A Clustering Approach for Unsupervised Chinese Coreference Resolution

Chi-shing Wang **Grace NGAI**

Department of Computing

Hong Kong Polytechnic University

Kowloon, HONG KONG

{cscswang, csgngai}@comp.polyu.edu.hk

Abstract

Coreference resolution is the process of identifying expressions that refer to the same entity. This paper presents a clustering algorithm for unsupervised Chinese coreference resolution. We investigate why Chinese coreference is hard and demonstrate that techniques used in coreference resolution for English can be extended to Chinese. The proposed system exploits clustering as it has advantages over traditional classification methods, such as the fact that no training data is required and it is easily extended to accommodate additional features. We conduct a set of experiments to investigate how noun phrase identification and feature selection can contribute to coreference resolution performance. Our system is evaluated on an annotated version of the TDT3 corpus using the MUC-7 scorer, and obtains comparable performance. We believe that this is the first attempt at an unsupervised approach to Chinese noun phrase coreference resolution.

1 INTRODUCTION

Noun phrase coreference resolution is the process of detecting noun phrases (NPs) in a document and determining whether the NPs refer to the same *entity*, where an entity is defined as “a construct that represents an abstract identity”. The NPs that refer to the entity are known as *mentions*. Mentions can be antecedents or anaphors. An anaphor is an expression that refers back to a previous expression in a discourse. In Figure 1, 克林頓總統 (President Clinton) refers to 克林頓 (Clinton) and is described as an anaphoric reference to克林頓 (Clinton). 克林頓總統 (President Clinton) is described as the antecedent of 他 (he). 克林頓 (Clinton), 克林頓總統 (President Clinton) and the second 他 (he) are all mentions of the same entity that refers to former U.S. president Bill Clinton.

[克林頓$_1$]説，華盛頓將逐步落實對[韓國$_2$]的經濟援助。[金大中$_3$]對[克林頓$_1$]的講話報以掌聲。[他$_3$]説：「[克林頓總統$_1$]在會談中重申，[他$_1$]堅定地支持[韓國$_2$]擺脱經濟危機。」

[Clinton$_1$] said that Washington would progressively follow through on economic aid to [Korea$_2$]. [Kim Dae-Jung$_3$] applauded [Clinton$_1$]’s speech. [He$_1$] said, “[President Clinton$_1$] reiterated in the talks that [he$_1$] would provide solid support for [Korea$_2$] to shake off the economic crisis.

Figure 1: An excerpt from the text, with coreferring noun phrases annotated. *English translation in italics.*

NP coreference resolution is an important subtask in natural language processing (NLP) applications such as text summarization, information extraction, data mining and question answering. This task has attracted much attention in recent years (Cardie and Wagstaff, 1999; Harabagiu et al., 2001; Soon et al., 2001; Ng and Cardie, 2002; Yang et al., 2004; Florian et al., 2004; Zhou et al., 2005), and has been included as a subtask in the MUC (Message Understanding Conferences) and ACE (Automatic Content Extraction) competitions.

Coreference resolution is a difficult task for various reasons. Firstly, a list of features can play a role to support coreference resolution such as

Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing, pages 40–47,
Sydney, July 2006. ©2006 Association for Computational Linguistics

gender agreement, number agreement, head noun matches, semantic class, positional information, contextual information, appositive, abbreviation etc. Ng and Cardie (2002) found 53 features which are useful for this problem. However, no single feature is completely reliable since there are always exceptions: e.g. the number agreement test returns false when **這個部隊** (*this army*, singular) is matched against **眾隊員** (*army members*, plural), despite the two phrases being coreferential. Secondly, identifying features automatically and accurately is hard. Features such as semantic class come from named entity recognition (NER) systems and ontologies and gazetteers, but they are not always accurate, especially where new terms are concerned. Thirdly, coreference resolution subsumes the pronoun resolution problem, which is already difficult since pronouns carry limited lexical and semantic information.

