. In the table, there are four entities displayed: {*Goldwater, nominee, he*}, {*Republican*}, {*president*} *and* {*extremist*}. The latter two are 'generic' by which we mean they do not refer to a specific person and are discarded when computing the ACE metric used in the September 2002 evaluation. Figure 1 shows the above example where the algorithm is about to link the mention 'nominee'. The mention 'Clinton' occurs in a previous sentence, and the first occurrence of 'he' is to the right of the 'nominee' mention and the second occurrence of 'he' is in a subsequent sentence.

The relevancy model indicating the linkage of a pair of mentions is similar to the one presented by Ittycheriah (2001) for question answering. The method we use to identify the entities is basically a clustering strategy: initially, entities are created for each mention, and then 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 'not linked'. The algorithm examines the mentions in *document order* and from the view of each mention (m_c) there are:

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

The algorithm is as follows:

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

For m_i **in** $\mathcal{L}$
  If $p(l|m_i$**, NULL)** $<$ **thresh_single_mention**
    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 $p(l|m_c, m_i) >$ **thresh_merge_**[**type**]
    Merge(m_c, m_i**)**
    Merged = 1
    break

If (!Merged)
  For m_i **in** $\mathcal{R}$
    If $p(l|m_c, m_i) >$ **thresh_merge_**[**type**]
      DiscourseNew(m_c**)**
      discourseNew = 1
      break

If (!discourseNew)
  SingleMention(m_c)
}

The routines 'Merge', 'DiscourseNew' and 'SingleMention' simply mark in a data structure the current status of the mention. The 'Add' routine adds a mention to the list $\bar{\mathcal{L}}$. The 'Rank' routine creates $\bar{\mathcal{L}}'$ in order to improve linking performance: when comparing to entities to the left we make the comparison to only 'canonical' mentions of the entity. To compute these 'canonical' mentions, the mentions of the partially formed clusters to the left are ranked by their level (NAME, NOMINAL, PRONOUN) and mentions of the highest level are selected. Thus, if an entity to the left has a single name mention in the cluster, then the comparison is made to that mention only. If there are more mentions at the same priority level in an entity, the maximum probability among the mentions for linking to the current mention is selected for comparison to the threshold. In the current system, NAME to NAME comparisons are much more accurate than other pairings because they rely primarily

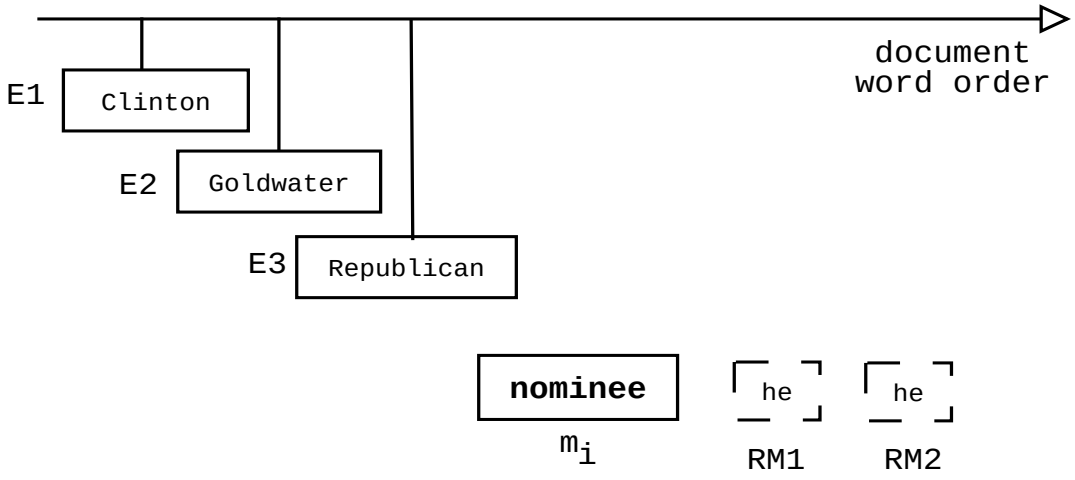

Figure 1: Determining where to link the mention 'nominee'.

on string comparisons; comparing NAME mentions to nominal or pronominal mentions when there are already NAME mentions in an entity can mistakenly attract two distinct names into an entity.

