# Extracting Bacteria Biotopes with Semi-supervised Named Entity Recognition and Coreference Resolution

**Nhung T. H. Nguyen** and **Yoshimasa Tsuruoka**
School of Information Science
Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi, Ishikawa 923-1292 Japan
{nthnhung,tsuruoka}@jaist.ac.jp

## Abstract

This paper describes our event extraction system that participated in the bacteria biotopes task in BioNLP Shared Task 2011. The system performs semi-supervised named entity recognition by leveraging additional information derived from external resources including a large amount of raw text. We also perform coreference resolution to deal with events having a large textual scope, which may span over several sentences (or even paragraphs). To create the training data for coreference resolution, we have manually annotated the corpus with coreference links. The overall F-score of event extraction was 33.2 at the official evaluation of the shared task, but it has been improved to 33.8 thanks to the refinement made after the submission deadline.

## 1 Introduction

In this paper, we present a machine learning-based approach for bacteria biotopes extraction of the BioNLP Shared Task 2011 (Bossy et al. , 2011). The task consists of extracting bacteria localization events, namely, mentions of given species and the place where it lives. Places related to bacteria localization events range from plant or animal hosts for pathogenic or symbiotic bacteria to natural environments like soil or water[1]. This task also targets specific environments of interest such as medical environments (hospitals, surgery devices, etc.), processed food (dairy) and geographical localizations.

[1] https://sites.google.com/site/bionlpst/home/bacteria-biotopes

The task of extracting bacteria biotopes involves two steps: Named Entity Recognition (NER) and event detection. The current dominant approach to NER problems is to use supervised machine learning models such as Maximum Entropy Markov Models (MEMMs), Support Vector Machines (SVMs) and Conditional Random Fields (CRFs). These models have been shown to work reasonably well when a large amount of training data is available (Nadeau and Sekine, 2007). However, because the annotated corpus delivered for this particular subtask in the shared task is very small (78 documents with 1754 sentences), we have decided to use a semi-supervised learning method in our system. Our NER module uses a CRF model with enhanced features created from external resources. More specifically, we use additional features created from the output of HMM clustering performed on a large amount of raw text, and word senses from WordNet for tagging.

The target events in this shared task are divided into two types. The first is Localization events which relates a bacterium to the place where it lives. The second is PartOf events which denotes an organ that belongs to an organism. As in Bossy et al. (2010), the largest possible scope of the mention of a relation is the whole document, and thus it may span over several sentences (or even paragraphs). This observation motivated us to perform coreference resolution as a pre-processing step, so that each event can be recognized within a narrower textual scope. There are two common approaches to coreference resolution: one mainly relies on heuristics, and the other employs machine learning. Some



*Proceedings of BioNLP Shared Task 2011 Workshop*, pages 94–101,
Portland, Oregon, USA, 24 June, 2011. 

instances of the heuristics-based approach are described in (Harabagiu et al., 2001; Markert and Nissim, 2005; Yang and Su, 2007), where they use lexical and encyclopedic knowledge. Machine learning-based methods (Soon and Ng, 2001; Ng and Cardie, 2002; Yang et al. , 2003; Luo et al. , 2004; Daume and Marcu, 2005) train a classifier or search model using a corpus annotated with anaphoric pairs. In our system, we employ the simple supervised method presented in Soon and Ng (2001). To create the training data, we have manually annotated the corpus with coreference information about bacteria.

Our approach, consequently, has three processes: NER, coreference resolution of bacterium entities, and event extraction. The latter two processes can be formulated as classification problems. Coreference resolution is to determine the relation between candidate noun phrases and bacterium entities, and the event extraction is to detect the relation between two entities. It should be noted that our official submission in the shared task was carried out without using a coreference resolution module, and the system has been improved after the submission deadline.

Our contribution in this paper is two-fold. In the methodology aspect, we use an unsupervised learning method to create additional features for the CRF model and perform coreference resolution to narrow the scope of events. In the resource aspect, the manual annotations for training our coreference resolution module will be made available to the research community.

The remainder of this paper is organized as followed. Section 2, 3 and 4 describe details about the implementation of our system. Section 5 presents the experimental results with some error analysis. Finally, we conclude our approach and discuss future work in section 6.

