Coreference Resolution System using Maximum Entropy Classifier

Weipeng Chen,Muyu Zhang,Bing Qin
Center for Information Retrieval
Harbin Institute of Technology
{wpchen,myzhang,bing.qin}@ir.hit.edu.cn

Abstract

In this paper, we present our supervised learning approach to coreference resolution in ConLL corpus. The system relies on a maximum entropy-based classifier for pairs of mentions, and adopts a rich linguisitically motivated feature set, which mostly has been introduced by Soon et al (2001), and experiment with alternaive resolution process, preprocessing tools,and classifiers. We optimize the system's performance for MUC (Vilain et al, 1995), BCUB (Bagga and Baldwin, 1998) and CEAF (Luo, 2005) .

1. Introduction

The coreference resolution is the task in which all expressions refer to the same entity in a discourse will be identified. As the core of natural language processing, coreference resolution is significant to message understanding, information extraction, text summarization, information retrieval, information filtration, and machine translation.

A considerable engineering efforts is needed for the full coreference resolution task, and a significant part of this effort concerns feature engineering. The backbone of our system can be split into two subproblems: mention detection and creation of entitly. We train a mention detector on the training texts. Once the mentions are identified, coreference resolution involves partitioning them into subsets corresponding to the same entity. This problem is cast into the binary classification problem of deciding whether two given mentions are coreferent. Our system relies on maximum entropy-based classifier for pairs of mentions. Our system relies on a rich linguistically motivated feature set. Our system architecture makes it possible to define other kinds of features: atmoic word and markable features. This approach to feature engineering is suitable not only for knowledge-rich but also for knowledge-poor datasets. Finally, we use the best-first clustering to create the coreference chains.

2. System Description

This section briefly describes our system. First the mention detection is presented. Next, the features which we import are described. Finally, we describled the learning and encoding methods.

2.1 Mention Detector

The first stage of the coreference resolution process try to identify the occurrence of mentions in document. To detect system mention from a test text, we train a mention detector on the training data. We formulate the mention problem as a classification, by assigning to each token in the text a label, indicating whether it is a mention or not. Hence, to learn the detector, we create one training text and derive its class value (one of **b**, **i**, **o**) from the annotated data. Each instance represents the w_i, the token under consideration, and consists of 19 linguistic features, many of which are modeled after the systems of Bikel et al. (1999) and Florian et al. (2004) , as described below.

(1) **Lexical:** Tokens in the windows of three words before and after the target word: $\{w_{i-3}, \ldots, w_{i+3}\}$.

(2) **Capitalization:** Determine whether w_i is IsAllCaP (all the characters of word are capitalized, such as "BBN"), IsInitCap (the word starts with a capitalized character,

Proceedings of the 15th Conference on Computational Natural Language Learning: Shared Task, pages 127–130,
Portland, Oregon, 23-24 June 2011. ©2011 Association for Computational Linguistics

such as “Sally”), IsCapPeriod (more than one characters of word are capitalized but not all, and the first character is not capitalized too, such “M.”), and IsAllLower (all the character of word aren’t capitalized, such as “can”) (see Bikel et al. (1999)).

(3) **Grammatical:** The single POS tags of the tokens in the window of three words before and after the target word{t_{i-3},…,t_{i+3}}.

(4) **Semantic:** The named entity (NE) tag and the Noun Phrase tag of w_i.

We employ maximum entropy-based classifier, for training the mention detector. These detected mentions are to be used as system mentions in our coreference experiment.

2.2 Features

To determine which mentions belong to same entitly, we need to devise a set of features that is useful in determining whether two mentions corefer or not. All the feature value are computed automatically, without any manual intervention.

(1) **Distance Feature:** A non-negative integer feature capture the distance between anaphor and antecedent. If anaphor and antecedent are in the same sentence, the value is 0; If their sentence distance is 1, the value is 1, and so on.

(2) **Antecedent-pronoun Feature:** A Boolean feature capture whether the antecedent is pronoun or not. True if the antecedent is a pronoun. Pronouns include reflexive pronouns, personal pronouns, and possessive pronouns.