2.1 An Example

In the example of Figure 1, we compute
$p(\text{linked}|\text{'nominee'}, \text{'Clinton'})$
$p(\text{linked}|\text{'nominee'}, \text{'Goldwater'})$
$p(\text{linked}|\text{'nominee'}, \text{'Republican'})$
If the one of probablities to the left exceeds the appropriate threshold for merging, then the decision to merge is taken. If the candidate is not merged, then the probabilities for mentions to the right are computed ($p(\text{linked}|\text{'nominee'}, \text{'he'})$) and if this exceeds the threshold a discourse new entity is created. If neither decision is satisfied, then the candidate is considered as a single-mention entity. Since name and nominal mentions may occur in longer documents outside of the local window of mentions, we do a second pass of clustering by comparing entity clusters and combining them when they satisfy the above thresholds. In the results section, we discuss the performance with and without the second pass algorithm as well as removing the search over mentions to the right of the current mention.

3 Training Data

As a starting point, we assume the existence of the following information for each mention within a document:

- the mention heads (in English usually the last word of a mention, except in the case of proper nouns where the full name is taken LDC (2002)),
- mention types taking the values of NAME[1], NOMINAL[2], or PRONOUN[3] (entity attributes),
- entity type: PERSON , ORGANIZATION, GEOPOLITICAL ENTITY, LOCATION and FACILITY,
- determination of whether a mention is generic[4]

The training data for the algorithm is pairs of mentions together with the annotated decision whether the pair of entities is linked or not. The ACE training corpus has 240K words of data, which we split into 190K words of training and 50K words development data (DEV test). We report current results on the ACE evaluation data of Feb. 2002, Sept. 2002 and Sept. 2003.

4 Maximum Entropy Model

Maximum entropy algorithms have been developed for a number of natural language processing tasks. A thorough description of the MaxEnt algorithm is presented in Berger et al. (1996). Here, we use the MaxEnt algorithm for modelling the distribution $p(l|m_i, m_j)$, where $i \neq j$. The MaxEnt algorithm assigns weights to each feature such that the feature expectation with respect to the model matches its expectation with respect to the empirical distribution $\tilde{p}(l, m_i, m_j)$ while simultaneously minimizing the model divergence from the uniform distribution. We

[1] An entity that is mentioned by name. ACE (2002)
[2] An entity mentioned by a common name, e.g. president.
[3] An entity that is mentioned only by pronoun, e.g. five of them.
[4] We use generic to refer to mentions which we can not link to a physical entity.

will primarily focus our attention on the features used in estimating this distribution. The training corpus of 190K words yields 1.4M instances of mention pairs. The unconditional training data probability for linking any two pairs of mentions is 0.21 and thus if no feature fires for a mention pair, the default decision is to not link them (in our system the threshold for linking names is 0.5). We select all features which occur at least 20 times in the training corpus to build the baseline model.

5 Features

Our binary-valued features are defined as functions of the form $f(\text{link decision}, m_i, m_j)$ and provide the support for estimating the distribution $p(l|m_i, m_j)$. We categorize our features into the following streams:

- **Common Features** {CF} These are features which are functions of the mention pair ($f(\text{link decision}, g(m_i, m_j))$) we are comparing.
  - *Similarity* Exact_Match: Exact match on heads (either true or false); Substr_Match: Substring match of heads; Overlap_Heads: overlap of heads (number of words overlapping between the two mentions).
  - *Capitalization* diffCapitals: number of words with different case; Acronym: whether one mention is an acronym of the other.
  - *Distance* Edit_Distance: calculated in terms of the number of operations (insertion, deletion, substitution) needed to transform the first mention into the second one; Word_Distance and Sentence_Distance: the number of words or sentences between the two mentions.
  - *Syntactic Features* Appositive and a Comma Feature (instantiated to be true if the two mentions are separated by just a comma). For example, in sentence (1) below, the two mentions "girlfriend" and "Sherri Weiss" are in apposition or equivalence relation and the apposition feature fires. In example (2), although our apposition feature mistakenly does not fire for 'Hillary' and 'wife' (because "whose wife" is a relative phrase according to our parser), the comma exclusive rule applies in this instance. The apposition feature is more stringent than the comma feature and the system weights them differently in linking mentions.

