Coreference Resolution with Decision Tree

CS224N: Final Project

Stanford University, Spring 2008

Sanghoon Kwak Takahiro Aoyama

sanghoon@stanford.edu taoyama@stanfod.edu

Abstract

Coreference resolution is the task to determine whether two expressions in text refer to the same entity. In this paper, we present an approach to coreference resolution of noun phrases of newswire based on machine learning approach with decision tree. We designed 12 features such as plurality and gender, and modified the C4.5 decision tree builder to generate a decision tree based on our training data. We trained and evaluated our approach on the Automatic Content Extraction (ACE) 2004 dataset and achieved encouraging results.

1. Introduction

Coreference Resolution, which is also known as Anaphora Resolution, is a much heralded topic in natural language processing. Coreference resolution is one of essential techniques used in many areas such as information extraction (IE) and question and answering systems. Thus, the coreference resolution problem is being tackled by many NLP researchers and various approaches have been proposed. For example, in the early 90s, Aone and Bonnet (1993) built a decision tree based on annotated Japanese news articles, focusing on zero-anaphora. Also, in 1995, Mccarthy, Lehnert and RESOLVE improved another decision tree approach that concentrated on business-related data. Recently, Wee Meng Soon et al. developed a machine learning approach for coreference resolution of noun phrases (2001). Moreover, Gildea and Jurafsky designed technique for semantic role labeling for coreference resolution (2005).

In this paper, we first provide details of parsing a large training data from news articles in the ACE (Automatic Content Extraction) data set. We use various features of noun phrases to build this decision tree training set. In the next section, we describe the training and testing results of our parsed data set. For the training, we used the C4.5 decision tree builder that has been modified to suit our objectives.

2. Parser

In this section, I would like to describe the ACE dataset we used for this project and the features of noun phrases that we extracted from the dataset.

2.1. ACE 2004 Data Set

The ACE dataset from the 2004 corpus was used for this project. The ACE data set has been designed to help people to develop automatic content extraction algorithms. The dataset is consisted of separate XML files for each news article, and each XML file provides information for each noun phrases such as position in the article, semantic class, and other coreferent noun phrases. The ACE data set that we used is consisted of 100 news articles from the *Broadcast News Program* and the *Newswire of AP and NYTimes*. The following is a snippet from the XML data file.

```
<entity ID="APW20001018.1350.0453-E1" TYPE="PER" CLASS="SPC">
  <entity_mention ID="1-3" TYPE="NAM" LDCTYPE="NAM" LDCATR="TRUE">
    <extent>
      <charseq START="832" END="856">Texas Gov. George W.
Bush</charseq>
    </extent>
    <head>
      <charseq START="843" END="856">George W. Bush</charseq>
    </head>
  </entity_mention>
  <entity_mention ID="1-4" TYPE="NOM" LDCTYPE="NOM" LDCATR="TRUE">
    <extent>
      <charseq START="859" END="893">the Republican presidential
nominee</charseq>
    </extent>
    <head>
      <charseq START="887" END="893">nominee</charseq>
    </head>
  </entity_mention>
```

<Figure 1. Sample data>

- **ENTITY ID** specifies the ID of all noun phrases that coreferences each other.
- **TYPE** specifies the semantic class of noun phrases under the current entity ID.
- **ENTITY MENTION** specifies a noun phrase under the current entity ID. More than one entity mention means that there are at least one pair of coreferent noun phrases. For example, in the above data, “Texas Gov. George W. Bush” and “the republican presidential nominee” are coreferent.

2.2. Selected Features

Based on this corpus, we extracted and designed 12 features in order to check if the antecedent noun phrase *REi* is coreferent to the noun phrase *REj*. Few of the features include word distance, gender match, plurality match, and so on. The full list of features are listed and discussed in detail below.

- **Distance Feature**

This feature denotes the distance between *REi* and *REj* by the number of sentences that separate the two noun phrases. We extracted this feature by first searching *REi* and *REj* and counting the number of stop marks between them. Since there were cases such as “Oct. 10” and “Mr. Bush” where the stop mark did not actually end a sentence, we handpicked several patterns in which the stop mark should not be considered as the end of the sentence.

