2013

Resolving pronominal anaphora using commonsense knowledge

Seyedeh Leili Javadpour
Louisiana State University and Agricultural and Mechanical College

RESOLVING PRONOMINAL ANAPHORA USING COMMONSENSE KNOWLEDGE

A Dissertation

Submitted to the Graduate Faculty of the
Louisiana State University and
Agricultural and Mechanical College
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

in

The Interdepartment Program in
Engineering Science

by
Leili Javadpour
B.S., Isfahan University of Technology, 2007
M.S., University of Liverpool, 2009
August 2013

ACKNOWLEDGEMENTS

I would like to take the time to acknowledge those individuals who have been guiding and supporting influences throughout my academic career.

Special thanks and sincere appreciation are extended to my advisor Dr. Gerald M. Knapp for his support, guidance, patience and constructive criticism during the past three and half years of graduate study. I am also thankful to Dr. Jianhua Chen and Dr. Laura Ikuma for being on my committee and for the help with references. I thank Ms. Wendy Luedtke for her encouragement and cooperation.

I would like to thank my fellow graduate students Dr. Ricardo Calix, Soha Arabkhazaeli and Jamie Guidry for the help they gave and the knowledge they shared, and the entire semantic analysis research group at LSU.

I cannot thank Naomi and Mark Valliolahi and their children Amy and Matthew enough for allowing me and Mehdi to be part of their loving family, helping us though rough times, motivating us with their positive energy and celebrating our accomplishments with us. We never felt lonely and they have made our time in Louisiana such an enjoyable memory, which we will always treasure.

My gratitude and appreciation goes to Dr. Merrikh Ramazanian and Dr. Mahmood Sabahi for their generosity and unconditional love. Along the educational path and throughout my life they have always offered words of encouragement and advice. Their kindness will never be forgotten.

The most substantial contributions to my entire education are, of course, from my parents Mahin and Hossein who have sacrificed greatly throughout my life. They have taught lessons of life

with their unconditional love and faith. My appreciation and love goes to my sisters Maria and Roya for the love, motivation, and emotional support that they have always provided.

And last but not least my deepest appreciation and love goes to Mehdi, the best friend, best husband and best team mate that anyone can ask for. He has been by my side throughout this entire process, providing words of encouragement and giving me a shoulder to lean on when I most needed it.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Coreference resolution is the task of resolving all expressions in a text that refer to the same entity. Such expressions are often used in writing and speech as shortcuts to avoid repetition. The most frequent form of coreference is the anaphor. To resolve anaphora not only grammatical and syntactical strategies are required, but also semantic approaches should be taken into consideration. This dissertation presents a framework for automatically resolving pronominal anaphora by integrating recent findings from the field of linguistics with new semantic features.

Commonsense knowledge is the routine knowledge people have of the everyday world. Because such knowledge is widely used it is frequently omitted from social communications such as texts. It is understandable that without this knowledge computers will have difficulty making sense of textual information.

In this dissertation a new set of computational and linguistic features are used in a supervised learning approach to resolve the pronominal anaphora in document. Commonsense knowledge sources such as ConceptNet and WordNet are used and similarity measures are extracted to uncover the elaborative information embedded in the words that can help in the process of anaphora resolution.

The anaphoric system is tested on 350 Wall Street Journal articles from the BBN corpus. When compared with other systems available such as BART (Versley et al. 2008) and Charniak and Elsner 2009, the current system performed better and also resolved a much wider range of anaphora. The system was able to achieve a 92% F-measure on the BBN corpus and an average of 85% F-measure when tested on other genres of documents such as children stories and short stories selected from the web.

CHAPTER 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 that all reference the same entity. Such expressions are often used in writing and speech as shortcuts to avoid repetition. The discourse element (noun phrase) on which a coreference's interpretation depends upon is called its antecedent.

The most frequent form of coreference is the anaphor. Anaphora comes from an ancient Greek word meaning 'the act of carrying back', and indicates the antecedent precedes the referring expression. Less commonly, the antecedent may follow the referring expression, in which case it is called a cataphor. In the following example, there are two anaphoric references: "his"→"Jones" and "the place"→"his country mansion"

> "Jones sold his country mansion. Guess who bought the place?" (Bosch 1983)

Coreferences are ubiquitous in writing and speech. A research paper on 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. 1993). As a consequence, coreference resolution is a fundamental preprocessing step in text understanding (semantic) applications, such as dialog 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. 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 resolving complex references.

This research focused on pronominal resolution, a subset of coreference resolution where the referent expression is a pronoun:

"Jack fell down and broke his crown"

The most widely known reference resolution systems are summarized in Table 1 (Charniak and Elsner 2009), along with their performance in pronominal resolution on annotated document corpora. The systems were tested on different corpora with different writing styles, so direct comparison cannot be made. 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.

How people resolve pronouns has been extensively studied in both computational studies and linguistics and psycholinguistics studies.

Table 1: Performance comparison between reference resolution systems (source: Charniak and Elsner 2009)

Program	**Percent pronouns correctly resolved**	**Authors**
BART	<40	Versley et al. 2008
JavaRAP	52.9	Giu et al. 2004
GUITAR	53.4	Poesio and Kabadjov 2004
OpenNLP	59.3	Morton et al. 2005

Computational linguistics researchers have primarily focused on identifying features for classification (Soon et al. 2001; Ng and Cardie 2002). The feature vector in Soon et al's work are representative, consisting of 12 features and are derived based on each potential antecedent and anaphor combination. The features consider word distance between a pronominal and candidate antecedents, number and gender agreement, semantic class agreement (classes are female, male, person, organization, location, date, time, money, percent, object) and string matching (after removing the articles (a, an, the) and the demonstrative pronouns (this, these, that, those) the

remaining string should match e.g. *the license* and *this license* have the same string match). Other features which have been used in computational studies include 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, and it also considers whether any of the pairs is an alias of the other. Others who worked with features for classification used these features and added new semantic and grammatical features (described in detail in Section 2.1.3) to improve the performance (Ng and Cardie 2002; Ponzetto and Strube 2006; Culotta et al. 2007; Versley et al. 2008; Stoyanov et al. 2010).

Researchers have been working on resolving pronominal pronouns using supervised learning methods (Hobbs 1978; Ge et al. 1998; Tetreault 1999;Mitkov et al. 2002; Strube and Muller 2003; Bergsma and Lin 2006) and unsupervised learning methods (Kehler et al. 2004; Cherry and Bergsma 2005; Charniak and Elsner 2009). Because of the complexity in resolving pronominal pronouns the focus of the researchers were either on resolving personal pronouns (Miltsakaki 2010; Charniak and Elsner 2009), resolving pronouns using the parse tree (Bergsma and Lin, 2006; Yang et al, 2006) or determining non-anaphoric pronouns (Bergsma et al. 2008; Li 2010). There has also been work done on resolving pronouns in other languages such as Chinese (Ning and Jun-Feng 2010; Manjuan and Ping 2010), Korean (Park et al. 2010), Arabic (Abdul-Mageed 2011) and Hindu (Pala and Begum 2011).

Linguistics research has focused attention on defining rules for anaphora resolution. The main constraint for anaphora resolution is the Precede-command (Ross-Langacker) constraint (Langacker 1969; Ross 1967). For a pronoun to refer to a noun phrase (NP) at least one of the following must be fulfilled: (NP(a)=the antecedent, NP(p)= pronoun)

1. NP(a) must precede NP(p)
2. NP(a) must command NP(p)[1]

In the first sentence of the following example, ‘he’ refers to ‘that man’ but not in the second sentence.

That man can sing and *he* can dance.

He can sing and that man can dance.

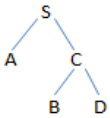
Pause and stress on a pronoun which can be presented by having commas after the pronoun or having the pronoun in uppercase letters, are parameters that affect the anaphoric relation (Akmajian and Jackendoff 1970; Bolinger 1979). Also, if a pronoun is in the preceding clause then the number of words in the subordinate clause (a clause which is not complete by itself)

[1] Constituent-command (c-command) is a relationship between nodes of a tree structure and it is defined as follows (T. Reinhart 1976):

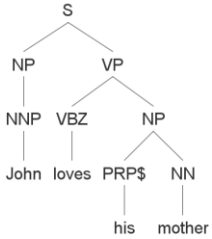
Node A c-commands node B in a tree structure, if:

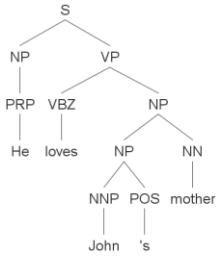
1. Neither A nor B dominates the other (there is no downward path between the two nodes), and
2. The first upper node that most immediately dominates A also dominates B.

For example in the following graph, ‘S’ dominates all other nodes; ‘C’ dominates ‘B’ and ‘D’; ‘A’ commands ‘C’,’B’,’D’; ‘A’ and ‘C’ command each other (Bosch 1983):



As it is shown in the first sentence ‘John’ c-commands ‘his’ but not in the second example:





containing a NP may be of influence for the interpretation of that NP as coreferential (Lakoff 1968). As is shown in the following, 'him' in the first sentence preferably doesn't refer to 'Julius' but likely does in the second sentence due to a longer subordinate clause that the NP is in:

"Martha hit *him* before *Julius* left."

"Martha hit *him* before *Julius* left in his Rolls Royce for a dinner engagement at the Ritz"

(Bosch 1983)

The grammatical roles of the NPs have an impact on their likelihood of being an antecedent. Entities evoked[2] 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 1997).

In the following example 'John' is more salient because is in subject position while 'Tom' is less salient because it is realized in object position. Therefore 'he' is most likely to refer to 'John' than 'Tom'.

John hit Tom. Then *he* ran home.

Little work has been done in transferring the results of linguistics studies to computational models. Although Bosch suggests that there are "no structurally stable restrictions on pronoun-antecedent pairs" and the grammatical formulae that have been proposed can fail in conditions (Bosch 1983), they do provide evidence for resolutions and may be incorporated in an evidence

[2] Evoke covers entities which have been activated (being in current short term memory) and in-focus (not only are in short term memory but also at the current center of attention).

fusion model to improve performance compared to the current state of art systems (summarized in Table 1).

Commonsense knowledge is the routine knowledge people have of the everyday world, and because such knowledge is widely known it is frequently omitted from social communications such as texts. It is understandable that without this knowledge computers will have difficulty making sense of textual information (Liu and Singh 2004). WordNet (Stark 1998) is a widely used semantic resource in the computational linguistics community today (Liu and Singh 2004) and has been widely applied in feature extraction and information retrieval (Soon et al. 2001; Hsu et al. 2008; Szarvas et al. 2011; Hobbs and Montazeri 2011; Li 2010) and also coreference resolution (Ponzetto and Strube 2006), but results show that WordNet alone doesn't give a high performance in a machine learning based coreference resolution system (Ponzetto and Strube 2007). ConceptNet is a more recently developed commonsense knowledge base with a Natural Language Toolkit (NLTK) by which knowledge is extracted and a semantic network produced. It has extended semantic relations and contextual reasoning compared to WordNet and is generated from an open mind commonsense corpus (Liu and Singh 2004). Researchers have recently started using ConceptNet for semantic processing and summarizing (Tonelli and Delmarte 2010; Szarvas et al. 2011) and emotion detection (Lu et al. 2010; Balahur et al. 2011).

In resolving anaphors, people utilize background (commonsense) knowledge to assign the most likely reference. The following examples illustrate this:

a. The soldiers shot at the women and they fell.
b. The soldiers shot at the women and they missed.

(Mitkov 2002)

To resolve these pronouns one must know that those who get shot fall and those who shoot can miss. For computer algorithms to correctly identify the references, they will need to be able to incorporate commonsense knowledge into the resolution process.

People appear to process natural language expressions incrementally, meaning that they will make choices immediately, but may switch interpretation as additional words provide evidence for or against the choice of referent (Fernandez 2011). Recent work has been done in incremental reference interpretation (Poesio and Rieser 2011; Fernandez 2011). There has been considerable work done in dialogue interpretation to explain how expressions are interpreted incrementally in dialogue (Poesio and Traum 1997; Poesio and Rieser 2010). If the information about the possible anaphoric antecedents is not taken into account early on in processing the text, then expensive backtracking becomes necessary.

Researchers have been using linguistic theories (Lobner 1985; Barwise and Perry 1983; Poestio and Traum 1997; Cann et al. 2005) for implementation of their models. Many sources of information might be needed to provide classifications (Bezdek et al. 1999) and this is known as information fusion. The objective of all decision support systems (DSS) is to create a model, which given a minimum amount of input data/information, and is able to produce correct decisions (Ruta and Gabrys 2000).

There are different fusion methods that have been used in different fields. A group of these methods operate on the classifier and try to improve the classification rate by optimizing the classifier. For different data sets, different classifiers might give good results; therefore it can be useful to train a set of different classifiers on a data set and merge their outputs into a combined decision (Kittler et al. 1998; Polikar 2006). Some researchers used weight preferences to

calculate the overall score and select the best method (Lappin and Leass 1994). Another method is to have one main classifier and in cases that the performance is low use the existing auxiliary classifiers along with the main classifier to obtain the ending result (Jia et al. 2009).

1.1 Objectives

The major objectives of this dissertation were:

- Develop a rule-based and feature-based resolution model incorporating latest research from both linguistics and computational science.
- Develop an efficient "Commonsense Knowledge" model for resolution.
- Develop an evidence fusion model combining evidence from the above models into a unified resolution model.
- Evaluate performance of the model against the annotated corpus, as well as against existing state of the art methods and also against different types of document (story, fairytale and news).

1.2 Organization of this Dissertation

This dissertation is organized as follows: Chapter 1 presents the introduction and motivation for the work. Chapter 2 presents the literature review and background for the methods used. Chapter 3 describes an overview of the methodology developed in this work. Chapter 4 describes the classification framework using the computational and linguistic features. Chapter 5 addresses the use of commonsense knowledge in the classification process and proposes an evidence fusion model for resolving pronominal anaphora. Chapter 6 provides a summary of contributions and potential future work.

CHAPTER 2: LITERATURE REVIEW

2.1 Pronominal Anaphoric Resolution Systems

Pronominal anaphoric resolution systems can be categorized in four major groups. The first group consists of the early attempts in this area. These naïve approaches did not exploit knowledge sources, mainly used syntactic information, and were relatively easy to implement. In these approaches the parse tree is searched for the potential antecedents.

The systems in the second category are those that use extensive heuristics rules for determining antecedents. Instead of using syntactic information and relying on the parse tree, these approaches use different weighted factors.

The third group consists of systems that use machine learning approaches. The two major models in the machine learning process are supervised and unsupervised learning methods.

The last group consists of the approaches that focus on the role of semantics in anaphora resolution. Although the progress has been slow, there have been studies done in this area to use semantic information to resolve anaphora.

2.1.1 Naïve Approaches

Hobbs' naïve algorithm (Hobb 1978) is one of the most influential pronoun resolution systems and has been used by many researchers (Lappin and Leass 1994; Baldwin 1997; Mitkov 2002). Hobbs proposed two approaches for pronoun resolution. The first approach is simple and efficient which traverses the surface of the parse tree[3] in a particular order. The algorithm searches for a NP of correct number and gender. The branches of tree are traversed in a left-to-

[3] The parse tree describes the grammatical structure of sentences by having the sentence, verb, noun, etc. specified.

right, breadth-first[4] manner. In his second approach the importance of world knowledge has been taken into consideration and knowledge is available in the form of predicate calculus axioms for semantic analyzing of the texts.

Centering Theory (Grosz et al. 1995) is a family of models that have a representation of the discourse model. The basic idea of this theory is that for having coherence in a discourse, at least one of the entity mentions should have been mentioned earlier in the discourse.

According to Centering Theory, each utterance (U) has a backward looking center, C_b (U), and a set of forward-looking centers, C_f (U). C_b (U_n) represents the entity currently being focused on after U_n is interpreted and $C_b(U_n)$ is a list of all entities mentioned in U_n that can serve as C_b for the following utterance[5] (Brennan et al. 1987).

Left-Right Centering (LRC) algorithm (Tetreault 1999) is a model based on the rules and constraints of Centering Theory, which tries to resolve the lack of incremental processing of the previous models. In this algorithm the search is done for finding the antecedent in the current utterance. If nothing is found then the previous C_f is searched left to right for an antecedent.

2.1.2 Heuristic Approaches

The Resolution of Anaphora Procedure (RAP) is an algorithm (Lappin and Leass 1994) in which there are seven weighted preferences that are used for computing the score of each potential antecedent. The most highly weighted preferences are recency (which is weighted 100) and grammatical function (subject=80, indirect object=40). Each of the potential antecedents is evaluated against the seven preferences and receives a weighted-sum. The system also employs a

[4] A breath-first search of a tree is one in which every node of depth *n* is visited before any node of depth *n+1* (Hobbs1978)

[5] In other words, $C_b(U_{n+1})$ is the most highly ranked element of $C_f(U_n)$

number of constraints such as number and gender agreement, binding requirements and non-anaphoric pronoun detection. The RAP algorithm was evaluated on computer manual text and obtained an overall F-measure of 86%.

Miktov's algorithm has a list of weighted antecedent indicators (Barbu and Mitkov 2001). This list also includes semantic information embedded in a list of indicator verbs such as *'discuss'*, *'identify'* and *'present'*. When a potential antecedent is evaluated, each indicator is matched against the antecedent and the matching ones add their weight to the candidate's score. The aggregate score of each candidate is then compared to decide which potential antecedent is chosen. Beside the indicators, gender and number agreement is also used as a constraint. The only external tool that this algorithm uses is a part of speech (POS) tagger and a noun phrase extractor. The evaluation of Mitkov's approach showed a success rate of 89.7% on a collection of texts with data from Portable Style Writer (PSW) (Mitkov et al. 2002).

Miktov's approach was extended for automatic operation (Mitkov et al. 2002). The additions were a new parser, three new indicators and a non-anaphoric pronoun classifier (for those pronouns that don't have a definite antecedent, e.g. 'it'). The system was evaluated on different groups of texts, including PSW and it gave a success rate of 59.35% when using the non-anaphoric classifier and accuracy of 61.82% when not using the classifier (Mitkov et al. 2002).

2.1.3 Machine Learning Approaches

Machine learning approaches are divided into supervised and unsupervised methods. In the supervised methods the classifier is given a set of labeled data. The system requires both positive (correctly classified data) and negative (Incorrectly classified data) examples, which the positive

examples occur in the labeled training data. The process of supervised learning method is shown in Figure 1.

Unsupervised machine learning algorithms deal with a set of unlabeled data and there is no class information about the instances. The goal is to group them based on their similarities. By applying these unsupervised algorithms, researchers hope to discover unknown, but useful, classes of items (Jain et al. 1999). Unsupervised learning is preferred over supervised learning when labeled data is expensive or difficult to provide.

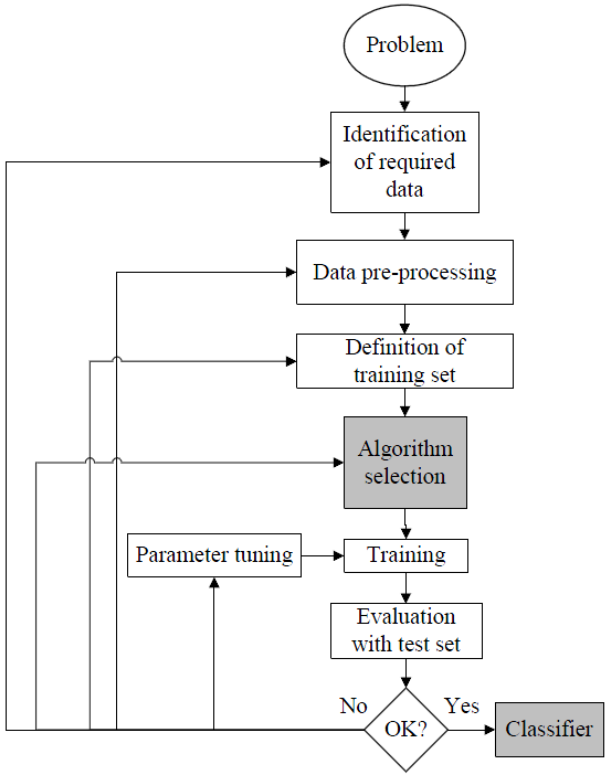


Figure 1: The process of supervised machine learning (source: Kotsiantis 2007)

Soon et al. (2001) presented a learning approach using Hidden Markov Models (HMM) for coreference resolution of noun phrases. The decision tree algorithm uses 12 features. The features include number and gender agreement, part of speech tags, semantic class, alias

resolution, string matching of the head nouns, distance and syntactic features. The training set contains both positive and negative data from a manually annotated dataset. The positive instance contains the anaphor and its closest antecedent and the negative instances are the incorrect entities between these two. Many researchers have been extending the work of Soon et al. Ng and Cardie (2002) extended the work by adding 41 features (summarized in Table 2) which were mostly syntactic features.