(1) When his *girlfriend*, *Sherri Weiss*, asks him to go with her, ...
(2) Blunt and colorful, Goldwater was admired at the end by leaders of both political parties, including President Clinton, whose *wife* , *Hillary*, worked for the Goldwater campaign as a teenager.

- **Mention Specific Features** {M1, M2} These are features inherited from the two mentions being compared, including entity and mention type, the quantized length of the mention head (number of words), as well as pronominal features (i.e. number, gender, and reflexive).
- **WordNet Features** {WN1,WN2} Miller (1990) We extracted the hypernyms of the heads of the two mentions (in order to class the mentions and improve on the data sparsity), as well as indicated whether the WordNet morph algorithm had to be applied to the mention to get the entry (indicating the use of the morph algorithm informs the system whether the mention was inflected).
- **Pattern Features** {PT} These are lexical features which define the context surrounding the two mentions. When the two mentions are sufficiently close (6 words), we annotate the words with '*'. Otherwise, we indicate the words as being to the left of the first mention by x_- and to right by x_+ and additionally indicate the position of the word. An example is given below in the error analysis section.

During feature generation, we considered the following compositions of feature streams:

Model	Feature Templates	Number of features
Base {BM}	{CF}, {CF}x{CF}, {M1}x{M2}	1292
WordNet {WNM}	{WN1}x{WN2}	3988
Cross-stream {CSM}	{M1}x{WN2}, {M2}x{WN1}	6519
Pattern-based {PM}	{PT}x{PT}	22664
Full {FM}	All above	34463

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).

6 Single Mention Modeling

In the training corpus, there are 30492 mentions comprising 12630 entities. Of these, 7855 are single mention entities (62% of all entities). If we consider only those mentions which are not generic, there are 5975 (47% of all entities) such entities. A similar observation has been reported by Poesio and Vieira (1998) for the newspaper domain for discourse new entities. However, single mention entities are not simply discourse new, they are discussed only once in the current document. Thus, single mention entities are a major portion of the entities to be detected and we developed specialized features to model the linking behaviour of single mentions.

The probability model $p(l|m_i, m_j)$ allows a simple extension to handle single mentions. The training data was modified to add a special mention called NULL, to which a mention is linked if the mention is not linked elsewhere in the document. Then, at decode time, we measure explicitly the link probability to the NULL mention and if it exceeds a threshold we form a single mention entity.

In addition, we measure the following two features for single mention events.

- **SameHeadCount** This counts the number of occurrences of this head in the current document. The intuition of this feature is that heads of single mention entities typically occur only once per document. We binned this feature into the following categories: 1,2,3,4,5-10,10plus. When measured on the training corpus, we obtain the distribution shown in Table 2. Here, 'linked' refers to being a single mention (linked to the NULL mention). As an example, if a mention has count 3 then it has only 8% chance of being a single mention. The occurrences of single mentions with repeating heads (especially in the case of 5-10) consist mostly of pronouns such as 'I' and 'We' but also include a fair number of annotation errors.

- **SameWordCount** This feature counts the maximum number of times any word contained in this mention occurs. Often mentions are introduced with a title and susequent mentions only have the title word (for example 'president').

	not linked	linked
1	5774	4639
2	4045	807
3	2201	199
4	1735	132
5-10	4281	157
10plus	2811	41
Total	20487	5975

Table 2: SameHeadCount feature distribution on training data

7 Evaluation Criteria

The ACE metric aligns system outputs to reference entities by maximizing the mention overlap, which allows this metric to measure performance on *true* mentions (mentions from the gold standard) as well as *System* mentions (mentions returned by a mention detector). A set of weights which are designed to reflect the cost of misses and false alarms to a hypothetical application was used ACE (2002) to compute a value of the system for the application. These weights reflect the choice that either missing or generating a false alarm of a name entity is 5 times worse than a nominal entity and 25 times worse than missing a pronominal entity. Since this metric applies on an entity basis and these weights are skewed to named entities, it is possible to have significant errors at the mention level and still obtain a reasonable score.

