Comparing Supervised and Unsupervised Classification of Messages in the Enron Email Corpus

A THESIS
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Apurva Padhye

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

August 2006

UNIVERSITY OF MINNESOTA

This is to certify that I have examined this copy of master's thesis by

Apurva Padhye

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Dr. Ted Pedersen

_____

Name of Faculty Adviser

_____

Signature of Faculty Advisor

_____

Date

GRADUATE SCHOOL

Abstract

The Enron e-mail corpus has recently become available to the research community as a consequence of various legal actions involving the Enron Corporation. It consists of 517,431 messages sent and received by 151 ex-employees of Enron over a period of three and a half years. This is a unique resource, since in general e-mail is considered very private, and real-life e-mail is usually not available for study.

We have manually annotated a 3,021 message subset of this data, and carried out supervised and unsupervised learning experiments on it to see how well existing methods of text classification apply to e-mail. We find that supervised methods are able to perform with relatively high accuracy when making coarse grained distinctions in topics, but have difficulty with more fine grained distinctions. In particular, we find that Support Vector Machines are consistently among the most accurate.

Contents

List of Tables

List of Figures

1 Introduction

In today's world, communicating with others via e-mail has become an integral part of life. It is hard to find a college student, professional, or any educated person for that matter, who does not send and/or receive e-mails. Also, it is an established fact that a lot of the communication that occurs within companies and organizations is nowadays done by e-mails, rather than memos or common bulletin boards. However, the use of e-mail is not only restricted to professional or academic settings. It is widely used in more informal sorts of environments too. In fact, such is the impact of e-mail in our daily lives that even people living in the same house tend to send each other notes via e-mail. As such, any average person today heavily depends on e-mail as a fast, reliable and efficient means of communication.

Given the wide scale usage of e-mail, it is evident that some form of sorting or categorizing of e-mail is necessary. In fact, as deduced by Klimt and Yang [1] most e-mail users adopt some kind of foldering strategy. The granularity of the categorization and the depth of the folders tend to vary from user to user. The sorting of e-mails into various user-defined folders also varies from person to person. This categorization of e-mails into different folders defined by individual users can be a tiresome task if it is not done regularly. If there were some way to automate this cumbersome task, we would save a lot of human effort.

This sets the stage for our thesis. Our thesis focuses on *Comparing Supervised and Unsupervised Classification of Messages in the Enron Email Corpus*. At this point, let us first see what text classification is. Traditional text classification is defined as the task of automatically assigning a piece of text to one of many categories, based on its content. It is important to note that the automatic categorization of e-mails is significantly different from traditional text classification. This is because unlike structured text documents, e-mail often does not follow any fixed structure. E-mail is usually written in the way that

people normally speak to each other. There are numerous colloquial terms, slang words, abbreviations, and other such components present in e-mail messages that are not found in other written documents, used in traditional text classification. What makes this task interesting and challenging is the fact that the foldering strategies employed by various users can be as unique and different as the individual users themselves. Whereas some people might have few, highly populated folders, others may have numerous folders, with fewer messages and a fine-grained distinction between individual folders. Thus, there is no generic rule that can be applied to solve the problem. Any system that is created for this purpose has to adapt to the foldering style of every user.

It should be noted here that different people find different uses for automatic classification of e-mail messages. Bekkerman et al., [3] see widespread applications of automatic e-mail classification in spam filtering, extraction of e-mail threads and automatic e-mail foldering, as per user-defined folders. On the other hand, Shetty and Adibi [4] find it useful for link analysis, social network analysis, fraud detection and countering terrorism. These two applications differ in that one is from the point of view of the individual user (Bekkerman) and the other is a more global view (Shetty and Adibi) where you analyze the overall flow of e-mail. Whatever its purpose and application, e-mail classification has been widely accepted as a complex and interesting research problem.

As mentioned previously, one basic point of distinction between traditional text classification and classification of e-mail the way in which the actual classification is done. In almost all cases, text categorization is fairly standard and done according to topic. However, classification of e-mail cannot be approached with this assumption. There can be various ways in which an individual chooses to classify his/her e-mail messages. E-mails may be classified according to the topic, sender, group, importance, timeline, etc. Also, over time, a certain topic may become obsolete, grow in prominence, branch out, or just be forgotten.

