**Co-reference Resolution Project**

**Introduction :**

The project is based on one of state of art paper for co-reference system *(“A Machine Learning Approach to Coreference Resolution of Noun Phrases” by Soon*) . The paper presents an approach to extract markables and further use machine learning algorithm to train the model to determine, if a pair of markable is co-referent or not. The algorithm defines markables as the union of all the noun phrases, named entities and nested noun phrases found.

**Methodology:**

We are given data which is in a particular defined structure. It has sentences tokenized into phrases and labeled with other meta data . The data is in \*\_conll file format which contain data in a tabular structure. The data structure can be further examined by ReadMe file provided with the data source.

The markables are determined using the pos tag sequence. At times,there arenested markables too.

After markable extraction , we have to get training data for the model ,which will have positive and negative instances. Below example shows how to get training data example.

* ((Eastern Airlines)5 executives)6 notified ( (union)7 leaders)8 that (the carrier)9 wishes to discuss (selective (wage)10 reductions)11 on (Feb. 3)12.
* ( (Union)13 representatives)14 who could be reached said (they)15 hadn't decided whether (they)16 would respond.

For the above example , the positive and negative training examples will be:

|  |  |
| --- | --- |
| +ve examples | -ve examples |
| *((union)7 , (union)13)* | * *((the carrier)9,(union)13)* * *((wage)10,(union)13)* * *((selective wage reductions)11,(union)13)* * *((Feb 13)12, (union)13)* |

Here are the steps performed during the process :

* Read the data file and ReadME file given , to understand the data structure.
* Extract markables , that is noun phrases , using POS tag pattern . At time ,there are nested markables too.
* Store the markables in a file on disk , so to reduce the space complexity on original data file. The file content includes doc\_id, part number , sentence index, start index and end index ,tag pattern, phrase and co-referent number for each markable instance.
* Use the markables file , to extract positive and negative training instances as explained above.
* Extract feature vectors for training instances , which for now is limited to just 6 features but can be increased anytime in future for improving end results. The features used are:
  + Distance Feature (DIST)
  + i-Pronoun Feature (I\_PRONOUN)
  + j-Pronoun Feature (J\_PRONOUN)
  + String Match Feature (STR\_MATCH)
  + Definite NP Feature (DEF\_NP)
  + Demonstrative Noun Phrase Feature (DEM\_NP)
* Feature vectors are then written in .arff file format.
* Weka library is used for training the model. We use J48 decision tree classifier for getting the classification model.
* Testing :
  + In testing phase , we repeat the same steps of reading the test data file and then extracting markables.
  + Further , extract testing data instances and feature extraction for those. It then writes the feature vector in .arff file format.
  + It uses the model ,created in training phrase to get predictions and other accuracy measures.

**Results:**

Here are the results for the testing phase :

Correctly Classified Instances 11054 95.17 %

Incorrectly Classified Instances 561 4.83 %

Kappa statistic 0.4308

Mean absolute error 0.0822

Root mean squared error 0.2124

Relative absolute error 73.0063 %

Root relative squared error 86.6446 %

Total Number of Instances 11615

**=== Detailed Accuracy By Class ===**

**Precision** **Recall** **F-Measure** MCC ROC Area PRC Area **Class**

0.833 0.309 0.451 0.490 0.662 0.328 true

0.955 0.996 0.975 0.490 0.662 0.956 false

Weighted Avg. 0.947 0.952 0.941 0.490 0.662 0.915

**=== Confusion Matrix ===**

a b <-- classified as

230 515 | a = **true**

46 10824 | b = **false**

**References** :

* *A Machine Learning Approach to Coreference Resolution of Noun Phrases by Soon, http://anthology.aclweb.org/J/J01/J01-4004.pdf*