# A High-Performance Coreference Resolution System using a Constraint-based Multi-Agent Strategy

**ZHOU GuoDong** **SU Jian**
Institute for Infocomm Research
21 Heng Mui Keng Terrace
Singapore 119613
Email: zhougd@i2r.a-star.edu.sg

## Abstract

This paper presents a constraint-based multi-agent strategy to coreference resolution of general noun phrases in unrestricted English text. For a given anaphor and all the preceding referring expressions as the antecedent candidates, a common constraint agent is first presented to filter out invalid antecedent candidates using various kinds of general knowledge. Then, according to the type of the anaphor, a special constraint agent is proposed to filter out more invalid antecedent candidates using constraints which are derived from various kinds of special knowledge. Finally, a simple preference agent is used to choose an antecedent for the anaphor form the remaining antecedent candidates, based on the proximity principle. One interesting observation is that the most recent antecedent of an anaphor in the coreferential chain is sometimes indirectly linked to the anaphor via some other antecedents in the chain. In this case, we find that the most recent antecedent always contains little information to directly determine the coreference relationship with the anaphor. Therefore, for a given anaphor, the corresponding special constraint agent can always safely filter out these less informative antecedent candidates. In this way, rather than finding the most recent antecedent for an anaphor, our system tries to find the most direct and informative antecedent. Evaluation shows that our system achieves Precision / Recall / F-measures of 84.7% / 65.8% / 73.9 and 82.8% / 55.7% / 66.5 on MUC-6 and MUC-7 English coreference tasks respectively. This means that our system achieves significantly better precision rates by about 8 percent over the best-reported systems while keeping recall rates.

## 1 Introduction

Coreference accounts for cohesion in texts. Especially, a coreference denotes an identity of reference and holds between two expressions, which can be named entities, definite noun phrases, pronouns and so on. Coreference resolution is the process of determining whether two referring expressions refer to the same entity in the world. The ability to link referring expressions both within and across the sentence is critical to discourse and language understanding in general. For example, coreference resolution is a key task in natural language interfaces, machine translation, text summarization, information extraction and question answering. In particular, information extraction systems like those built in the DARPA Message Understanding Conferences (MUC) have revealed that coreference resolution is such a crucial component of an information extraction system that a separate coreference task has been defined and evaluated in MUC-6 (1995) and MUC-7 (1998).

There is a long tradition of work on coreference resolution within computational linguistics. Many of the earlier works in coreference resolution heavily exploited domain and linguistic knowledge (Carter 1987; Rich and LuperFoy 1988; Carbonell and Brown 1988). However, the pressing need for the development of robust and inexpensive solutions encouraged the drive toward knowledge-poor strategies (Dagan and Itai 1990; Lappin and Leass 1994; Mitkov 1998; Soon, Ng and Lim 2001; Ng and Cardie 2002), which was further motivated by the emergence of cheaper and more reliable corpus-based NLP tools such as part-of-speech taggers and shallow parsers alongside the increasing availability of corpora and other resources (e.g. ontology).

Approaches to coreference resolution usually rely on a set of factors which include gender and number agreements, c-command constraints, semantic consistency, syntactic parallelism, semantic parallelism, salience, proximity, etc. These factors can be either “constraints” which discard invalid ones from the set of possible candidates (such as gender and number agreements, c-command constraints, semantic consistency), or “preferences” which gives more preference to certain candidates and less to others (such as syntactic parallelism, semantic parallelism, salience, proximity). While a number of approaches use a similar set of factors, the computational strategies (the way antecedents are determined, i.e. the algorithm and formula for assigning antecedents) may differ, i.e. from simple co-occurrence rules (Dagan and Itai 1990) to decision trees (Soon, Ng and Lim 2001; Ng and Cardie 2002) to pattern induced rules (Ng and Cardie 2002) to centering algorithms (Grosz and Sidner 1986; Brennan, Friedman and Pollard 1987; Strube 1998; Tetreault 2001).