- **IsPronoun Feature**

This feature is set to true if a noun phrase is a pronoun. We compared the noun phrase with possible pronouns such as personal pronouns (*he, him, you*), possessive pronouns (*your, her*), and reflexive pronouns (*yourself, herself*). This feature was extracted for both *REi* and *REj*.

- **String Match Feature**

This feature is set to true if REi and REj have the same character sequence.

- **Definite NP Feature**

This feature checks if both noun phrases are definite nouns. We basically check if both noun phrases begin with the word "the".

- **Demonstrative NP Feature**

This feature checks if both noun phrases are demonstrative. If *REj* is a noun phrase which starts with articles *a* or *an*, and demonstrative pronouns (*this, that, these, those*) this feature is set to true.

- **Number Agreement Feature**

This feature checks if both noun phrases are plural or singular. To determine plurality or singularity of a noun phrase, we first check if it is a pronoun. If so, we compare it with the list of known plural and singular pronouns. If the noun phrase is a proper noun, we simply determine its plurality by checking if it ends with "s". Otherwise, we have to determine the noun phrase's plurality by determining the morphological root of the noun. For morphological analysis, we utilized PCKIMMO, an open source software which, given a lexicon, grammar, and rules, determines the morphological root of a word.

- **Semantic Class Agreement Feature**

This feature checks if both of *REi* and *REj* belong to the same semantic class. The class information is directly obtained from the ACE dataset. Semantic classed denoted in the dataset are "person(PER)", "organization (ORG)", "location (LOC)", "geo-political entity (GEO)", "facility (FAC)", "vehicle (VEH)", and "weapon (WEA)."

- **Gender Agreement Feature**

This feature checks if *REi* and REj agree in gender (i.e. both of them are male, or both are female). To determine the gender of a noun phrase, we first check if a noun phrases is a pronoun. If so, we compare it with hand generated list of male and female pronouns. Few heuristics were used for non-pronouns. First of all, we checked if the phrase contained well known male and female references such as "father", "brother", "actress", and so on. Furthermore, we obtained the list of common American male and female first names and checked if the phrase contained any of these names. We stored the name list in a hashmap for fast reference.

- **IsProperNoun Feature**

This feature checks if a noun phrase is a proper noun. This is done by simply checking if the first character is uppercase. If the noun phrase contains multiple words, we check if any of the words is a proper noun. This is valid since even if a word is the beginning of a sentence, the corpus stores it with a lower case first character if it is not a proper noun.

- **Appositive Features**

This feature checks if *REj* is in apposition to *REi*. To illustrate the importance of appositive features, consider the phrase “Bill Gates, the chairman of Microsoft Corp”. In the example, we say *the chairman of Microsoft Corp*., and *Bill Gates* are in apposition. To determine if REj is in apposition to REi, we first check if either phrase is a proper noun. If either phrase is a proper noun, we then check if REi is followed by a comma. If so, we finally check if REj is followed by that comma. Since there are cases such as “Adam, 56, is a …” we allow REj to be few words away from the comma.

- **Alias Features**

This feature tests If *REj* is an alias to *REi*, or vice versa. To determine this feature, we use couple of heuristics. First of all, if RE contains “Mr.” or “Mrs.”, we check if the other word contains words beyond “Mr.” or “Mrs.” For example, if one noun phrase is Mr. Brown and the other phrase is Tom Brown, we regard these two phrases as aliases. We also check for acronyms by checking if a noun phrase has only uppercase characters or has uppercase characters separated by a stop mark. If so, we check if the other noun phrase has the same set of uppercase characters that appear in the same order. For example, “IBM (or I.B.M)” is an acronym for “International Business Machines Corporation”.

We gather above data for each possible noun phrase pair in a news article. To be precise, since we want to know if an antecedent, REi is coreferent to a future phrase REj, we only consider pairs (REi, REj) such that REi’s appears before REj. The position information of these phrases is stored in the ACE dataset. After running this parser in approximately 70 New York Times and Broadcast News Program articles, we were able to construct a training data of approximately 400,000 noun phrase pairs.

3. Decision Tree

With the constructed training data, we trained a decision tree that will help us decide whether to unseen noun phrases are coreferent. To construct a decision tree, we used the open source decision tree builder C4.5. We then constructed a test dataset from unseen articles to test how well our features captured the characteristics of corefernt noun phrases.