## 2 Semi-supervised NER

According to the task description, the NER task consists of detecting the phrases that denote bacterial taxon names and localizations which are broken into eight types: Host, HostPart, Geographical, Food, Water, Soil, Medical and Environment. In this work, we use a CRF model to perform NER. CFRs (Lafferty et. al., 2001) are a sequence modeling framework that not only has all the advantages of MEMMs but also solves the label bias problem in a principled way. This model is suitable for labeling sequence data, especially for NER. Based on this model, our CRF tagger is trained with a stochastic gradient descent-based method described in Tsuruoka et al. (2009), which can produce a compact and accurate model.

Due to the small size of the training corpus and the complexity of their category, the entities cannot be easily recognized by standard supervised learning. Therefore, we enhance our learning model by incorporating related information from other external resources. On top of the lexical and syntactic features, we use two additional types of information, which are expected to alleviate the data sparseness problem. In summary, we use four types of features including lexical and syntactic features, word cluster and word sense features as the input for the CRF model.

### 2.1 Word cluster features

The idea of enhancing a supervised learning model with word cluster information is not new. Kamaza et. al. (2001) use a hidden Markov model (HMM) to produce word cluster features for their maximum entropy model for part-of-speech tagging. Koo et al. (2008) implement the Brown clustering algorithm to produce additional features for their dependency parser. For our NER task, we use an HMM to produce word cluster features for our CRF model.

We employed an open source library[2] for learning HMMs with the online Expectation Maximization (EM) algorithm proposed by Liang and Klein (2009). The online EM algorithm is much more efficient than the standard batch EM algorithm and allows us to use a large amount of data. For each hidden state, words that are produced by this state with the highest probability are written. We use this result of word clustering as a feature for NER. The optimal number of hidden states is selected by evaluating its effectiveness on NER using the development set.

To prepare the raw text for HMM clustering, we downloaded 686 documents (consisting of both full documents and abstracts) about bacteria biotopes

[2] http://www-tsujii.is.s.u-tokyo.ac.jp/~hillbig/ohmm.htm

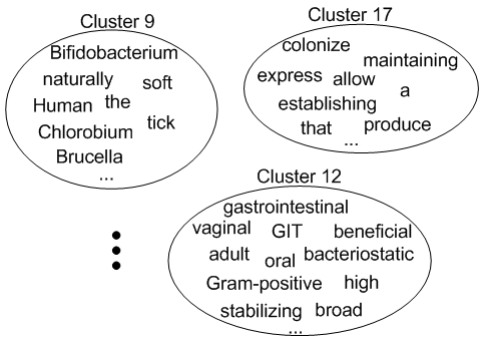


Figure 1: Sample of HMM clustering result.

| Word | POS | Sense |
|---|---|---|
| chromosome | NN | body |
| colonize | VBP | social |
| detected | VBN | perception |
| fly | NN | animal |
| gastrointestinal | JJ | pert |
| infant | NN | person |
| longum | FW | bacterium |
| maintaining | VBG | stative |
| milk | NN | food |
| onion | NN | plant |
| proterins | NNS | substance |
| USA | NNP | location |

Table 1: Sample of word sense features given by WordNet and NCBI dictionary.

from MicrobeWiki, JGI Genome Portal, Genoscope, 2Can bacteria pages at EBI and NCBI Genome Project (the training corpus is also downloaded from these five webpages). In addition, we use the 100,000 latest MEDLINE abstracts containing the string "bacteri" in our clustering. In total, the raw text consists of more than 100,000 documents with more than 2 million sentences.

A part of the result of HMM clustering is shown in Figure 1. According to this result, the word "*Bifidobacterium*" belongs to cluster number 9, and its feature value is "*Cluster-9*". The word cluster features of the other words are extracted in the same way.

### 2.2 Word sense features

We used WordNet to produce additional features on *word senses*. Although WordNet[3] is a large lexical database, it only comprises words in the general genre, to which only the localization entities belong. Since it does not contain the bacterial taxon names, the most important entities in this task, we used another dictionary for bacteria names. The dictionary was extracted from the genomic BLAST page of NCBI [4]. To connect these two resources, we simply place all entries from the NCBI dictionary under the 'bacterium' sense of WordNet. Table 1 illustrates some word sense features employed in our model.

### 2.3 Pre-processing for bacteria names

In biomedical documents, the bacteria taxon names are written in many forms. For example, they are presented in a full name like "Bacillius cereus", or in a short form such as "B. cereus", or even in an abbreviation as "GSB" (green sulfur bacteria). Moreover, the bacteria names are often modified with some common strings such as "strain", "spp.", "sp.", etc. "Borrelia hermsii ***strain*** DAH", "Bradyrhizobium ***sp.*** BTAi1", and "Spirochaeta ***spp.***" are examples of this kind. In order to tackle this problem, we apply a pre-processing step before NER. Although there are many previous studies solving this kind of problem, in our system, we apply a simple method for this step.