This paper proposes a simple constraint-based multi-agent system to coreference resolution of general noun phrases in unrestricted English text. For a given anaphor and all the preceding referring expressions as the antecedent candidates, a common constraint agent is first presented to filter out invalid antecedent candidates using various kinds of general knowledge. Then, according to the type of the anaphor, a special constraint agent is proposed to filter out more invalid antecedent candidates using constraints which are derived from various kinds of special knowledge. Finally, a simple preference agent is used to choose an antecedent for the anaphor form the remaining antecedent candidates, based on the proximity principle. One interesting observation is that the most recent antecedent of an anaphor in the coreferential chain is sometimes indirectly linked to the anaphor via some other antecedents in the chain. In this case, we find that the most recent antecedent always contains little information to directly determine the coreference relationship with the anaphor. Therefore, for a given anaphor, the corresponding special constraint agent can always safely filter out these less informative antecedent candidates. In this way, rather than finding the most recent antecedent for an anaphor, our system tries to find the most direct and informative antecedent.

In this paper, we focus on the task of determining coreference relations as defined in MUC-6 (1995) and MUC-7 (1998). In order to evaluate the performance of our approach on coreference resolution, we utilize the annotated corpus and the scoring programs from MUC-6 and MUC-7. For MUC-6, 30 dry-run documents annotated with coreference information are used as the training data. There are also 30 annotated training documents from MUC-7. The total size of 30 training documents is close 12,400 words for MUC-6 and 19,000 for MUC-7. For testing, we utilize the 30 standard test documents from MUC-6 and the 20 standard test documents from MUC-7.

The layout of this paper is as follows: in Section 2, we briefly describe the preprocessing: determination of referring expressions. In Section 3, we differentiate coreference types and discuss how to restrict possible types of direct and informative antecedent candidates according to anaphor types. In Section 4, we describe the constraint-based multi-agent system. In Section 5, we evaluate the multi-agent algorithm. Finally, we present our conclusions.

## 2 Preprocessing: Determination of Referring Expressions

The prerequisite for automatic coreference resolution is to obtain possible referring expressions in an input document. In our system, the possible referring expressions are determined by a pipeline of NLP components:

- Tokenization and sentence segmentation
- Named entity recognition
- Part-of-speech tagging
- Noun phrase chunking

Among them, named entity recognition, part-of-speech tagging and noun phrase chunking apply the same Hidden Markov Model (HMM) based engine with error-driven learning capability (Zhou and Su 2000). The named entity recognition component (Zhou and Su 2002) recognizes various types of MUC-style named entities, that is, organization, location, person, date, time, money and percentage. The HMM-based noun phrase chunking component (Zhou and Su 2000) determines various noun phrases based on the results of named entity recognition and part-of-speech tagging.

## 3 Coreference Types

Since coreference is a symmetrical and transitive relation, it leads to a simple partitioning of a set of referring expressions and each partition forms a coreference chain. Although any two referring expressions in the coreference chain is coreferential, some of conference pairs may be direct while others may be indirect since they only become conferential via other referring expressions in the same coreference chain. This indicates that the most recent antecedent of an anaphor in the coreferential chain is sometimes indirectly linked to the anaphor via some other antecedents in the chain. In these indirect cases, we find that the most recent antecedent always contains little information to directly determine the coreference relationship with the anaphor. Generally, direct and informative coreference pairs are much easier to resolve than indirect and less informative ones. In the following example[1],

*Microsoft Corp. (i)* announced *its (i)* new CEO yesterday. *Microsoft (i)* said …

[1] The italic markables with the same identification symbol are coreferential.

"Microsoft Corp.", "its" and "Microsoft" form a coreference chain. Among the three coreference pairs in the chain,

1) The coreference pair between "Microsoft Corp." and "Microsoft" is direct.
2) The coreference pair between "Microsoft Corp." and "its" is direct.
3) The coreference pair between "its" and "Microsoft" is *indirect*. This coreference pair only becomes coreferential via another referring expression "Microsoft Corp." Direct resolution of this coreference pair is error-prone and not necessary since it can be indirectly linked by the other two coreference pairs in the coreference chain.

Therefore, for a given anaphor, we can always safely filter out these less informative antecedent candidates. In this way, rather than finding the most recent antecedent for an anaphor, our system tries to find the most direct and informative antecedent. This also suggests that we can classify coreference types according to the types of anaphors and restrict the possible types of antecedent candidates for a given anaphor type as follows:

- **Name alias coreference**

This is the most widespread type of coreference which is realised by the name alias phenomenon. The success of name alias coreference resolution is largely conditional on success at determining when one referring expression is a name alias of another referring expression. Here, the direct antecedent candidate of a named entity anaphor can only be the type of named entity. For example,

> *Microsoft Corp. (i)* announced its new CEO yesterday. *Microsoft (i)* said …

- **Apposition coreference**

This is the easiest type of coreference. A typical use of an appositional noun phrase is to provide an alternative description for a named entity. For example

> *Julius Caesar (i)*, *the well-known emperor (i)*, was born in 100 BC.