Ponzetto and Strube (2006) used WordNet and Wikipedia for adding semantic features. Stoyanov et al. (2010) increased the set of features to 88 and implemented Reconcile (a coreference resolution research platform).

Kehler et al. (2004) implemented a system for pronoun interpretation that is self-trained and uses raw data. The self-trained algorithm uses Maximum Entropy (MaxEnt) with a set of features and a shallow parser.

Cherry and Bergsma (2005), proposed an unsupervised Expectation Maximization (EM) approach to pronoun resolution. The training set consists of (*p, k, C*) triples. Where *p* is the pronoun to be resolved, *k* is the parsed sentence of the pronoun and *C* is the list of all noun candidates. EM's role is to induce a probability distribution over candidates to maximize the likelihood of the (*p, k*) pairs observed in our training set.

Poon and Domingos (2008), presented a system that is based on Markov Logic Network (MLN) (Richardson and Domingos 2006) which is a weighted first-order knowledge base serving as a template to create Markov networks. A cluster-based model is used in their system that considers all the mentions of the name entity as a cluster rather that comparing pairs of antecedents and anaphors.

Table 2: Features for NP coreference based on Ng and Cardie's work (source: Ng and Cardie 2002)

Lexical	PRO_STR	C if both NPs are pronominal and are the same string; else I.
	PN_STR	C if both NPs are proper names and are the same string; else I.
	WORDS_STR	C if both NPs are non-pronominal and are the same string; else I.
	SOON-STR-NONPRO	C if both NPs are non-pronominal and the string of NP_i matches that of NP_j, else I.
	WORD_OVERLAP	C if the intersection between the content words in NP_i and NP_j is not empty, else I.
	MODIFIER	C if the prenominal modifiers of one NP are a subset of the prenominal modifiers of the other; else I.
	PN_SUBSTR	C if both NPs are proper names and one NP is a proper substring (w.r.t. content words only) of the other; else I.
	WORDS_SUBSTR	C if both NPs are non-pronominal and one NP is a proper substring (w.r.t. content words only) of the other; else I.
Grammatical	BOTH_DEFINITES	C if both NPs start with "the;" I if neither start with "the;" else NA.
	BOTH_EMBEDDED	C if both NPs are prenominal modifiers; I if neither are prenominal modifiers; else NA.
	BOTH_IN_QUOTES	C if both NPs are part of a quoted string; I if neither are part of a quoted string; else NA.
	BOTH_PRONOUNS	C if both NPs are pronouns; I if neither are pronouns, else NA.
	BOTH_SUBJECTS	C if both NPs are grammatical subjects; I if neither are subjects; else NA.
	SUBJECT_1	Y if NP_i is a subject; else N.
	SUBJECT_2	Y if NP_j is a subject; else N.
	AGREEMENT	C if the NPs agree in both gender and number; I if they disagree in both gender and number; else NA.
	ANIMACY	C if the NPs match in animacy; else I.
	MAXIMALNP	I if both NPs have the same maximal NP projection; else C.
	PREDNOM	C if the NPs form a predicate nominal construction; else I.
	SPAN	I if one NP spans the other; else C.
	BINDING	I if the NPs violate conditions B or C of the Binding Theory; else C.
	CONTRAINDICES	I if the NPs cannot be co-indexed based on simple heuristics; else C. For instance, two non-pronominal NPs separated by a preposition cannot be co-indexed.
	SYNTAX	I if the NPs have incompatible values for the BINDING, CONTRAINDICES, SPAN or MAXIMALNP constraints; else C.
	INDEFINITE	I if NP_j is an indefinite and not appositive; else C.
	PRONOUN	I if NP_i is a pronoun and NP_j is not; else C.
	CONSTRAINTS	C if the NPs agree in GENDER and NUMBER and do not have incompatible values for CONTRAINDICES, SPAN, ANIMACY, PRONOUN, and CONTAINS PN; I if the NPs have incompatible values for any of the above features; else NA.
	CONTAINS_PN	I if both NPs are not proper names but contain proper names that mismatch on every word; else C.
	DEFINITE_1	Y if NP_i starts with "the;" else N.
	EMBEDDED_1	Y if NP_i is an embedded noun; else N.
	EMBEDDED_2	Y if NP_j is an embedded noun; else N.
	IN_QUOTE_1	Y if NP_i is part of a quoted string; else N.
	IN_QUOTE_2	Y if NP_j is part of a quoted string; else N.
	PROPER_NOUN	I if both NPs are proper names, but mismatch on every word; else C.
	TITLE	I if one or both of the NPs is a title; else C.

(Table 2 continued)

Sematic	CLOSEST_COMP	C if NP_i is the closest NP preceding NP_j that has the same semantic class as NP_j and the two NPs do not violate any of the linguistic constraints; else I.
	SUBCLASS	C if the NPs have different head nouns but have an ancestor-descendent relationship in WordNet; else I.
	WNDIST	Distance between NP_i and NP_j in WordNet (using the first sense only) when they have an ancestor-descendent relationship but have different heads; else infinity.
	WNSENSE	Sense number in WordNet for which there exists an ancestor-descendent relationship between the two NPs when they have different heads; else infinity.
POS	PARANUM	Distance between the NPs in terms of the number of paragraphs.
Other	PRO_RESOLVE	C if NP_j is a pronoun and NP_i is its antecedent according to a naive pronoun resolution algorithm; else I.
	RULE_RESOLVE	C if the NPs are coreferent according to a rule-based coreference resolution algorithm; else I.

The features in this system consist of gender and number agreement, distance measurement and head noun determination. Charniak and Elsner (2009), also used Expectation Maximization for resolving pronoun anaphora using an unsupervised approach. The accuracy for their model is around 68.7%. Their model does not handle cataphora and only allows antecedents to be at most two sentences back.

Feature Based Approaches

The approaches that are based on extracting features mainly start with Soon et al's basic features and add additional features to it. The Soon et al's features are listed below with a description for each (the features are derived based on two extracted NPs, *i* and *j)*:

1. Distance feature (values are 0,1,2,3 …): is the number of sentences *i* and *j* are apart.
2. i-pronoun feature (true or false value): returns true if *i* is a pronoun.
3. j-pronoun feature (true or false value): returns true if *j* is a pronoun.
4. String match feature (true or false value): First the articles (*a, an, the*) and demonstrative pronouns (*this, these, that, those*) are removed and the remaining strings are compared.

5. Definite noun phrase feature (true or false value): Definite noun phrase is a noun phrase that starts with the word *the*. If *j* is a definite noun phrase it returns true.
6. Demonstrative noun phrase feature (true or false value): A demonstrative noun phrase is the one that starts with *this*, *that*, *these*, or *those*. If *j* is a demonstrative noun phrase it returns true.
7. Number agreement feature (true or false value): Returns true if *i* and *j* both agree in number.
8. Semantic class agreement feature (true or false or unknown value): The semantic classes are 'female', 'male', 'person', 'organization', 'location', 'date', 'time', 'money', 'percent' and 'object'.
9. Gender agreement feature (true or false or unknown value): Returns true if *i* and *j* both agree in gender.
10. Both-Proper-Names feature (true or false value): A proper name is based on capitalization. It returns true if *i* and *j* both are proper names.
11. Alias feature (true or false value): Returns true if *i* is an alias of *j* or vice versa.
12. Appositive feature (true or false value): If *j* is an apposition to *i*, returns true.

Ng and Cardie (2002) also extended the features by adding more semantic, grammatical and lexical features. They considered quotations in a text by adding the following features:

1. InQuote1: If the first NP is part of a quoted string returns Y, else N.
2. InQuote2: If the second NP is part of a quoted string returns Y, else N.

Stoyanov et al. (2010) extended the features to 76 and implemented Reconcile (a coreference resolution platform). They used WordNet and extracted the following 5 features:

1. WNSynonyms: If the NPs are WordNet synonyms returns ’C’, else ’I’.
2. WordNetClass: If both NPs have the same WordNet class returns ’C’, else ’I’.
3. WordNetDist: The distance in the WordNet Synset tree between the two NPs.
4. WordNetSense: Returns the first WordNet sense that both NPs share.
5. WordOverlap: If the intersection of the content words of the two NPs is not empty, then ’C’, else ’I’ (Stoyanov et al. 2010).

Ponzetto and Strube (2006) also added 4 new features that 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.

Rule Based Approaches

Most of the methods presented as heuristic methods are considered to be rule based. The most basic rules that are used by researchers are gender and number agreement.

C-command constraint which was explained in chapter 1 is a relationship between nodes of a tree structure and it is defined as follows (T. Reinhart 1976). In the following there are some examples to the general constraint.

In the following sentences the pronoun is in the main clause in direct object position and the potential antecedent is in a prepositional clause or phrase (PP).

a. I saw *him* when *John* got in.
b. I briefly saw *him* when *John* arrived home from hospital.

(Bosch 1983)

In the first sentence there is no anaphoric relationship but in the second sentence there is. The reason is that in the tree structure of the first sentence the PP is part of the verbal phrase (VP), whereas in the second sentence they are independent and are attached to a higher node.

In the following examples the pause that is indicated by a comma has an impact on the referential relations in which in the first sentence ‘he’ doesn’t refer to ‘John’ but in the second sentence it does.

a. *He* lied to me and *John* was my friend.
b. *He* lied to me, and *John* was my friend.

(Bosch 1983)

Chomsky’s binding theory provides a set of syntactic rules for intra-sentential anaphora (Chomsky 1993). Chomsky's binding theory contains three principles which govern the distribution of reflexive and reciprocal pronouns, ordinary pronouns, and full noun phrases. The principles can be described as follows:

1. The antecedent of a reflexive pronoun expression (e.g. each other) can usually be obtained by travelling up the parse tree, searching for the closest clause or noun phrase that has a subject. If the match is successful, the subject is the antecedent.
2. Exactly the opposite of reflexive, a personal pronoun cannot corefer to any entity residing within the clause or noun phrase as identified using the previous rule.
3. A noun phrase cannot be considered as coreferential with a definite description anaphor if its parent phrase also contains the anaphor.

The system proposed by Li (2010) approaches pronominal anaphora resolution with a rule-based algorithm that operates on coreference clusters. Besides gender/number agreement, other major

factors considered by the system include syntax-based salience, guidance provided by the centering theory, and semantic-based restrictions.

2.1.4 Semantic Centric Systems

There have been early approaches incorporating semantic knowledge in the decision process and Dagan and Itai's approach (1990) is one of the earliest automatic ones that apply the knowledge to the coreference resolution. It determines the preference of candidates based on predicate argument frequencies (Yang et al. 2006). The approach collects statistics from a large corpus of tuples (anchor, mention) in which the anchor is a combination of lemma functioning as either a verb or an adjective and a grammatical function of subject-verb, verb-object, or adjective-noun. It then counts the number of occurrences of each of these tuples and then uses a threshold to determine its validity.

Another approach is an extension to Lappin and Leass' (1994) RAP system and is called RAPSTAT. In this approach the scores from the RAP is examined and if the lexical statistics strongly suggest otherwise the decision is overridden. Using this approach there was 2.5% improvement in accuracy.

Bean and Rillof's (2004) approach is a supervised machine learning system which populates its knowledge from a training set in a certain domain and by applying this knowledge is able to resolve coreference in the same domain.

Bergsma and Lin's (2006) approach focuses on finding coreference and non-coreference paths. These are the paths that usually lead to coreferential/non-coreferential mentions of the two words. In a simple way, the paths are learnt by scanning a large corpus for dependency paths that have pronouns attached to both ends. If the pronouns are the same then the path is marked as

likely being coreferent, otherwise it is marked as non-coreferent. Their system showed an accuracy of 71.6% over third person pronouns on MUC-7 data set.

2.2 Use of Commonsense in Language Processing Systems

Most of the reference resolution systems rely of shallow features, such as the distance between the coreference expressions, string matching and so on. But the relevance of world knowledge and inference for reference resolution systems is something that should be taken into consideration (Charniak 2000). Recently researchers have been working on incorporating semantic knowledge in reference resolution. Ponzetto and Strube (2006) investigated the use of WordNet and Wikipedia taxonomies for extracting semantic similarity and relatedness measures between NPs in the coreference chain. They used Maximum Entropy (MaxEnt) for learning their model and for preventing over-fitting their model they used Gaussian as a smoothing method. Coreference resolution is viewed as a binary classification task in which given a pair of words the classifier has to decide whether they corefer or not. The MaxEnt model produces a probability for each candidate pair, taking into consideration the context in which the candidates occur. WordNet features were able to improve an average of 10% the accuracy rate for common nouns on a set of datasets, whereas using Wikipedia features resulted in smaller improvements in accuracy on the same datasets.

Li designed a system that queried documents on the web to obtain the probability distribution of a word's gender and also offered a web-based method for detecting non-anaphoric 'it' which eliminated up to 4% of errors in the anaphora resolution system (Li 2010).

WordNet::Similarity is freely available software that returns the similarity measures and relatedness between a pair of concepts (Pedersen et al. 2004). These measures are based on the

structure of WordNet. Measures of similarity are based on the information in *is-a* hierarchy of concepts and can only be used for words in the same part of speech boundaries. Measures of relatedness are more general and can be used for words across part of speech boundaries and are not limited to *is-a* relations.

Kim and Baldwin incorporated the WordNet::Similarity software on Penn Treebank corpus and proposed a methodology to classify the test noun compounds in a text and were able to achieve 53.3% accuracy (Kim and Baldwin 2005). Noun compounds are made up of two or more words. The rightmost word is the head noun and the remaining words are the modifiers. Kim and Baldwin worked on recognizing the semantic relationship between the head noun and the modifiers using WordNet.

Nard's dissertation focuses on a methodology for resolving coreference using sematic constraints (Nard 2012). He used the interpretation of knowledge to identify the antecedents. WordNet is used to extract more information for the document that can help in anaphora resolution process. In this process a multi pass approach was implemented in which if an anaphor is difficult to resolve it would be transferred to a semi resolved state and would be analyzed later, instead of immediately making a decision based on the current information. He was able to achieve a precision rate of 78%.

Bundanitsky and Hirst used the similarity measures in WordNet to detect and correct spelling errors in real-word examples (Budanitsky and Hirst 2006). They used both measures of similarity and measures of relatedness in their methodology.

Spagnola and Lagoze studied the usefulness of additional pathway variables such as edge type and user-rated scores in ConceptNet (Spagnola and Lagoze 2011). The experiments show an

improved performance in conceptual similarity calculations in ConceptNet when the new features were added.

2.3 Use of Evidence Fusion Models

Often many sources of information are combined to achieve high classification accuracies (Bezdek et al. 1999; Bezdek et al. 2005). This strategy has been called data fusion, information fusion, multistage classifier design, classifier fusion, or sensor fusion. The main idea of fusion approaches is that the results of multiple sources of information are combined together to reach a better final decision than any component classifier.

The use of information fusion techniques have increased in recent years. The goal of using these techniques is to get better results by combining the existing well performing methods. Different methods produce different errors on different data and by assuming that the individual methods perform well, combing them should reduce the overall classification error and return correct outputs (Ruta and Gabrys 2000). Fusion of information can be categorized in three levels that are connected with the classification process: data level fusion, feature level fusion and classifier fusion (Bezdek et al. 1999; Bezdek et al. 2005).

Data level fusion involves combining sensor outputs directly. Feature level fusion is much more general and directly takes advantage of the ability of different sensors to measure complementary information. This level of fusion involves combining multi-dimensional, quantitative feature vectors derived from sensor measurements, possibly together with qualitative information. Classifier fusion is generally considered to be at a higher level, such as combining the outputs of several classifiers. The basic assumption is that the classifier algorithms are imperfect and

therefore one way of enhancing the performance of the classification system is to construct multiple independent systems and then combine the results.

Eisenstein et al. (2008) worked on incorporating gestures of the speaker in coreference resolution. They presented a fusion model that learns to predict which gestures are salient for this purpose (Eisenstein et al. 2008). Danesh et al. (2007) also used Naïve-Bayes, KNN and Rocchio[6] as their classifiers and combined them using fusion methods for text classification (Danesh et al. 2007). The work done by Jia et al. focuses on fusing multiple classifiers for text categorization. They used one main classifier and set up a certain number of classifiers as the auxiliary classifiers. If the results of the main classifier were credible the other classifiers wouldn't be used, otherwise the results of the auxiliary classifiers along with the main classifier will give the last decision (Jia et al. 2009).

2.4 Corpora

There are many corpora available for natural language processing and reference resolution. Message Understanding Conference (MUC) 6 (Chinchor and Sundheim 2003) contains 318 annotated Wall Street Journal articles. The MUC corpora are mainly used for information extraction. BBN Pronoun Coreference and Entity Type Corpus (Weischedel and Brunstein 2005) supplements the one million word Penn Treebank corpus of Wall Street Journal texts and contains manually annotation of pronoun coreference that is indicated by sentence and token number. Automatic Content Extraction (ACE) (Mitchell et al. 2004) is used for developing extraction technology for automatic processing of language data. The Treebank (Marcus et al. 1993&1999) is another corpus that is widely used for NL processing, parsing and tagging.

[6] Rocchio is the classic method for text classification in information retrieval.

Brown Laboratory for Linguistic Information Processing (BLLIP) 1987-89 WSJ Corpus Release 1 (Charniak et al. 2000) contains a Treebank-style parsing of Wall Street Journal collection. The part of speech (POS) tagging was done using statistically-based methods developed by Charniak et al.

2.4.1 Knowledge Corpora

Knowledge corpora refer to databases that include information about words or concepts. There are several implementations such as: ConceptNet (Havasi et al. 2007), Opencyc, WordNet (Miller et al. 1990) and WordNet-Affect, FrameNet (Fillmore et al. 2002) and YAGO (Suchanaek et al. 2007).

ConceptNet gathers commonsense knowledge thorough ordinary people in its site. The data is represented in the form of semantic network and is available for use in natural language processing. It also has a Python implementation which gives access to a copy of ConceptNet database. The version of ConceptNet 3.0, for instance, contains over one million assertions collected by human annotators from the World Wide Web. And as it is shown in Table 3 the semantic relations are embedded in different categories.

The format of output produced by ConceptNet for a word such as “cat” is shown in Figure 2.

ConceptNet represents the information as a directed graph (Figure 3). The nodes of the graph are the concepts and the labeled edges are assertions of commonsense that connect two concepts. Each assertion is associated with a frequency value that defines whether people said that the relationship is sometimes, generally or always true. The frequency value can also be negative defining that the relationship in rarely or never true.

Table 3: ConceptNet semantic relations

Category	Semantic relations
Things	Is A Property Of Has Property Part Of Made Of Has A
Events	First Sub Event Of, Last Sub Event Of Has Prerequisite Event For Goal Event Event For Goal State Event Requires Object Has Sub Event
Actions	Effect Of Effect Of Is State Capable Of Receives Action Causes
Spatial	Often Near Location Of At Location
Goals	Desires Event Desires Not Event Motivated By Goal
Functions	Used For
Generic	Can Do Conceptually Related To

AtLocation (cat, lap)
AtLocation (cat, bed)
AtLocation (cat, windowsill)
CapableOf (cat, hunt mouse)
CapableOf (cat, eat mouse)
CapableOf (cat, drink water)
CapableOf (cat, corner mouse)
HasA (cat, four leg)
HasA (cat, whisker)
HasA (cat, fur)
IsA (cat, carnivore)

Figure 2: ConceptNet selected output for "cat"

Since the information is gathered from humans, the system needs to handle noise and incorrect information and imprecision. Therefore AnalogySpace process was developed that represented the knowledge in a semantic network as a sparse matrix. The concepts are along one axis and the

features along another axis. By using singular value decomposition (SVM) the dimensionality is reduced and the result represents the most salient aspects of the knowledge (Speer et al. 2008).