In addition to the aforementioned, Chinese coreference resolution is also made more difficult due to the lack of morphological and orthographic clues. Chinese words contain less exterior information than words in many Indoeuropean languages. For example, in English, number agreement can be detected through word inflections and part-of-speech (POS) tags, but there are no simple rules in Chinese to distinguish whether a word is singular or plural. Proper name and abbreviations are identified by capitalization in English, but Chinese does not use capitalization. Moreover, written Chinese does not have word boundaries, so word segmentation is a crucial problem, as we cannot get the true meaning of the sentence based on characters alone. A simple sentence can be segmented in several different ways to get different meanings. This characteristic affects the performance of all parts and leads to irrecoverable errors. In addition, there are very few Chinese coreference data sets available for research purposes (none of them freely available) and as a result, no easily obtainable benchmarking dataset for training and measuring performance. Building a reasonably large coreference corpus is a labor-consuming task.

To our knowledge, there have only been two Chinese coreference systems in previously published work: Florian et al. (2004), which presents a statistical framework and reports experiment results on Chinese texts; and Zhou et al. (2005), which proposed a unified transformation based learning framework for Chinese entity detection and tracking. It consists of two models: the detection model locates possibly coreferring NPs and the tracking model links the coreference relations.

This paper presents research performed on Chinese noun phrase coreference resolution. Since there are no freely available Chinese coreference resources, we used an unsupervised method that partially borrows from Cardie and Wagstaff's (1999) clustering-based technique, with features that are specially designed for Chinese. In addition, we perform and present the results of experiments designed to investigate the contribution of each feature.

2 Experiment Setup

Identifying coreferent NPs in an unannotated document actually involves two tasks: mention detection, which identifies the anaphors and antecedents in a document, followed by noun phrase coreference resolution. In order to reduce the complexity of the final system, we follow the usual approach in handling these two phases separately.

2.1 Corpus

Even though we are using an unsupervised approach, a gold standard corpus is still needed for experiment evaluation. Since we did not have access to the ACE multilingual entity tracking corpus, we created our own corpus by selecting 30 documents from the TDT3 Chinese corpus. This resulted in a corpus of approximately 36K Chinese characters, about the same size as the MUC dryrun test sets. We then had our corpus annotated by a native Chinese speaker following the MUC-7 (Hirschman and Chinchor, 1997) and ACE Chinese entity guidelines (LDC, 2004) by picking out noun phrase mentions corresponding to one of the following nine types of entities: Person, Organization, Location, Geo-Political Entity (GPE), Facility, Vehicle, Weapon, Date and Money, and for each pair of mentions, deciding whether they refer to the same entity following MUC-7 definitions. According to the guidelines, each mention participates in exactly one entity, and all mentions in the same entity are coreferent. The NPs that are marked include proper nouns, nominal nouns and pronouns and the entity types are a superset of those used in the MUC and ACE competitions. The resulting corpus includes 1640 mentions, referring to 410 entities.

Once our corpus had been determined, the first step was to determine the possible mentions in a plain text. We first used a dictionary-based word segmentation system (Lancashire, 2005) to segment the Chinese characters into words. The segmented words are then labeled with POS tags by a statistical POS tagging system (Fung et al., 2004).

3 Mention Detection

After the corpus has been preprocessed, mention detection involves the identification of NPs in the corpus that refer to some entity. Most of these NPs correspond to non-recursive NPs, which makes this task simpler as most syntactic parsers identify NPs as part of the parsing process. This approach, however, suffers from two problems: firstly, the parser itself is unlikely to be 100% accurate; and secondly, the boundaries of the NPs identified by the parser may not correspond exactly with those of the entities identified by the human annotator.

Another approach is simply to use heuristics based on the POS tag sequence to identify potential NPs of interest. The advantage of this method is that the NPs thus extracted should be closer to the human-annotated entities since the heuristics will be constructed specifically for this task.

To investigate the effect of different approaches on the result of the coreference resolution, we applied both methods separately to our corpus. The corpus was parsed with a state-of-the-art multilingual statistical parser (Bikel 2004), which is trained on the Chinese Penn Treebank. After parsing, we extracted all non-recursive NP chunks tagged by the parser as possible mentions.