3.1 C4.5 Decision Tree

In order to use a machine learning algorithm to learn a classifier based on the feature vectors, we used the C4.5 decision tree builder (Quilan 1993) that is available as an open-source software. A leaf node in the resulting decision tree indicates whether two noun phrases are coreferent or not (0 or 1) and the nodes indicate the features that will further subdivide the decision tree. In building this tree, the C4.5 algorithm examines the difference in entropy that result from choosing a feature when generating sub lists. For example, if the target attribute takes on c different values, then the entropy of S relative to this c classification is expressed as

$$\mathrm{Entropy(S)} = \sum_{i=1}^{c}(-p_i \log_2 P_i)$$

Given the entropy as a measurement of the impurity in a collection of training data set, we can now define a measure, information gain, of the effectiveness of an attribute in classifying the training data.

$$\mathrm{Gain(S, A)} = \mathrm{Entropy(S)} - \sum \frac{|\mathrm{S_v}|}{|\mathrm{S}|}\mathrm{Entropy(S_v)}$$

3.2 C4.5 Decision Rules

The C4.5 tree builder further helped us to analyze our results by providing commonly used and high precision decision rules based on the training data. One example of such rules is, if the string match feature is true and the distance between two phrases is zero, those two nouns are coreferent. We will provide detailed examples and discussion of rules in the following section.

4. Results and Error Analysis

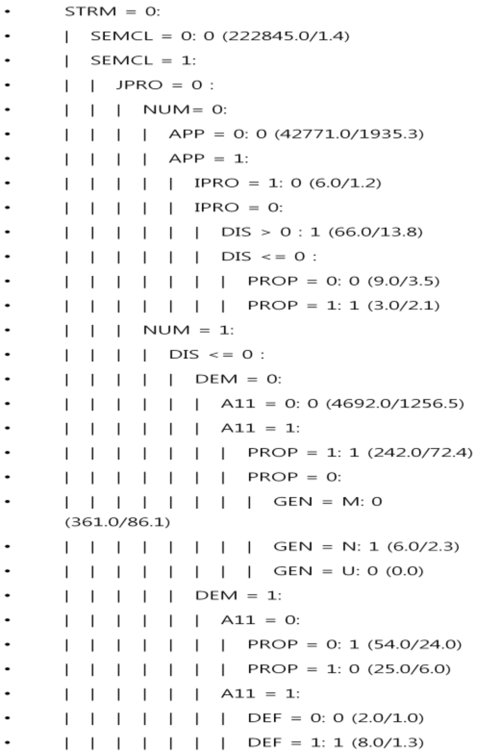
We used ACE 2004 corpus to create a test dataset and tested this data on the decision tree we trained as described above. In the following sections, we provide examples of matched noun phrases, and commonly used rules generated by the decision tree.

4.1 Test Data

We created a test data with unseen 30 New York Times and Broadcast News Program articles. When we generated the test data with the same format as the training data (which does not contain the phrase pairs), we also generated the actual noun phrase pairs in a separate file for future reference. The total number of phrase pairs generated amounted to 200,000. We then modified the C4.5 tree builder to find the corresponding phrase pair in that file indicating whether the decision tree was correct or not. Via this modification, we were able to not only learn the general error rate but also specific phrase pairs that were correctly or incorrectly classified.

