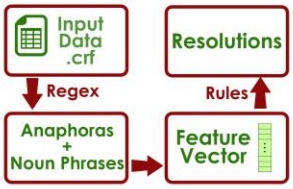
Team Leaders

Sarvagya Shastri Sonam Choudhary

CS-6340

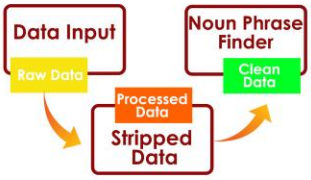
INTRODUCTION

• Extracted all anaphoras given in the input. Other Noun Phrases in the document are identified and arranged according to increasing order of occurence.
• Each anaphora is checked with all possible anaphoras and NPs prior to its occurence.
• Anaphora is resolved by comparing its features with that of all possible antecendents.



DATA PREPROCESSING

• Data is input as a string.
• All the anaphoras are extracted using regular expressions. All the text occuring before first <COREF>, in between first </COREF> and last <COREF>, and text appearing after the last </COREF> are obtined using regular expressions.
• The text other than anaphoras is stripped of any leading/trailing space, brackets and other abnormal symbols.
• This clean data is sent to nltk.regexpparser to find all possible Noun Phrases in the document.

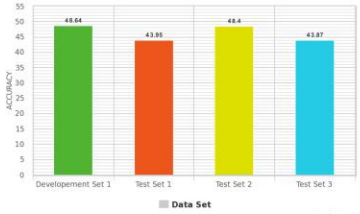


SYSTEM COMPONENTS

• **Feature Vector Generator:** Feature Vector for each item contained - Number, Gender, Pronoun Type, Animacy, Article, Head Noun, Proper Name. Most of the effort made was on this part.
• **Appositive Finder:** Anaphoras existing in appositive structure are identified and resolved.
• **Word Substring:** All anaphoras that are found entirely in an antecedent are resolved to that antecedent.
• **Pronouns Resolver:** Possible pronoun anaphora is primarily checked for its type. Antecendets occuring above it, that are valid for pronoun's type are considered. Resolution is made for antecedent with best match and least distance.
• **Head Nouns Matcher:** Head nouns of anaphora and antecedent are matched. Resolution is done for closest match.
• **Semantic Class:** Head nouns of both the anaphora and possible antecedent are checked for semantic class similarity. Resolution is made for similarity score more than 70%.

From the above components, team member Sarvagya Shastri developed - Feature Vector Generator, Word Substring and Semantic Class. Team member Sonam Choudhary worked on - Appositive Finder, Pronoun Resolver and Semantic Class. We believe equal work was done.

PERFORMANCE



• **Accuracy:** Developement Dataset: 48.64. Test Set 1: 43.95, Test Set 2: 48.4, Test Set 3: 43.87.
• Accuracy as high as 70% was touched in Dev Set 1, 64% in Test Set 2, and 65% in Test Set 3.
• Our System performed better on larger files than on small files.

SUCCESS and REGRETS

• Appositive matcher, Word Substring, Semantic CLass and to an extent Pronouns Resolver worked well in most cases. Head Nouns matcher was unreliable sometimes.

Some of our successful matches are:

• **Semantic Class Matching**:
{'ID':12,'text':'discussions','referrent':[u'9']}->{'ID':9,'text':'talks','referrent':[u'X1803']}
• **Appositive Matching:**
{'ID':67,'text':'president of the North American Securities Administrators Association','referrent':[u'X850']}->{'ID':X850,'text':'James C. Meyer,','referrent':['']}
• **Pronouns Resolver:**
•{'ID':38,'text':'their','referrent':[u'37']}->{'ID':37,'text':'employees','referrent':[u'X253']}

We think references based on gender and animacy were the least successful, mainly because of the difficulty in their determinination. Looking back we can say we could have added some more features to our system to resolve Proper Nouns.

EXTERNAL RESOURCES

• **NLTK:** Regexpparser for Noun Phrase Identification, WordNet, pos_tagger.
• **Datasets:** Male Names, Female Names, Types of Pronouns. www.cs.cmu.edu/Groups/AI/util/areas/nlp/corpora/names/0.html
• **Noun Phrase Coreference as Clustering , Cardie and Wagstaff, EMNLP 2000** was followed partly.