For the heuristic-based approach, we applied a few simple heuristics, which had been previously developed during unrelated work for English named-entity resolution (i.e. they were not written with foreknowledge of the gold standard entities) and which are based on the part-of-speech tags of the words. Some examples of our heuristics were to look for pronouns, or to extract all noun sequences, or sequences of determiners followed by adjectives and nouns.

Table 1 shows the performance of the parsing-based approach versus the heuristic-based approach. The parser-based approach suffers mainly because the NPs that it extracts tend to be on the long side, resulting in recall errors when the boundaries of the parser-identified NPs mismatch with the human-annotated entities. In addition, the parser also tends to extract more NPs than needed, which results in a hit to precision.

	Recall	Precision	F-Measure
Heuristics	83	59.3	69.2
Parser-Based	62.7	28.7	39.4

Table 1: Mention Detection Results

4 Coreference Resolution

The final step after the mention detection phase is to determine which of the extracted phrases refer to the same entity, or are coreferent.

The small size of our corpus made it quite obvious that we would not be able to perform supervised learning, as there would not be enough data for generalization purposes. Therefore we chose to use an unsupervised clustering approach for this step. Clustering is a natural choice as it partitions the data into groups; used on coreference resolution, we expect to gather coreferrent NPs into the same cluster. Furthermore, most clustering methods can easily incorporate both context-dependent and independent constraints into their features.

4.1 Features

Our features use both lexical and syntactic information designed to capture both the content of the phrase and its role within the sentence. With the exception of the last three features, which are defined with respect to a noun phrase pair, all our features describe various aspects of a single noun phrase:

Lexical String – This is just simply the string of words in the phrase.

Head Noun – The head noun in a phrase is the noun that is not a modifier for another noun.

Sentence Position – This measures the position of the phrase within the document.

Gender – For each phrase, we use a gazetteer to assign it a gender. The possible values are male (e.g. **先生**, *mister*), female (e.g. **小姐**, *miss*), either (e.g. **團長**, *leader*) and neither (e.g. **工廠**, *factory*).

Number – A phrase can be either singular (e.g. **一隻貓**, *one cat*), plural (e.g. **兩隻狗**, *two dogs*), either (e.g. **產品**, *product*) or neither (e.g. **安全**, *safety*).

Semantic Class – To give the system more information on each phrase, we generated our own gazetteer from a combination of gazetteers compiled from web sources and heuristics. Our gazetteer consists of 4700 entries, each of which is labeled with one of the following semantic classes: person, organization, location, facility, GPE, date, money, vehicle and weapon. Phrases in the corpus that are found in the gazetteer are given the same semantic class label; phrases not in the gazetteer are marked as UNKNOWN.

Proper Name – The part-of-speech tag "NR" and a list of common proper names were used to label each noun phrase as to whether it is a proper name (values: true/false).

Pronoun – Determined by the part-of-speech "PN". Values: true/false.

Demonstrative Noun Phrase – A demonstrative noun phrase is a phrase that consists of a noun phrases preceded by one of the characters [這那該] (*this/that/some*).

Appositive – Two noun phrases are in apposition when the first phrase is headed by a common noun while the second one is a proper name with no space or punctuation between them. e.g. [美國總統][克林頓]上星期到朝鮮訪問。(*[US president] [Clinton] visited Pyongyang last week.*) This differs from English where two nouns are considered to be in apposition when one of them is an anaphor and separated by a comma from the other phrase, which is the most immediate proper name. (e.g. "Bill Gates, the chairman of Microsoft Corp")

Abbreviation – A noun phrase is an abbreviation when it is formed by using part of another noun phrase, e.g. 朝鮮中央通訊社 (*Pyongyang Central Communications Office*) is commonly abbreviated as 朝中社. Since name abbreviations in Chinese are often given in an ad-hoc manner, it would be infeasible to generate a list of nam and abbreviations in advance. We therefore use the following heuristic: given two phrases, we test if one is an abbreviation of another by extracting each successive character from the shorter phrase and testing to see if it is included in the corresponding word from the longer phrase. Intuitively, we know that this is a common way of abbreviating terms; empirically, it usually gives us a correct result.