- **Predicate nominal coreference**

Predicate nominal is typically coreferential with the subject. For example,

> *George W. Bush (i)* is *the president of the United States (i)*.

- **Pronominal coreference**

This is the second widespread type of coreference which is realised by pronouns. Pronominal coreference has been widely studied in literature of traditional anaphora resolution. The direct antecedent candidate of a pronoun anaphor can be any type of referring expressions. For example,

> *Computational linguistics (i)* from different countries attended the tutorial. *They (i)* took extensive note.

- **Definite noun phrase coreference**

This is the third widespread type of coreference which is realised by definite noun phrases. It has also been widely studied in the literature of traditional anaphora resolution. A typical case of definite noun phrase coreference is when the antecedent is referred by a definite noun phrase anaphor representing either same concept (repetition) or semantically close concept (e.g. synonyms, super-ordinates). The direct antecedent candidate of a definite noun phrase anaphor can be any type of referring expressions except pronouns. For example,

> *Computational linguistics (i)* from different countries attended the tutorial. *The participants (i)* took extensive note.

- **Demonstrative noun phrase coreference**

This type of coreference is not widespread. Similar to that of definite noun phrase coreference, the direct antecedent candidate of a demonstrative noun phrase anaphor can be any type of referring expressions except pronouns. For example,

> Boorda wants to limit the total number of sailors on *the arsenal ship (i)* to between 50 and 60. Currently, *this ship (i)* has about 90 sailors.

- **Bare noun phrase coreference**

The direct antecedent candidate of a bare noun phrase anaphor can be any type of referring expressions except pronouns. For example,

> The price of *aluminium (i)* siding has steadily increased, as the market for *aluminium (i)* reacts to the strike in Chile.

## 4 Constraint-based Multi-Agent System for Coreference Resolution

In accordance with the above differentiation of coreference types according to the anaphor types, a constraint-based multi-agent system is developed.

### 4.1 Common Constraint Agent

For all coreference types described in Section 3, a common constraint agent is applied first using following constraints:

**Morphological agreements**

These constraints require that an anaphor and its antecedent candidate should agree in gender and number. These kinds of morphological agreements has been widely used in the literature of anaphora resolution

**Semantic consistency**

This constraint stipulates that the anaphor and its antecedent candidate must be consistent in semantics. For example, the anaphor and its antecedent candidate should contain the same sense or the anaphor contains a sense which is parental to the antecedent candidate. In this paper, WordNet (Miller 1990) is used for semantic consistency check.

For example,

*IBM (i)* announced its new CEO yesterday. *The company (i)* said …

### 4.2 Special Constraint Agents

For each coreference type described in Section 3, a special constraint agent is applied next using some heuristic rules mainly based on the accessibility space, which is learnt from the training data as follows:

For a given coreference type and a given valid antecedent type, all the anaphors of the given coreference type are identified first from left to right as they appear in the sentences. For each anaphor, its antecedent is then determined using the principle of proximity. If the most recent antecedent candidate has the given antecedent type, meet the morphological agreements and semantic consistency and is in the same coreference chain as the anaphor, this coreference pair is counted as a correct instance for the given conference type and the given antecedent type. Otherwise, it is counted as an error instance. In this way, the precision rates of the coreference type over different valid antecedent types and different accessibility spaces are computed as the percentage of the correct instances among all the correct and error instances. Finally, the accessibility space for a given coreference type and a given antecedent type is decided using a precision rate threshold (e.g. 95%).