Another factor that distinguishes e-mail foldering from general text classification is the time-dependent nature of e-mail. As such, the task of e-mail foldering is extremely sensitive to time. Whereas written text hardly varies over a considerable amount time, e-mail messages are highly sensitive to time. The content of the e-mails, the people involved in the interaction, the thread under discussion, etc. may change over time. Folders created by the users may undergo a lot of changes with time. Messages and topics that originally belonged to a certain folder may later be found to be more relevant to another folder, which may be newly created and hence had not been considered when the user first foldered the message. This is when the categorization gets tricky. New folders may need to be created and older ones may become obsolete.

One more factor that makes this task interesting is the lack of real-life e-mail messages for research purposes. Due to the highly personal and sometimes sensitive nature of e-mail, it is understandable that not many people are keen to allow their e-mail messages to be used for research purposes. As such, finding a large enough corpus of e-mail messages for research purposes is a task by itself. In fact, the Enron corpus (which we use in our research) is the only widely available, real-world and large enough corpus that is currently available. Some individual researchers have created a smaller collection of e-mail messages by collating e-mail messages of various colleagues, fellow researchers and students, but these corpora are generally smaller and not representative of general e-mails.

Another factor that contributes to make e-mail classification very different from other text classification is the amount of prior work required before we actually proceed with the actual classification. Whereas little preprocessing is necessary for a corpus used in simple text classification, the corpora used for e-mail classification typically tend to be extremely noisy and disorganized. It may contain numerous HTML tags, MIME attachments, special characters, etc. present in real-world e-mail messages. Therefore, there is a lot of cleaning and preprocessing involved before we make use of any real-world e-mail corpus.

One final issue that makes e-mail classification different from general text classification is that of performance evaluation. Bekkerman et al., [3] mention that current evaluation methods will not suffice for e-mail classification and some new evaluation method will have to be developed, specifically for e-mail classification. The reasons that they give for arriving at this conclusion are similar to the ones described above when we mentioned that e-mail classification is different from automatic text classification.

All the above-mentioned factors lead us to one conclusion. Any method developed for the automatic classification of e-mail will have to be flexible enough to adapt to the individuality of every e-mail user, and intuitive enough to understand a particular user's foldering strategy and try to simulate it. We also need to come up with an effective evaluation method that can aid researchers in this field.

The overall contributions of this thesis are outlined as below:

1. The creation of a manually annotated corpus of 3,021 e-mail messages, with the messages categorized according to topic or context.

2. The creation of a detailed hierarchical structure for the topic-wise categorization of e-mail messages.

3. The creation of an e-mail specific list of stopwords, that can be appended to any stoplist when working on e-mail related corpora.

4. The development of a package of PERL programs to convert the e-mail messages from XML to Senseval-2 format.

5. A comparison of the output obtained by applying supervised and unsupervised learning methods on the created corpus.

6. A comparison of the effectiveness of three supervised learning methods with regards to the problem of automatic classification of e-mail messages.

2 Background

As mentioned previously, one of the major problems faced by people researching automatic classification of e-mail is the unavailability of a sizeable corpus of real-life e-mail messages that can be analyzed. The recent release of the Enron corpus for research purposes was thus significant for such researchers. The Enron corpus has been extensively used for various purposes since it was made available to the public in March 2004. Since then, researchers have modified the corpus as per their needs or used a particular subset of the corpus for different purposes, thereby creating different versions of the original Enron corpus. In this chapter, we shall first see the events leading to the availability of the Enron corpus and then some of the different versions of the Enron corpus.

2.1 A Brief History of the Enron Corporation leading to the availability of the corpus

In this section, we shall summarize the journey of Enron from a small company in the mid-west to a corporate super-power, which employed thousands of people worldwide. Also, we shall see what circumstances led to the eventual bankruptcy of the corporation and how and why the e-mail corpus was made public.

The origins of the Enron Corporation can be traced back to 1930, when it began as a modest company in Omaha named the Northern Natural Gas Company [15]. The Northern Natural Gas Company, in its turn, was a consortium of Northern American Power and Light Company, Lone Star Gas Company, and United Lights and Railways Corporation. For some time, it was a private company, owned by the board members of the consortium. The Northern Natural Gas Company gradually became public over a

period of seven years between 1941 and 1947. The company existed as the Northern Natural Gas Company for many years. In 1979, the company was restructured and became known as InterNorth Incorporated.