Edit Distance – Abbreviations and nicknames

Feature *f*	**Function**
Noun Phrase Match	-1 if the string of NP_i matches the string of NP_j; else 0
Head Noun Match	-1 if head noun of NP_i matches the head noun of NP_j; else 0
Sentence Distance	0 if NP_i and NP_j are in the same sentence; For non-pronouns: 1/10 if they are one sentence apart; and so on with maximum value 1; For pronouns: if more than two sentences apart, then 1
Gender Agreement	1 if they do not match in gender; else 0
Number Agreement	1 if they do not match in number; else 0
Semantic Agreement	1 if they do not match in semantic class or unknown; else 0
Proper Name Agreement	1 if both are proper names, but mismatch on every word; else 0
Pronoun Agreement	1 if either NP_i or NP_j is pronoun and mismatch in gender or number; else 0
Demonstrative Noun Phrase	-1 if NP_i is demonstrative and NP_i contains NP_j; else 0
Appositive	-1 if NP_i and NP_j are in an appositive relationship; else 0
Abbreviation	-1 if NP_i and NP_j are in an abbreviative relationship; else 0
Edit Distance	0 if NP_i and NP_j are the same, 1/(length of longer string) if one edit is needed to transform one to another, and so on.

Table 2: Features and functions used in clustering algorithm

are very commonly used in Chinese and even though the previous feature will work on most of them, there are some common exceptions. To make sure that we catch those as well, we introduced a Chinese-specific feature as a further test. Since abbreviations and nicknames are not usually substrings of the original strings, but will still share some common characters, we measure the Levenshtein distance, defined as the number of character insertions, deletions and substitutions, between every potential antecedent-anaphor pair.

4.2 Distance Metric

In order for the clustering algorithm to be able to group instances together by similarity, we need to determine a distance metric between two instances – in our case, two noun phrases. For our system, we borrowed a simple distance metric from Cardie and Wagstaff (1999) that sums up the results of a series of functions over the two phrases:

$$dist(NP_i, NP_j) = \sum_{f \in F} function_f(NP_i, NP_j)$$

Table 2 presents the features and the corresponding functions that were used in our system. Each function calculates a distance between the two phrases that is an indicator of the degree of incompatibility between the two phrases with respect to a particular feature. The NOUN PHRASE, HEAD NOUN, DEMONSTRATIVE, APPOSITIVE and ABBREVIATIVE functions test for compatibility and return a negative value when the two phrases are compatible for that term's feature. The reason for the negative value returned is that if the two phrases match on this particular feature, then it is a strong indicator of coreference. Therefore, we reduce the distance between two phrases, making it more likely that they will be clustered together into the same entity. When there is a mismatch, however, it does not necessarily indicate that the two NPs are non-coreferential, so we leave the distance between the NPs unchanged.

Conversely, there are some features where a mismatch would indicate that the two NPs are absolutely non-compatible and will definitely not refer to the same entity. The DISTANCE, GENDER, NUMBER, SEMANTIC, PROPER NAME, PRONOUN and EDIT_DISTANCE functions return a positive value when the two phrases mismatch on that particular feature. A positive value results in a greater distance between two phrases, which makes it less likely for them to be clustered together.

4.3 Clustering Algorithm

Most of the previous work in clustering-based noun phrase coreference resolution has centered around the use of bottom-up clustering methods, where each noun phrase is initially assigned to a singleton cluster by itself, and clusters which are "close enough" to each other are merged (Cardie & Wagstaff, 1999; Angheluta et al., 2004).

In our system, we use a method called modified k-means clustering (Wilpon & Rabiner 1985), which takes the opposite approach and uses a top-down approach to split clusters, interleaved with a k-means iterative phase. Modified k-means clustering has been successfully applied to speech recognition and it has the advantage of always being able to come to the optimal clustering (i.e. it is not dependent upon the starting state or merging order).