(3) **Anaphor-pronoun Feature:** A Boolean feature capture whether the anaphor is pronoun or not. True if the anaphor is a pronoun.

(4) **String Match Feature:** A non-negative integer feature. If one candidate is a substring of another, its value is 0, else the value is 0 plus the edit distance.

(5) **Anaphor Definite Noun Phrase Feature:** A Boolean feature capture whether the anaphor is a definite noun phrase or not. True if the anaphor is a pronoun. In our definition, a definite noun phrase is someone that start with the word “the”.

(6) **Anaphor Demonstrative Noun Phrase Feature:** A Boolean feature capture whether the anaphor is a demonstrative noun or not. True if the anaphor is a demonstrative noun. In our definition, a demonstrative noun is someone that start with the word, such as this, that, those, these.

(7) **ProperName Feature:** A Boolean feature. True if anphor and antecedent both are proper name.

(8) **Gender Feature:** Its value are true, false or unknow. If gender of pair of instance matches, its value is true,else if the value is umatches, the value is false; If one of the pair instance’s gender is unknown, the value is uknown.

(9) **Number Feature:** A Boolean feature. True if the number of pair of instance is matches;

(10) **Alias Feature:** A Boolean feature. True if two markables refer to the same entity using different notation(acronyms, shorthands, etc), its value is true.

(11) **Semantic Feature:** Its value are true, false, or unknown. If semantic class relateness of a pair instance is the same, or one is the parent of other, its value is true; Else if they are unmatch,the value is false; If one of the the pair instance’s semantic class is unknown, the value is unknown.

2.3 Learning

We did not make any effort to optimize the number of training instances for the pair-wise learner: a positive instance for each adjacent coreferent markable pair and negative training instances for a markable *m* and all markables disreferent with *m* that occur before *m* (Soon et al.,2001). For decoding it generates all the possible links inside a window of 100 markables.

Our system integrate many machine learning methods, such as maximum entropy (Tsuruoka, 2006) , Descision Tree,Support Vector Machine (Joachims, 2002) . We compare the result using different method in our system, and decide to rely on maximum entropy-based classifier, and it led to the best results.

2.4 Decoding

In the decoding step, the coreference chains are created by the best-first clustering. Each mention is compared with all of its previous mentions with probability greater than a fixed threshold, and is clustered with the one hightest probability. If none has probability greater than the threshold, the mention becomes a new cluster.

3. Setting and data

3.1 Setting

Our system has participated in the closed settings for English. Which means all the knowledge required by the mention detector and feature detector is obtained from the annotation of the corpus(see Pradhan et al. (2007)), with the exception of WordNet.

3.2 Data

We selecte all ConLL training data and development data, contain “gold” files and “auto” file, to train our final system. The "gold" indicates that the annotation is that file is hand-annotated and adjudicated quality, whereas the second means it was produced using a combination of automatic tools. The training data distribution is shown in Table 1.

Category	bc	bn	mz	nw	wb
Quantity	40	1708	142	1666	190

Table 1: Final system’s training data distribution

In this paper, we report the results from our development system, which were trained on the training data and tested on the development set. The detail is shown in Table 2,3.

Category	bc	bn	mz	nw	wb
Quantity	32	1526	128	1490	166

Table 2: Experiment system’s training data distribution

Category	bc	bn	mz	nw	wb
Quantity	8	182	14	176	24

Table 3: Experiment system’s test set distribution

4. Evaluation

First, we have evaluated our mention detector module, which is train by the ConLL training data. It regards all the token as the candidate, and cast it into the mention detector, and the detector decides it is mention or not. The mention detector’s result is shown in Table4.

Metric	R	P	F
Value	63.6	55.26	59.14

Table 4: Performance of mention detector on the development set

Second, we have evaluated our system with the system mention, and we use the previous mention detector to determine the mention boundary. As follow, we list the system perfomance of using MUC, B-CUB,CEAF (E) , CEAF (M) , BLANC (Recasens and Hovy, in prep) in Table 5 .