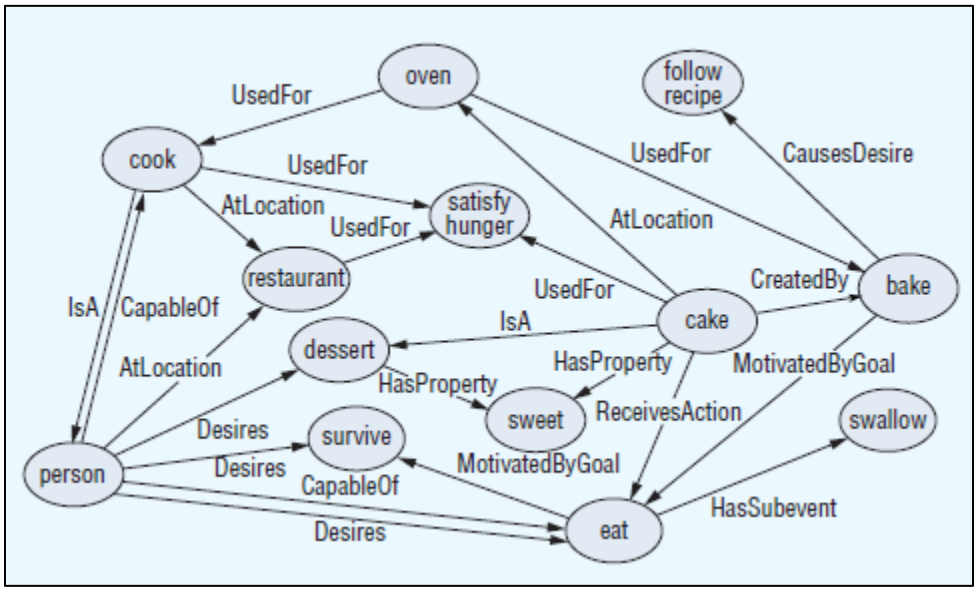


Figure 3: ConceptNet diagram for "cake"(Source: Speer et al. 2008)

WordNet, which is a large lexical database of English, was first introduced at Princeton University. The main relationship between words is synonymy. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations. The format of output produced by WordNet for a word such as “dog” is shown in Figure 4.

```
>>> wn.synset('dog.n.01')
Synset('dog.n.01')
>>> wn.synset('dog.n.01').definition
'a member of the genus Canis (probably descended from the common wolf) that has been domesti
>>> wn.synset('dog.n.01').examples
['the dog barked all night']
>>> wn.synset('dog.n.01').lemmas
[Lemma('dog.n.01.dog'), Lemma('dog.n.01.domestic_dog'), Lemma('dog.n.01.Canis_familiaris')]
>>> [lemma.name for lemma in wn.synset('dog.n.01').lemmas]
['dog', 'domestic_dog', 'Canis_familiaris']
>>> wn.lemma('dog.n.01.dog').synset
Synset('dog.n.01')
```

Figure 4: WordNet selected output for "dog"

The majority of the WordNet's relations connect words from the same part of speech (POS). Therefore, WordNet consists of four main parts, nouns, verbs, adjectives and adverbs, with few cross-POS pointers (Fellbaum 2010; Miller 1990). Cross-POS relations include the links that hold among semantically similar words sharing the same meaning. For example, observe (verb), observant (adjective), observation and observatory (nouns).

FrameNet has been in operation in Berkeley since 1997 and is a lexical database of English. The basic idea is that the meanings of most of the words can best be understood on the basis of a semantic frame[7]. The database has three major components which are lexicon, frame database and the annotated example sentences. Lexicon is composed of entities which are linked to a dictionary-type data, formulas for capturing the morphosyntactic ways that elements of the semantic frames can be realized, semantically annotated example sentences, frame database and other resources such as WordNet. Frame database contains descriptions of each frame's structure. Annotated example sentences are those that are marked up to exemplify the semantic and morphosyntactic properties of the lexical items (Baker et al. 1998).

YAGO2 is a huge semantic base, derived from Wikipedia, WordNet and GeoNames; which was originally introduced by Suchanek et al. 2007. YAGO2 has knowledge of more than 10 million entities (such as persons, organizations, cities, etc.) and contains more that 80 million facts about these entities. The different relations are listed in Table 4.

The approach developed here gathers and integrates temporal, spatial and semantic information from Wikipedia, WordNet and GeoNames. The extractors also catch keywords associated to

[7] A description of a type of event, relation, or entity

entities from the Wikipedia articles and incorporate multilingual information (Hoffart et al. 2009).

Table 4: Relations specified in YAGO2

Acted In	Happened In	Has Expenses	Has ISBN	Has Pages	Has Website	Is Interested In	Plays For
created	Happened On Date	Has Export	Has Imdb	Has Population	Has Weight	Is Known For	produced
Deals With	Has Academic Advisor	Has Family Name	Has Import	Has Population Density	Has Won Prize	Is Leader Of	Started On Date
Died In	Has Area	Has GDP	Has Inflation	Has Poverty	Holds Political Position	Is Located In	Subclass Of
Died On Date	Has Budget	Has Gender	Has Latitude	Has Preferred Meaning	imports	Is Married To	type
directed	Has Capital	Has Gini	Has Longitude	Has Preferred Name	In Language	Is Politician Of	Was Born In
edited	Has Child	Has Given Name	Has Motto	Has Revenue	influences	Lives In	Was Born On Date
Ended On Date	Has Currency	Has HDI	Has Musical Role	Has TLD	Is Affiliated To	means	Was Created On Date
exports	Has Duration	Has Height	Has Number Of People	Has UTC Offset	Is Called	owns	Was Destroyed On Date
Graduated From	Has Economic Growth	hasICD10	Has Official Language	Has Unemployment	Is Citizen Of	Participated In	Works At
Wrote Music For							

2.5 Feature Selection Techniques

There are many feature extraction methods in machine learning. The approaches consist of those which take a set of features and try to map them to a new set of transformed features (such as Principal Component Analysis) and those which try to reduce the number of features without transforming the original set of features (such as chi-square ranking) (Witten and Frank 2005).

Some of the main feature reduction techniques used in literature include Document Frequency (DF), Information Gain (IG), Mutual Information (MI) and Chi-Square statistic.

DF is a method based on the number of documents in which the term occurs. The basic assumption of this method is that rare terms are either non informative or not influential. DF is the simplest technique for vocabulary reduction. However it is typically not used for cases where low DF-terms are assumed to be relatively informative and therefore should not be removed aggressively (Yang and Pederson 1997).

IG measures the number of bits of information obtained for category prediction by knowing the presence or absence of a term in a document (Yang and Pederson 1997).

MI is a method that calculates the two-way contingency between words and categories. A weakness of this method is that the score is strongly influenced by the marginal probabilities of terms. For terms with an equal conditional probability, rare terms will have a higher score than common terms (Yang and Pederson 1997).

Chi-square statistics measures the lack of independence between terms. The major difference between Chi-square and MI is that Chi-square values are normalized and therefore comparable across terms for the same category (Yang and Pederson 1997).

Feature selection with chi-square keeps the original features without transformation. This allows for insights into the types of features that help for a particular prediction task. Additionally, feature selection techniques like chi-square ranking are less computationally expensive than other approaches such as Principal Component Analysis (PCA) and can therefore be performed quickly.

2.6 Overview of Machine Learning Approaches

Machine Learning (ML) is essential for automated systems to make decisions. This section mentions the most important methodologies currently being used in the area of machine learning. Machine learning approaches can be divided into supervised and unsupervised learning methods. The methods presented in this dissertation focus on supervised learning techniques.

2.6.1 Classifiers

Classifiers are machine learning approaches that given a set of input features produce a specific class as an output. Important classifiers include Support Vector Machines (Burges 1998) commonly implemented using LibSVM (Chang and Lin 2001) and Naïve Bayes, decision trees, random forests, and the k-nearest neighbor classifier (Witten and Frank 2005). A list of different classification techniques and their characteristics are given in Table 5.

Table 5: Machine Learning Techniques (Source: Calix 2011)

Technique	Definition	Pros	Cons
Support Vector Machines	Supervised learning approach that optimizes the margin that separates data.	SLT Confidence Characteristic (expected risk) Can handle non-linearly separable data	Class imbalance issues
Decision Trees	This method performs classification by constructing trees where branches are separated by decision points.	Easy to understand	Not flexible
Random Forest	This method performs classification by constructing a number of decision trees and returning a final class by combining the outputs of the individual trees.	Runs efficiently on large data bases	Doesn't over fit Slow process time
Neural Networks	Model represents the structure of the human brain with neurons and links to the neurons.	Versatile	Can obscure the underlying structure of the model
K Nearest Neighbor (KNN)	This method classifies the objects based on closest how close they are to training examples in feature space	Doesn't require parameter tuning	Simple Slow process time
Linear Discriminant Analysis (LDA)	Creates linear function of features to classify data	Simple yet robust classification method	Normality assumptions of the classes

(Table 5 continued)

Gaussian Mixture models (GMM)	This probabilistic method represents signals as weighted sums of normal distributions	Can be used to represent non-normal distributions	Initialization is important for optimization
Naïve Bayes	Probabilistic Learning to calculate the probability of seeing a certain condition in the world selecting the most probable class given the feature vector	Fast, easy to understand the model	Bayes assumptions of independence
Maximum Likelihood Estimation (MLE)	Calculates the likelihood that an object will be seen based on its proportion in the sample data	Simple	Too simplistic for some applications
Expectation Maximization	Similar to MLE but is used when there is missing data in the training set	Very useful when missing data	Too simplistic
Hidden Markov models (HMM)	A Markov Chain is a weighted automaton consisting of nodes and arcs where the nodes represent states and the arcs represent the probability of going from one state to another.	Probabilistic. Good for sequence mining	Combinatorial complexity/ needs prior knowledge
Bayesian Networks	Probabilistic networks	Graphical representation Improves understanding	Requires knowledge of probabilities
Bootstrap Aggregation (Bagging)	This classification method divides the training set in multiple samples and each set is used to train a different component classifier[8].	High performance	Usually applied to decision tree models

[8] All HMMs, all neural networks and all decision trees

CHAPTER 3: APPROACH

Pronominal anaphora resolution in text requires multiple stages of preprocessing. This chapter provides an overview of the methodologies used in this dissertation and how they fit together.

Chapter 4 focuses on identifying an affective feature set for accurately detecting pronouns in text. For this purpose, studies done in computational, linguistics and psycholinguistics studies were analyzed and a set of new features identified for use in the automated resolution processing. The process of feature extraction requires different stages of preprocessing which are explained in detail in this chapter.

Chapter 5 focuses on the use of commonsense knowledge in detecting pronouns in text. ConceptNet and WordNet are used to generate similarity measure features between each pronoun and NP. An evidence fusion model is presented that combines the information from the previous stages in one system.

Each chapter includes experimental results, discussion and conclusion for the proposed methodologies.

3.1 Tools

The following tools are used for performing the tasks required in this methodology:

Python 2.6 and NLTK (Bird et al. 2009); Matlab; Stanford Parser (De Marneffe et al. 2006); Charniak Parser (Charniak 2000); ConceptNet (Havasi et al. 2007); WordNet (Miller et al. 1990); Machine Learning with LibSVM; WEKA (Witten and Frank 2005) and Gender Database (Bergsma and Lin 2006).

3.2 Corpus

The corpus used is the BBN Pronoun Coreference and Entity Type Corpus (Weischedel and Brunstein 2005). The corpus contains two components:

- Pronoun coreference: Pronoun coreference of the Wall Street Journal (WSJ) corpus (an example of the input text is shown in Figure 5). Pronouns and antecedents are indexed by sentence and token numbers as shown in Figure 6.
- Entity types: The corpus includes annotation of 12 named entity types (Person, Facility, Organization, GPE, Location, Nationality, Product, Event, Work of Art, Law, Language, and Contact-Info), nine nominal entity types (Person, Facility, Organization, GPE, Product, Plant, Animal, Substance, Disease and Game), and seven numeric types (Date, Time, Percent, Money, Quantity, Ordinal and Cardinal). Several of these types are further divided into subtypes. Annotation for a total of 64 subtypes is provided.

```
(WSJ0006
        S1: Pacific First Financial Corp. said shareholders approved its acquisi
tion by Royal Trustco Ltd. of Toronto for $ 27 a share , or $ 212 million .
        S2: The thrift holding company said it expects to obtain regulatory appr
oval and complete the transaction by year-end .
)
```

Figure 5: Example of sentences of WSJ

```
(WSJ0006
    (
        Antecedent -> S1:1-4 -> Pacific First Financial Corp.
        Pronoun -> S1:8-8 -> its
    )
    (
        Antecedent -> S2:1-4 -> The thrift holding company
        Pronoun -> S2:6-6 -> it
    )
)
```

Figure 6: The pronoun coreference output file

For easier use of the pronoun coreference output, one vector is specified for each pronoun-antecedent pair, which follows the format shown in Figure 7. An algorithm in Python is written which takes the pronoun coreference information (Figure 6) and returns it in the format shown in Figure 7. The corpus is for both training and testing processes of the methodology.

```
0001
0002
0003
Antecedent -> S1:2-2 -> form :: Pronoun -> S1:27-27 -> it
Antecedent -> S2:3-3 -> fiber :: Pronoun-> S2:21-21 -> it
Antecedent -> S2:3-3 -> fiber :: Pronoun-> S2:11-11 -> it
Antecedent -> S3:1-2 -> Lorillard Inc. :: Pronoun -> S3:20-20 -> its
Antecedent -> S16:21-21 -> Talcott :: Pronoun -> S17:34-34 -> he
Antecedent -> S29:9-9 -> events :: Pronoun -> S30:1-1 -> It
0004
Antecedent -> S5:2-2 -> maturities :: Pronoun -> S5:11-11 -> they
Antecedent -> S8:14-14 -> yields :: Pronoun -> S8:21-21 -> they
Antecedent -> S14:1-3 -> Dreyfus World-Wide Dollar :: Pronoun-> S15:1-1 -> It
Antecedent -> S14:1-3 -> Dreyfus World-Wide Dollar :: Pronoun-> S15:17-17 -> its
```

Figure 7: Format of antecedent and pronoun annotations

For further evaluation of the proposed method and comparing it with the available methods, the Message Understanding Conference (MUC) 7 corpus (Chinchor 2001) is used. The tasks performed in MUC-7 consist of named entity extraction and coreference chains. On the level of entity extraction person, organization, location, dates, times, percentages, and monetary amounts are marked.

The corpus contains the following components:

- The input file: Is a text file containing a collection of sentences. A part of it is shown in Figure 8.

```
<DOC>
<DOCID> nyt960102.0516 </DOCID>
<STORYID cat=w pri=u> A0264 </STORYID>
<SLUG fv=ttj-z> BC-PEARL-HARBOR-HNS </SLUG>
<DATE> &LR; </DATE>
<NWORDS> 01-02 </NWORDS>
<PREAMBLE>
BC-PEARL-HARBOR-HNS
PENTAGON BLOCKS MOVE TO PROMOTE PEARL HARBOR COMMANDERS
(For use by New York Times News Service clients.)
By CHARLES J. LEWIS
c.1996 Hearst Newspapers
</PREAMBLE>
<TEXT>
<p>
   WASHINGTON  &MD;  The Pentagon has denied a request that top U.S.
commanders in Hawaii in 1941 be absolved of blame for failing to be
on alert for the Japanese attack on Pearl Harbor, but the military
agreed that top Washington officials also must share the blame.
<p>
   A Pentagon study re-affirmed the conclusion of previous
government investigations that both Rear Admiral Husband E. Kimmel
and his Army counterpart, Maj. Gen. Walter C. Short, ``committed
errors of judgment'' leading up to the Dec. 7, 1941, debacle.
<p>
```

Figure 8: An example of raw text in MUC7

- Entity types: The named entities for the sentences in the previous figure are shown in Figure 9.

```
<DOC>
<DOCID> nyt960102.0516 </DOCID>
<STORYID cat=w pri=u> A0264 </STORYID>
<SLUG fv=ttj-z> BC-<ENAMEX TYPE="LOCATION">PEARL-HARBOR</ENAMEX>-HNS </SLUG>
<DATE> &LR; </DATE>
<NWORDS> <TIMEX TYPE="DATE">01-02</TIMEX> </NWORDS>
<PREAMBLE>
BC-<ENAMEX TYPE="LOCATION">PEARL-HARBOR</ENAMEX>-HNS
<ENAMEX TYPE="ORGANIZATION">PENTAGON</ENAMEX> BLOCKS MOVE TO PROMOTE <ENAMEX TYPE="LOCATION">P
(For use by <ENAMEX TYPE="ORGANIZATION">New York Times News Service</ENAMEX> clients.)
By <ENAMEX TYPE="PERSON">CHARLES J. LEWIS</ENAMEX>
c.<TIMEX TYPE="DATE">1996</TIMEX> <ENAMEX TYPE="ORGANIZATION">Hearst Newspapers</ENAMEX>
</PREAMBLE>
<TEXT>
<p>
   <ENAMEX TYPE="LOCATION">WASHINGTON</ENAMEX>  &MD;  The <ENAMEX TYPE="ORGANIZATION">Pentagon
commanders in <ENAMEX TYPE="LOCATION">Hawaii</ENAMEX> in <TIMEX TYPE="DATE">1941</TIMEX> be ab
on alert for the Japanese attack on <ENAMEX TYPE="LOCATION">Pearl Harbor</ENAMEX>, but the mil
agreed that top <ENAMEX TYPE="LOCATION">Washington</ENAMEX> officials also must share the blam
<p>
   A <ENAMEX TYPE="ORGANIZATION">Pentagon</ENAMEX> study re-affirmed the conclusion of previou
government investigations that both Rear Admiral <ENAMEX TYPE="PERSON">Husband E. Kimmel</ENAM
and his <ENAMEX TYPE="ORGANIZATION">Army</ENAMEX> counterpart, Maj. Gen. <ENAMEX TYPE="PERSON"
errors of judgment'' leading up to the <TIMEX TYPE="DATE">Dec. 7, 1941</TIMEX>, debacle.
```

Figure 9: Name entities for the input text

- Coreference chains: An example is shown in Figure 10. Each NP has a corresponding ID which is used to identify the coreference chain.

```
<DOC>
<DOCID> nyt960102.0516 </DOCID>
<STORYID cat=w pri=u> A0264 </STORYID>
<SLUG fv=ttj-z> BC-<COREF ID="175" TYPE="IDENT" REF="3">PEARL-HARBOR</COREF>-HNS </SLUG>
<DATE> &LR; </DATE>
<NWORDS> 01-02 </NWORDS>
<PREAMBLE>
BC-<COREF ID="3">PEARL-HARBOR</COREF>-HNS
<COREF ID="5">PENTAGON</COREF> BLOCKS MOVE TO PROMOTE <COREF ID="7" MIN="COMMANDERS"><COREF
(For use by New York Times News Service clients.)
By CHARLES J. LEWIS
c.1996 Hearst Newspapers
</PREAMBLE>
<TEXT>
<p>
   WASHINGTON  &MD;  <COREF ID="4" TYPE="IDENT" REF="5">The Pentagon</COREF> has denied <CO
commanders in <COREF ID="26">Hawaii</COREF> in 1941</COREF> be absolved of <COREF ID="11" M
on alert for <COREF ID="17" MIN="attack">the Japanese attack on <COREF ID="8" TYPE="IDENT"
agreed that <COREF ID="24" MIN="officials">top Washington officials</COREF> also must share
<p>
   <COREF ID="19">A <COREF ID="12" TYPE="IDENT" REF="9">Pentagon</COREF> study</COREF> re-a
government investigations that <COREF ID="22">both <COREF ID="49" MIN="Husband E. Kimmel">R
and <COREF ID="61" MIN="counterpart"><COREF ID="15" MIN="counterpart">his Army counterpart,
errors of judgment'' leading up to <COREF ID="16" TYPE="IDENT" REF="17" MIN="debacle">the <
<p>
```

Figure 10: Coreference chains for the input text

3.3 Assumptions

To reduce the complexity of the task this study will not address relative pronouns (*who, whom which, whose*) and demonstrative pronouns (*this, that, these, those*).

3.4 Performance Assessment

This section discusses how the system's performance is measured and how the accuracy of the system is determined.

3.4.1 System Accuracy on Test Corpora

The corpus used for training and testing is the BBN Pronoun Coreference and Entity Type Corpus (Weischedel and Brunstein 2005). For the purpose of evaluation the corpus is divided into two sections: one section for training purposes and the other section for testing purposes. The training set consists of 80% of the corpus and testing consists of the remaining 20%. 10 Fold cross validation is performed to increase the accuracy of the result.

The objective is to determine how accurate the system is compared to human annotators. For classification tasks, metrics such as precision, recall and F-measure are calculated and used for evaluating the predicted results (Table 6).

Table 6: Classification outcomes

	Actual Class	
Predicted Class	**True Positive (tp)** *Correct result* Correctly classified as positive	**False Positive (fp)** *Unexpected result* A negative data which has been classified as positive
	False Negative (fn) *Missing results* A positive data which has been classified as negative	**True Negative (tn)** *Correct absence of result* Correctly classified as negative

$$precision = \frac{tp}{tp + fp} \quad \textbf{(Eq. 1)}$$

$$recall = \frac{tp}{tp + fn} \quad \textbf{(Eq. 2)}$$

$$F\_measure = \frac{2 \times precision \times recall}{precision + recall} \quad \textbf{(Eq. 3)}$$

3.4.2 Generalization of the Method

Documents from different genres such as story, fiction and news were used to test the accuracy of the classification model in a more generalized environment. The accuracy scores on different genres are then compared to the results from the BBN corpus to determine how the model generalizes to other domains.

For this purpose three different types of documents are analyzed.

- 2 Short stories from the web.

- 4 Children's stories from the UIUC (University of Illinois at Urbana-Champaign) affect corpus (Alm 2008).
- 2 News from the MUC-7 corpus.

The system was tested on the above stories. The stories were hand annotated and the pronouns were resolved manually and compared to classification results. The annotations were done by one person and double checked by another person. There was complete agreement between the two annotators.