- *Retrieving the full form of bacteria names*. We assume that (a) both short form and full form must occur in the same document; (b) a token is considered as an abbreviation if it is written in upper case and its length is shorter than 4 characters. When a token satisfies condition (b) (which means it is an abbreviation), the processing retrieves its full form by identifying all sequences containing tokens initialized by its abbreviated character. In case of short form like "B. cereus", the selected sequence must include the right token (which is "cereus" in "B. cereus").

- *Making some common strings transparent*. As our observation on the training data, there are 8 common strings in bacteria names, including "strain", "str", "str.", "subsp", "spp.", "spp", "sp.", "sp". All of these strings will be removed before NER and recovered after that.

[3] http://wordnet.princeton.edu/

[4] http://www.ncbi.nlm.nih.gov/sutils/genom_table.cgi

## 3 Coreference Resolution as Binary Classification

Coreference resolution is the process of determining whether different nominal phrases are used to refer to the same real world entity or concept. Our approach basically follows the learning method described in Soon and Ng (2001). In this approach, we build a binary classifier using the coreferring entities in the training corpus. The classifier takes a pair of candidates and returns *true* if they refer to the same real world entity and *false* otherwise. In this paper, we limit our module to detecting the bacteria's coreference, and hence the candidates consist of noun phrases (NPs) (starting by a determiner), pronouns, possessive adjective and name of bacteria.

In addition to producing the candidates, the preprocessing step creates a set of features for each anaphoric pair. These features are used by the classifier to determine if two candidates have a coreference relation or not.

The following features are extracted from each candidate pair.

- *Pronoun*: 1 if one of the candidates is a pronoun; 0 otherwise.
- *Exact or Partial Match*: 1 if the two strings of the candidates are identical, 2 if they are partial matching; 0 otherwise.
- *Definite Noun Phrase*: 1 if one of the candidates is a definite noun phrases; 0 otherwise.
- *Demonstrative Noun Phrase*: 1 if one of the candidates is a demonstrative noun phrase; 0 otherwise.
- *Number Agreement*: 1 if both candidates are singular or plural; 0 otherwise.
- *Proper Name*: 1 if both candidates are bacterium entities or proper names; 0 otherwise.
- *Character Distance*: count the number of the characters between two candidates.
- *Possessive Adjective*: 1 if one of the candidates is possessive adjective; 0 otherwise.

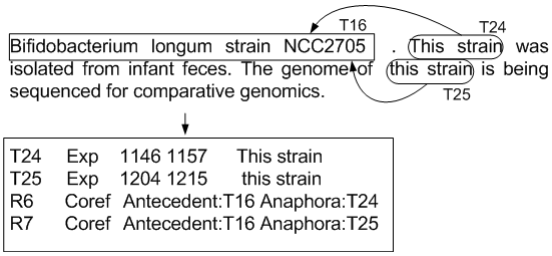


Figure 2: Example of annotating coreference resolution. T16 is a bacterium which is delivered in *.a2 file, T24 and T25 are anaphoric expressions. There are two coreference relations of T16 and T24, T16 and T25.

- *Exist in Coreference Dictionary*: 1 if the candidate exists in the dictionary extracted from the training data; 0 otherwise. This feature aims to remove noun phrases which are unlikely to be related to the bacterium entities.

The first five features are exactly the same as those in Soon and Ng (2001), while the others are refined or added to make it suitable for our specific task.

In the testing phase, we used the ***best-first clustering*** as in Ng and Cardie (2002). Rather than performing a right-to-left search from each anaphoric NP for the first coreferent NP, a right-to-left search for a *highly likely antecedent* was performed. Hence, the classifier was modified to select the antecedent of NP with the coreference likelihood score above a threshold. This threshold was tuned by evaluating it on the development set.

### 3.1 Corpus annotation

To create the training data for coreference resolution, we have manually annotated the corpus based on the gold-standard named entity annotations delivered by the organizer. Due to our decision to focus on bacteria names, only the coreference of these entities are labeled. We use a format similar to those of the organizer, i.e. the standoff presentation and textbound annotations. The coreference annotation file consists of two parts, one part for anaphoric expressions and the other for coreference relation. Figure 2 shows an example of a coreference annotation with the original text.