An alternative metric (the cluster F-measure) is the F-measure of each mention's cluster and averaged for all mentions detected in a document. In this metric attaching a pronoun to a wrong cluster of mentions has the same penalty as with a named mention. The purity of a cluster of mentions can be improved without reflection in the ACE value score since there the emphasis is on names and entities. However, in order to measure the cluster F-measure for system mentions, the entities have to be mapped to their reference entities and this is already done by the ACE metric and thus we present only the ACE value number for system outputs. Our use of the cluster F-measure is primarily as a diagnostic to measure improvements in nominal and pronoun mention tracking and using the ACE metric will allow systems to be compared across sites.

8 Results

Tables 3 and 4 present the results of running the algorithm. Given the relative large number of feature types, we ran the algorithm with different feature subsets, to facilitate the analysis of the contribution of each feature type to the overall system performance. Since the ACE metric is focused on names (see Section 7), we devel-

Test Set	Model	ACE Value	Cluster Metric		
			Precision	Recall	F
DevTest	Rule model {RM}	82.6	82.3	40.5	54.2
	Base model {BM}	84.9	59.0	49.0	53.5
	WordNet model {WNM}	-1.0	47.3	20.9	29.0
	Cross Stream model {CSM}	-8.6	47.5	28.1	35.3
	Pattern based model{PM}	28.2	38.2	24.4	29.8
	Full model {FM}	88.8	60.3	61.6	60.9
Feb 2002	Rule model {RM}	83.5	78.0	47.0	58.6
	Full model {FM}	86.6	61.2	64.9	63.0

Table 3: Results on true mentions.

Model	Dev	Feb 2002	Sept 2002	Sept 2003
Rule model {RM}	63.1	60.0	61.8	48.7
Full model {FM}	69.4	68.6	68.7	71.2

Table 4: ACE Value results on system output.

oped a set of rules for matching mentions based only on spelling is taken as the baseline for our experiments. The mention merging rules include (1) mentions with the same spelling, (2) mentions which are substrings (e.g., "Bill Clinton" and "Clinton" are matched), and (3) mentions which are acronyms of the other. All pronouns and nominals are merged with the last named mention of the same semantic type.

The models tested are presented in Section 5. The BM model performs better in terms of ACE value but slightly worse in terms of the cluster metric compared to the rule based system. The next three models provide complementary information but individually perform much worse than the baseline. Combining the models {BM, WNM, CSM, PM} yields the full model resulting in a 12.4% improvement in the cluster F-measure over the rule based system and a similar increase over the base model. On the blind February evaluation set, we get an increased score in the cluster metric but the ACE value numbers are slightly lower than in the development test.

Combining both mention detection and mention tracking yields the results of Table 4. The performance is considerably lower (from 88.8% drops to 69.4% on the development test, and 86.6% to 68.6% on the Feb. evaluation) and this drop is largely due to the noise introduced by the mention detector. However, we still see a corresponding improvement in applying the statistical model over the rule based system in the real world mentions, although a smaller effect in the February 2002 test shown as is shown in Tables 4. The results indicate a similar improvement in the ACE Sept. 2002 result; however, the performance of the rule model in the ACE Sept. 2003 result has degraded significantly due to higher occurrence of nominal entities. The results verify the robustness of the full model {FM} to different test sets.

Additionaly we computed the scores for models based on specific subsets of features. The results show that the strongest features yielding most improvement included string matches (ACE value of 83.3%) and capitalization features (ACE value of 82.5%). The other features are not significant individually, but they contribute to the overall system.

In order to measure the contribution of different parts of the algorithm, we ran an experiment that removed the second pass algorithm from the full model (which obtained ACE value 88.8%). The resulting ACE value was 88.1%. Searching over the right mentions starts many entities which need a second pass of clustering to remove them.

