IJAMML 5:1 (2016) 81-98 September 2016
ISSN: 2394-2258
Available at http://scientificadvances.co.in
DOI: http://dx.doi.org/10.18642/ijamml_7100121700

MACHINE LEARNING APPROACH FOR PRONOMINAL ANAPHORA RESOLUTION BASED ON LINGUISTIC AND COMPUTATIONAL FEATURES

Leili Javadpour[a], Mehdi Khazaeli[b] and Gerald M. Knapp[c]

[a]Eberhardt School of Business, University of the Pacific Stockton, 95211, USA

[b]Department of Civil Engineering, University of the Pacific Stockton, 95211, USA

[c]Department of Mechanical and Industrial Engineering, Louisiana State University, Baton Rouge, 70803, USA

Abstract

Anaphora resolution is the problem of resolving references of pronouns to antecedents (previously mentioned noun phrases) in text documents. It is a fundamental preprocessing step in text understanding (semantic) applications, including dialogue and story understanding, document summarization, information extraction, machine translation, and recognizing entailment relations in text. We propose a set of computational and linguistic features to resolve the pronominal anaphora in text documents for a machine learning approach. The system was evaluated on the BBN Pronoun Coreference and Entity Type Corpus, and an F-measure of 89% was obtained. The system was also tested on different genre of document and the performance is compared with the result of the annotated corpus.

---

*Corresponding author.
E-mail address: ljavadpour@pacific.edu (Leili Javadpour).

2010 Mathematics Subject Classification: 91F20, 03B65, 68T50.
Submitted by Johar M. Ashfaque.
Received August 4, 2016

Keywords: anaphora resolution, coreference, pattern recognition, machine learning, natural language processing.

---

1. Introduction

Coreference resolution is the task of resolving all expressions in a text that refer to the same entity, grouping the expressions into chains. Such expressions are often used in writing and speech as shortcuts to avoid repetition. The noun phrase which its interpretation depends upon is called its antecedent. Coreference is ubiquitous in writing and speech. A study of news articles from the Wall Street Journal Corpus found that 30% of nominal expressions (words or phrases functioning as nouns) were anaphoric (Marcus et al. [6]). Coreference resolution is a critical preprocessing step in text understanding (semantic) applications, such as dialogue and story understanding, document summarization, information extraction, machine translation, recognizing and understanding relationships between individuals in a social network, and recognizing entailment relations in text. Past work in automating this task, however, has had limited success and not achieved high levels of accuracy.

For such applications to be successful, it is critical that it be clear who or what is being referred to in the text from sentence to sentence. Coreference resolution poses difficult problems for automated systems, most of which are largely unsolved. In fact, people often have difficulty in resolving complex references. How people resolve pronouns has been extensively studied in both computational, linguistics and psycholinguistics studies.

The most frequent form of coreference is the anaphor and indicates the antecedent precedes the referring expression in the text. The goal of this work is to develop and test an automated system that can find the references of the personal pronouns in the text by using computational and linguistic features.

The methodology proposed in this work analyzes the text and creates feature vector for all the noun phrase and pronoun combinations. The methodology is trained and tested on the BBN Pronoun Coreference and Entity Type Corpus (Weischedel and Brunstein [15]). This corpus contains articles from the Wall Street Journal texts and contains annotation of pronoun coreference, indicated by sentence and token numbers.

2. Literature Review

The most widely known automated coreference resolution systems are summarized in Table 1 (Charniak and Elsner [4]), along with their performance in pronominal resolution on annotated documents of MUC-6 corpus. The systems have different restrictions (e.g., JavaRAP only resolves third person pronouns, GUITAR does not resolve possessive pronouns and BART and OpenNLP resolve all NP anaphora) and different output conventions and therefore direct comparison should be done with caution. However, the comparison does show that performance is still far from sufficient for practical applications, pointing out the need for additional research in this area.