Modified k-means starts off with all the instances in one big cluster. The system then iteratively performs the following steps:

1. For each cluster, find its centroid, defined as the instance which is the closest to all other instances in the same cluster.
2. For each instance:
    a. Calculate its distance to all the centroids.
    b. Find the centroid with the minimum distance, and join its cluster.
3. Iterate 1-2 until instances stop moving between clusters.
4. Find the cluster with the largest intra-cluster distance. (Call this $\text{Cluster}_{\text{max}}$ and its centroid, $\text{Centroid}_{\text{max}}$.) If this distance is smaller than some threshold *r*, stop.
5. From the instances inside $\text{Cluster}_{\text{max}}$, find the pair that are the furthest apart from each other.
    a. Add the pair of instances to the list of centroids and remove $\text{Centroid}_{\text{max}}$ from the list.
    b. Repeat from Step 2.

The algorithm thus alternates traditional k-means clustering with a step that adds new clusters to the pool of existing ones. Used for coreference resolution, it splits up the instances into clusters in which the instances are more similar to each other than to instances in other clusters.

The only thing left to do is to determine a suitable threshold. As functions that check for compatibility return negative values while positive distances indicate incompatibility, a threshold of 0 would separate compatible and incompatible

	Recall	Precision	F-Measure
Gold Standard Entities	78	88.5	82.9
Baseline (Heuristic-based Entities)	80.9	44.1	57.1
Baseline (Noun Phrase Match Only)	50.9	77.2	61.3
Heuristic-Based Entity Recognition	62.9	77.1	69.3
Parsing-Based Entity Recognition	42.5	62.9	50.7

Table 3: Coreference Resolution Performance

elements. However, since the feature extraction will not be totally accurate, (especially for the GENDER and NUMBER features which test for incompatibility) we chose to be more lenient with deciding whether two phrases should be clustered together, and used a threshold of $r = 1$ to allow for possible errors.

5 Evaluation

Evaluation of coreference resolution systems has traditionally been performed with precision and recall. The MUC competition defines recall as follows (Vilain et al., 1995):

$$R = \frac{\sum(|C_i| - |p(C_i)|)}{\sum(|C_i| - 1)}$$

Each C_i is a gold standard cluster (i.e. a set of phrases which we know refer to the same entity), and $p(C_i)$ is the partitioning of C_i by the automatically-generated clusters. For precision, the role of the automatic and gold standard clusters are reversed. Our results were evaluated using the MUC scoring program which reports recall, precision and F-measure, where the F-measure is defined as the harmonic mean of precision and recall:

$$F = \frac{2PR}{P + R}$$

Table 3 presents the results of our coreference resolution system on the outputs of both the parsing-based and heuristic-based entity detection systems, as measured by the MUC-7 scoring program. For the purposes of comparison, we also present results of our clustering algorithm on the gold standard entities. This gives us a sense of the upper bound that we could potentially achieve if we got 100% accuracy on our mention detection phase. An additional baseline is generated by implementing a system that assumes that all phrases refer to the same entity – i.e. it takes all the heuristically-generated phrases and puts them into one big cluster. This gives us an upper bound on the recall of the system. Yet another baseline, to see how easy the task is, is to merge mentions together if the “Noun Phrase Match” function tests true.

From the results, it can be seen that our system achieves a performance gain of over 10 F-Measure points over the simplest baseline, and over 8 F-Measure points over the more sophisticated baseline. Unfortunately, due to corpus differences, we cannot conduct a comparison with results found in previous work.

An interesting observation is the fact that the heuristic-based entity recognizer achieves better performance than the one based on statistical parsing. The parser is trained on the Chinese Penn Treebank, which tends to have relatively longer noun phrases, and as result, the phrases generated by the parser also tend to be on the long side. This causes errors at the entity recognition phase, which results in a performance hit for the overall system.