Then, in 1985, InterNorth Inc. took over Houston-based Houston Natural Gas Company. The transaction was engineered by the Chief Executive Officer of Houston Natural Gas Company, Kenneth Lay [15]. Lay handled the acquisition in such a way, that he emerged as the Chief Executive Officer of the new company too. Initially, the company was to be named "Enteron," formed by combining the two words, "Enter" and "On," both of which have positive connotations.

However, it was later learned that in biological vocabulary, "Enteron" means "intestine" – and intestine has other connotations for a natural gas company. Hence, the name "Enteron" was shortened to "Enron." Thus was born one of the largest energy companies, whose operations were not limited only to the U.S. but were spread all over the world.

Initially, Enron was only involved in the transmission and distribution of energy and gas throughout the United States and the development, construction and operation of power plants and other infrastructure worldwide. Over time, it branched into other areas. Also, looking at the changing market conditions, Enron gradually changed from being a production company to a services-oriented company. In short, rather than producing the energy that it sold, Enron was more of a middleman, earning huge commissions in the process.

In 1998, Enron decided to move into the water sector. It founded a new company called Azurix Corporation, which focused on water energy. Throughout 1998 and 1999, Azurix acquired many smaller water companies and contracts in Mexico, Argentina and other areas in South America. Also, in 1999 Enron launched Enron Online. It was the first web-based transaction system that allowed buyers and sellers to buy, sell and trade commodity

products globally. The idea was simple – Enron insisted on being the middleman; users did not know each other. They could only do business with Enron [16].

Thus, over the years, through its pioneering marketing strategies Enron became a force to reckon with in the energy market. It continued to make amazing strides due to its market-leading business strategies and the promotion of power and communication bandwidth commodities and related risk management derivatives as tradable securities. It had set up operations in many countries across the globe, and was doing very well in the domestic market too. Enron Corporation was named as "America's Most Innovative Company" for five consecutive years (from 1996 to 2000) by Fortune magazine.

However, things were not as good as they seemed on the outside. Enron's much-publicized venture, the Azurix Corporation, was not doing well. It was a large-scale money loser, and ended up being one of the first "Special Purpose Entities" of the Enron Corporation. We shall momentarily speak about Special Purpose Entities or SPEs, as they are popularly known.

Beginning around 1999, Enron began to lie about its profits. Huge losses were accrued, which were then bundled off on SPEs. Enron created a number of such SPEs, including Azurix Corporation. The prime reason for the creation of these smaller companies was to keep Enron's balance sheets looking good, and transferring all losses to the SPEs. The global reputation of Enron Corporation also suffered a setback in this period, due to persistent rumors of bribery and political pressure, so as to secure contracts in Central and South America, Africa, Philippines and other areas in South-East Asia, including India. One of the most controversial contracts was the three billion dollar Dabhol Power project in Maharashtra, India. It was alleged that Enron officials used political connections to pressurize the Maharashtra State Electricity Board [17].

In early 2001, Kenneth Lay resigned as the Chief Executive Officer of Enron Corporation. Jeffrey Skilling succeeded him, as the CEO [18]. However, Jeff Skilling

surprised everyone by mysteriously resigning from the position of Chief Executive Officer in August 2001, hardly six months after taking over the helm of the company. He cited "personal reasons" for his resignation. It was later confirmed that this was around the same time that Sherron Watkins sent an anonymous e-mail to Kenneth Lay, advising him that things were not as rosy as they seemed. Sherron eventually became widely known as "Enron's Whistleblower" [14].

From then on, things only got worse. By October 2001, losses transferred to the Special Purpose Entities were more than $618 million, and could not remain hidden any more. For the first time ever, Enron reported a huge loss and a reduction in the value of shareholder stake in the company, in its third-quarterly report.

Alerted by the sudden change from previous reports, the Securities and Exchange Commission started a formal inquiry into the affairs of Enron Corporation on October 31st, 2001 [19]. Needless to say, the value of Enron shares, which had been steadily lowering throughout 2001, plunged to an all-time low. In January 2001, the value of an Enron share was $90 and by November 2001, it went as low as 30 cents. Finally, on December 2nd, 2001 Enron Corporation officially filed for bankruptcy and announced that thousands of its employees would be laid off.