Table 1. Performance comparison between reference resolution systems (source: Charniak and Elsner [4])

Program	% pronouns correctly resolved
BART	< 40
JavaRAP	52.9
GuiTAR	53.4
OpenNLP	59.3

How people resolve pronouns has been extensively studied in both computational studies, linguistics and psycholinguistics studies. Computational linguistics researchers have primarily focused on identifying features for classification (Soon et al. [10]; Ng and Cardie [7]). The feature vector proposed by Soon et al. consists of 12 feature derived based on each potential antecedent and anaphor combination. The

features represent word distance between a pronominal and candidate antecedents, number and gender agreement, string matching and semantic class agreement. The classes are female, male, person, organization, location, date, time, money, percent, object. (Ng and Cardie [7]; Ponzetto and Strube [8]; Versley et al. [14]; Stoyanov et al. [11]) added new grammatical features including parameters that indicate whether one of the elements of the coreference pair is a pronoun, or a definite noun phrase, or a demonstrative noun phrase, or a proper noun to improve the performance.

Others who worked with features for classification used these features and added new semantic and grammatical features to improve the performance (Ponzetto and Strube [8]; Stoyanov et al. [11]).

In the work done by Ponzetto and Strube new features were added which were based on the information extracted from Wikipedia. This considers the Wikipedia pages of the potential antecedent and the potential anaphor, and looks for overlaps between the titles or in the context (Ponzetto and Strube [8]). In another work, WordNet was used to generate more semantic information for refereeing noun phrases together. In this study, the words are searched in WordNet for possible similarities between class, syntax or synonyms (Stoyanov et al. [11]).

Saha et al. [9] worked on developing a system using multi-object optimization techniques for resolution anaphora and showed that optimizing using multiple metrics resulted in higher accuracy. Corry is a coreference resolution that uses a set of 64 features and is able to achieve an average accuracy of around 70% on a set of different corpora (Uryupina [13]).

This research focuses on developing an automated system that can resolve personal pronoun references using linguistic features. The BBN Pronoun Coreference and Entity Type Corpus (Weischedel and Brunstein [15]) was used for supervised training and performance testing. This corpus contains articles from the Wall Street Journal texts and contains annotation of pronoun coreference, indicated by sentence and token numbers. The experimental results show that our proposed approach

achieved competitive resolution with other systems available such as BART (Versley et al. [14]; Charniak and Elsner [4]), our proposed system performed better and also resolved a much wider range of anaphora. We were able to achieve a 89% F-measure on the BBN corpus.

3. Methodology

The methodology uses a rule based approach to resolve pronouns in sentences by first detecting noun phrases (NPs) and pronouns and then generating feature vectors for all the pronoun and NP combinations. Computational and linguistic features are combined in a feature vector of 15 features to train and test the anaphora resolution model.

The BBN Pronoun Coreference and Entity Type Corpus is designed for the purpose of pronominal anaphora resolution and the pronouns and their antecedents are indexed by sentence and token numbers.

After generating the feature vectors, different classification methods are used and the best result is presented.

Figure 1 provides a summary of the distances between pronouns and their antecedent for the documents in the BBN Corpus. As shown the maximum distance between a pronoun and its antecedent was 5 sentences. Although the researches indicate that 90% of antecedents are at most 2 sentences apart from their pronouns, but to ensure that the antecedent is among the NPs selected, NPs that are at most 5 sentences apart from the pronouns are being taken into consideration.

The pronominal pronouns that are considered are subjective (he, she, it, they), objective (him, her, it, them), reflexive (himself, herself, itself, themselves), and possessive (his, hers, its, their, theirs) personal pronouns.

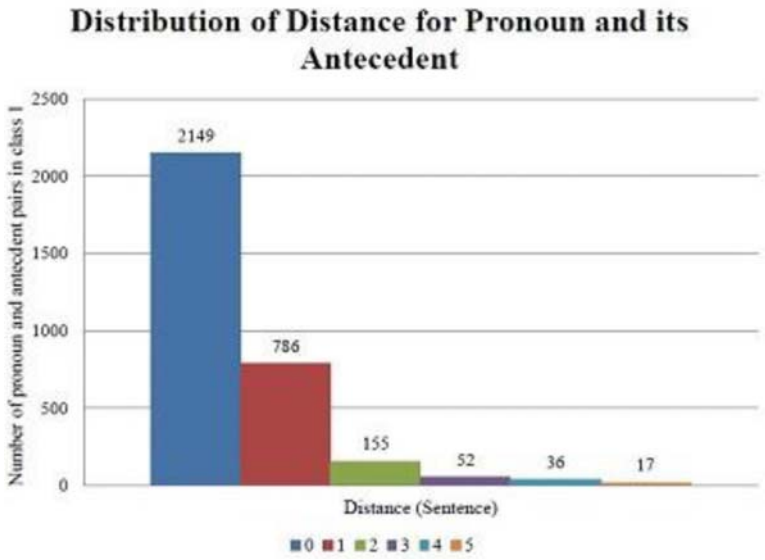

Figure 1. Distribution of distance for pronoun and its antecedent.