4.2. Decision Results

With the above test data, we obtained several results and rules from the C4.5 decision tree algorithm. A snippet of the generated decision tree looks like the following.



```
•       STRM = 0:
•       |   SEMCL = 0: 0 (222845.0/1.4)
•       |   SEMCL = 1:
•       |   |   JPRO = 0 :
•       |   |   |   NUM= 0:
•       |   |   |   |   APP = 0: 0 (42771.0/1935.3)
•       |   |   |   |   APP = 1:
•       |   |   |   |   |   IPRO = 1: 0 (6.0/1.2)
•       |   |   |   |   |   IPRO = 0:
•       |   |   |   |   |   |   DIS > 0 : 1 (66.0/13.8)
•       |   |   |   |   |   |   DIS <= 0 :
•       |   |   |   |   |   |   |   PROP = 0: 0 (9.0/3.5)
•       |   |   |   |   |   |   |   PROP = 1: 1 (3.0/2.1)
•       |   |   |   NUM = 1:
•       |   |   |   |   DIS <= 0 :
•       |   |   |   |   |   DEM = 0:
•       |   |   |   |   |   |   A11 = 0: 0 (4692.0/1256.5)
•       |   |   |   |   |   |   A11 = 1:
•       |   |   |   |   |   |   |   PROP = 1: 1 (242.0/72.4)
•       |   |   |   |   |   |   |   PROP = 0:
•       |   |   |   |   |   |   |   |   GEN = M: 0
        (361.0/86.1)
•       |   |   |   |   |   |   |   |   GEN = N: 1 (6.0/2.3)
•       |   |   |   |   |   |   |   |   GEN = U: 0 (0.0)
•       |   |   |   |   |   DEM = 1:
•       |   |   |   |   |   |   A11 = 0:
•       |   |   |   |   |   |   |   PROP = 0: 1 (54.0/24.0)
•       |   |   |   |   |   |   |   PROP = 1: 0 (25.0/6.0)
•       |   |   |   |   |   |   A11 = 1:
•       |   |   |   |   |   |   |   DEF = 0: 0 (2.0/1.0)
•       |   |   |   |   |   |   |   DEF = 1: 1 (8.0/1.3)
```

<**Figure 2. Decision Trees for learning ACE 2004 Training dataset>**

Each node in the tree indicates a feature and each leaf node indicates whether the test data is coreferent or not. For example "STRM" is the string match feature, "NUM" is the number agreement feature, "ALI" is the alias feature, and so on. Also, the 0 value on the leaf indicates a not coreferent phrase pair while 1 indicates a coreferent pair.

4.2.1 Error Rate

	Training	**Testing**
Items	406400	212865
Error	24762	14213
Rate	6.10%	6.70%

<Table 1. Error Rate>

Table 1 shows that Error rate for the training and testing dataset. One can notice that, based the training set generated using our features, we were able to reach a test accuracy of 93.3%. In other words, given an arbitrary noun phrase pair and their feature values, we can correctly decide whether they are correct or not with an accuracy of 93.3%.

4.2.2 Confusion Rate

Not Coreferent	**Coreferent**
190557	3363
10716	8229

<Table 2. Confusion Matrix>

Table 2 is the confusion matrix based on the test set decision results. This matrix has one row and column for the decision result (coreferent or not coreferent). The number shown in row i, columns j is the number of noun phrase pairs that we classified into class i but whose true class was j. For example, 190447 indicates the number of pairs that we correctly classified as not coreferent. On the other hand, 3363 indicates the number of phrase pairs that we classified as coreferent but which should have been classified as not coreferent.

4.2.3 Rules

As mentioned previously, C4.5 also generates rules that are used with high accuracy. These rules provide up with insight on which features are used in most of the highest accurate rules and which were generally insignificant in deciding whether two phrases are coreferent or not. Following are sample rules generated based on our decision tree. We denote the antecedent as REi and REj as a possible coreferring noun.

-
  ```
  Rule 43:
          DIS > 20
          IPRO = 0
          DEF = 1
          SEMCL = 1
          APP = 1
          ->  class 1  [97.7%]
  ```

This rule classifies nouns as coreferent (class 1) if the antecedent is not a pronoun (IPRO = 0), both are definite nouns (DEF = 1), both are in the same semantic class (SEMCL = 1), REj is appositive to REi (APP = 1), and if REi and REj are separated by more than 20 sentences. This rule has been used on 60 pairs with an accuracy of 97.7%. The high accuracy in this rule indicates that the distance values in our training data might have been erroneous because of reasons described in the distance feature section (2.2).

-
  ```
  Rule 141:
          STRM = 1
          SEMCL = 1
          ->  class 1  [93.4%]
  ```

This rule classifies nouns as coreferent (class 1) if they have the same character sequence (STRM = 1), and are in the same semantic class (SEMCL = 1). This rule has been used in 14531 pairs and obtrained an accuracy of 93.4%. The high accuracy of this rule emphasizes the importance of the semantic class feature in determining coreference. However, in the next section, we will see examples of incorrectly classified pairs that have the same string sequence are in the same semantic class.

-
```
Rule 86:
       DIS <= 0
       JPRO = 1
       NUM = 1
       SEMCL = 1
       ->  class 1  [80.3%]
```

This rule classifies noun pairs as coreferent (class 1) if REj is a pronoun (JPRO = 1), if they agree in number (NUM = 1), are in the same semantic class (SEMCL = 1), and are in the same sentence (DIS <= 0).

-
```
Rule 65:
       IPRO = 1
       STRM = 0
       DIS > 0
       ->  class 0  [98.6%]
```

This rule classifies noun pairs as not coreferent (class 0) if REi is a pronoun (IPRO = 1), if they do not have the character sequence (STRM = 0), and if they are not in the same sentence (DIS > 0).