6 Analysis

One interesting question to ask about the results is the contribution of any given individual feature to the result of the overall system. We have already investigated the effect of entity recognition, and in this section, we take a look at the features for the clustering algorithm. **Error! Reference source not found.** presents the results of a series of experiments in which one feature at a time was removed from the clustering algorithm. The last entry in the table shows the results of the full system; the drop in performance when a feature is removed is indicative of its contribution. Judging from the results, the 3 features that contribute the most to performance are the NOUN PHRASE MATCH, SEMANTIC AGREEMENT and EDIT DISTANCE features. Two out of the three, NOUN PHRASE and EDIT DISTANCE, operate on lexical information. The importance of string matching to coreference resolution is consistent with findings in previous work (Yang et al. 2004), which arrived at the same conclusion for English.

In addition, we note that the two Chinese-specific features that were introduced, ABBREVIATION and EDIT DISTANCE, both contribute significantly (as measured by a student’s t-test) to the performance of the final system.

Removed feature	Recall	Precision	F-measure
Noun Phrase Match	**59.8**	**75.9**	**66.9**
Head Noun Match	60.4	76.2	67.4
Sentence Distance	63.2	73.3	67.8
Gender Agreement	62.9	76.3	68.9
Number Agreement	63.2	75.9	69
Semantic Agreement	**60.5**	**73**	**66.2**
Proper Name Agreement	63	76.2	69
Pronoun Agreement	61.3	76.9	68.2
Demonstrative Noun Phrase	62.2	77.9	69.2
Appositive	60.1	76.9	67.5
Abbreviation	61.6	77	68.4
Edit Distance	**62.4**	**72.8**	**67.2**
None (All Features)	62.9	77.1	69.3

Table 4: Contribution of individual features to overall performance.

Of our features, those that contribute the least to the overall performance are the GENDER, NUMBER and DEMONSTRATIVE NOUN PHRASE features. For DEMONSTRATIVE NOUN PHRASE, the reason is because of data sparsity – there are just simply not enough examples that it would make any significant impact. For the GENDER and NUMBER features, we find that the problem is mostly with errors in feature generation.

To our knowledge, this is the first published result on unsupervised Chinese coreference resolution. Due to differences in data, it is not possible to conduct a comparison of our work with previous results.

7 Related Work

Coreference resolution has attracted much attention in recent years, especially as a result of the MUC and ACE competitions. The approaches taken have exhibited a shift from knowledge-based approaches to learning-based approaches. Many of the learning-based approaches recast coreference resolution as a binary classification task, which, given a pair of NPs, uses a trained classifier to determine whether they are coreferent. Soon et al. (2001) used this approach with a 12-feature decision tree-based classifier and Ng and Cardie (2002) extended this approach with extra machine learning frameworks and a larger set of features. Yang et al. (2004) extended this approach into an NP-cluster based approach, which considers the relationships between phrases and coreferential clusters.

In addition, several unsupervised approaches have been proposed. Cardie and Wagstaff (1999) re-cast the problem as a clustering task which applied a set of incompatibility functions and weights in the distance metric. Bean and Riloff (2004) used information extraction patterns to identify contextual clues that would determine the compatibility between NPs.

All of the previously mentioned work has been for English. There has been relatively little work in Chinese: Florian et al. (2004) provides results using a language-independent framework on the Entity Detection and Tracking task (EDT). They formulate the detection subtask as a classification problem using a Robust Risk Minimization classifier combined with a Maximum Entropy classifier. Their system performs significantly well on English, Chinese and Arabic, however, the system suffers from small amount of training data (90K characters for Chinese, in contrast with 340K words for English). Their system obtained an ACE value of 58.8 on the ACE evaluation data on Chinese. Finally, Zhou et al. (2005) proposed a unified Transformation-Based Learning framework on Chinese EDT. The TBL tracking model looks at pairs of NPs at a time and classifies them as being coreferent or not based on the values of six features. They report an ACE score of 63.3 on their dataset.

8 Conclusions and Future Work

In this paper, we have presented an unsupervised approach to Chinese coreference resolution. Our approach performs resolution by clustering, with the advantage that no annotated training data is needed. We evaluated our approach using a corpus which we developed using standard annotation schemes, and find that our system achieves an error reduction rate of almost 30% over the baseline. We also analyze the performance of our system by investigating the contribution of individual features to our system. The analysis illustrates the contribution of the new language-specific features.