For extracting the potential antecedents, the documents is also passed through a series of natural language processors, such as tokenizers, part-of-speech taggers, syntactic parsers, etc. For this purpose, the NLTK library, Stanford Parser and Charniak Parser are used. The output of the Stanford Parser is used to tag all noun phrases in the text. Each noun phrase is a potential antecedent that may have pronouns referring to it. Charnaik Parser[1] and Stanford Dependencies are used to generate a set of features.

3.1. Feature vector

The feature vector consists of a combination of different semantic, grammatical and linguistic features. Some of these features, such as number and gender agreement and the distance features, have been used in other resolution systems. The major difference between the features used here and those implemented in other systems is the use of linguistic rules.

[1]ftp://ftp.cs.brown.edu/pub/nlparser/

Our feature vector consists of a total of 15 features, which are described in the following. The 15 features consist of three main groups. One group is the features that are generated based on both the pronoun and NP (F1, F2, F3, F10, and F12), the other groups are either based on only the pronoun (F6 and F7) or only the NP (F4, F5, F8, F9, F11, F13, F14, and F15). A feature vector is generated for each potential antecedent and pronoun combination (only noun phrases within the last 5 sentences from the pronoun are considered). The information needed for deriving the feature vectors is extracted from the Stanford Parsed tree, Stanford Dependencies, and Charniak Parser clause generation.

Number and gender agreement (F1 and F2): The possible values for number and gender agreement are 0 and 1. The gender and number for the pronoun is selected from Table 2.

Table 2. List of pronouns and their number and gender

Pronoun	Number	Gender
He, him, himself, his	Singular	Male
She, her, herself, hers	Singular	Female
It, itself, its	Singular	Neutral
They, them, themselves, their, theirs	Plural	Neutral

The number and gender features are specified using the following rules:

(1) The noun phrase is first checked for designators of Mr., Mrs., Ms., and Miss. If found number and gender is specified.

(2) In cases where rule #1 does not apply the head noun of the NP is extracted and used for identifying the gender and number. In cases where the NP consists of more than one word the head noun is the rightmost word in the phrase.

(3) The tag of the head noun is first checked and if:

(a) Tag = 'NNP' then number = 'Singular';

(b) Tag = 'NNPS' then number = 'Singular';

(c) Tag = 'NNS' then number = 'Plural';

(d) Tag = 'NN' then number = 'Singular'.

(4) In other cases, the Gender Data Base[2] will be used and the head noun will be queried to find the gender and number. The gender with the most counts in the database will be specified as the gender of the NP. And if the probability of being plural is greater than 50% the number feature will be plural, otherwise singular. In cases where the word is not found the system returns 'NOTFOUND' and later it is added manually.

Distance feature (F3): This feature captures the number of sentences the pronoun and NP are apart and therefore the possible values can be 0, 1, 2, 3, and 4. If the pronoun and NP are in the same sentence the value will be 0.

Proper name feature (F4): For the noun phrase to be a proper name if prepositions such as of and appear in the name they should not be uppercase (Soon et al. [10]). If the noun phrase is a proper name returns 1, otherwise returns 0.

Definite noun phrase feature (F5): A definite noun phrase is a noun phrase that starts with 'the'. Therefore if the noun phrase starts with 'the', returns 1, otherwise returns 0.

Demonstrative noun phrase feature (F6): A demonstrative noun phrase is a noun phrase that starts with one of the demonstrative pronouns this, that, these, or those. If the noun phrase is demonstrative it returns 1, otherwise returns 0.