3.4.3 Ranking Analysis of Features

An analysis is performed to determine which features have the highest contribution to pronominal anaphora resolution. The Chi-square feature selection method (Witten and Frank 2005) is used to determine the ranking of different features in this methodology.

CHAPTER 4: CLASSIFICATION USING LINGUISTIC FEATURES

In this chapter the methodology for extracting computational and linguistic features for resolving pronominal anaphors is explained. The overall steps include preprocessing the text using natural language processing toolkits, and then extracting a feature vector for all the combinations of pronouns and noun phrases. The corpus is then divided into training and testing sections and supervised classification methods are used to resolve the pronouns. Following this is results of analysis of feature importance and classification performance for the methodology.

4.1 Preprocessing

In preprocessing the document is passed through a series of linguistic processors such as tokenizers, part-of-speech taggers and syntactic parsers. These components produce annotations of the input text. For the preprocessing step the build in functions in NLTK library of Python and Stanford Parser are used. Stanford Parser is used to perform preprocessing on the document.

4.1.1 Stanford Parser

A natural language parser is a program that specifies the grammatical structure of a sentence; for example it specifies the subject or objects of a verb, or determines which group of words go together as a phrase. These probabilistic parsers use the knowledge from hand parsed sentences and try to produce the most accurate analysis of a new sentence. These statistical parsers still make some mistakes, but commonly work well.

For the preprocessing steps Stanford Parser (De Marneffe et al. 2006) which is a Java based statistical parser is used. The Natural Language Processing Group at Stanford University first implemented Stanford Parser. The parser can read various forms of plain text input and can output various analyses formats, including part-of-speech (POS) tagged text, phrase structure

trees, and a grammatical relations (typed dependency) format. An example of Stanford Parser POS tagged text is shown in Figure 11.

```
(ROOT
  (S
    (NP
      (NP (NNP Bell))
      (, ,)
      (VP (VBN based)
        (PP (IN in)
          (NP (NNP Los) (NNP Angeles))))
      (, ,))
    (VP (VBZ makes)
      (CC and)
      (VBZ distributes)
      (NP
        (UCP (JJ electronic) (, ,) (NN computer)
          (CC and)
          (NN building))
        (NNS products)))
    (. .)))
```

Figure 11: Parsed tree

The Stanford dependencies provide a representation of grammatical relations between words in a sentence. It is designed to be used by those who want to extract textural relations. The current representation contains approximately 53 grammatical relations. The dependencies are binary grammatical relations between a governor (also known as a regent or a head) and a dependent (De Marneffe and Manning 2008).

As shown in Figure 12 the grammatical relations stand in a hierarchy. If a precise relation does not exist or cannot be found, the most generic grammatical relation will be used.

root - root
dep - dependent
    aux - auxiliary
        auxpass - passive auxiliary
        cop - copula
    arg – argument
        agent - agent
        comp - complement
            acomp - adjectival complement
            attr - attributive
            ccomp - clausal complement with internal subject
            xcomp - clausal complement with external subject
            complm - complementizer
            obj - object
                dobj - direct object
                iobj - indirect object
                pobj - object of preposition
            mark - marker (word introducing an advcl)
            rel - relative (word introducing a rcmod)
        subj - subject
            nsubj - nominal subject
                nsubjpass - passive nominal subject
            csubj - clausal subject
                csubjpass - passive clausal subject
    cc - coordination
    conj - conjunct
    expl - expletive (expletive “there”)
    mod - modifier
        abbrev - abbreviation modifier
        amod - adjectival modifier
        appos - appositional modifier
        advcl - adverbial clause modifier
        purpcl - purpose clause modifier
        det - determiner
        predet - predeterminer
        preconj - preconjunct
        infmod - infinitival modifier
        mwe - multi-word expression modifier
        partmod - participial modifier
        advmod - adverbial modifier
            neg - negation modifier
        rcmod - relative clause modifier
        quantmod - quantifier modifier
        nn - noun compound modifier
        npadvmod - noun phrase adverbial modifier
            tmod - temporal modifier
        num - numeric modifier
        number - element of compound number

Figure 12: Hierarchy of typed dependencies (Source: De Marneffe and Manning 2008)

prep - prepositional modifier
poss - possession modifier
possessive - possessive modifier ('s)
prt - phrasal verb particle
parataxis - parataxis
punct - punctuation
ref - referent
sdep - semantic dependent
xsubj - controlling subject

The graphical representation of the Stanford Dependencies for the sentence: "Bell, based in Los Angeles, makes and distributes electronic, computer and building products." (De Marneffe and Manning 2008) is shown in Figure 13.

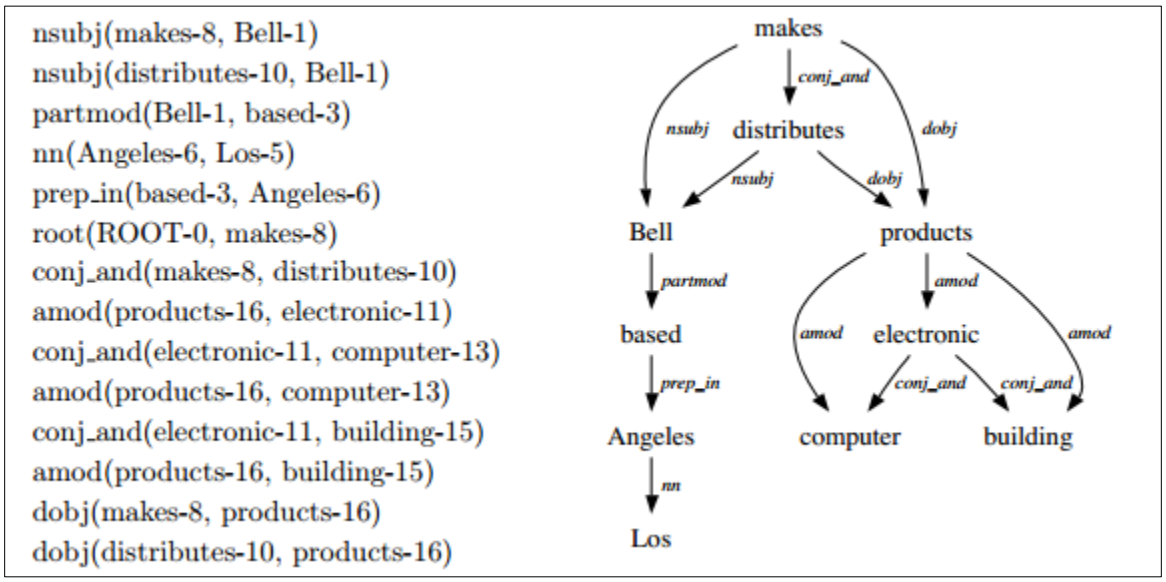


Figure 13: Standard Stanford dependencies (Source: De Marneffe and Manning 2008)

4.1.2 Charniak Parser

In 1999 Eugene Charniak presented a parser (ftp://ftp.cs.brown.edu/pub/nlparser/) for parsing sentences down to Penn tree-bank style parse tree and the parser achieved an average of 90% precision/recall for sentences. The main innovation of this parser was the use of "maximum-entropy-inspired" model for conditioning and smoothing.

Among many output that this parser gives is the clause annotation of sentences. The reason Charniak Parser was used as part of the preprocessing steps is for extracting the clauses that are used for generating feature vectors (explained later in section 4.2). An example of the clause annotation is shown in Figure 14.

```
<P>
<S>
<C>The/DT White/NNP House/NNP said/VBD Mr./NNP Bush/NNP
decided/VBD to/TO grant/VB duty-free/JJ status/NN for/IN 18/CD
categories/NNS ,/, </C>
<C>but/CC turned/VBN down/RP such/JJ treatment/NN for/IN other/JJ
types/NNS of/IN watches/NNS `/`` `/`` because/IN of/IN the/DT
potential/NN for/IN material/JJ injury/NN to/TO watch/VB
producers/NNS located/VBN in/IN the/DT U.S./NNP and/CC the/DT
Virgin/NNP Islands/NNP ./. '/'' '/'' </C>
</S>
</P>
```

Figure 14: Clause annotation using Charniak Parser

In Table 7 the five popular state-of-art parsers have been compared in terms of speed and accuracy. The parsers are trained using the standard training set of the Penn Treebank consisting of sections 2 through 21.

The results show that parsers have an average error of 10% when parsing a document. This has to be taken into consideration when analyzing the results. Parsers, which are part of the classification method, take time and are also not free of errors. Any error in this stage will cause more errors throughout the process.

Table 7: Different parsers' F score (%) and time (min:seconds) to parser sample text (Source: Cer et al. 2010)

Parser	F-measure (%)		Parse Time
	Unlabeled	Labeled	
Stanford englishPCFG v1.6.2 (Klein and Manning, 2003)	87.2	84.2	10:04
Charniak 05Aug16 (Charniak, 2000)	90.5	87.8	11:09
Charniak-Johnson June06 (CJ) (Charniak and Johnson, 2005)	91.7	89.1	10:18
Bikel v1.2 (Bikel, 2004)	88.7	85.3	28:57
Berkeley v1.1 (Petrov et al., 2006)	90.5	87.9	9:14

4.1.3 Pronoun Identification

The pronominal pronouns that are considered are subjective (*he, she, it, they*), objective (*him, her, it, them*), reflexive (*himself, herself, itself, themself*) and possessive (*his, hers, its, their, theirs*) personal pronouns. A python processor is developed to perform this identification. The input is the original text file; the output is a text file with entries for each pronoun identified, including the pronoun, sentence and word position (Figure 15).

```
S1: wordnum:2 PRP:it
S1: wordnum:6 PRP:their
S3: wordnum:14 PRP:its
S4: wordnum:7 PRP:its
S5: wordnum:29 PRP:it
S9: wordnum:14 PRP:their
S10: wordnum:6 PRP:it
S14: wordnum:8 PRP:they
S14: wordnum:13 PRP:their
S16: wordnum:4 PRP:it
S17: wordnum:21 PRP:them
S18: wordnum:10 PRP:they
S19: wordnum:11 PRP:them
S20: wordnum:5 PRP:they
S20: wordnum:10 PRP:their
```

Figure 15: Pronoun extraction for each sentence

4.1.4 Antecedent Detection

In this step the candidates for the antecedent are specified. The parsed tree of Stanford Parser is used to extract the NPs from the text. A Python processor is developed that inputs the parse tree

and returns a text file with entries for each NP, including its sentence number and start and end word position (Figure 16).

```
S1: 28-29 Hot Springs
S1: 22-23 resort towns
S1: 9-11 the nation 's
S1: 4-4 time
S1: 25-26 Boca Raton
S1: 18-20 the sunny confines
S1: 6-8 their biannual powwow
S2: 2-3 this year
S3: 14-17 its fall board meeting
S3: 8-10 the Hoosier capital
S3: 1-3 The National Association
S3: 12-12 Indianapolis
S3: 5-5 Manufacturers
S4: 16-17 factory owners
S4: 7-8 its guests
S4: 18-20 royalty rock stars
S4: 2-3 the city
S5: 33-36 a company to expand
S5: 14-15 the buckle
S5: 4-4 course
S5: 29-31 a good place
S5: 17-19 the Rust Belt
S5: 9-12 125 corporate decision makers
S5: 1-2 The idea
```

Figure 16: NP extraction for each sentence

4.2 Features

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 distance features have been used in all the resolution systems. The major difference between the features used here and those implemented in other systems is the use of linguistic rules.

Our feature vector consists of 15 grammatical and linguistics features which are described in the following. Each feature vector is derived based on two extracted parts, a potential antecedent (which is an NP) and a pronoun. The information needed for deriving the feature vectors is provided by pre-processing the text in advance (see Section 4.1).

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 8.

Table 8: List of pronouns and their number and gender

Pronoun	**Number**	**Gender**
he, him, himself, his	Singular	Male
she, her, herself, hers, her	Singular	Female
it, itself, its	Singular	Neutral
they, them, themselves, their, theirs	Plural	Neutral

The number and gender for each NP is specified using the following rules:

1. The noun phrase is first checked for designators such as *Mr.*, *Mrs.*, *Ms.*, and *Miss*, If found number and gender is specified.
2. In cases where rule #1 doesn't 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 (Bergsma and Lin 2006) (explained in the following) is used and the head noun is queried to find the gender and number. The gender with the most counts in the database is specified as the gender of the NP. If the probability of being Plural is greater than 50% the number feature is plural, otherwise singular. In cases where the word is not found the system returns “NOTFOUND” and later it is added manually.

Gender DB (Bergsma and Lin 2006) 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:

Nounphrase [TAB] Masculine_Count [SPACE] Feminine_Count [SPACE] Neutral_Count [SPACE] Plural_Count

As with everything in statistical NLP, it should be taken into consideration that nouns with few observations are more likely to have misleading gender counts, while those with higher counts are generally more accurate (Bergsma and Lin 2006). An example of probabilistic gender and number counts using Gender DB is given in Table 9.

Table 9: Probabilistic Gender Examples from Gender DB (Source: Bergsma and Lin 2006)

Word	Masculine	Feminine	Neutral	Plural
company	0.6	0.1	98.1	1.2
condoleeza rice	4.0	92.7	0.0	3.2
pat	58.3	30.6	6.2	4.9
president	94.1	3.0	1.5	1.4
wife	9.9	83.3	0.8	6.1
coincidence	28.3	4.3	44.6	22.7
middle river	0.0	50	50	0.0
bookseller	12.9	4.7	79.4	2.9

Distance Feature (F3): This feature captures the distance between the pronoun and the noun phrase and therefore the possible values can be 0, 1, 2, 3… If the pronoun and NP are in the same sentence the value is 0. This feature is believed to be among the most important features

and research indicates that 90% of antecedents are at most 2 sentences apart from their pronouns. Figure 17 provides a summary of the distances between pronouns and their antecedent for the documents in the BBN Corpus. The annotations for the corpus were based on each pair of pronoun and antecedent and therefore the pronoun chain is not taken into consideration. As shown the maximum distance between a pronoun and its antecedent was 5 sentences. 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.

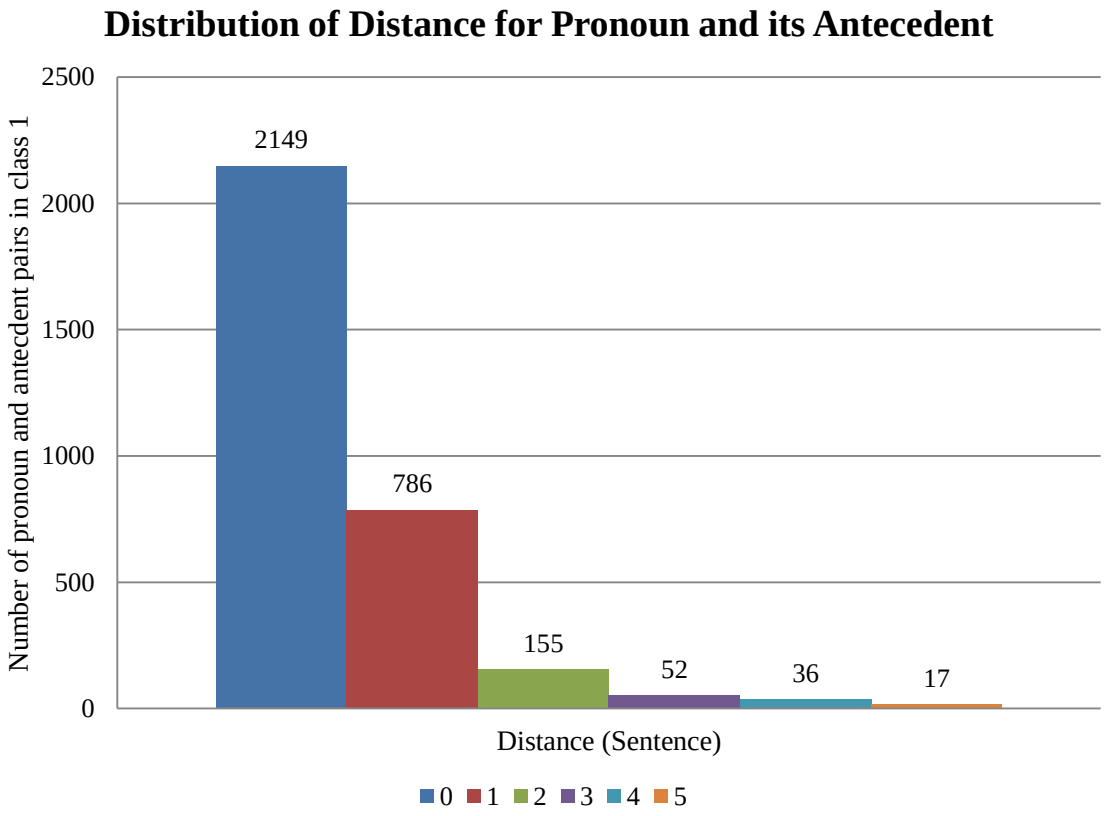


Figure 17: Illustration of distribution of distance for a pronoun and its antecedent

Proper Name Feature (F4): For the noun phrase to be a proper name, if prepositions *'of'* and *'and'* appear in the name they shouldn't be uppercase (Soon et al. 2001).

The POS tag of the NP is checked and if it's either NNP or NNPS it returns 1, otherwise returns 0.

Definite Noun phrase Feature (F5): A definite noun phrase is a noun phrase that starts with *'the'*. If the NP is definite 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. The reason behind using F5 and F6 is the Givenness Hierarchy (Webber 1988) (explained in the following).

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 (Table 10). (Webber 1988)

Table 10: Givenness Hierarchy (Source: Webber 1988)

In focus	Activated	Familiar	Uniquely identifiable	Referential	Type identifiable
It	*That, This, This*	*That*	*The*	Indefinite *this*	*a, an*

In the above hierarchy each status entails all lower statuses.

- ***Type identifiable:*** A representation of the type of object described can be accessed.
- ***Referential:*** The speaker intends to refer to a particular object or objects.
- ***Uniquely Identifiable:*** The speaker's intended referent on the basis of the referent alone can be identified. If the referent doesn't already exist in the memory, a representation for it is constructed.

- ***Familiar:*** The referent can be uniquely identified since there is already a representation of it in memory.
- ***Activated:*** The referent is represented in current short-term memory.
- ***In Focus:*** The referent is not only in short-term memory, but is also at the current center of attention

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 1997).

Pronoun with 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 with 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 with 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 with 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 pronoun and NP are in the same sentence they are checked to see whether they are in the same clause or not. Charniak Parser is used in the preprocessing steps for extracting the clauses in each sentence (Section 4.1.2). If they are in the same clause it returns 1 otherwise 0.

NP in the Prepositional Clause (F12): A prepositional clause is a clause that starts with any of the prepositions (listed in Table 11). The clause, in which the noun phrase is part of, is checked and if it's a prepositional clause it returns 1 otherwise 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 applied to those 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 1970). 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 1970; Bolinger 1979).

Table 11: List of prepositions

about	Because of	Except	Like	Through
above	Before	Except for	Near	Throughout
according	Behind	Excepting	Next	Till
to	Below	For	Of	to
across	Beneath	From	Off	Toward
After	Beside	in	On	Under
Against	Between	In addition to	Onto	Underneath
Along	beyond	In back of	On top of	Unlike
Along with	But	In case of	Out of	Until
among	By	In front of	Outside	Up
Apart from	By means of	In place of	Over	Upon
Around	Concerning	Inside	Past	Up to
As	Despite	In spite of	Regarding	With
As for	Down	Instead of	Round	Within
At	During	Into	Since	without

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. According to Lakoff if a pronoun is in the preceding clause then the number of words in the subordinate clause

containing a NP may be of influence for the interpretation of that NP as coreferential (Lakoff 1968).

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 0.

Excitation Feature (F15): This feature indicates how much the NP is in focus by taking into consideration the number of times the NP has been mentioned in the previous sentences. The previous sentences are checked and the excitation feature is incremented by one when any of the following is found in all the previous sentences:

1. The NP
2. The head noun of NP
3. For NPs longer than 2 words that start with designators such as *Mr.*, *Mrs.*, *Ms.*, and *Miss*, the first word after the title is also searched.

This way an NP that has been mentioned in previous sentences is in focus and therefore has a higher chance of being referred to a pronoun than the NP that hasn't been mentioned.