- **Agent for name alias coreference**

A named entity is co-referred with another named entity when the formal is a name alias of the latter. This type of coreference has an accessibility space of the whole document. In this paper, it is tackled by a named entity recognition component, as in Zhou and Su (2002), using the following name alias algorithm in the ascending order of complexity:

1) The simplest case is to recognize full identity of strings. This applies to all types of entity names.
2) The next simplest case is to recognize the various forms of location names. Normally, various acronyms are applied, e.g. "NY" vs. "New York" and "N.Y." vs. "New York". Sometime, partial mention is also applied, e.g. "Washington" vs. "Washington D.C.".
3) The third case is to recognize the various forms of personal proper names. Thus an article on Microsoft may include "Bill Gates", "Bill" and "Mr. Gates". Normally, the full personal name is mentioned first in a document and later mention of the same person is replaced by various short forms such as acronym, the last name and, to a less extent, the first name, of the full person name.
4) The most difficult case is to recognize the various forms of organizational names. For various forms of company names, consider a) "International Business Machines Corp.", "International Business Machines" and "IBM"; b) "Atlantic Richfield Company" and "ARCO". Normally, various abbreviation forms (e.g. contractions and acronym) and dropping of company suffix are applied. For various forms of other organizational names, consider a) "National University of Singapore", "National Univ. of Singapore" and "NUS"; b) "Ministry of Education" and "MOE". Normally, acronyms and abbreviations are applied.

- **Agent for apposition coreference**

If the anaphor is in apposition to the antecedent candidate, they are coreferential. The MUC-6 and MUC-7 coreference task definitions are slightly different. In MUC-6, the appositive should be a definite noun phrase while both indefinite and definite noun phrases are acceptable in MUC-7.

- **Agent for predicate nominal coreference**

If the anaphor is the predicate nominal and the antecedent candidate is the subject, they are coreferential. This agent is still under construction.

- **Agent for pronominal coreference**

This agent is applied to the most widely studied coreference: pronominal coreference. 6 heuristic rules are learnt and applied depending on the accessibility space and the types of the antecedent candidates:

1) If the anaphor is a person pronoun and the antecedent candidate is a person named entity, they are coreferential over the whole document.
2) If the anaphor is a neuter pronoun and the antecedent candidate is an organization named entity, they are coreferential when they are in the same sentence.
3) If the anaphor is a neuter plural pronoun and the antecedent candidate is a plural noun

phrase, they are coreferential over the whole document.
4) If both the anaphor and the antecedent candidate are third person pronouns, they are coreferential over the whole document.
5) If both the anaphor and the antecedent candidate are first or second person pronouns, they are coreferential when they are in the same paragraph.
6) If both the anaphor and the antecedent candidate are neuter pronouns, they are coreferential when they are in the same paragraph or the antecedent candidate is in the previous paragraph of the anaphor.

- **Agent for definite noun phrase coreference**

The agent for definite noun phrase coreference is mainly based on the accessibility space. This agent is based on the following 3 heuristic rules:
1) The definite noun phrase will be coreferential with a named entity if they are in same paragraph or the entity name is in the previous paragraph of the definite noun phrase.
2) The definite noun phrase will be coreferential with a named entity if the head word of the definite noun phrase is only modified by the determiner "the". That is, the definite noun phrase is of type "the HEADWORD", e.g. "the company".
3) The definite noun phrase will be coreferential with a definite/demonstrative/indefinite noun phrase if they string-match[2].

- **Agent for demonstrative noun phrase coreference**

The agent for demonstrative noun phrase coreference is similar to the agent for definite noun phrase coreference except that the anaphor is a demonstrative noun phrase.

- **Agent for base noun phrase coreference**

This is the most complicated and confusing coreference in MUC coreference task definitions. Although this type of coreference occupies a large portion, it is hard to find heuristic rules to deal with it. In our system, only one heuristic rule is applied: If the anaphor and the antecedent candidate string-match and include at least two words except the determiner, they are coreferential over the whole document.

[2] The determiners, e.g. "a", "an" and "the", are removed from the strings before comparison. Therefore, "the company" string-matches "a company".

### 4.3 Common Preference Agent

For a given anaphor, invalid antecedents are first filtered out using the above common constraint agent and the special constraint agent. Then, the strategy has to choose which of the remaining candidates, if any, is the most likely antecedent candidate. In our strategy, this is done through a common preference agent based on the principle of proximity. That is, our common preference agent takes advantages of the relative locations of the remaining antecedent candidates in the text. Among the antecedent candidates:
1) First it looks for those occurring earlier in the current sentence, preferring the one that occurs earliest in the natural left-to-right order.
2) If there are no antecedent candidates occurring earlier in the current sentence, look to those occurring in the immediately preceding sentence of the same paragraph, again preferring the one that occurs earliest in that sentence in left-to-right order.
3) If nothing comes up, look back at those occurring in the earlier sentences of the same paragraph, moving back a sentence at a time, but now, within a given sentence preferring the most rightward candidate that occurs later in the sentence.
4) Finally, if the scope extends back beyond a paragraph boundary, it looks to those that occur in the sentences of the preceding paragraph, again preferring later to earlier occurrences.