---

[2]This data was generated by Shane Bergsma from a large amount of online news articles while he was doing an engineering internship at Google Inc. The file contains an alphabetical listing of extracted noun phrases and their gender and number counts. The number of times each noun is connected to a masculine, feminine, neutral, or plural pronoun is specified. This is taken as the gender probability estimate for that noun. In each line, the noun phrase is followed by a tab and then four columns holding the counts for the corresponding gender/number.

http://www.cs.ualberta.ca/ bergsma/gender

The reason behind using F5 and F6 is the Givenness Hierarchy (Webber [12]). When entities are introduced into a discourse by a clause (or other non-nominal expressions), they are accessible to immediate subsequent reference with demonstrative pronouns, but comparatively less accessible to reference with personal pronouns. This can be explained on the basis of the observation that such entities are typically activated, but not brought into focus, upon their introduction to a discourse (Webber [12]).

Features 7-10 (explained in the following) are grammatical features based on the fact that entities evoked from the subject position are considered to be more salient than those evoked from the object position, which in turn are considered to be more salient than those evoked from other grammatical positions such as subordinate clauses or prepositional phrases (Kameyama [5]).

Pronoun having a subject role (F7): The pronoun is checked in the dependencies output of the Stanford Parser and if the tag is NSubj then it returns 1, otherwise returns 0.

Pronoun having an object role (F8): The pronoun is checked in the dependencies output of the Stanford Parser and if the tag is DObj then it returns 1, otherwise returns 0.

NP having a subject role (F9): The head noun of the NP is checked in the dependencies output of the Stanford Parser and if the tag is NSubj then it returns 1, otherwise returns 0.

NP having an object role (F10): The head noun of the NP is checked in the dependencies output of the Stanford Parser and if the tag is DObj then it returns 1, otherwise returns 0.

Pronoun and NP in the same clause (F11): For cases where the pronoun and NP are in the same sentence they are checked to see whether they are in the same clause or not. The clauses are extracted using the Charniak's parser. The raw text is fed into the parser and an annotation indicating the sentence and clauses are returned. If they are the same it returns 1, otherwise returns 0.

NP in the prepositional clause (F12): A prepositional clause is a clause that starts with any of the prepositions such as about, around, since, on, to, etc. The clause in which the noun phrase is part of will be checked and if it's a prepositional clause it returns 1, otherwise returns 0.

Existence of a comma between the pronoun and NP (F13): The sentence is checked and if there is a comma between the pronoun and noun phrase returns 1, otherwise returns 0. This feature only returns 1 in cases where both the pronoun and noun phrase are in the same sentence. Stress on a pronoun is one of the parameters that effect the anaphoric relation (Akmajian and Jackendoff [1]). Pause and stress on a pronoun which can be presented by having commas after the pronoun or having the pronoun in uppercase letters, are parameters that effect the anaphoric relation (Akmajian and Jackendoff [1]; Bolinger [2]).

NP part of a long subordinate clause (F14): A subordinate clause (also known as dependent clause) starts with a subordinate conjunction and contains both subject and verb. Stanford Parser's SBAR tag is used for extracting the subordinate clauses and if the noun phrase is part of a subordinate clause with length of 5 or more words, then it returns 1, otherwise returns 0.

Excitation feature (F15): This feature indicates how much an NP is in focus by taking into consideration the number of times the NP has been mentioned recently. However, one should note that in order to measure the focus on a specific NP in a sentence, we cannot count the number of times it has been appeared since the beginning of the text up to that sentence. This is due to the fact that it might have been in focus for one part but lately the focus has been moved to other NP's. To properly take this fact into account, we incorporate α for getting factor in a formulation. We suggest to use a first order auto regressive (AR(1)) filter which is applied on the numbers of times the NP is appeared in the sentences. We measure an excitation of an NP in a sentence by

$$y[n] = (1 - \alpha)x[n] + \alpha y[n - 1] \text{ for } n = 1, 2, 3, \ldots, N, \quad (1)$$

where n is an index for sentence, N is the total number of sentences in the text, $x[n]$ is the number of times that the NP is appeared in sentence

n, $y[n]$ is the excitation of the NP in sentence n, and α is the forgetting factor. This way an NP that with more in previous sentences is in focus and therefore has a higher chance of being referred to a pronoun than the NP that has not been mentioned.

3.2. Classification model

The feature vectors are input to a trained classification model, which decides whether the pronoun and antecedent candidate NP are coreferent (class = 1) or not (class = 0).