15 features are considered for each pronoun and NP pair. In the feature extraction process first the features that are related to each pronoun is extracted. These features include sentence and word position, gender and number of each pronoun and pronoun having a subject (F7) or object (F8) role in the sentence. An example of the extracted feature for a couple of pronouns is shown in Figure 18.

```
>>> PRPVector
['S1: wordnum:2 PRP:it G:N N:S F7:1 F8:0 ',
 'S1: wordnum:6 PRP:their G:N N:P F7:0 F8:0 ',
 'S3: wordnum:14 PRP:its G:N N:S F7:0 F8:0 ',
 'S4: wordnum:7 PRP:its G:N N:S F7:0 F8:0 ',
 'S5: wordnum:29 PRP:it G:N N:S F7:1 F8:0 ',
 'S9: wordnum:14 PRP:their G:N N:P F7:0 F8:0 ',
 'S10: wordnum:6 PRP:it G:N N:S F7:0 F8:1 ',
 'S14: wordnum:8 PRP:they G:N N:P F7:1 F8:0 ',
 'S14: wordnum:13 PRP:their G:N N:P F7:0 F8:0 ',
 'S16: wordnum:4 PRP:it G:N N:S F7:1 F8:0 ',
 'S17: wordnum:21 PRP:them G:N N:P F7:1 F8:0 ',
 'S18: wordnum:10 PRP:they G:N N:P F7:1 F8:0 ',
 'S19: wordnum:11 PRP:them G:N N:P F7:1 F8:0 ',
 'S20: wordnum:5 PRP:they G:N N:P F7:1 F8:0 ',
 'S20: wordnum:10 PRP:their G:N N:P F7:0 F8:0 ']
```

Figure 18: An example of features extracted for pronouns

The next step is generating features that are related to each NP. This includes the sentence and word position of the NP, the gender and number of the head noun, NP being a proper name (F4) or definite noun phrase (F5) of a demonstrative noun phrase (F6), NP having a subject (F9) or object (F10) role, NP being part of a prepositional clause (F12) or part of a long subordinate clause (F14), and the last feature is the number of times the NP has been mentioned in the previous sentences (F15). (An example of the extracted features are shown in Figure 19)

The last step is generating the final 15 features using the features extracted for each pronoun and NP.

```
>>> NPVector
['S1: 28-29 Hot Springs G:M N:S F4:1 F5:0 F6:0 F9:0 F10:0 F12:0 F14:0 F15:0',
 'S1: 22-23 resort towns G:M N:P F4:0 F5:0 F6:0 F9:0 F10:0 F12:0 F14:0 F15:0',
 "S1: 9-11 the nation 's G:N N:S F4:0 F5:1 F6:0 F9:0 F10:0 F12:0 F14:0 F15:0",
 'S1: 4-4 time G:N N:S F4:0 F5:0 F6:0 F9:1 F10:0 F12:0 F14:1 F15:0',
 'S1: 25-26 Boca Raton G:N N:S F4:1 F5:0 F6:0 F9:0 F10:0 F12:0 F14:0 F15:0',
 'S1: 18-20 the sunny confines G:N N:P F4:0 F5:1 F6:0 F9:0 F10:0 F12:0 F14:0 F15:0',
 'S1: 6-8 their biannual powwow G:N N:S F4:0 F5:1 F6:0 F9:0 F10:0 F12:0 F14:1 F15:0',
 'S2: 2-3 this year G:N N:S F4:0 F5:0 F6:1 F9:0 F10:0 F12:0 F14:0 F15:0',
 'S3: 14-17 its fall board meeting G:N N:S F4:0 F5:0 F6:0 F9:0 F10:0 F12:0 F14:0 F15:0',
 'S3: 8-10 the Hoosier capital G:N N:S F4:0 F5:1 F6:0 F9:0 F10:0 F12:0 F14:0 F15:0',
 'S3: 1-3 The National Association G:N N:S F4:1 F5:1 F6:0 F9:1 F10:0 F12:0 F14:0 F15:0',
 'S3: 12-12 Indianapolis G:N N:S F4:1 F5:0 F6:0 F9:0 F10:0 F12:0 F14:0 F15:0',
 'S3: 5-5 Manufacturers G:N N:S F4:1 F5:0 F6:0 F9:0 F10:0 F12:0 F14:0 F15:0',
 'S4: 16-17 factory owners G:N N:P F4:0 F5:0 F6:0 F9:0 F10:0 F12:0 F14:0 F15:0',
 'S4: 7-8 its guests G:M N:P F4:0 F5:0 F6:0 F9:0 F10:1 F12:0 F14:0 F15:0',
 'S4: 18-20 royalty rock stars G:N N:P F4:0 F5:0 F6:0 F9:0 F10:0 F12:0 F14:0 F15:0',
 'S4: 2-3 the city G:N N:S F4:0 F5:1 F6:0 F9:1 F10:0 F12:0 F14:0 F15:0',
 'S5: 33-36 a company to expand G:N N:S F4:0 F5:0 F6:0 F9:1 F10:0 F12:0 F14:1 F15:0',
 'S5: 14-15 the buckle G:M N:S F4:0 F5:1 F6:0 F9:1 F10:0 F12:0 F14:1 F15:0',
 'S5: 4-4 course G:N N:S F4:0 F5:0 F6:0 F9:0 F10:0 F12:0 F14:0 F15:0',
 'S5: 29-31 a good place G:N N:S F4:0 F5:0 F6:0 F9:1 F10:0 F12:0 F14:1 F15:0',
 'S5: 17-19 the Rust Belt G:N N:S F4:0 F5:1 F6:0 F9:0 F10:0 F12:0 F14:1 F15:0',
 'S5: 9-12 125 corporate decision makers G:N N:P F4:0 F5:0 F6:0 F9:0 F10:0 F12:0 F14:0 F15:0',
 'S5: 1-2 The idea G:N N:S F4:1 F5:1 F6:0 F9:0 F10:0 F12:0 F14:0 F15:0',
```

Figure 19: An example of features extracted for NPs

The class is also extracted using the annotations provided for the BBN corpus. As shown in the following for each story in the BBN the antecedents and pronouns are extracted with their sentence and word positions specified (Figure 20). This information is used to indicate whether the NP or pronoun go together or not. If they match the class is 1, otherwise is 0.

```
0001
0002
0003
Antecedent -> S1:2-2 -> form :: Pronoun -> S1:27-27 -> it
Antecedent -> S2:3-3 -> fiber :: Pronoun-> S2:21-21 -> it
Antecedent -> S2:3-3 -> fiber :: Pronoun-> S2:11-11 -> it
Antecedent -> S3:1-2 -> Lorillard Inc. :: Pronoun -> S3:20-20 -> its
Antecedent -> S16:21-21 -> Talcott :: Pronoun -> S17:34-34 -> he
Antecedent -> S29:9-9 -> events :: Pronoun -> S30:1-1 -> It
0004
Antecedent -> S5:2-2 -> maturities :: Pronoun -> S5:11-11 -> they
Antecedent -> S8:14-14 -> yields :: Pronoun -> S8:21-21 -> they
Antecedent -> S14:1-3 -> Dreyfus World-Wide Dollar :: Pronoun-> S15:1-1 -> It
Antecedent -> S14:1-3 -> Dreyfus World-Wide Dollar :: Pronoun-> S15:17-17 -> its
```

Figure 20: Processed annotations

4.3 Classification

WEKA (Waikato Environment for Knowledge Analysis) (Bouckaert et al. 2010) is used for applying classification methods such as Naïve Bayes, Support Vector Machines and Decision Trees. WEKA is a machine learning software with a collection of machine learning algorithms for data mining tasks.

After the features have been extracted, a model is trained to detect the antecedents for pronouns in a sentence. In the classification process, LibSVM, Decision Tree, Naïve Bayes and bagging classifiers are used.

To perform the classification task, feature vectors were extracted for all the combination of NP and pronouns that are at most 5 sentences apart in the text. Each NP-pronoun pair is classified into true or false classes. True class indicates that the pronoun refers to the NP and the false class indicates it doesn't. Class is also extracted using the annotation provided for the BBN corpus. Feature vectors are a combination of binary and numeric and therefore normalization was performed on the features.

350 documents from the BBN corpus are used. After the NPs were extracted and feature vectors were created the total number of data added up to 195,929. Since the number of data of training vectors in the false class was much higher than the ones in the true class, a program was written that would randomly pick equal numbers of vector for each class. The final number of vectors after equalization of classes was 6,390. Performance is measured using precision and recall accuracy scores.

An example of sample data is shown in Figure 21.

After extracting the feature vectors for all the pronoun and NP pairs, classification is performed using classifiers. The result of the classifier for each pair is either 1 (if the pronoun and NP refer to each other) or 0 (if they don't).

```
NP->142:14-16->S&P 500 stocks , Pronoun->145:17-17->his, 0,0,3,1,0,0,0,0,0,0,0,0,1,0,0
NP->27:5-6->Mr. McFall , Pronoun->28:19-19->he, 1,1,1,1,0,0,1,0,1,0,0,0,0,0,1
NP->28:1-2->Mr. Dinkins , Pronoun->29:17-17->he, 1,1,1,1,0,0,1,0,1,0,0,0,0,0,3,1
NP->31:31-33->the health system , Pronoun->26:1-1->He, 1,0,5,0,1,0,1,0,0,0,0,0,0,0,0
NP->43:5-6->a close-up , Pronoun->38:7-7->his, 1,0,5,0,0,0,0,0,1,1,0,0,0,1,0,0
NP->22:24-26->three-digit price tags , Pronoun->26:11-11->its, 0,1,4,0,0,0,0,0,0,1,0,0,0,0,0,0
NP->31:3-4->Section 8 , Pronoun->32:6-6->them, 0,1,1,1,0,0,0,0,1,0,0,0,0,0,2,0
NP->19:2-4->the two leaders , Pronoun->19:18-18->their, 1,1,0,0,1,0,0,0,1,0,0,0,1,1,0,1
NP->9:13-15->the next year , Pronoun->9:1-1->He, 1,0,0,0,1,0,1,0,0,0,0,0,0,0,0,0
NP->19:6-7->the least , Pronoun->19:18-18->he, 1,1,0,0,1,0,1,0,1,0,0,0,1,1,0,0
NP->25:17-17->front , Pronoun->20:47-47->her, 1,0,5,0,0,0,1,0,0,0,0,0,0,1,0,0
NP->107:1-2->Mrs. Ward , Pronoun->108:12-12->she, 1,1,1,1,0,0,1,0,1,0,0,0,0,0,6,1
NP->19:13-16->escrow and record-keeping rules , Pronoun->19:7-7->they, 1,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0
NP->27:1-2->Mr. Phelan , Pronoun->29:8-8->he, 1,1,2,1,0,0,1,0,1,0,0,0,0,0,6,1
NP->35:4-5->Judge O'Kicki , Pronoun->31:3-3->his, 1,1,4,1,0,0,0,0,1,0,0,0,0,0,3,0
NP->44:15-17->the Porche 944 , Pronoun->40:11-11->They, 0,1,4,0,1,0,1,0,1,0,0,0,0,0,0,0
NP->7:7-11->morning and prime-time news shows , Pronoun->8:21-21->it, 0,1,1,0,0,0,1,0,0,0,0,0,0,0,0,0
NP->22:12-13->program trading , Pronoun->22:34-34->it, 1,1,0,0,0,0,1,1,0,1,0,0,0,0,7,1
NP->10:8-12->a highly polished jam session , Pronoun->9:1-1->His, 1,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0
NP->15:12-12->brokers , Pronoun->17:7-7->their, 1,1,2,0,0,0,0,0,1,0,0,0,0,1,0,0
NP->20:5-6->magazine publishing , Pronoun->15:22-22->it, 1,1,5,0,0,0,0,1,0,0,0,0,0,0,0,0
NP->11:1-2->The company , Pronoun->11:17-17->its, 1,1,0,1,1,0,0,1,1,0,0,0,1,0,2,1
NP->16:1-2->Mr. White , Pronoun->19:2-2->his, 1,1,3,1,0,0,0,0,1,0,0,0,0,0,0,1
NP->18:29-29->anything , Pronoun->21:8-8->them, 0,1,3,0,0,0,1,0,0,1,0,0,0,1,1,0
NP->2:2-3->the Treasury , Pronoun->2:12-12->it, 1,1,0,0,1,0,1,1,1,0,0,0,0,0,0,1
NP->40:14-14->then , Pronoun->41:6-6->he, 1,1,1,0,1,0,1,0,0,0,0,0,0,0,3,0
NP->17:1-2->Mr. Chase , Pronoun->17:10-10->his, 1,1,0,1,0,0,0,0,1,0,0,0,0,0,1,1
NP->3:13-15->Mr. Bush 's , Pronoun->3:20-20->his, 1,1,0,1,0,0,0,0,0,0,0,0,0,1,0,1
NP->21:1-1->Chrysler , Pronoun->21:3-3->its, 1,1,0,1,0,0,0,0,1,0,0,0,0,0,2,1
NP->4:12-13->Norberto Mehl , Pronoun->6:17-17->he, 1,1,2,1,0,0,1,0,1,0,0,0,0,0,0,1
NP->69:26-28->college-bowl type competitions , Pronoun->65:9-9->she, 0,0,4,0,0,0,1,0,0,1,0,0,0,0,0,0
NP->84:28-29->the judge , Pronoun->89:14-14->his, 1,1,5,0,1,0,0,0,1,0,0,0,0,0,23,0
NP->8:1-2->Philip Morris , Pronoun->8:7-7->its, 1,0,0,1,0,0,0,0,1,0,0,0,0,0,3,1
NP->20:1-2->Mr. Nixon , Pronoun->21:2-2->he, 1,1,1,1,0,0,1,0,1,0,0,0,0,0,10,1
NP->26:1-2->A bomb , Pronoun->30:17-17->himself, 1,0,4,1,0,0,0,1,1,0,0,0,0,0,0,0
NP->31:14-15->design teachers , Pronoun->36:34-34->he, 0,0,5,0,0,0,1,0,0,1,0,0,0,0,0,0
NP->40:1-2->The demonstrators , Pronoun->40:24-24->their, 1,0,0,1,1,0,0,0,1,0,0,0,1,0,0,1
NP->3:27-28->the company , Pronoun->3:5-5->its, 1,1,0,0,1,0,0,0,1,0,0,0,1,1,0,1
NP->93:1-2->The judge , Pronoun->93:24-24->he, 1,1,0,1,1,0,1,0,1,0,0,0,0,0,7,1
NP->12:4-4->Genentech , Pronoun->12:12-12->it, 1,1,0,1,0,0,0,0,1,0,0,0,0,0,3,1
```

Figure 21: Sample data

After the classification task, a feature analysis is performed to identify feature contribution to accuracy. The classification results are shown in Table 12. Both the SVM method (89%) and Bagging using SVM (89%) achieved better accuracy scores.

But as it is shown the overall accuracy for all the classifiers used here are above 88% and shows promising results compared to other methodologies.

Table 12: Classification results

Naïve 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**

The confusion matrix for LibSVM classifier is also shown in Table 13.

Table 13: Confusion matrix for LibSVM classification

Classified as / **Actual class**	**a**	**b**
a=0	2806	389
b=1	313	2882

In the following misclassification data are analyzed and the possible reasons for causing the error are stated.

4.3.1 Feature Analysis

Feature analysis was performed using chi-square feature selection techniques (Witten and Frank 2005). Chi-square feature ranking is a technique used to calculate the likelihood that a feature is correlated with a class. Based on the annotations in the corpus, this technique can estimate likelihoods per feature and rank the features that are most useful in the classification. This helps identify which features are important for anaphora resolution.

Feature analysis is particularly important in this case since we have added many new features which were theoretically proven in linguistic studies, but the effectiveness of them in machine classification approaches has not yet been analyzed.

The results for features analysis (Figure 22) show that the top 8 features have a significantly higher Chi value compared to other features and among these features are three of 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

Preposition clause and *same clause* feature have a chi score of 0 and the reason is that the number of data that have this feature are small compared to other features, and since we are randomly selecting data these tend to get eliminated. This doesn't necessary mean that the features are not important.

```
=== Attribute Selection on all input data ===

Search Method:
        Attribute ranking.

Attribute Evaluator (supervised, Class (nominal): 16 CLASS):
        Chi-squared Ranking Filter

Ranked attributes:
3073.12175   3 DISTANCE
1148.94702   9 NPSUBJ
1087.23434   1 GENDER
1007.0155    4 PROPERNAME
 792.92103   2 NUMBER
 409.16043  13 COMMA
 380.68516  15 EXCITATION
 289.00998  10 NPOBJ
  48.6328   14 SUBORDINATECLAUSE
   1.35285   5 DEFINITENP
   0.68825   7 PRPSUBJ
   0.5794    6 DEMONSTRATIVENP
   0.06286   8 PRPOBJ
   0        12 PREPOSITIONALCLAUSE
   0        11 SAMECLAUSE

Selected attributes: 3,9,1,4,2,13,15,10,14,5,7,6,8,12,11 : 15
```

Figure 22: Attribute ranking using Chi-squared ranking filter

The contribution of our work lies in showing that a machine learning approach, when combined with the linguistic studies done in this area, is able to achieve accuracy competitive with that of state-of-the-art systems.

4.3.2 Analysis and Results

In this section, the misclassified cases are analyzed. As shown in Figure 23, 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 was done to study the reasons that caused these errors.

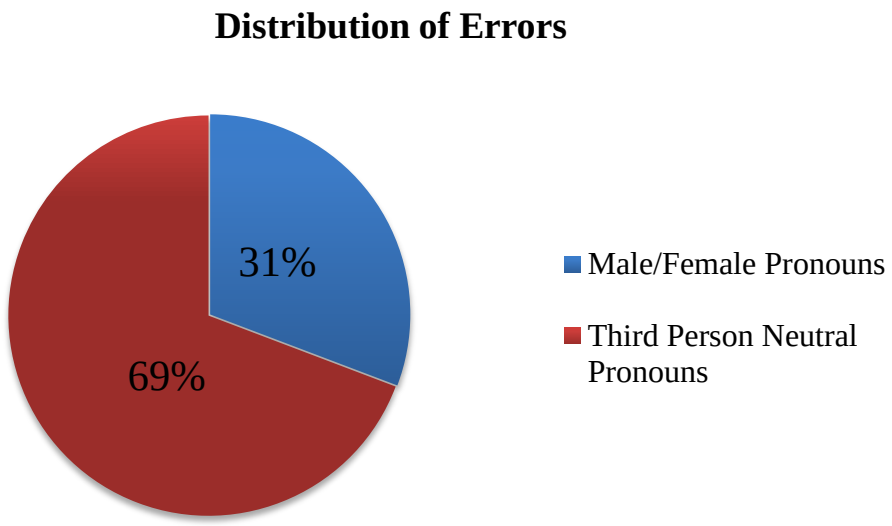


Figure 23: Error classification

The errors caused by misclassifying third person neutral pronouns were first analyzed to find the main reasons for causing the errors. Figure 24 shows the main groups of errors in resolving third person neutral pronouns.

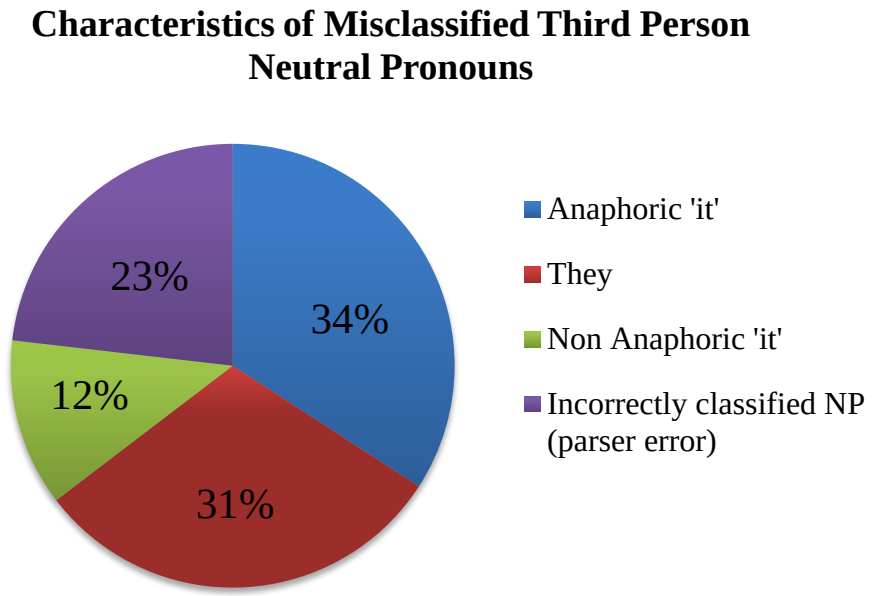


Figure 24: Distribution of errors in resolving third person neutral pronouns

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. Since gender has a high rank in classification, in cases were the pronoun and NP referred together but had disagreements in gender misclassifications were occurred.

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 is generating the feature vector and therefore caused misclassification.

Figure 25 shows the main groups of errors in resolving third person male and female pronouns.