9 Error Analysis

The errors the system makes can be attributed to

- insufficient features which go beyond surface forms
- annotation issues

We have induced a large number of features (34K) automatically from the training data, but the system still lacks features such as the gender information for proper names. As an example to show both annotation problems and problems the system has, consider sentence (2) from Section 5, where we are trying to link 'teenager' to 'Hillary'. The system incorrectly links 'teenager' to 'Goldwater'. Note that in this instance, 'Goldwater' is a denominal modifier and is marked as a PERSON named mention in contrast to the ACE annotation guidelines LDC (2002). The 'campaign' is considered as a nominal ORG in many other instances in the training data, but in this instance

Feature Stream	Value
Common Features	appositive_n comex_n sentDist_0 editDist_far wordDist_5_10
Mention 1 Features	PERSON NAME lengthHd1_1 10plus_sentence_number_m1
Mention 2 Features	PERSON NOMINAL lengthHd2_1
WordNet 1 Features	Hillary WN_person_1 WN_being_1 wnwm_1_1
WordNet 2 Features	teenager WN_person_2 WN_being_2 wnwm_2_1
Pattern Features	whose_x- wife_x- ,_x- <men1> ,_x+ worked_x+ for_x+ campaign_y- as_y- a_y- <men2> ._y+ That_y+ he_y+

Table 5: Features relating 'Hillary' and 'teenager'.

Feature Stream	Value
Common Features	appositive_n comex_n sentDist_0 editDist_far wordDist_3
Mention 1	PERSON NAME lenHd1_1 10plus_sentence_number_m1
Mention 2	PERSON NOMINAL lenHd2_1
WordNet 1	Goldwater NONE_1
WordNet 2	teenager WN_person_2 WN_being_2 wnwm_2_1
Pattern	<men1> campaign* as* a* <men2>

Table 6: Features relating 'Goldwater' and 'teenager'.

has not been marked by the annotators. The features relating 'Hillary' to 'teenager' are shown in Table 5 and from 'Goldwater' to 'teenager' are shown in Table 6. The probability of linking to 'Hillary' is 0.147, whereas the probability of linking to 'Goldwater' is 0.233. The difference comes from the proximity feature wordDist_3 versus wordDist_5_10. In both tables below, the symbols represent individual features and their values.[5] An additional parse-dependent feature (such as head identification for the mentions) might allow the system to correct this link since 'teenager' and 'campaign' are of different semantic types.

10 Related Work

Coreference resolution continues to be the subject of extensive NLP research. Until recently, most papers reporting quantitative evaluation results have not relied on machine learning from annotated corpora (e.g. Baldwin (1997), Kameyama (1997), Lappin and Leass (1994), Mitkov (1998)) and the vast majority deals with pronominal resolution in English. In order to find some common ground for comparison to our work, we focus on probabilistic and machine-learning approaches reporting quantitative results. The probabilistic method of Ge (2000) used a statistical framework for the resolution of third person anaphoric pronouns.

[5]Legend: comex: comma and only comma between the two related mentions; sentDist: sentence distance; editDist: distance or measure of string similarity; lengthHd1/lengthHd2: lengths of heads in terms of words; wnwm: refers to whether the item was found in WordNet directly or had to be morphed in order to get the WordNet entry

Aone and Bennett (1995), McCarthy and Lehnert (1995), Vieira and Poesio (2000), and W. Soon and Lim (2001) are among the proponents of machine learning methods for reference resolution using decision tree systems to classify coreferential noun phrases. Aone and Bennett (1995)'s machine learning resolver deals solely with Japanese texts and the evaluation focused on noun phrases denoting organizations yielding an F-measure of 77.27%. McCarthy and Lehnert (1995)'s RESOLVE system carried out on the MUC-5 English Joint Venture corpus achieved an F-measure of 85.8%. However, the genre was restricted and the task involved specific types of noun phrases, i.e. organizations and business entities. W. Soon and Lim (2001) evaluated their coreference resolution system in fully automatic mode against the background of pre-processing errors. The evaluation resulted in balanced F-measures of 60.4% and 62.6% for MUC-7 and MUC-6, respectively.