Several classification engines were investigated, including support vector machines using LIBSVM and Naïve Bayes, Random Forest and Bagging classifier. Results of the classification of the methodology are presented in Section 4. An analysis of the methodology is performed to determine the contribution of new features to the classification problem.

4. Results and Discussions

Results of the classification of the methodology are presented in Table 4 and shows promising results. The experiments were conducted on 350 articles from the BBN Pronoun Coreference and Entity Type Corpus. The corpus is randomly divided into 80% of the samples for training and 20% for testing. An analysis of the methodology is performed to determine the contribution of new features to the classification problem.

The confusion matrix result for SVM classifier is also shown in Table 3.

Table 3. Confusion matrix for LibSVM classification

Classified as	0	1
0	2806	389
1	313	2882

Feature analysis was performed by using Chi-square feature selection techniques (Witten and Frank [16]). Chi-square feature ranking is a technique used to calculate the likelihood that a feature is correlated with a class.

Table 4. Classification results

Naive Bayes (time taken to train model: 0.04 seconds)			
Class	**Precision**	**Recall**	**F-Measure**
1	0.879	0.881	0.88
0	0.881	0.879	0.88
All	0.88	0.88	**0.88**
SVM (time taken to train model: 3.51 seconds)			
Class	**Precision**	**Recall**	**F-Measure**
1	0.881	0.902	0.891
0	0.9	0.878	0.889
All	0.89	0.89	**0.89**
Random Forest (time taken to train model: 0.16 seconds)			
Class	**Precision**	**Recall**	**F-Measure**
1	0.879	0.896	0.887
0	0.894	0.876	0.885
All	0.886	0.886	**0.886**
Bagging using SVM classifier (time taken to train model: 43.42 seconds)			
Class	**Precision**	**Recall**	**F–Measure**
1	0.879	0.905	0.892
0	0.902	0.876	0.889
All	0.891	0.89	**0.89**

Table 5 lists the top 10 features and among these features are three of the five new features:

(1) Existence of comma between pronoun and noun phrase when in one sentence.

(2) The number of times the noun phrase have been mentioned in the previous sentences.

(3) Noun phrase being part of a subordinate clause with more than 5 words.

Table 5. Attribute ranking using Chi-squared ranking filter

Chi-square Rank	Feature
3073.1	Distance
1148.9	NPsubj
1087.2	Gender
1007	Proper Name
792.9	Number
409.1	Comma
380.6	Excitation
289	NPobj
48.6	Subordinate Clause
1.35	Definite NP

4.1. Analysis of misclassifications

In this section, the misclassified cases are analyzed. As shown in Figure 2, 69% of the errors were in resolving third person neutral pronouns (it, its, them, they, their, themselves). The remaining 31% of misclassified cases were of male and female pronouns. These pronouns tend to be classified better than third person pronouns since they have specific gender and number. Therefore, further analysis is done to study the reasons that caused these errors.

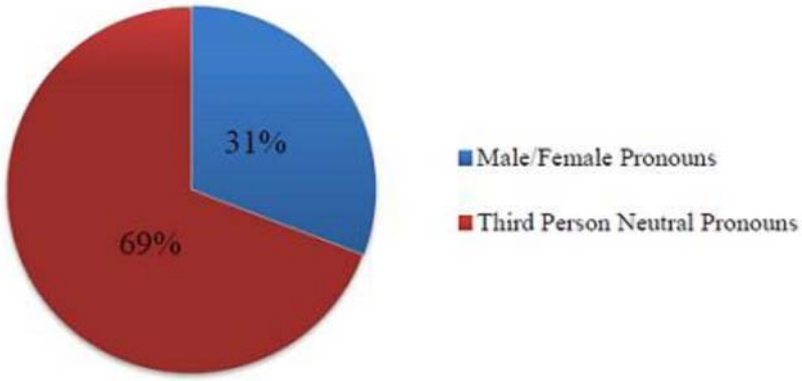

Figure 2. Error classification.

The errors caused by misclassifying third person neutral pronouns were first analyzed to find the main reasons for causing the errors. Figure 3 shows the main groups of errors in resolving third person neutral pronouns.