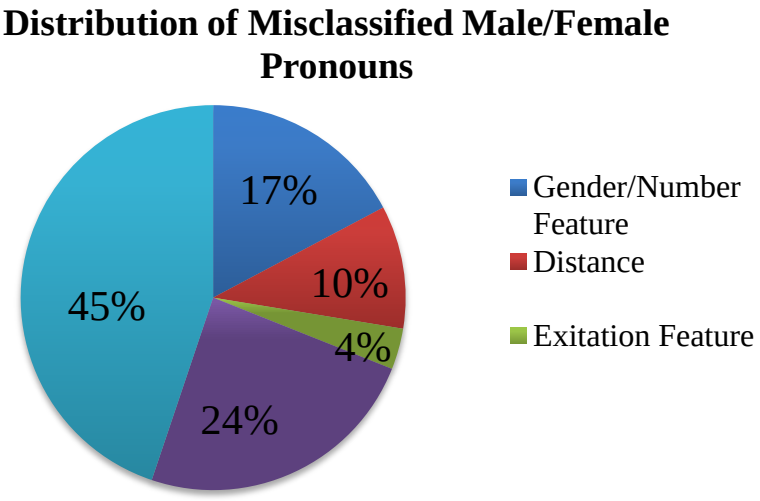


Figure 25: Distribution of errors in resolving male and female pronouns

When analyzing the errors caused by misclassifying male and female pronouns we discovered that 43% of errors were false negative[9] and the remaining 57% of errors were true positive[10].

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 doesn't classify them together.

Excitation Feature: This group of errors is caused when the excitation feature is very high but the pronoun doesn't 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 1983).

[9] This group consists of those pronoun and NPs that were a match but the system didn't classify them together.
[10] This group consists of those pronoun and NPs were the system has classified them as a match but and they don't match

To reduce the errors and improve the performance of the system more features need to be added that can capture semantic information from the text. In the next chapter the use of commonsense knowledge sources such as WordNet and ConceptNet is studied.

CHAPTER 5: CLASSIFICATION USING COMMONSENSE KNOWLEDGE

5.1 Commonsense knowledge

Commonsense knowledge may provide additional clues for deciphering coreferent chains. In particular, the verbs and adjectives applied to a noun phrase may help determine possible matches with other noun phrases. For example, "It ran into the woods" would tend to rule out inanimate NPs such as a house; "She had a beautiful collar" would tend to favor pet NPs over human story participants. In this chapter, we develop the methods for extracting potentially useful common sense information, and analyze the benefit of this information in the anaphor resolution task.

There are many commonsense knowledge sources available. The ones that are incorporated in here are the most widely known and used, ConceptNet and WordNet.

For named entities, it is possible to extract information directly from commonsense sources on the entities. For personal pronouns, however, we must rely strictly on the relations between the verbs, objects, subject, etc. of the pronouns and the candidate antecedents.

After preprocessing, we used the Stanford Parsers Dependencies to generate a list of related words that can help extract the information that is embedded in pronouns and NPs. The dependency tags that are used to extract relative words are *'nsubj'*, *'dobj'*, *'amod'*, *'conj'* and *'nn'*. An illustration of the idea is shown in Figure 26 and Figure 27 on the following example:

> "In the new position, he will oversee Mazda's U.S. sales, service, parts and marketing operations." (BBN Pronoun Coreference and Entity Type Corpus)

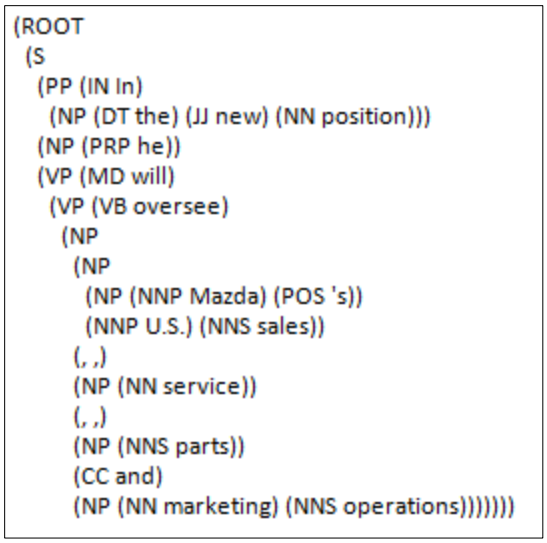


Figure 26: Stanford Parser parsed tree

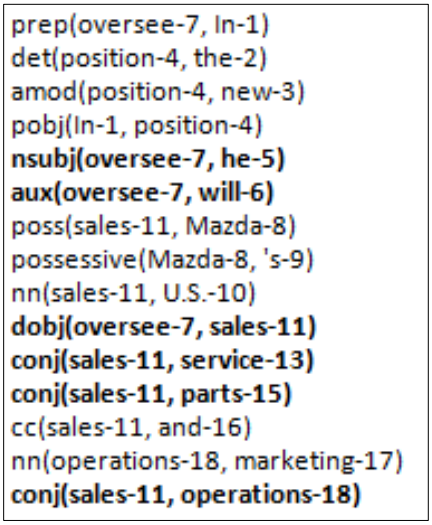


Figure 27: List of dependencies

In Figure 27 the important dependencies are specified to be used in further analysis, and in Figure 28 the new tree with the important chains is shown.

A Python processor was developed to process the dependencies list and extract any word with the specified tags (*'nsubj'*, *'dobj'*, *'amod'*, *'conj'* and *'nn'*) and add them to the list of related

words for each pronoun and NP. Each pair of word is then queried in WordNet and ConceptNet and the similarity measures are generated and combined for each pronoun and NP.

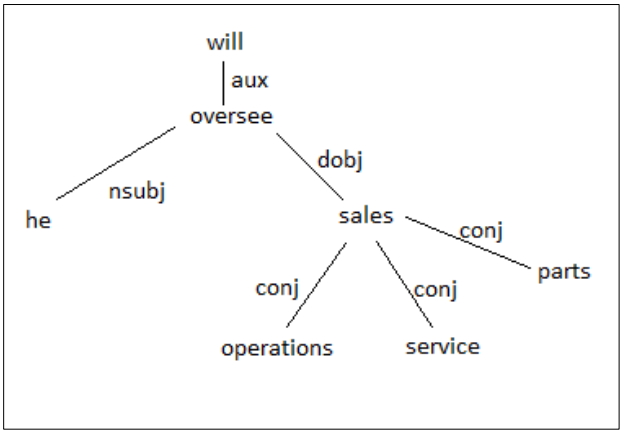


Figure 28: The new tree after extracting information

5.1.1 ConceptNet

The information in ConceptNet is gathered from people, and as a consequence contains some conflicting and imprecise relations. The AnalogySpace process (explained in Section 2.4.1) was developed to reduce noise and represent interrelations between concepts in a semantic network as a sparse matrix (Figure 29). AnalogySpace uses data from ConceptNet and represents knowledge as a matrix of objects or concepts along one axis, and features of the objects along another, creating a sparse matrix of very high dimension (Speer et al. 2008). By using singular value decomposition (SVD) the dimensionality is reduced and the result represents the most salient aspects of the knowledge (Speer et al. 2008).

Each concept can be associated with a vector in the space of possible features. The values of this vector are positive for features that produce positive meaning when combined with that concept, negative for features that produce negative meaning, and zero when nothing is known about the combination (Speer et al. 2008).

AnalogySpace was applied to generate a sparse matrix representation of ConceptNet using the Divisi toolkit. Divisi is a toolkit for Python that is particularly designed for working with semantic networks (Speer et al. 2008). Using Divisi with ConceptNet, the results will include relationships that are not expressed in the original data but related by common sense. Divisi is used to build an AnalogySpace. Dimensionality reduction is used to automatically discover large-scale patterns in the data from ConceptNet. These patterns are called ‘eigenconcepts’ or ‘axes’ and are used to classify the knowledge and predict new knowledge by filling in the gaps (Speer et al. 2008).

```
SparseMatrix (11685 by 85730)
         AtLocati   fawn\AtL   IsA/deer   fawn\IsA   AtLocati ...
fawn     0.500000      ---     0.792481      ---     0.500000
wood        ---     0.500000      ---        ---     0.792481
deer     0.500000      ---        ---     0.792481   0.500000
forest   0.500000   0.500000      ---        ---        ---
animal   0.792481      ---        ---        ---     0.500000
yellow      ---        ---        ---        ---        ---
blue        ---        ---        ---        ---        ---
colour r    ---        ---        ---        ---        ---
color       ---        ---        ---        ---     0.500000
sun         ---        ---        ---        ---    -0.500000
colour      ---        ---        ---        ---        ---
daffodil    ---        ---        ---        ---        ---
primary     ---        ---        ---        ---        ---
submarin    ---        ---        ---        ---        ---
primary     ---        ---        ---        ---        ---
bottle      ---        ---        ---        ---        ---
compassi    ---        ---        ---        ---        ---
presiden    ---        ---        ---        ---        ---
banana      ---        ---        ---        ---        ---
red         ---        ---        ---        ---        ---
...
```

Figure 29: SparseMatrix output from divisi2

Eigenconcepts are the axes that define the AnalogySpace. In ConceptNet, concepts are described using the feature they have, for example, “people want it”, “it is kind of animal” (Speer et al. 2008). In AnalogySpace, these features are summarized by a smaller number of eigenconcepts. How correlated each concept is with these eigenconcepts specified a concept’s coordinates in AnalogySpace. In Figure 30 an example of eigenconcepts for “desirable’ and “undesirable”

concepts are given. The concepts on left are undesirable concepts such as “ignore” and the concepts on the right are desirable concepts such as “feel loved”.

Figure 30: Eigenconcepts for ‘desirable’ and ‘undesirable’ concepts (Source: Speer et al. 2008)

Using the similarity measure, the concepts that are similar to each other are specified. Similarity scale ranges from 1(exactly similar) to -1(exactly dissimilar). But in some cases we need to know whether two concepts are related to each other or not. For example, concepts “sad” and “cry” are only a bit similar but they are very much related to each other. In this case we use the *reconstruct_activation* function in Divisi2 that takes the results from the SVD and spreads the activation from one concept to another. The result shows how much activation would spread from one concept to another (with a maximum of 1). In other words it shows how related the two concepts are together.

5.1.2 WordNet

WordNet is a good source of information for getting word sense and synonyms. WordNet gets each word’s synset[11] and can calculate similarities for words in the same Part of speech group.

There are different similarity measures in WordNet and the ones that are used in this approach are explained in the following.

Path length

This score is calculated based on a simple node-counting scheme (path). The relatedness score is inversely proportional to the number of nodes along the shortest path between the synsets. The

[11] Synonym set

shortest possible path occurs when the two synsets are the same, in which case the length is 1. Path_similarity assigns a score in the range 0–1 based on the shortest path that connects the concepts in Is-A (hypernym) taxonomy (-1 is returned in those cases where a path cannot be found).

Leacock & Chodorow

The relatedness measure proposed by Leacock and Chodorow (lch) is -log (length / (2 * D)), where length is the length of the shortest path between the two synsets (using node-counting) and D is the maximum depth of the taxonomy.

The fact that the lch measure takes into account the depth of the taxonomy in which the synsets are found means that the behavior of the measure is profoundly affected by the presence or absence of a unique root node. If there is a unique root node, then there are only two taxonomies: one for nouns and one for verbs. All nouns, then, will be in the same taxonomy and all verbs will be in the same taxonomy. Leacock Chodorow similarity returns a score denoting how similar two word senses are, based on the shortest path that connects the senses and the maximum depth of the taxonomy in which the senses occur.

Wu & Palmer

The Wu & Palmer measure (wup) calculates relatedness by considering the depths of the two synsets in the WordNet taxonomies, along with the depth of the LCS. The similarity score is a valued between 0 and 1 and is calculated using Equation 4.

$$Score = 2 \times \frac{depth(lch)}{depth(s_1) + depth(s_2)} \quad \textbf{(Eq. 4)}$$

The score can never be zero because the depth of the LCS is never zero (the depth of the root of a taxonomy is one). The score is one if the two input synsets are the same.

Wu-Palmer similarity returns a score denoting how similar two word senses are, based on the depth of the two senses in the taxonomy and that of their most specific ancestor node.

An example of calculating similarities is shown in Figure 31. For generating similarities between two words a synset of word is used and to make the process fast the first synset in the list of synsets is used to calculate similarities.

```
>>> car=wn.synset('car.n.01')
>>> teacher=wn.synset('teacher.n.01')
>>> professor=wn.synset('professor.n.01')
>>> teacher.path_similarity(professor)
0.25
>>> teacher.path_similarity(car)
0.0666666666666666
>>> teacher.lch_similarity(professor)
2.2512917986064953
>>> teacher.lch_similarity(car)
0.92953595862417571
>>> teacher.lch_similarity(teacher)
3.6375861597263857
```

Figure 31: Getting sense similarity from WordNet

In Figure 32 the list of related words extracted for each NP is shown. In some classes where no words with the specified tags are found then the list of words will be empty. Therefore the commonsense knowledge cannot be used to calculate any similarities. The group of words for each NP will be compared with the group of words for each pronoun and similarity measures are calculated.

```
'Doc 3:: 1:1-2->A form; caused, form, percentage',
'Doc 3:: 1:14-16->a high percentage; caused, form, filters, cigarette, make, percentage, high, deaths, cancer, reported, researchers',
'Doc 3:: 1:18-19->cancer deaths; deaths, cancer, percentage, high, caused',
'Doc 3:: 1:21-22->a group; caused, form, filters, cigarette, make, percentage, high, deaths, cancer, reported, researchers',
'Doc 3:: 1:24-24->workers; ',
'Doc 3:: 1:28-31->more than 30 years; ',
'Doc 3:: 1:33-33->researchers; reported, researchers',
'Doc 3:: 1:4-4->asbestos; caused, form, make, filters, percentage',
'Doc 3:: 1:9-11->Kent cigarette filters; filters, Kent, cigarette, make',
'Doc 3:: 2:1-3->The asbestos fiber; fiber, asbestos, resilient',
'Doc 3:: 2:1-5->The asbestos fiber crocidolite; fiber, asbestos, resilient',
'Doc 3:: 2:11-12->the lungs; enters, it, lungs',
'Doc 3:: 2:14-16->even brief exposures; exposures, brief, enters, it, lungs, causing, symptoms',
'Doc 3:: 2:20-20->symptoms; causing, it, symptoms, show, decades',
'Doc 3:: 2:24-25->decades later; show, decades',
'Doc 3:: 2:26-26->researchers; said, researchers',
'Doc 3:: 2:4-4->crocidolite; fiber, asbestos, resilient',
'Doc 3:: 3:1-2->Lorillard Inc.; Lorillard, stopped, makes, unit',
'Doc 3:: 3:12-13->Kent cigarettes; cigarettes, Kent, makes, unit',
'Doc 3:: 3:16-16->crocidolite; using, crocidolite, filters, Micronite, cigarette',
'Doc 3:: 3:18-21->its Micronite cigarette filters; filters, Micronite, cigarettes, Kent, makes, cigarette, using, crocidolite',
'Doc 3:: 3:23-23->1956; using, crocidolite',
'Doc 3:: 3:3-4->the unit; makes, unit, cigarettes',
'Doc 3:: 3:6-9->New York-based Loews Corp.; based, New, Loews, makes, unit',
'Doc 3:: 4:31-32->the problem; bring, attention'])
```

Figure 32: Related words extracted for NPs

In this stage the commonsense knowledge sources (WordNet and ConceptNet) are used to generate three new similarity features.

WordNet Similarity Feature (F16): Path similarity, Leacock and Chodorow similarity (lch) and the Wu & Palmer measure (wup) are calculated and WordNet similarity is generated based on the Equation 5.

$$WNSim_{p,a} = PathSim_{p,a} \times lchSim_{p,a} \times WupSim_{p,a} \quad \textbf{(Eq. 5)}$$

Where *p* is the pronoun and *a* is the potential antecedent.

Path and wup similarities are both between 0 and 1 but lch similarity can be greater than one (as shown in Figure 31). Before using the lch score in Equation 5 it is important to normalize it. Normalized lch similarity between a and b is calculated using Equation 6.

$$NormalizedLchSim = \frac{lchSim_{a,b}}{lchSim_{a,a}} \quad \textbf{(Eq. 6)}$$

When calculating this feature the following rules are taken into consideration:

1. WordNet can only be used for words with the same part of speech tag such as nouns, adjective, adverbs and verbs. Therefore the similarity measure for words in different groups will be 0. If no path was found between the words the similarity measures will be -1 and therefore the WordNet similarity will return -1.
2. LCH similarity can return numbers greater than one (as shown in Figure 29), therefore they first need to be normalized and then combined with other measures.

ConceptNet Similarity Feature (F17): The result of ConceptNet similarity is normalized and ranges from 1 (exactly similar) to -1 (exactly dissimilar).

Activation Feature (F18): This feature uses the ConceptNet to indicate how related the two words are together by calculating the amount of activation that would spread from one concept to another (with a maximum of 1).

Figure 33 shows the new features added to the feature vectors. At this stage class feature will be added and then the data is ready for classification.

5.2 Classification

After the commonsense knowledge features have been extracted, they are combined with the linguistic features and a model is trained to detect the antecedents for pronouns in a sentence (Figure 33). Performance is measured using precision and recall accuracy score.

In the classification process, LibSVM, Decision Tree, Naïve Bayes and bagging classifiers are used. The classification results are shown in Table 14. Although the performance is still very high compared to the state of art systems, the performance has decreased slightly compared to when we only used the 15 features. The reduction in performance is 1% across the different classifiers. To analyze the reasons, feature analysis is performed to study the importance of the added features and generate solutions for solving this problem.

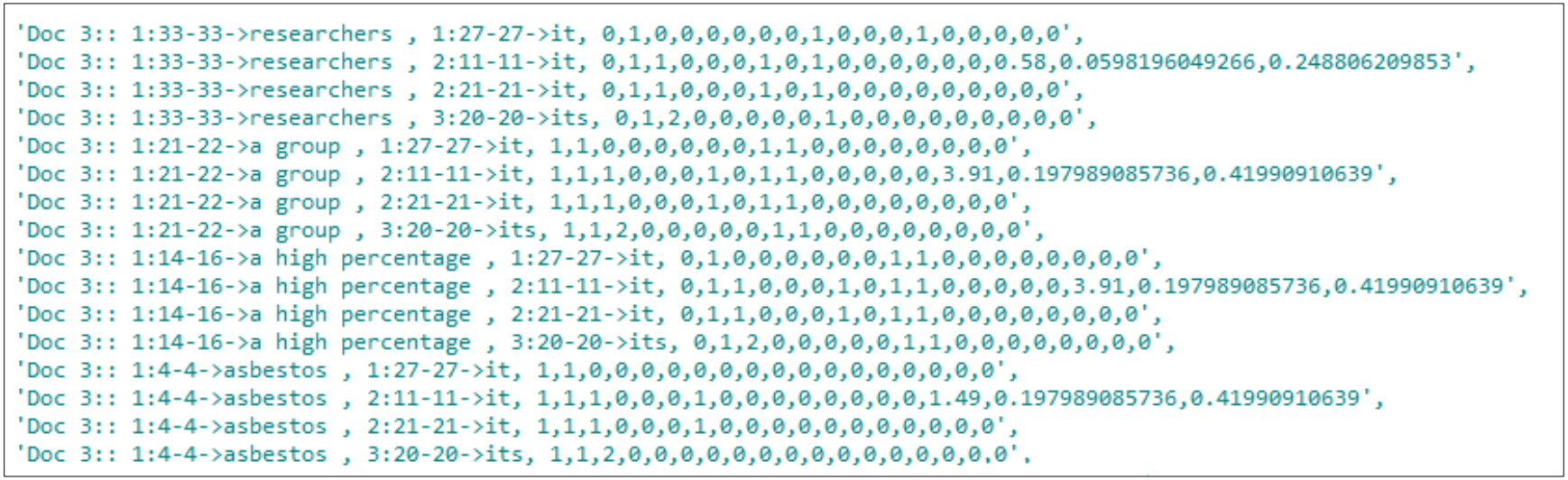


Figure 33: Feature vector after adding the three similarity features

Table 14: Classification results

Naïve Bayes *Time taken to train model:0.00 seconds*			
Class	Precision	Recall	F-Measure
1	0.869	0.875	0.872
0	0.874	0.869	0.871
All	0.872	0.872	**0.872**
SVM *Time taken to train model:3.52 seconds*			
Class	Precision	Recall	F-Measure
1	0.87	0.905	0.887
0	0.901	0.864	0.882
All	0.885	0.885	**0.884**
Random Forest *Time taken to train model:0.44 seconds*			
Class	Precision	Recall	F-Measure
1	0.862	0.888	0.875
0	0.884	0.858	0.871
All	0.873	0.873	**0.873**
Bagging using Libsvm *Time taken to train model:45.18 seconds*			
Class	Precision	Recall	F-Measure
1	0.874	0.896	0.885
0	0.893	0.871	0.882
All	0.883	0.883	**0.883**

5.2.1 Analysis and Result

In this section, the Chi-square feature selection technique is used to analyze the features used for classification. Feature analysis plays an important role since theoretically using commonsense knowledge is proven to improve the performance of reference resolution, but in here the performance has been reduced.

The results of the feature selection technique are given in Figure 34. The attribute evaluation shows that the features added in this chapter are highly ranked and therefore have a significant predictive power in the classification process.