Based on these rules we were able to find features that play an important role in determining coreference. In rules that classify noun pairs into class 1 mostly rely on the semantic class feature and IsPronoun feature (the antecedent REi should not be a pronoun and REj should). On the other hand, features such as definite and demonstrative nouns seemed to be relatively insignificant in the decision process.

4.3 Error Analysis

To evaluate the algorithm, we should not only know the error rate of the entire test set, but also have to examine the actual noun phrase pairs that were incorrectly classified. In this section, we examine several incorrectly classified phrase pairs and their contexts in the articles they appear in.

- Even if phrases are close enough and agree in number, it is still possible for them to be not coreferent when they appear in different paragraphs.

Example:

*….Vice President Al Gore and **President** Clinton.*

*The Texas governor's words made equally clear that **he** saw himself as the country's best hope for bridging ideological divides,……*

- When two nouns agree in number such as “we” and “them” (or “they” and “their”), and are in the same semantic class (person) it is possible to classify them as coreferent. However, these two nouns are irrelevant.

Example:

they *would ``lead to a renewal of negotiations.''*

…

*In Hebron, thousands of mourners crowded the streets for the funeral of Raed Mohtaseb, 27, chanting ``the blood of the martyrs is calling **us**.''*

- Although to nouns have the same character sequence, and are of the same semantic class, it is possible for them to be not coreferent if they are separated by many sentences.

Example

*``**I**'ll be that president,'' he added, a sentence that was equal parts promise and prediction…..*

…

`I *want you to understand that I can't win without you,'' Bush told a crowd of more than*

- Also, when noun pairs agree in semantic class, and the latter phrase is a pronoun, they can be classified as coreferent (although they do not agree in number). This demonstrates the weight the decision tree places on the semantic class feature and IsPronoun feature.

Example

George W. Bush *captured the votes of most men, whites, conservatives, Republicans, Southerners and white Protestants.*

…

*The category of independents includes respondents who indicated **they** considered themselves ``something else'' in 1972, 1992, 1996 and 2000*

5. Conclusion and Future Work

In this paper, we investigated on how to extract relevant information from a corpus to construct a training data, and also discussed the results obtained from training and testing the dataset with a C4.5 decision tree builder. As noted in the results section, we achieved a reasonably high accuracy (93.3%) on unseen data. Also, since the factor that affects very much on the coreference resolution is semantic class of a noun phrase, increasing the number of classes will give us a more accurate result. Also, we could implement additional heuristics for relatively inaccurate feature values such as distance, alias, and appositives. Furthermore, since NY newswire and the Broadcasting news source are more structured in terms of grammar, it would be more interesting to work on coreference resolution problem in unstructured text such as e-mail messages and web blogs.

6. References

[1] ACE 2004 Multilingal Traiing Data. LDC2005T09, Philadelphia, Penn: Linguistic Data Constorium

[2] Wee Meng Soon* & Hwee Tou Ng t (2001) A Machine Learning Approach to Coreference Resolution of Noun Phrase, *Computational Linguistics*,,27(3):521-544.

[3] Simone Paolo Ponzetto and Michael Strube (2006) Semantic Role Labeling for Coreference Resolution

[4] J Mccarthy and W.G Lehnert,(1995), Using Decision Trees for Coreference Resolution

[5] Mitkov, R. (1998), Robust Anaphora Resolution with Limited Knowledge.

[6] Gildea, D. & D. Jurfsky (2002), Automatic labeling of semantic roles, *Computational Linguistics*,,28(3):245-288.

[7] Nav, V & C. Crdie (2002) Improving machine learning approaches to coreference resolution. In *Proc. of* ACL-02. Pp.104-111.