## 4 Event Extraction

The bacteria biotopes, as mentioned earlier, are divided into two types. The first type of events, namely localization events, relates a bacterium to the place where it lives, and has two mandatory arguments: a Bacterium type and a localization type. The second type of events, i.e. PartOf events, denote an organ that belongs to an organism, and has two mandatory arguments of type HostPart and Host respectively. We view this step as determining the relationship between two specific entities. Because of no ambiguity between the two types of event, the event extraction can be solved as the binary classification of pairs of entities. The classifier is trained on the training data with four types of feature extracted from the context between two entities: distance in sentences, the number of entities, the nearest left and right verbs.

**Generating Training Examples**. Given the coreference information on bacterium entities, the system considers all the entities belonging to the coreference chains as real bacteria and generates event instances. Since about 96% of all annotated events occur in the same paragraph, we restrict our method to detecting events within one paragraph.

- ***Localization Event***. The system creates a relationship between a bacterium and a localization entity with ***minimum distance*** between them by the following priorities:

  (1) The bacterium *precedes* the localization entity *in the same sentence*.

  (2) The bacterium *precedes* the localization entity *in the same paragraph*.

- ***PartOf Event***. All possible relationships between Host and HostPart entities are generated if they are in the same paragraph.

## 5 Experiments and Discussion

The training and evaluation data used in these experiments are provided by the shared task organizers. The token and syntactic information are extracted from the supporting resources (Stenetorp et. al. , 2011). More detail, the tokenized text was done by GENIA tools, and the syntactic analyses was created by the McClosky-Charinak parser (McClosky and Charniak, 2008), trained on the GENIA Treebank corpus (Tateisi et al., 2005), which is one of the most accurate parsers for biomedical documents.

| Experiment | Acc. | Pre. | Re. | F-score |
|---|---|---|---|---|
| Baseline | 94.28 | 76.32 | 35.51 | 48.47 |
| Word cluster | 94.46 | 78.23 | 39.59 | 52.57 |
| Word sense | 94.63 | 74.15 | 44.49 | 55.61 |
| **All Features** | **94.70** | **77.62** | **45.31** | **57.22** |

Table 2: Performance of Named Entity Recognition in terms of Accuracy, Precision, Recall and F-score with different features on the development set.

For both classification of anaphoric pairs in coreference resolution and determining relationship of two entites, we used the SVM$^{light}$ library [5], a state-of-the-art classifier, with the linear kernel.

In order to find the best parameters and features for our final system, we conducted a series of experiments at each step of the approach.

### 5.1 Named Entity Recognition

We evaluated the impact of additional featues on NER by running four experiments. The Baseline experiment was conducted by using the original CRF tagger, which did not use any additional features derived from external resources. The other three experiments were conducted by incrementally adding more features to the CRF tagger. Table 2 shows the results on the development set[6].

Through these experiments we have realized that using the external resources is very effective. The *word cluster* and *word sense* features are used like a dictionary. The first one can be considered as the dictionary of specific classes of entity in the same domain with this task, which mainly supports the precision, whereas the latter is a general dictionary boosting the recall. With regard to F-score, the *word sense* features outperform the *word cluster* features. When we combine all of them, the F-score is improved significantly by nearly 9 points.

The detailed results of individual classes in Table 3 show that the Environment entities are the hardest to recognize. Because of their general characteristic, these entities are often confused with Host

[5] http://svmlight.joachims.org/

[6] These scores were generated by using the CoNLL 2000 evaluation script.

| Class | Gold | Pre. | Re. | F-score |
|---|---|---|---|---|
| Bacterium | 86 | 70.00 | 40.23 | 51.09 |
| Host | 78 | 78.57 | 56.41 | 65.67 |
| HostPart | 44 | 91.67 | 50.00 | 64.71 |
| Geographical | 8 | 71.43 | 62.50 | 66.67 |
| Environment | 8 | 0.00 | 0.00 | 0.00 |
| Food | 0 | N/A | N/A | N/A |
| Medical | 2 | 100.00 | 50.00 | 66.67 |
| Water | 17 | 100.00 | 17.65 | 30.00 |
| Soil | 1 | 100.00 | 100.00 | 100.00 |
| All | 244 | **77.62** | **45.31** | **57.22** |

Table 3: Results of NER using all features on the development set. The "Gold" column shows the number of entities of that class in the gold-standard corpus. The score of Food entities is not available because there is no positive instance in the development set.

| | Detection | Linking |
|---|---|---|
| Precision | 24.18 | 20.48 |
| Recall | 91.36 | 33.71 |
| F-score | 38.24 | 25.48 |

Table 4: Result of coreference resolution on the development set achieved with gold-standard named entity annotations.

or Water. In contrast, the Geographical category is easier than the others if we have gazetteers and administrative name lists.