By looking at the feature vectors and analyzing the NPs and pronouns related with each, many of the NPs and pronouns have no related words associated with them. This indicates that in the

dependency output of Stanford Parser, the tags that the program looks for are not found for some of the NPs and pronouns. For 24% of the data, commonsense knowledge was used and values were generated for F16, F17, and F18. In the cases where these features were not generated, a value of 0 doesn't necessarily mean that the words are not similar or related but it simply means that due to lack of information no values could have been calculated for these features.

This led to further analyzing those data that have a value for features 16-18. After studying the data the following rules are determined when dealing with commonsense knowledge sources:

1. ConceptNet similarity between each two word pair can have values for 1 to -1. 1 meaning exactly similar, and -1 meaning exactly dissimilar. By analyzing the data 80% of data in class 0 have negative or small values (less than 0.2) for this feature.
2. ConceptNet spread features shows how much activation would spread from one concept to another with a max of 1. By analyzing the data 70% of data in class 0 have negative or small values (less than 0.4) for this feature.
3. 66% of times where ConceptNet similarity and spread features were negative values, the data belonged to class 0.

The above points show that using this information can improve the performance of the system. But this information is not available for all the data and there are cases were due to lack of related words, ConceptNet and WordNet cannot be used. To solve this problem, the two models were fused together in a way that increases the performance of the system.

```
Attribute Evaluator (supervised, Class (nominal): 19 CLASS ):
        Chi-squared Ranking Filter

Ranked attributes:
2998.7387   3 DISTANCE
1126.6708   9 NPSUBJ
1085.7996   1 GENDER
 944.3385   4 PROPERNAME
 776.7695   2 NUMBER
 636.3614  18 ACTIVATION
 483.6253  17 CONCEPTNET
 469.9948  16 WORDNET
 430.8916  15 EXCITATION
 411.3461  13 COMMA
 233.0817  10 NPOBJ
  51.3661  14 SUBORDINATECLAUSE
   1.4575   7 PRPSUBJ
   1.2429   6 DEMONSTRATIVENP
   0.6738   5 DEFINITENP
   0.0437   8 PRPOBJ
   0       12 PREPOSITIONALCLAUSE
   0       11 SAMECLAUSE

Selected attributes: 3,9,1,4,2,18,17,16,15,13,10,14,7,6,5,8,12,11 : 18
```

Figure 34: Attribute ranking using Chi-squared ranking filter

5.3 Fused Model

For combining the multiple learning models that we have in our system, an evidence fusion model is used to give the final result. Using fusion methods will reduce the level of uncertainty.

The data is divided into two sections. The first section consists of those data that commonsense knowledge was able to calculate similarity features based on the group of related words extracted for each NP and pronoun. The second group consist of those data that due to lack of information commonsense knowledge returned zero and was not able to calculate similarity measures.

For the first group of data, the three commonsense knowledge features are combined with the 15 linguistic features for classification. For the second group of data, only the 15 linguistic features are used for classification.

Table 15 shows the classification results for the first group of data that have calculated values for the commonsense knowledge features using the 18 features. Commonsense knowledge features were able to increase the overall accuracy by 3%.

Table 15: Fused classification results

Naïve Bayes *Time taken to train model:0.05 seconds*			
Class	Precision	Recall	F-Measure
1	0.891	0.901	0.896
0	0.9	0.89	0.895
All	0.895	0.895	**0.895**
SVM *Time taken to train model:0.46 seconds*			
Class	Precision	Recall	F-Measure
1	0.847	0.918	0.881
0	0.91	0.834	0.87
All	0.878	0.876	**0.876**
Random Forest *Time taken to train model:0.94 seconds*			
Class	Precision	Recall	F-Measure
1	0.891	0.903	0.897
0	0.902	0.89	0.896
All	0.896	0.896	**0.896**
Bagging w/ Random Forest *Time taken to train model:7.13 seconds*			
Class	Precision	Recall	F-Measure
1	0.907	0.931	0.919
0	0.927	0.905	0.917
All	0.918	0.918	**0.918**

To summarize, in cases where there is enough information to use commonsense knowledge sources the performance is 92% F-Measure; in the remaining cases the methodology presented in Chapter 4 is used for resolving pronouns with a performance of 89% F-Measure.

Both classification methods were used on the BBN corpus and the overall F-measure was 90%.

5.3.1. Analysis

In this section the misclassified data are analyzed for the fused model. An overview of the error classifications is shown in Figure 35. More than 70% of the errors occur when resolving third person neutral pronouns (*it, its, them, they, their, themselves*). The remaining errors are caused when resolving male and female third person pronouns (*he, she*).

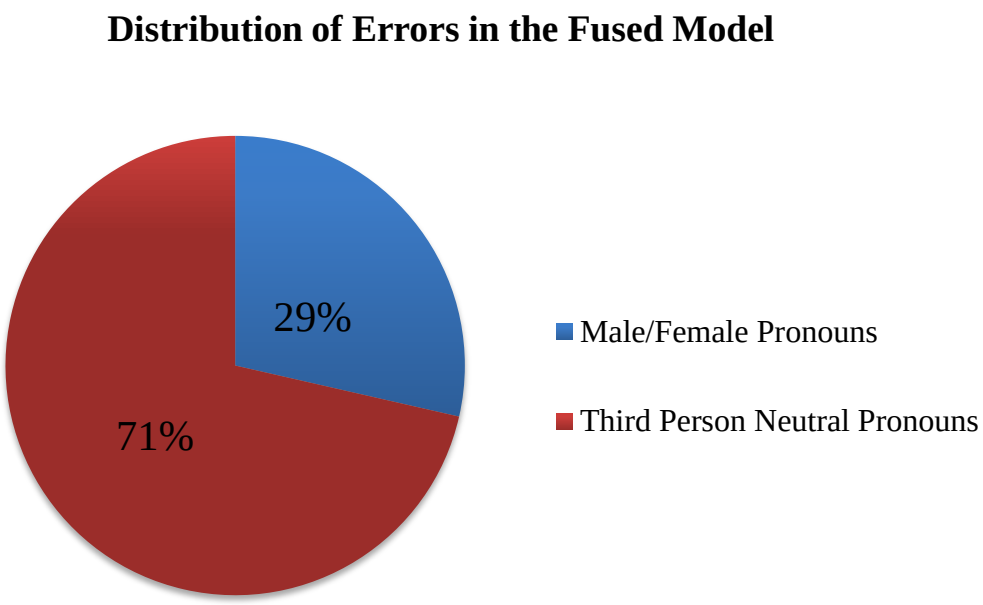


Figure 35: Fused error classification

The errors in third person neutral pronouns have been further analyzed to specify the reasons behind this misclassification. Different reasons causing misclassification are shown in Figure 36.

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. To solve this problem it is important to develop a stage in which non-anaphoric pronouns are first detected and removed from the list of

pronouns that need to be resolved. We are not considering detecting non anaphoric pronouns but if solved the performance will be improved.

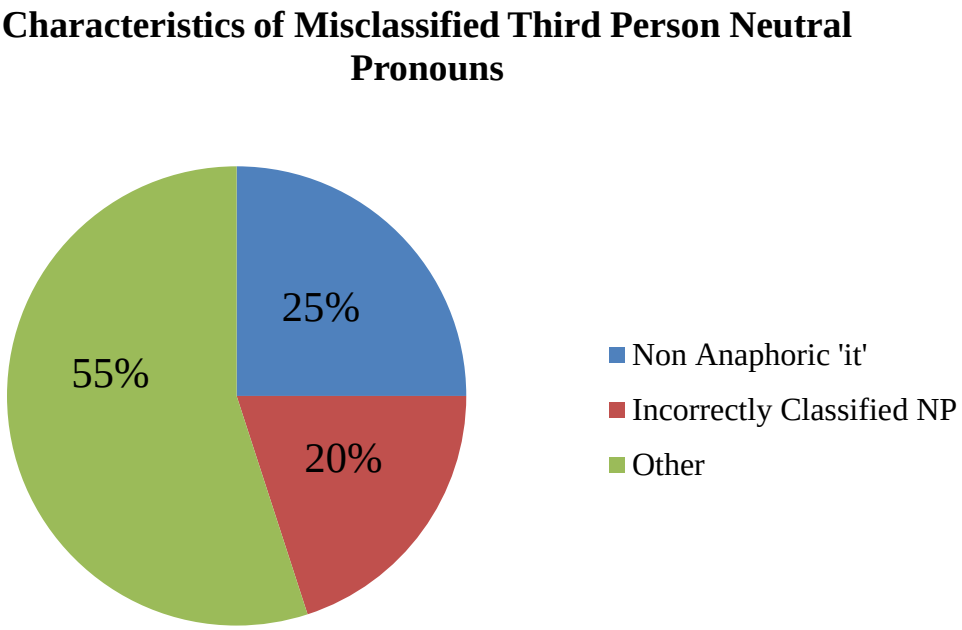


Figure 36: Distribution of errors in resolving third person neutral pronouns

Incorrectly Classified NP: Another reason that causes misclassification is related to the errors in the preprocessing engines. Stanford Parser and Charniak Parser are the preprocessing engines used in this system. These parsers are not 100% accurate, and therefore result in errors in parse tree and annotations. This leads to detecting incorrect Noun Phrases (NPs) and therefore misclassification occurs

The second group of errors occurs when resolving male/female pronouns. Figure 37 shows the main reasons causing misclassification. .

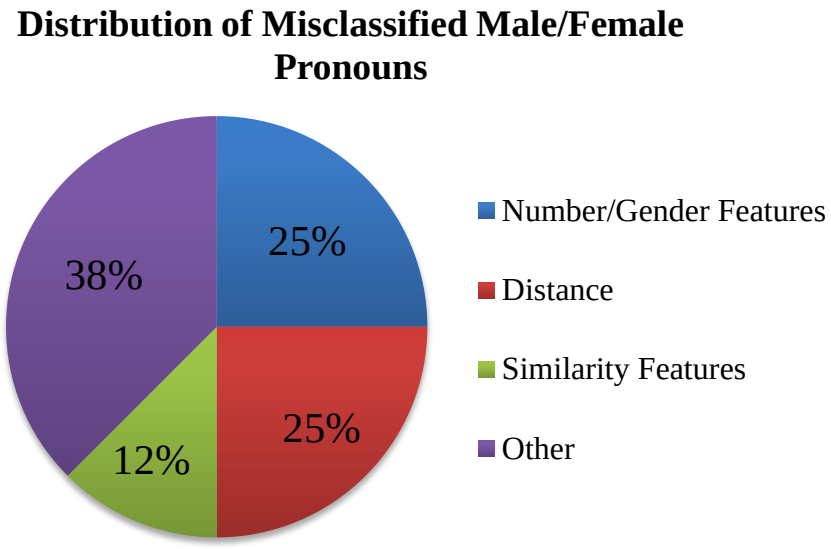


Figure 37: Distribution of errors in resolving male and female pronouns

Gender/Number Feature: 25% of errors 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: 25% of errors belong 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 doesn't classify them together.

Similarity Features: 12% of the errors are caused when the pronoun and NP agree in number and gender and have high similarity features, but don't refer together.

The mentioned points approve Bosch's saying that there are "no structurally stable restrictions on pronoun-antecedent pairs" and the grammatical formulae that have been proposed can fail in conditions (Bosch 1983).

5.4 Generalized Results

In this section the fusion model is used for anaphora resolution on text from different genres. This step will show the usability of the proposed methodology and whether it can be applied to different documents.

For this purpose samples from the following documents are used:

- Short informal stories from the web.
- Children's stories from the UIUC (University of Illinois at Urbana-Champaign) affect corpus (Alm 2008).
- News from the MUC-7 corpus.

For the above samples the system is trained on the feature vectors from the BBN corpus and then tested on each. Since Bagging Classifier showed the highest performance therefore is used for training and testing the data. Since the annotations for pronouns and their antecedents were not available the above documents were annotated manually and then compared to the results of the classifier.

5.4.1 Short Stories from Web

Random short stories were selected from the web to test the system. For this purpose the following stories have been selected:

- ***"The Worst Way to Go" by Dan Morgan (F-measure: 85%)***
    - 150 sentences (shortest sentence 3 words, longest sentence 76 words)
    - Total number of pronouns: 832

        - First person pronouns[12] (*I, me, you, myself*): 507.
        - Third person male and female pronouns (*he, him, she, her*): 156
        - Third person neutral pronouns (*it, they, them*): 169

- ***"Return to Paradise" by Eliza Riley (F-measure: 83%)***
    - 49 sentences (shortest sentence 7 words, longest sentence 45 words)
    - Total number of pronouns: 128
        - First person pronouns (*I*): 10
        - Third person male and female pronouns (*he, him, his, she, her*): 89
        - Third person neutral pronouns (*it, itself, they, them*): 29

Since the style of writing was not as formal as news documents used for training, and the sentences were usually very long, Stanford Parser generated incorrect noun phrases. The following are examples of sentences with incorrectly labeled NPs (shown in underlined italics).

Sentence 1: "Back when I was working nights at the downtown bus station for *a while sweeping floors* and scrubbing out toilets and pinching gum off the bottom of seats and picking up other people's trash and having lunch at the counter where they had good meatloaf and a waitress named Holly that I tried to screw but never got to, one night when I just got off work something happened that I will not never forget."

Sentence 2: "He used to tell me all about going off places with his biker friends and how him and his old lady would sometimes take off on a weekend and take blankets with them and ride off over in the Hill Country where they'd camp out by *a river and smoke dope* and screw a lot and just watch the sun coming up."

[12] This group of pronouns are not studied here.

Sentence 3: "I sure wished lots of times that I had my own Harley that I could jump on when things got a bit too tight or there was too many people yelling at *me or maybe just* when the toilet was all backed up at the station and they sent me in to clean it all out and mop up the piss on the floors."

5.4.2 Stories from UIUC

Another genre of document that was selected for testing the system was children's fairy tale stories from the UIUC affect corpus. The following stories were selected:

- ***The Story of a Fierce Bad Rabbit (F-measure: 78%)***
  - 18 sentences (between 3 to 18 words for each sentence)
  - Total number of pronouns: 20
    - Third person male and female pronouns (*he, him, his*): 15
    - Third person neutral pronouns (*it, its*): 5
- ***The Story of Miss Moppet (F-measure: 81%)***
  - 18 sentences (between 5 to 25 words for each sentence)
  - Total number of pronouns: 22
    - Third person male and female pronouns (*he, him, she, her*): 20
    - Third person neutral pronouns (*it*): 2
- ***The Tale of Mr. Jeremy Fisher (F-measure: 84%)***
  - 47 sentences (between 1 to 46 words for each sentence)
  - Total number of pronouns: 84
    - First person pronouns (*I, my*): 26
    - Third person male and female pronouns (*he, him*): 46
    - Third person neutral pronouns (*it, they, them, their*): 12

- ***The Tale of Tom Kitten (F-measure: 87%)***
    - 49 sentences (between 1 to 41 words for each sentence)
    - Total number of pronouns: 72
        - First person pronouns (*I*): 5
        - Third person male and female pronouns (*he, him, she, her*): 32
        - Third person neutral pronouns (*it, they, them*): 35

Among the selected stories, there were stories that had short sentences with many characters which were usually different animals. This caused errors in specifying the gender of these NPs. For this reason the accuracy of the system decreased on this group of documents. The performance for the above stories ranged from 78% to 87% and was mainly related on the length of the story and how easy the characters were distinguishable.

Unless the characters were specified as for example "Miss Rabbit" or "Mr. Bear", it was hard to specify the gender of the NP. Also in many cases, due to lack of related words, little information was extracted from commonsense knowledge sources. But in cases where the stories were longer and the gender of actors was easier to specify, the performance increased.

5.4.3 MUC-7

The system was also tested on the MUC-7 corpus, which is widely used among researchers for comparison between different coreference resolution methodologies. The following samples were used from the corpus to test the system:

- ***"FAA Underestimated Number of Flights over Plutonium Storage Area in Panhandle",*** by Hollace Weiner---- Doc ID: nyt960214.0765 ***(F-measure: 57%)***
    - 26 sentences (between 7 to 46 words for each sentence)

- Total number of pronouns: 7
    - Third person male and female pronouns (*he*): 1
    - Third person neutral pronouns (*it, its, their*): 6

- ***"BC-F14-Crashes-Bloom Northrop Grumman F-14s to Stand Down for Review of Accidents",*** by Bill Arthur---- Doc ID: nyt960222.0269 ***(F-measure: 83%)***
    - 25 sentences (between 5 to 37 words for each sentence)
    - Total number of pronouns: 8
        - First person pronouns (*me, we*): 2
        - Third person male and female pronouns (*he*): 4
        - Third person neutral pronouns (*they*): 6

MUC-7 is prepared in a way that is used for coreference resolution. Although pronominal anaphora resolution is a subset of coreference resolution, but the documents used in this corpus don't have many personal pronouns. As stated in the above the number of pronouns is very low compared to children stories and short stories. Therefore this corpus was not a particularly good evaluator for our system since the number of personal pronouns is very low. The average performance of the system on the above documents was 70% which is lower that what was expected.

Table 17 shows the summary of the results on the different genres of document.

5.5 Time Analysis

In this section a time analysis is done on the different stages of the anaphora resolution to make sure that the system can perform in real time. Table 16 shows the time break down for different stages of the process.

Table 16: Time breakdown for each stage

Stage	**Time**
Preprocessing	3min 16 sec
Feature Generation	1 min 5 sec
Similarity Features	2 min 34 sec
Class Generation	0 min 56 sec
Training the model	0 min 0.2 sec
Classification on the testing data	0 min 0.1 sec
Total Time	8 min 3 sec

The results of time analysis show that 41% of the total time is taken in the preprocessing stage. The computer used for testing the system is a desktop computer with 4GB Ram and Intel dual core processor. The time can be reduced by using a faster processor. Also the parsers are queried through Python and this causes increase in preprocessing time. The other stage is generating similarity features which take 32% of the total time. Connecting to WordNet and ConceptNet and searching words and calculating similarities is a time consuming task. But this time can also be reduced by using faster processors.

Table 17: Summary of results on additional test documents

Name	Type	# of sent	# of pronouns	# of Male female pronouns	# of neutral pronouns	F-measure
"The Worst Way to Go" by Dan Morgan	Short story	150	832	156	169	**85%**
"Return to Paradise" by Eliza Riley	Short story	49	128	89	29	**83%**
The Story of a Fierce Bad Rabbit	UIUC	18	20	15	5	**78%**
The Story of Miss Moppet	UIUC	18	22	20	2	**81%**
The Tale of Mr. Jeremy Fisher	UIUC	47	84	46	12	**84%**
The Tale of Tom Kitten	UIUC	49	72	32	35	**87%**
FAA Underestimated Number of Flights	MUC7	26	7	1	6	**57%**
BC-F14-Crashes-Bloom	MUC7	25	8	4	6	**83%**

CHAPTER 6: CONCLUSIONS AND FUTURE WORK

This dissertation studies the automated pronominal anaphora resolution system in text. The aim was to bridge the gap between the theory and practice and incorporate the linguistic knowledge in an anaphora resolution system. Computational and linguistic studies were combined and an evidence fusion model was developed that combined learning-based and rule-based algorithm with commonsense knowledge.

Theoretical studies show that there are different linguistic factors relevant for anaphora resolution and not all the state of art approaches incorporate these factors. Reference resolution task is an important topic and has been addressed in the literature widely, but the existing algorithms for both anaphora resolution and 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 and that the state of art reference resolution systems cannot successfully handle them.

Two resolution methodologies were developed and the results were discussed:

First, 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. The results show major improvement compared to the state of art systems.

Second, commonsense knowledge sources such as WordNet and ConceptNet were used to extract more information from the document and use it to uncover elaborative information embedded in the anaphor.

Finally, the two methodologies developed were combined in an evidence fusion system and were tested and evaluated on BBM Pronoun Coreference corpus as well as sample short stories from web, children's stories from UIUC and samples from MUC 7 corpus, to investigate its usability and performance with other available algorithms. The methodologies developed in this work can serve as a guide for future developments in information extraction and text analysis systems.

The important contributions of this work include:

- The development of a learning-based and rule-based algorithm using features based on theoretical and linguistic studies on anaphora.
- The development of an algorithm using commonsense knowledge sources to gain the information needed to perform as close to a human brain as possible.
- The development of a fusion system that combines the information from the two approaches to resolve pronominal pronouns in text.

By improving the task of anaphora resolution the research aims to have an effective impact on other tasks such as text understanding, document summarization, information extraction, machine translation and etc.

6.1 Recommendations for Future Work

The two main groups of errors in our system were caused by incorrectly generated NPs and when the system tried to resolve non-anaphoric 'it'. The parsers available are not 100% accurate and therefore using any parser will result in generating incorrect NPs. To solve the problem a stage should be specified that would check the NPs and filter the incorrect one.

Future work will focus on improving the process of detecting and resolving non-anaphoric pronouns in a sentence. By incorporating a stage that will detect these pronouns the accuracy will be increased. This can be done by either specifying rules that can distinguish between anaphoric and non-anaphoric pronouns or by developing a set of features that can be used for identifying non-anaphoric pronouns.