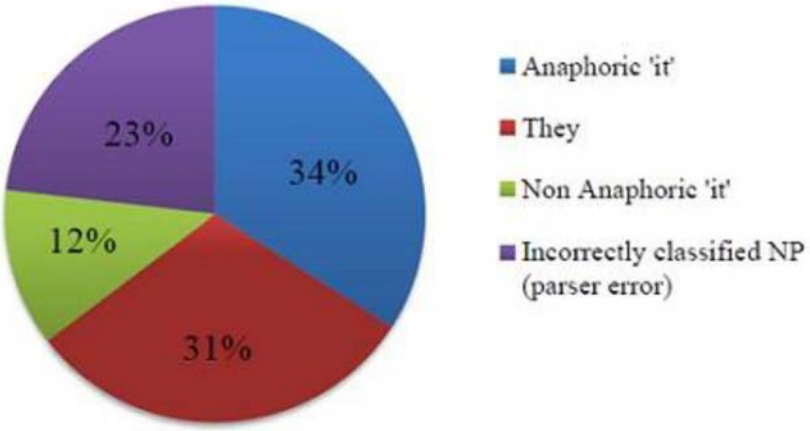

Figure 3. Distribution of errors in resolving third person neutral pronouns.

Anaphoric 'it' and 'they': 34% of the errors occurred when resolving anaphoric 'it' and 31% of the errors occurred when resolving anaphoric 'they'. Errors in this group were caused due to errors in gender agreement. The gender for third person neutral pronouns is always neutral but they can refer to NPs with male, female, and neutral gender.

Non anaphoric 'it': 12% of the errors are caused by resolving non anaphoric 'it'. The system doesn't distinguish between anaphoric and non-anaphoric pronouns and therefore errors are made when trying to find antecedents for these pronouns.

Incorrectly classified NP: This group of errors is caused due to errors in the preprocessing stage. Incorrectly classified NPs from Stanford Parser led to difficulties in generating the feature vector and therefore caused misclassification.

Figure 4 shows the main groups of errors in resolving third person male and female pronouns. When analyzing the errors caused by misclassifying male and female pronouns we discovered that 43% of errors were false negative[3] and the remaining 57% of errors were true positive[4].

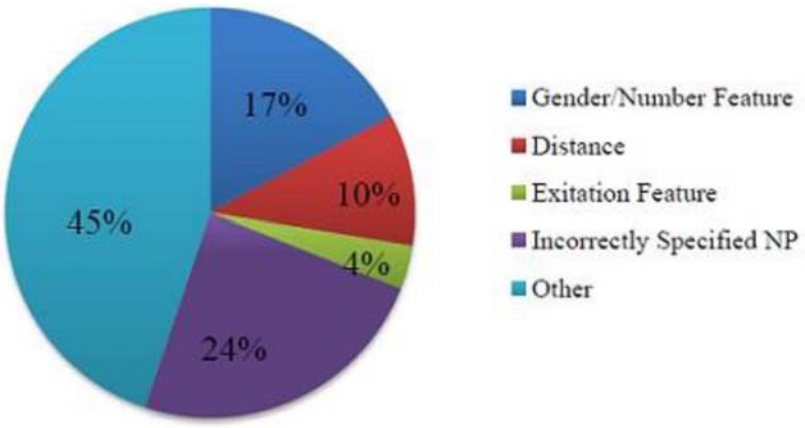

Figure 4. Distribution of errors in resolving male and female pronouns.

Gender/number feature: The errors in true positive group are caused by pronoun and NPs that are in the same sentence and also agree in number and gender but do not refer together. The reason also lies in the fact that distance, number and gender have a high rank in classification.

Distance: The main reason for errors in the false negative group belongs to the pronouns and their relative antecedents that are more than 3 sentences apart. Since distance has the highest Chi score, it plays a great role in classification and therefore when the pronoun and antecedent are more than 3 sentence apart the system does not classify them together.

[3]This group consists of those pronoun and NPs that were a match but the system didn't classify them together.

[4]This group consists of those pronoun and NPs were the system has classified them as a match but and they don't match.

Excitation feature: This group of errors is caused when the excitation feature is very high but the pronoun does not refer to the NP.

Incorrectly classified NP: As explained earlier errors in the parsers used is the reason behind these misclassification data. The features used in this system are mainly those that have proven to help the process of anaphora resolution but as Bosch suggests "there are no structurally stable restrictions on pronoun-antecedent pairs and the grammatical formulae that have been proposed can fail in conditions" (Bosch [3]).