### 5.2 Coreference Resolution

We next evaluated the accuracy of coreference resolution for bacterium entities. The evaluation[7] is carried out in two steps: evaluation of mention detection, and evaluation of mention linking to produce coreference links. The exact matching criterion was used when evaluating the accuracy of the two steps. Table 4 shows the performance of the coreference resolution module when taking annotated entites as input. As mentioned in section 3, the first step of this module considers all NPs beginning with a determiner and bacterium entities as candidates. Therefore, the number of the candidate NPs is vastly larger than that of the positive ones. This is the reason why the precision of mention detection is low, while the recall is high. This high recall leads to a large number of generated linkings and raises the complexity of linking detection. In order to obtain more accurate results, we had to remove weak linkings whose classification score is under 0.7 (this is the best threshold on the development set). However, as shown in Table 4, the performance of mention linking was not satisfactory.

[7] http://sites.google.com/site/bionlpst/home/protein-gene-coreference-task

| Experiment | Pre. | Re. | F-score |
|---|---|---|---|
| No Coref. | 42.11 | 27.34 | 33.15 |
| With Coref. | 43.40 | 27.64 | 33.77 |

Table 5: Comparative results of event extraction with and without coreference information on the test set.

| Type of event | Num. of addition | | Num. of ruled out | |
|---|---|---|---|---|
| | True | False | True | False |
| Localization | 17 | 1 | 6 | 20 |
| PartOf | 6 | 5 | 1 | 0 |
| *Total* | 29 | | 27 | |

Table 6: Contribution of coreference resolution to event extraction.

### 5.3 Event Extraction

Finally, we carried out two experiments on the test set to investigate the effect of coreference resolution on event extraction. The results shown in Table 5 indicate that the contribution of coreference resolution in this particular experiment is not significant. The coreference information helps the module to add 29 more events (23 true and 6 false events) and rule out 27 events (20 false and 7 true events) compared with the experiment with no coreference resolution. Detail about this contribution is presented in Table 6.

We further analyzed the result of event extraction and found that there exist two kinds of Localization events, which we call *direct* and *indirect* events. The *direct* events are the ones that are easily recognizable on the surface level of textual expressions. The three Localization events in Figure 3 belong to this type. Our module is able to detect most of the direct events, especially when we have the coreference information on bacteria – it is straight-forward because the two arguments of the event occur in the same sentence. In constrast, the *indirect* events are more complicated. They appear implicitly in the document and we need to infer them through an intermediate agent. For example, a bacterium causes a disease, and this disease infects the humans or an-

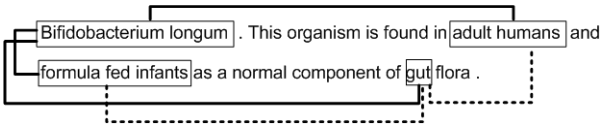


Figure 3: Example of direct events. The solid line is the Localization event, the dash line is the PartOf event.

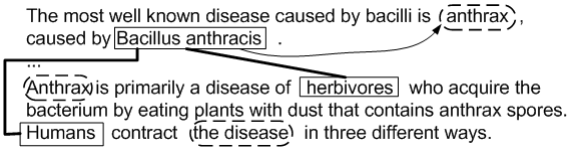


Figure 4: Example of indirect events. The solid line is the Localization event, the arrow shows the causative relation.

imals. Therefore, it can be considered that the bacterium locates in the humans or animals. Figure 4 illustrates this case. In this example, the Bacillus anthracis causes Anthrax, Humans contract the disease (which refers to Anthrax), and the Bacillus anthracis locates in Humans. These events are very difficult to recognize since, in this context, we do not have any information about the disease. Events of this type provide an interesting challenge for bacteria biotopes extraction.

## 6 Conclusion and Future Work

We have presented our machine learning-based approach for extracting bacteria biotopes. The system is implemented with modules for three tasks: NER, coreference resolution and event extraction.

For NER, we used a CRF tagger with four types of features: lexical and syntactic features, the word cluster and word sense extracted from the external resources. Although we achieved a significant improvement by employing WordNet and the HMM clustering on raw text, there is still much room for improvement. For example, because all extracted knowledge used in this NER module belongs to the general knowlegde, its performance is not as good as our expectation. We envisage that the performance of the module will be improved if we can find useful biological features.