### 4.4 Multi-Agent Algorithm

The coreference resolution algorithm is implemented based on the previous multi-agents. First, all the anaphors are identified from left to right as they appear in the sentences. Then, for a given anaphor,
1) All the referring expressions occurred before the anaphor are identified as antecedent candidates.
2) The common constraint agent is applied to filter out the invalid antecedent candidates using various general constraints, such as morphological agreements and semantic consistency constraints.
3) The corresponding special constraint agent (if exists) is recalled to first filter out indirect and less informative antecedent candidates and then check the validity of the remaining antecedent candidates by using some heuristic rules. In this way, more invalid antecedent candidates are discarded using various special constraints, such as the accessibility space.
4) The antecedent is chosen from the remaining antecedent candidates, if any, using the

common preference agent based on the principle of proximity.

## 5 Experimentation

Table 1 shows the performance of our constraint-based multi-agent system on MUC-6 and MUC-7 standard test data using the standard MUC evaluation programs while Table 2 gives the comparisons of our system with others using the same MUC test data and the same MUC evaluation programs. Here, the precision (P) measures the number of correct coreference pairs in the answer file over the total number of coreference pairs in the answer file and the recall (R) measures the number of correct coreference pairs in the answer file over the total number of coreference pairs in the key file while F-measure is the weighted harmonic mean of precision and recall: $F = \frac{(\beta^2+1)RP}{\beta^2 R + P}$ with $\beta^2$=1.

Table 1: Results of our baseline multi-agent coreference resolution system on MUC-6 and MUC-7

| Performance | MUC-6 | | | MUC-7 | | |
|---|---|---|---|---|---|---|
| | R | P | F | R | P | F |
| **Overall** | **65.8** | **84.7** | **73.9** | **55.7** | **82.8** | **66.5** |
| • Agent for name alias coreference | 32.7 (35) | 92.3 | - | 33.6 (36) | 89.0 | - |
| • Agent for apposition coreference | 4.3 (5) | 95.5 | - | 2.6 (3) | 84.6 | - |
| • Agent for predicate nominal coreference[3] | - (2) | - | - | - (3) | - | - |
| • Agent for pronominal coreference | 18.6 (22) | 77.5 | - | 10.8 (16) | 72.3 | - |
| • Agent for definite noun phrase coreference | 9.4 (15) | 80.0 | - | 7.0 (20) | 85.0 | - |
| • Agent for demonstrative noun phrase coreference | 0.1 (2) | 50.0 | - | 0.2 (2) | 66.7 | - |
| • Agent for bare noun phrase coreference | 1.9 (19) | 63.0 | | 1.7 (20) | 61.1 | - |

Table 2: Comparison of our system with the best-reported systems on MUC-6 and MUC-7

| Performance Comparison | MUC-6 | | | MUC-7 | | |
|---|---|---|---|---|---|---|
| | R | P | F | R | P | F |
| **Ours** | **65.8** | **84.7** | **73.9** | 55.7 | **82.8** | **66.5** |
| Ng and Cardie 2002 (C4.5) | 64.1 | 74.9 | 69.1 | **57.4** | 70.8 | 63.4 |
| Ng and Cardie 2002 (RIPPER) | 64.2 | 78.0 | 70.4 | 55.7 | 72.8 | 63.1 |

Table 1 shows that our system achieves F-measures of 73.9 and 66.5 on MUC-6 and MUC-7 standard test data, respectively. The figures outside the parentheses show the contributions of various agents to the overall recall while the figures inside the parentheses show the frequency distribution of various coreference types in the answer file. It shows that the performance difference between MUC-6 and MUC-7 mainly comes from the significant distribution variation of pronominal coreference. It also shows that there are much room for improvement, especially for the types of pronominal coreference and definite noun pronoun resolution. Table 2 shows that our system achieves significantly better F-measures by 3.1~4.8 percent over the best-reported systems (Ng and Cardie 2002). Most of the contributions come form precision gains. Our system achieves significantly better precision rates by 6.7~10.0 percent over the best-reported systems (Ng and Cardie 2002) while keeping recall rates. One reason behind such high performance is the restriction of indirect and less informative antecedent candidates according to the type of the anaphor. Another reason is differentiation of various types of coreference and the use of multi-agents. In this way, various types of coreference are dealt with effectively by different agents according to their characteristics. The recall difference between our system and the RIPPER system in (Ng and Cardie 2002) maybe come from the predicate nominal coreference, which can be easily resolved using a machine learning algorithm, e.g. (Cohen 1995). Completion of the agent for predicate nominal coreference can easily fill the difference.