4.2. Time analysis

In this system, first the text is preprocessed and Stanford Parser and NLTK toolkit is used to generate the parsed text. Running Stanford Parser is time consuming especially that we are both generating parsed tree and dependencies from it. Charniak parser is also used to generate the clauses in the text, which is also a timely process.

On the other hand, we are considering the pronoun and NPs that are 5 sentences apart. Therefore processing and generating the data takes time. All the mentioned reasons will cause the system to take time to create the feature vectors. When the features are ready classification is quick. The breakdown of time for each step is shown in Table 6.

Table 6. Breakdown of time for a 21 sentence document

Stage	Time
Preprocessing	3 min 16 sec
Feature generation	1 min 5 sec
Class generation	0 min 56 sec
Classification	0 min 0.1 sec
Total Time	5 min 17 sec

5. Conclusion

Reference resolution task is an important topic and have been addressed in the literature widely, but the existing algorithms for coreference resolution have demonstrated only moderate accurate performance. The reason can be those hard to interpret anaphors which need better knowledge or a better model to be resolved. A learning based and rule based algorithm for detecting pronominal pronouns using computational and linguistic features was developed. The features used in the methodology were proven in theoretical studies but were never tested and used in an automated system. We were able to achieve F-measure of 89%. Results show that by combining the linguistic and computational studies higher accuracy can be obtained.

References

[1] Adrian Akmajian and Ray Jackendoff, Coreferentiality and stress, Linguistic Inquiry 1(1) (1970), 124-126.

[2] Dwight Le Merton Bolinger, Pronouns and Repeated Nouns, Ind. Univ. linguistics Club, 1977.

[3] Peter Bosch, Agreement and Anaphora: A Study of the Role of Pronouns in Syntax and Discourse, Academic Press London, 1983.

[4] Eugene Charniak and Micha Elsner Em, works for pronoun anaphora resolution, In Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics (2009), 148-156.

[5] Megumi Kameyama, Intrasentential centering: A case study, arXiv preprint cmp-lg/9707005, (1997).

[6] Mitchell P. Marcus, Mary Ann Marcinkiewicz and Beatrice Santorin, Building a large annotated corpus of English: The penn treebank, Computational Linguistics 19(2) (1993), 313-330.

[7] Vincent Ng and Claire Cardie, Improving machine learning approaches to coreference resolution, In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics (2002), 104-111.

[8] Simone Paolo Ponzetto and Michael Strube, Exploiting semantic role labeling, wordnet and wikipedia for coreference resolution, In Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, Association for Computational Linguistics (2006), 192-199.

[9] Sriparna Saha, Asif Ekbal, Olga Uryupina and Massimo Poesio, Single and multi-objective optimization for feature selection in anaphora resolution, In Proceedings of the International Joint Conference on Natural Language Processing (2011), 93-101.

[10] Wee Meng Soon, Hwee Tou Ng and Daniel Chung Yong Lim, A machine learning approach to coreference resolution of noun phrases, Computational Linguistics 27(4) (2001), 521-544.

[11] Veselin Stoyanov, Claire Cardie, Nathan Gilbert, Ellen Riloff, David Buttler and David Hysom, Coreference resolution with reconcile, In Proceedings of the ACL 2010 Conference Short Papers, Association for Computational Linguistics (2010), 156-161.

[12] Bonnie L. Webber, Discourse Deixis and Discourse Processing, 1988.

[13] Olga Uryupina, Corry: A System for Coreference Resolution, In Proceedings of the 5th International Workshop on Semantic Evaluation, Uppsala, Sweden, Association for Computational Linguistics (2010), 100-103.

[14] Y. Versley, S. P. Ponzetto, M. Poesio, V. Eidelman, A. Jern, J. Smith, X. Yang and A. Moschitti, BART: A modular toolkit for coreference resolution, In Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Demo Session, Association for Computational Linguistics (2008), 9-12.

[15] Ralph Weischedel and Ada Brunstein, BBN Pronoun Coreference and Entity Type Corpus, Linguistic Data Consortium, Philadelphia, 2005.

[16] Ian H. Witten and Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann, 2005.

■