Vieira and Poesio (2000) initially classified the definite descriptions into direct anaphors (the definite description and its antecedent share the head noun), bridging descriptions (the antecedent denotes the same entity but is represented by a different head noun) and discourse new (first-mentioned descriptions denoting objects not related to entities already introduced in discourse). Their system scored 62% recall and 83% precision for direct anaphora and a recall of 69% and precision of 72% for discourse new descriptions. The resolution of bridging descriptions proved to be much more difficult, because world knowledge became necessary. The system was based on manually developed decision trees.

Cardie and Wagstaff (1999) treat reference resolution as a clustering task and apply an unsupervised algorithm which yielded an F-measure of 53.6% on formal evaluation of MUC-6. Kehler's FASTUS system Kehler (1997) uses maximum entropy modeling to assign probability distribution to alternative sets of coreference relationships among noun phrase entity templates, but measures performance in terms of cross-entropies.

11 Conclusion

We presented an algorithm that uses a mention pair probability model for mention tracking. We ran this algorithm in the ACE evaluation of Sept. 2002 and 2003 and report results here on both a development set drawn from the Sept. 2002 training data and the Feb. 2002 evaluation data. The features in our system are automatically selected and the weights associated with each feature are computed from the training corpus. Relatively good performance is easily achieved using either rules (ACE Value 82.6%) or a statistical model incorporating distance and string submatch features (84.9%). Using features that go beyond the surface form (e.g., WordNet features) yields an increased performance of 88.8%.

References

ACE. 2002. Ace - edt phase 2 + rdc documentation. *www.nist.gov/speech/tests/ace/index.htm*.

C. Aone and S. Bennett. 1995. Evaluating automated and manual acquisiion of anaphora resolution strategies. In *Proceedings of the 33rd Annual Meeting of the ACL*, pages 122–129, June.

B. Baldwin. 1997. Cogniac: High precision coreference with limited knowledge and linguistic resources. In *Proceedings of the ACL/EACL Workshop on Operational Factors in Practical, Robust Anaphora Resolution*, pages 38–45.

Adam L. Berger, Vincent Della Pietra, and Stephen Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71.

C. Cardie and K. Wagstaff. 1999. Noun phrase coreference as clustering. In *Proceedings of the 1999 Joint SIGDAT Conference in Empirical Methods in NLP and Very Large Corpora*, pages 82–89.

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

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

Abraham Ittycheriah. 2001. *Trainable Question Answering Systems*. PhD Thesis, Department of Electrical and Computer Engineering, Rutgers - The State University of New Jersey.

M. Kameyama. 1997. Recognizing referential links: An information extraction perspective. In *Proceedings of the ACL/EACL Workshop on Operational Factors in Practical, Robust Anaphora Resolution*, pages 46–53.

Andy Kehler. 1997. Probabilistic coreference in information extraction. *EMNLP-2*, pages 163–173.

S. Lappin and H. Leass. 1994. An algorithm for pronominal anaphora resolution. *Computational Linguistics*, 20.

LDC. 2002. Automatic content extraction - phase 2 - annotation. *www.ldc.upenn.edu/Projects/ACE/PHASE2/Annotation*.

J. McCarthy and W. Lehnert. 1995. Using decision trees for coreference resolution. In *Proceedings of the 14th International Conference on Artificial Intelligence*, pages 1050–1055, June.

G. Miller. 1990. Wordnet: An on-line lexical database. *International Journal of Lexicography*, 3(4):235–244.

R. Mitkov. 1998. Robust pronoun resolution with limited knowledge. *Proceedings of the 17th International Conference on Computational Linguistics*, pages 869–875.

Ruslan Mitkov. 2002. *Anaphora Resolution*. Pearson Education Ltd., London.

M. Poesio and R. Vieira. 1998. A corpus-based investigation of definite description use. *Computational Linguistics*, 24.

R. Vieira and M. Poesio. 2000. An empirically-based system for processing definite descriptions. *Computational Linguistics*, 26.

T. Ng W. Soon and C. Lim. 2001. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics*, 27:521–544.