While the results produced by our system are impressive, it should be noted that all our features consider only the mention phrase itself. We consider this to be a rather simplistic and incomplete. In future work, we plan to investigate the use of more sophisticated features, including contextual features, to improve the performance of our system.

References

ANGHELUTA R., JEUNIAUX P., RUDRADEB M., MOENS M.F. 2004. Clustering Algorithms for Noun Phrase Coreference Resolution. *Proceedings of the 7th International Conference on the Statistical Analysis of Textual Data.*

BEAN, D. and RILOFF, E. 2004. Unsupervised learning of contextual role knowledge for coreference resolution. In *Proc. of HLT/NAACL*, pages 297–304.

BIKEL, D. M. 2004. A Distributional Analysis of a Lexicalized Statistical Parsing Model. In *Proceedings of EMNLP*, Barcelona

CARDIE, C. and WAGSTAFF, K. 1999. Noun phrase coreference as clustering. In *Proceedings of the 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 82-89.

FLORIAN, R., HASSAN, H., ITTYCHERIAH, A., JING, H., KAMBHATLA, N., LUO, X., NICOLOV, N., and ROUKOS, S. 2004. Statistical Model for Multilingual Entity Detection and Tracking. In *Proceedings of 2004 annual meeting of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2004).*

FUNG, P., NGAI, G., YANG, Y. S., and CHEN, B.F. 2004. A maximum-entropy Chinese parser augmented by Transformation-Based Learning. *ACM Transactions on Asian Language Information Processing (TALIP)*, 3(2), pp 159-168.

GAO J.F., LI M. and HUANG C.N. 2003. Improved souce-channel model for Chinese wordsegmentation. In *Proc. of ACL2003.*

HARABAGIU, S., BUNESCU, R.,and MAIORANO, S. 2001. Text and Knowledge Mining for Coreference Resolution, in *Proceedings of the 2nd Meeting of the North American Chapter of the Association of Computational Linguistics (NAACL-2001).*

HIRSCHMAN, L. and CHINCHOR, N. 1997. MUC7 Coreference Task Definition, http://www.itl.nist.gov/iaui/894.02/related_projects/muc/proceedings/co_task.html.

LANCASHIRE, D. 2005. Adsotrans Chinese-English annotation. http://www.adsotrans.com/.

LDC. 2004. Chinese Annotation Guidelines for Entity Detection and Tracking. http://www.ldc.upenn.edu/Projects/ACE/Data.

MUC-7. 1998. *Proceedings of the Seventh Message Understanding Conference (MUC-7).* Morgan Kaufmann, San Francisco, CA.

NG V. 2005. Machine learning for coreference resolution: From local classification to global ranking. *In Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL), 2005.*

NG, V. and CARDIE, C. 2002. Improving machine learning approaches to coreference resolution. In *Proceedings of the 40rd Annual Meeting of the Association for Computational Linguistics,* Pages 104-111.

NG V. and CARDIE C. 2003. Bootstrapping Coreference Classifiers with Multiple Machine Learning Algorithms. *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing (EMNLP-2003), Association for Computational Linguistics, 2003.*

SOON, W., NG, H., and LIM, D. 2001. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics*, 27(4):521-544.

VILAIN, M., BURGER, J., ABERDEEN, J., CONNOLLY, D., and HIRSCHMAN, L. 1995. A model-theoretic coreference scoring scheme. In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pages 45–52, San Francisco, CA. Morgan Kaufmann.

WILPON, J., AND RABINER, L. 1985. A modified K-means clustering algorithm for use in isolated word recognition. In *IEEE Transactions on Acoustics, Speech, Signal Processing*. ASSP-33(3), 587-594.

YANG, X., ZHOU, G., SU, J., and TAN, C. L. 2004. An NP-Cluster Based Approach to Coreference Resolution. *Proceedings of the 20th International Conference on Computational Linguistics (COLING2004).*

ZHOU Y., HUANG C., GAO J., WU L. 2005. Transformation Based Chinese Entity Detection and Tracking. *Proceedings of the Second International Joint Conference on Natural Language Processing.*