## 6 Conclusions

This paper presents a constraint-based multi-agent strategy to coreference resolution of general noun phrases in unrestricted English text.

The first contribution of this paper comes from the high performance of our system and its easy

[3] The agent for predicate nominal coreference is still under construction.

implementation. The second contribution is to filter out indirect and less informative antecedent candidates according to the anaphor type. The third contribution is the differentiation of various coreference types according to the anaphor types and the use of multi-agents.

Future work includes:

- The exploration of new constraints to improve the precision and new coreference types to increase the recall.
- The problem of type coercion or metonymy which is a general problem and accounts for much of the overall missing recall.
- The problem of cataphora, which is not handled in the current mechanism.

## References


Brennan S. E. Friedman M. W. and Pollard C. J. 1987. A centering approach to pronouns. *Proceedings of the 25th Annual Meeting of the Association for Computational Linguistics (ACL'1987),* pages 155-162.

Carbonell J. and Brown R. 1988. Anaphora resolution: a multi-strategy approach. *Proceedings of the 12th International Conference on Computational Linguistics (COLING'1988),* pages 96-101, Budapest, Hungary.

Carter D. M. 1987. *Interpreting Anaphors in Natural Language Texts.* Ellis Horwood, Chichester, UK.

Cohen W. 1995. Fast effective rule induction. *Proceedings of the Twelfth International Conference on Machine Learning (ICML'1995).* pages 115-123. Tahoe City, CA.

Dagan I. and Itai A. 1990. Automatic processing of large corpora for the resolution of anaphora references. *Proceedings of the 13th International Conference on Computational Linguistics (COLING'1990),* pages 1-3, Helsinki, Finland.

Grosz B. J. and Sidner C. L. 1986. Attention, intentions and the structure of discourse. *Computational Linguistics,* 12(3):175-204.

Lappin S. and Leass H. 1994. An algorithm for pronominal anaphora resolution. *Computational Linguistics.* 20(4):535-561.

Miller G.A. 1990. WordNet: An online lexical database. International Journal of Lexicography. 3(4):235-312.

Mitkov R. 1998. Robust pronoun resolution with limited knowledge. Proceedings of the 36th Annual Meeting for Computational Linguistics and the 17th International Conference on Computational Linguistics (COLING/ACL'1998), pages 869-875, Montreal, Canada.

MUC-6. 1995. Proceedings of the 6th Message Understanding Conference (MUC-6). Morgan Kaufmann, San Francisco, CA.

MUC-7. 1998. Proceedings of the 7th Message Understanding Conference (MUC-7). Morgan Kaufmann, San Mateo, CA.

Ng V. and Cardie C. 2002. Improving machine learning approaches to coreference resolution. *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'2002),* pages 104-111, Philadelphia, Penn.

Rich E. and LuperFoy S. 1988. An architecture for anaphora resolution. *Proceedings of the 2nd Conference on Applied Natural Language Processing (ANLP'1988),* pages 18-24, Austin, TX.

Soon W. M.., Ng H. T. and Lim C. Y. 2001. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics*, 27(4):521-544.

Strube M. 1998. Never look back: An alternative to centering. Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics, pages 1251-1257.

Tetreault J. R. 2001. A corpus-based evaluation of centering and pronoun resolution. *Computation Linguistics*, 27(4):507-520.

Zhou G. D. and Su Jian, 2000. Error-driven HMM-based chunk tagger with context-dependent lexicon. Proceedings of the Joint Conference on Empirical Methods on Natural Language Processing and Very Large Corpus (EMNLP/ VLC'2000). Hong Kong.

Zhou G. D. and Su Jian. 2002. Named Entity Recognition Using a HMM-based Chunk Tagger, *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'2002)*. Philadelphia.