We have attempted to use the information obtained from the coreference resolution of bacteria to narrow the event's scope. On the test set, although it does not improve the system significantly, the coreference information has shown to be useful in event extraction. [8]

In this work, we simply used binary classifiers with standard features for both coreference resolution and event detection. More advanced machine learning approaches for structured prediction may lead to better performance, but we leave it for future work.

## References

Robert Bossy, Claire Nedellec, and Julien Jourde. 2010. Guidelines for Annotation of Bacteria Biotopes.

Robert Bossy, Julien Jourde, Philippe Bessières, Marteen van de Guchte, and Claire Nédellec. 2011. BioNLP Shared Task 2011 - Bacteria Biotope, In *Proceedings of the BioNLP 2011 Workshop Companion Volume for Shared Task*. Portland, Oregon, Association for Computational Linguistics.

Hal Daumé III and Daniel Marcu. 2005. A Large-scale Exploration of Effective Global Features for a Joint Entity Detection and Tracking Model. In *Proceedings of HLT-EMNLP 2005*, pp. 97-104.

Sanda M. Harabagiu, Razvan C. Bunescu and Steven J. Maiorano. 2001. Text and Knowlegde Mining for Coreference Resolution. In *Proceedings of NAACL 2001*, pp. 1-8.

Jun'ichi Kazama, Yusuke Miyao, and Jun'ichi Tsujii. 2001. A Maximum Entropy Tagger with Unsupervised Hidden Markov Models. In *Proceedings of NLPRS 2001*, pp. 333-340.

Terry Koo, Xavier Carreras, and Michael Collins. 2008. Simple Semi-supervised Dependency Parsing. In *Proceedings of ACL-08: HLT*, pp. 595-603.

John Lafferty, Andrew McCallum and Fernando Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of ICML'01*, pp. 282-289.

Percy Liang and Dan Klein. 2009. Online EM for Unsupervised Models. In *Proceedings of NAACL 2009*, pp. 611-619.

Xiaoqiang Luo, Abe Ittycheriah, Hongyan Jing, Nanda Kambhatla and Salim Roukos. 2004. A Mention-Synchronous Co-reference Resolution Algorithm based on the Bell Tree. In *Proceedings of ACL 2004*, pp. 135-142.

Katja Markert and Malvina Nissim. 2005. Comparing Knowledge Sources for Nominal Anaphora Resolution. In *Computational Linguistics, Volume 31 Issue 3*, pp. 367-402.

[8] If you are interesting in the annotated corpus used for our coreference resolution model, please request us by email.

David McClosky and Eugene Charniak. 2008. Self-Training for Biomedical Parsing. *Proceedings of the Association for Computational Linguistics (ACL 2008, short papers)*, Columbus, Ohio, pp. 101-104.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Linguisticae Investigationes, Volume 30(1)*, pp. 326.

Vincent Ng and Claire Cardie. 2002. Improving Machine Learning Approach to Co-reference Resolution. In *Proceedings of ACL 2002*, pp. 104-111.

Wee Meng Soon and Hwee Tou Ng. 2001. A Machine Learning Approach to Co-reference Resolution of Noun Phrases. *Computational Linguistics 2001, Volume 27 Issue 4*, pp. 521-544.

Pontus Stenetorp, Goran Topić, Sampo Pyysalo, Tomoko Ohta, Jin-Dong Kim, and Jun'ichi Tsujii. 2011. BioNLP Shared Task 2011: Supporting Resources. In*Proceedings of the BioNLP 2011 Workshop Companion Volume for Shared Task*, Portland, Oregon, Association for Computational Linguistics.

Yuka Tateisi, Akane Yakushiji, Tomoko Ohta and Junichi Tsujii. 2005. Syntax Annotation for the GENIA corpus. In *Proceedings of IJCNLP 2005 (Companion volume)*, pp. 222-227.

Yoshimasa Tsuruoka, Jun'ichi Tsujii, and Sophia Ananiadou. 2009. Stochastic Gradient Descent Training for L1-regularized Log-linear Models with Cumulative Penalty. In *Proceedings of ACL-IJCNLP*, pp. 477-485.

Xiaofeng Yang, Guodong Zhou, Jian Su and Chew Lim Tan. 2003. Co-reference Resolution using Competition Learning Approach. In *Proceedings of ACL 2003*, pp. 176-183.

Xiaofeng Yang and Jian Su. 2007. Coreference Resolution Using Semantic Relatedness Information from Automatically Discovered Patterns. In *Proceedings of ACL 2007*, pp. 528-535.