Metric	R	P	F
MUC	45.53	47.00	46.25
BCUB	61.29	68.07	64.50
CEAF(M)	47.47	47.47	47.47
CEAF(E)	39.23	37.91	38.55
BLANC	64.00	68.31	65.81

Table 5 :Result using system mentions

Finally, we have evaluated our system with the gold mentions, which mention’s boundary is corect. The system performance is shown in Table 6:

Metric	R	P	F
MUC	50.15	80.49	61.78
BCUB	48.87	85.75	62.62
CEAF(M)	54.50	54.50	54.50
CEAF(E)	67.38	32.72	44.05
BLANC	66.03	78.41	70.02

Table6:Result using gold mentions

Result of system shows a big difference between using gold mentions and using system mentions. In comparison to the system using system mentions, we see that the F-score rises significantly by 4.21- 15.53 for the system using gold mentions. It is worth noting that the F-scorer when using the B-CUB metric, the system using system mention rises 2.12 for system using gold mention. Although this is surprising, in my opinion this correlation is because the mention detection recall more candidate mention, and the BCUB metric is benefit for the mention which is merge into the erroneous chain.

5. Conclusion

In this paper, we have presented a new modular system for coreference in English. We train a mention detector to find the mention's boundary based on maximum entropy classifier to decide pairs of mention refer to or not.

Due to the flexible architecture, it allows us extend the system to multi-language. And if it is necessary, we can obtain other modules to support the system. The results obtained confirm the feasibility of our system.

References

Wee Meng Soon,Hwee You Ng,and Daniel Chung Yong Lim.2001.A machine learing approach to coreference resolution of noun phrases.*Computational Linguistic(special Issue on Computational Anaphora Resolution)*,27(4):521-544

Marc Vilain,John Burger,John Aberdeen,Dennis Connolly,and Lynette Hirschman.1995.A modeltheoretic coreference scoring scheme.In *Proceedings of the 6th Message Understanding Conference,pages 45-52.*

Amit Bagga and Breck baldwin.1998.Algorithms for scoring coreference chains.In *Proceedings of the linguistic Coreference Workshoop at the International Conference on Language Resources and Evaluation(LREC-1998)*,pages 563-566.

Xiaoqiang Luo.2005.On coreference resoluton performance metrics.In *Proceeddings of the Annual Meeting of the North American Chapter of the Association for Computational Linguistics-Human Language Technology Conference(NAACL/HLY-2005)*,pages 25-32

Josef Steinberger,Massimo Poesio,Mijail A.kabadjovb,and Karel jezek.2007.Two uses of anaphora resolution in summarization.In *Information Processing and management,Special issue on Summarization*,pages 1663-1680

Bikel,R.Schwartz,and R.Weischedel.1999.An algorithm that learns what's in a name.Machine Learning,34(1-3):pages211-231

Florian,H.Hassan,A.Ittycheriah,H.Jing,N.Kambhatla, X. Luo,N.Nicolov,and I.Zitouni.2004.A statistical model for multilingual entity detection and tracking.In *Proc.of* HLA/NAACL.

Sameer Pradhan and Lance Ramshaw and Ralph Weischedel and Jessica MacBride and Linnea Micciulla. 2007.Unrestricted Coreference: Identifying Entities and Events in OntoNotes. In *Proceedings of the IEEE International Conference on Semantic Computing (ICSC),* Irvine, CA

Marta Recasens and Eduard Hovy.in prep.BLANC:Implementing the rand index for coreference evaluation.

Yoshimasa Tsuruoka.2006.A simple c++ library for maxium entropy classifiction.Ysujii laboratory,Department of Computer Science,University of Tokyo.

Throsten Joachims.1999.Making large-scale SVM learning practical.In B.Scholkopf,C.Burges,and A.Smola,editors,*Advances in Kernel Methods-Support Vector Learning*.MIT-Press.