The excitation feature helped in the process of identifying antecedents, but in some cases it also caused errors. This feature was incremented when the entire NP or its head noun was found in previous sentences. In the future this feature can be modified so only the times where the NP has been mentioned and also has a high semantic presence in the sentence.

Additionally, future work will focus on analyzing characteristics of different genre of documents and adding features such as speech features that will improve the performance of the system and can be used for any type of text with any style of writing.

Also, in the future the pronoun chains will be tracked. This will prevent losing any information regarding each pronoun. This way information will be built for each pronoun throughout the document and this will increase the performance of anaphora resolution.

REFERENCES

1. Abdul-Mageed, Muhammad. "Automatic Detection of Arabic Non-Anaphoric Pronouns for Improving Anaphora Resolution." ACM Transactions on Asian Language Information Processing (TALIP) 10, no. 1 (2011): 5.

2. Akmajian, Adrian, and Ray Jackendoff. "Coreferentiality and Stress." Linguistic Inquiry 1, no. 1 (1970): 124-26.

3. Alm, Ebba Cecilia Ovesdotter. "Affect in Text and Speech." UIUC, 2008.

4. Baker, Collin F., Charles J. Fillmore, and John B. Lowe. "The Berkeley Framenet Project." Paper presented at the 17th international conference on Computational linguistics, 1998.

5. Balahur, Alexandra, Jesús M. Hermida, Andrés Montoyo, and Rafael Muñoz. "Emotinet: A Knowledge Base for Emotion Detection in Text Built on the Appraisal Theories." Paper presented at the 16th international conference on Natural language processing and information systems, 2011.

6. Baldwin, Breck. "Cogniac: High Precision Coreference with Limited Knowledge and Linguistic Resources." Paper presented at the Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts, 1997.

7. Barbu, Catalina, and Ruslan Mitkov. "Evaluation Tool for Rule-Based Anaphora Resolution Methods." Paper presented at the 39th Annual Meeting on Association for Computational Linguistics, 2001.

8. Barwise, Jon, and John Perry. Situations and Attitudes. MIT Press 1983.

9. Bean, David, and Ellen Riloff. "Unsupervised Learning of Contextual Role Knowledge for Coreference Resolution." Paper presented at the HLT/NAACL, 2004.

10. Bergsma, Shane, and Dekang Lin. "Bootstrapping Path-Based Pronoun Resolution." Paper presented at the COLING/ACL, 2006.

11. Bergsma, Shane, Dekang Lin, and Randy Goebel. "Distributional Identification of Non-Referential Pronouns." Paper presented at the ACL-HLT, 2008.

12. Bezdek, James C., James Keller, Raghu Krishnapuram, and Mikhil R. Pal. Fuzzy Models and Algorithms for Pattern Recognition and Image Processing. Kluwer Academic Publishers Norwell, 1999.

13. Bezdek, James C., Mikhil R. Pal, James Keller, and Raghu Krisnapuram. Fuzzy Models and Algorithms for Pattern Recognition and Image Processing. Vol. 4: Kluwer Academic Publishers Norwell, 2005.

14. Bikel, Daniel M. "A Distributional Analysis of a Lexicalized Statistical Parsing Model." In Empirical Methods in Natural Language Processing (EMNLP), 2004.

15. Bird, Steven, Ewan Klein, and Edward Loper. Natural Language Processing with Python. O'Reilly Media, 2009.

16. Bolinger, D. "Pronouns and Repeated Nouns, Lulc." Bloomington, 1977.

17. Bosch, Peter Agreement and Anaphora: A Study of the Role of Pronouns in Syntax and Discourse. Acad. Press, 1983. The University of Michigan.

18. Bosch, Peter. "Pronouns under Control?". Journal of Semantics 5, no. 1 (1986): 65-78.

19. Bouckaert, Remco R., Eibe Frank, Mark A. Hall, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. "Weka---Experiences with a Java Open-Source Project." The Journal of Machine Learning Research 11 (2010): 2533-41.

20. Brennan, Susan E., Marilyn W. Friedman, and Carl J. Pollard. "A Centering Approach to Pronouns." Paper presented at the 25th annual meeting on Association for Computational Linguistics, 1987.

21. Budanitsky, Alexander, and Graeme Hirst. "Evaluating Wordnet-Based Measures of Semantic Distance." Computational Linguistics 32, no. 1 (2006): 13-47.

22. Burges, Christopher J.C. "A Tutorial on Support Vector Machines for Pattern Recognition." Data Mining and Knowledge Discovery 2, no. 2 (1998): 121-67.

23. Calix, Ricardo A. "Automated Semantic Understanding of Human Emotions in Writing and Speech." Louisiana State University, 2011.

24. Cann, Ronnie, Ruth M. Kempson, and Lutz Marten. The Dynamics of Language: An Introduction. Vol. 35: Elsevier Academic Press, 2005.

25. Cer, Daniel, Marie-Catherine De Marneffe, Daniel Jurafsky, and Christopher Manning. "Parsing to Stanford Dependencies: Trade-Offs between Speed and Accuracy." Paper presented at the Language Resources Evaluation Conference (LREC-10), 2010.

26. Chang, Chih-Chung, and Chih-Jen Lin. "Libsvm: A Library for Support Vector Machines." ACM Transactions on Intelligent Systems and Technology 2, no. 3 (2011): 1-27

27. Charniak, Eugene. "A Maximum-Entropy-Inspired Parser." Paper presented at the 1st North American chapter of the Association for Computational Linguistics conference, 2000.

28. Charniak, Eugene, and Micha Elsner. "Em Works for Pronoun Anaphora Resolution." Paper presented at the 12th Conference of the European Chapter of the Association for Computational Linguistics, 2009.

29. Charniak, Eugene, and Mark Johnson. "Coarse-to-fine N-Best Parsing and Maxent Discriminative Reranking." Paper presented at the 43rd Annual Meeting of the ACL, 2005.

30. Cherry, C., and S. Bergsma. "An Expectation Maximization Approach to Pronoun Resolution." Paper presented at the Ninth Conference on Computational Natural Language Learning, 2005.

31. Chinchor N., Sundheim B. "Message Understanding Conference (Muc) 6." edited by Linguistic Data Consortium. Philadelphia, 2003.

32. Chomsky, Noam. Lectures on Government and Binding: The Pisa Lectures. Vol. 9: Walter de Gruyter, 1993.

33. Culotta, Aron, Michael Wick, Robert Hall, and Andrew McCallum. "First-Order Probabilistic Models for Coreference Resolution." Paper presented at the HLT-NAACL, 2007.

34. Dagan, Ido, and Alon Itai. "Automatic Processing of Large Corpora for the Resolution of Anaphora References." Paper presented at the 13th conference on Computational linguistics, 1990.

35. Danesh, Ali, Behzad Moshiri, and Omid Fatemi. "Improve Text Classification Accuracy Based on Classifier Fusion Methods." Paper presented at the 10th International Conference on Information Fusion, 2007.

36. De Marneffe, Marie-Catherine, Bill MacCartney, and Christopher D. Manning. "Generating Typed Dependency Parses from Phrase Structure Parses." Paper presented at the LREC, 2006.

37. De Marneffe, Marie-Catherine, Bill MacCartney, and Christopher D Manning. "The Stanford Typed Dependencies Representation." Paper presented at the Cross-Framework and Cross-Domain Parser Evaluation. Association for Computational Linguistics, 2008.

38. Eisenstein, J., R. Barzilay, and R. Davis. "Gesture Salience as a Hidden Variable for Coreference Resolution and Keyframe Extraction." Journal of Artificial Intelligence Research 31, no. 1 (2008): 353-98.

39. Fellbaum, C. "Wordnet." Theory and Applications of Ontology: Computer Applications (2010): 231-43.

40. Fernández, R. "Incremental Resolution of Relative Adjectives: A Drt-Based Approach." Paper presented at the Constraints in Discourse, Agay, France, 2011.

41. Fillmore, Charles J., Collin F. Baker, and Hiroaki Sato. "The Framenet Database and Software Tools." Paper presented at the European Association for Lexicography, 2002

42. Ge, Niyu , John Hale, and Eugene Charniak. "A Statistical Approach to Anaphora Resolution." Paper presented at the Sixth Workshop on Very Large Corpora, 1998.

43. Grosz, B.J., S. Weinstein, and A.K. Joshi. "Centering: A Framework for Modeling the Local Coherence of Discourse." Computational Linguistics 21, no. 2 (1995): 203-25.

44. Havasi, C. "Conceptnet 3: A Flexible, Multilingual Semantic Network for Common Sense Knowledge." Paper presented at the 22nd Conference on Artificial Intelligence, 2007.

45. Hobbs J., Montazeri N. "The Deep Lexical Semantics of Event Words." edited by R. Osswald (ed.): University of Southern California, 2013.

46. Hobbs, J.R. "Resolving Pronoun References." Lingua 44, no. 4 (1978): 311-38.

47. Hoffart, J., F. Suchanek, K. Berberich, E. Kelham, G. de Melo, G. Weikum, G. Kasneci, M. Ramanath, and A. Pease. "Yago2: A Spatially and Temporally Enhanced Knowledge Base from Wikipedia." Commun. ACM 52, no. 4 (2009): 56-64.

48. Hsu, M.H., M.F. Tsai, and H.H. Chen. "Combining Wordnet and Conceptnet for Automatic Query Expansion: A Learning Approach." Paper presented at the 4th Asia information retrieval conference on Information retrieval technology, 2008.

49. Jain, A.K., M.N. Murty, and P.J. Flynn. "Data Clustering: A Review." ACM computing surveys (CSUR) 31, no. 3 (1999): 264-323.

50. Jia, M., D. Zheng, B. Yang, and Q. Chen. "Hierarchical Text Categorization Based on Multiple Feature Selection and Fusion of Multiple Classifiers Approaches." Paper presented at the Sixth International Conference on Fuzzy Systems and Knowledge Discovery, 2009.

51. Kameyama, Megumi. "Intrasentential Centering: A Case Study." Centering Theory in Discourse (1997).

52. Kehler, A., D. Appelt, L. Taylor, and A. Simma. "Competitive Self-Trained Pronoun Interpretation." Paper presented at the HLT/NAACL, 2004.

53. Kim, Su Nam, and Timothy Baldwin. "Automatic Interpretation of Noun Compounds Using Wordnet Similarity." Paper presented at the Second International Joint Conference on Natural Language Processing (IJCNLP-05), Jeju, South Korea, 2005.

54. Kittler, J., M. Hatef, R.P.W. Duin, and J. Matas. "On Combining Classifiers." IEEE Transactions on Pattern Analysis and Machine Intelligence 20, no. 3 (1998): 226-39.

55. Klein, Dan, and Christopher D. Manning. "Accurate Unlexicalized Parsing." Paper presented at the The 41st Meeting of the Association for Computational Linguistics, 2003.

56. Kotsiantis, S.B. "Supervised Machine Learning: A Review of Classification Techniques." Informatica 31 (2007): 249-68.

57. Lakoff, R.T. Abstract Syntax and Latin Complementation. The MIT Press, 1968.

58. Langacker, R.W. on Pronominalization and the Chain of Command. Englewood Cliffs, 1966.

59. Lappin, S., and H.J. Leass. "An Algorithm for Pronominal Anaphora Resolution." Computational Linguistics 20, no. 4 (1994): 535-61.

60. Li, Yifan. "Web-Assisted Anaphora Resolution." University of Alberta, 2010.

61. Liu, H., and P. Singh. "Conceptnet—a Practical Commonsense Reasoning Toolkit." BT technology journal 22, no. 4 (2004): 211-26.

62. Löbner, S. "Definites." Journal of Semantics 4, no. 4 (1985): 279.

63. Lu, C.Y., S.H. Lin, J.C. Liu, S. Cruz-Lara, and J.S. Hong. "Automatic Event-Level Textual Emotion Sensing Using Mutual Action Histogram between Entities." Expert Systems with Applications 37, no. 2 (2010): 1643-53.

64. Manjuan, D., and J. Ping. "An Empirical Study of Centering in Chinese Anaphoric Resolution." Paper presented at the Artificial Intelligence and Computational Intelligence (AICI), 2010.

65. Marcus, M.P., M.A. Marcinkiewicz, and B. Santorini. "Building a Large Annotated Corpus of English: The Penn Treebank." Computational Linguistics 19, no. 2 (1993): 313-30.

66. Marcus, Mitchell P., Beatrice Santorini, Mary Ann Marcinkiewicz, and Ann Taylor. "Treebank-3." edited by Linguistic Data Consortium. Philadelphia, 1999.

67. Markert, K., and U. Hahn. "On the Interaction of Metonymies and Anaphora." Paper presented at the Fifteenth international joint conference on Artifical intelligence, 1997.

68. Miller, G.A., R. Beckwith, C. Fellbaum, D. Gross, and K.J. Miller. "Introduction to Wordnet: An on-Line Lexical Database*." International journal of lexicography 3, no. 4 (1990): 235-44.

69. Miltsakaki, E. "Demo of Antelogue: Pronoun Resolution for Dialogues." Paper presented at the Semantic Computing, 2009.

70. Mitchell, Alexis , Stephanie Strassel, Mark Przybocki, JK Davis, George Doddington, Ralph Grishman, Adam Meyers, et al. "Tides Extraction (Ace) 2003 Multilingual Training Data." edited by Linguistic Data Consortium. Philadelphia, 2004.

71. Mitkov, R. Anaphora Resolution. Longman London, 2002.

72. Mitkov, R., R. Evans, and C. Orasan. "A New, Fully Automatic Version of Mitkov's Knowledge-Poor Pronoun Resolution Method." Computational Linguistics and Intelligent Text Processing (2002): 69-83.

73. Morton, T., J. Kottmann, J. Baldridge, and G. Bierner. "Opennlp: A Java-Based Nlp Toolkit." EACL, 2005.

74. Nard, P. "Resolving Co-Reference Anaphora Using Semantic Distance." 2012.

75. Ng, V., and C. Cardie. "Improving Machine Learning Approaches to Coreference Resolution." Paper presented at the 40th Annual Meeting on Association for Computational Linguistics, 2002.

76. Ning, P., and S. Jun-feng. "The Third Personal Pronoun Anaphora Resolution in the Paroxysmal Text of the Chinese Web." Paper presented at the Computer Application and System Modeling (ICCASM), 2010.

77. Pala, K., and R. Begum. "An Experiment on Resolving Pronominal Anaphora in Hindi: Using Heuristics." Information Systems for Indian Languages (2011): 267-70.

78. Park, K.S., D.U. An, and Y.S. Lee. "Anaphora Resolution System for Natural Language Requirements Document in Korean." Paper presented at the Information and Computing (ICIC), 2010.

79. Pedersen, Ted, Siddharth Patwardhan, and Jason Michelizzi. "Wordnet::Similarity - Measuring the Relatedness of Concepts." Paper presented at the HLT-NAACL, July 25-29, 2004 2004.

80. Petrov, S., L. Barrett, R. Thibaux, and D. Klein. "Learning Accurate, Compact, and Interpretable Tree Annotation." Paper presented at the Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, 2006.

81. Poesio, M., and M.A. Kabadjov. "A General-Purpose, Off-the-Shelf Anaphora Resolution Module: Implementation and Preliminary Evaluation." Paper presented at the LREC, 2004.

82. Poesio, M., and H. Rieser. "Completions, Coordination, and Alignment in Dialogue." Dialogue & Discourse 1, no. 1 (2010).

83. Poesio, M., and H. Rieser. "An Incremental Model of Anaphora and Reference Resolution Based on Resource Situations." Dialogue & Discourse 2, no. 1 (2011): 235-77

84. Poesio, M., and D.R. Traum. "Conversational Actions and Discourse Situations." Computational intelligence 13, no. 3 (1997): 309-47.

85. Polikar, R. "Ensemble Based Systems in Decision Making." Circuits and Systems Magazine 6, no. 3 (2006): 21-45.

86. Ponzetto, S.P., and M. Strube. "Deriving a Large Scale Taxonomy from Wikipedia." Paper presented at the 22nd national conference on Artificial intelligence, 2007.

87. Ponzetto, S.P., and M. Strube. "Exploiting Semantic Role Labeling, Wordnet and Wikipedia for Coreference Resolution." Paper presented at the HLT/NAACL, 2006.

88. Poon, H., and P. Domingos. "Joint Unsupervised Coreference Resolution with Markov Logic." Paper presented at the Empirical Methods in Natural Language Processing, 2008.

89. Reinhart, T. "The Syntactic Domain of Anaphora." Massachusetts Institute of Technology, 1976.

90. Richardson, M., and P. Domingos. "Markov Logic Networks." Machine Learning 62, no. 1 (2006): 107-36.

91. Ross, John Robert. "Constraints on Variables in Syntax." 1967.

92. Ruta, D., and B. Gabrys. "An Overview of Classifier Fusion Methods." Computing and Information systems 7, no. 1 (2000): 1-10.

93. Soon, W.M., H.T. Ng, and D.C.Y. Lim. "A Machine Learning Approach to Coreference Resolution of Noun Phrases." Computational Linguistics 27, no. 4 (2001): 521-44.

94. Spagnola S. , Lagoze C. "Edge Dependent Pathway Scoring for Calculating Semantic Similarity in Conceptnet." Paper presented at the The Ninth International Conference on Computational Semantics, Stroudsburg, PA, 2011.

95. Speer, R., C. Havasi, and H. Lieberman. "23rd National Conference on Artificial Intelligence." Paper presented at the AAAI Conf. Artificial Intelligence, 2008 2008.

96. Stark, M.M., and R.F. Riesenfeld. Wordnet: An Electronic Lexical Database. A Bradford Book, 1998.

97. Stoyanov, V., C. Cardie, N. Gilbert, E. Riloff, D. Buttler, and D. Hysom. "Coreference Resolution with Reconcile." Paper presented at the ACL Conference, 2010.

98. Stoyanov, V., C. Cardie, N. Gilbert, E. Riloff, D. Buttler, and D. Hysom. "Reconcile: A Coreference Resolution Research Platform." 2010.

99. Strube, M., and C. Müller. "A Machine Learning Approach to Pronoun Resolution in Spoken Dialogue." Paper presented at the 41st Annual Meeting on Association for Computational Linguistics, 2003.

100. Suchanek, F.M., G. Kasneci, and G. Weikum. "Yago: A Core of Semantic Knowledge." Paper presented at the 16th international conference on World Wide Web, 2007.

101. Szarvas, G., T. Zesch, and I. Gurevych. "Combining Heterogeneous Knowledge Resources for Improved Distributional Semantic Models." Computational Linguistics and Intelligent Text Processing (2011): 289-303.

102. Tetreault, J.R. "Analysis of Syntax-Based Pronoun Resolution Methods." Paper presented at the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics, 1999.

103. Tonelli, S., and R. Delmonte. "Venses++: Adapting a Deep Semantic Processing System to the Identification of Null Instantiations." Paper presented at the 5th International Workshop on Semantic Evaluation, 2010.

104. Versley, Y., S.P. Ponzetto, M. Poesio, V. Eidelman, A. Jern, J. Smith, X. Yang, and A. Moschitti. "Bart: A Modular Toolkit for Coreference Resolution." Paper presented at the Companion Volume of the Proceedings of the 46th Annual Meeting of the Association for Compuatational Linguistics, 2008.

105. Webber, B. L. "Discourse Deixis and Discourse Processing." University of Pennsylvania, 1988.

106. Weischedel, Ralph, and Ada Brunstein. "BBN Pronoun Coreference and Entity Type Corpus." edited by Linguistic Data Consortium. Philadelphia, 2005.

107. Witten, Ian H., Eibe Frank, and Mark A. Hall. Data Mining: Practical Machine Learning Tools and Techniques. The Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann, 2011.

108. Yang, X., J. Su, and C.L. Tan. "Kernel-Based Pronoun Resolution with Structured Syntactic Knowledge." Paper presented at the COLING/ACL 2006.

109. Yang, Xiaofeng , Jian Su, and Chew Lim Tan. "Kernel-Based Pronoun Resolution with Structured Syntactic Knowledge." Paper presented at the 21st International Conference

on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, 2006.

110. Yang, Y., Pedersen J.P. "A Comparative Study on Feature Selection in Text Categorization." Paper presented at the Fourteenth International Conference on Machine Learning, 1997.

VITA

Leili Javadpour received her bachelor's degree at Isfahan University of Technology in Industrial Engineering in 2007. Thereafter, she went to Liverpool to study Masters and received her degree in Product Design and Management at University of Liverpool in 2009. In January of 2010, she started graduate studies in the college of engineering at Louisiana State University (LSU) as a research assistant in the Industrial Engineering program. During her years at LSU she has worked as an instructor and teacher assistant for the Industrial Engineering program and as a graduate assistant at E.J. Ourso College of Business.

She is a candidate for the Doctor of Philosophy degree in Engineering Science with concentration in Information Technology and Engineering. The degree will be conferred at the summer commencement 2013.