As a result of this, Arthur Anderson, the accounting firm employed by Enron also witnessed a turn in their fortunes. Once, a member of the "Big Five" Accounting companies in the US, it is now believed to be a company with little or no business ethics. Arthur Anderson knew about the goings on at Enron, long before their official insolvency. As the auditors of Enron, it was expected that they would communicate to the public regarding the organization situation and the financial performance of the company. Since they failed to do so, they were also regarded as accomplices in the ensuing scandal.

In fact, Arthur Anderson kept on giving Enron a clean bill of health, in spite of possessing knowledge about many financial discrepancies. They knowingly and

intentionally categorized decreased shareholder equity as an increase. Hence, Arthur Anderson was also indicted for altering, destroying and concealing Enron-related material and persuading others to do the same. It is also said that senior Enron staff were advised by the company to get rid of any official e-mail messages, which may be stored on their machines and/or company mail addresses.

In the course of the investigation that followed, the Federal Energy and Regulatory Commission decided to make the e-mail corpus used during the investigation available to the general public. The corpus was put up on the Internet in May 2002. It contained around 92% of Enron e-mails ranging over a period of three and a half years, from early 1999 to mid-2002. The corpus consisted of a total of 619,449 e-mails from 158 Enron employees. However, attachments were not available.

Later on, Dr. Leslie Kaelbling, Professor of Computer Science and Engineering at the Massachusetts Institute of Technology, purchased the dataset for research purposes [20]. Also, people at the Stanford Research Institute, led by Dr. Melinda Gervasio fixed inherent integrity problems contained in the corpus, as a part of the CALO (Cognitive Agent that Learns and Organizes) project [10]. This team cleaned and filtered the e-mail headers, removed the HTML tags and converted invalid e-mail addresses to a standard format (user@enron.com, when user was specified and no_address@enron.com when the recipient was not specified), so as to facilitate further research. Most importantly, e-mail messages that were considered sensitive were deleted, "as part of a redaction effort due to requests from affected employees."

The dataset as it is currently available was put online by Dr. William W. Cohen, an Associate Research Professor at the Carnegie Mellon University, on March 2nd, 2004, solely for research purposes [7]. In later sections, we shall see more about the corpus as it was originally available and how others in this field have made use of the corpus. We shall also look at new contributions that have been made due to ongoing research.

2.2 Different versions of the Enron corpus

In this section, we shall briefly see the different versions of the Enron corpus. This is just a short compilation of the work done by other researchers on the Enron corpus. We shall begin with a description of the original corpus, followed by variations of the original corpus. For the sake of simplicity, we have named the different corpora after the researchers who worked on them.

2.2.1 Original Corpus (As distributed by William Cohen)

This corpus was originally distributed by William Cohen in March 2004 [7]. This corpus is almost identical to the one made public by the Federal Energy and Regulatory Commission, without the integrity problems that were present. It is a huge corpus, containing 517,431 distinct e-mail messages. Attachments have been excluded. The size of the tarred gzipped file is 400 MB. This gives us an idea of the volume of data that is contains.

This corpus maintains the original folder structure and their hierarchies. It contains e-mail messages exchanged between 151 users over a period of three and a half years. The e-mail messages have been organized into 150 user folders, with numerous sub-folders. The foldering has been done according to each user. Hence, every user has a folder named after him/her. Within this folder, the individual foldering strategy of the user has been maintained. The total number of folders present in the corpus exceeds 4700. The corpus is available for download at William Cohen's Enron page http://www.cs.cmu.edu/~enron/

2.2.2 Klimt and Yang corpus

Brian Klimt and Yiming Yang from Carnegie Mellon were amongst the first people to work on the Enron corpus. Klimt and Yang also wrote a paper introducing the Enron corpus [1]. Klimt and Yang went through the entire corpus and eliminated the duplicate messages that it contained. Most of the duplicate messages were found in computer-generated folders like “Inbox,” “Sent Items,” etc. Klimt and Yang removed these computer-generated folders from the corpus. Only those folders that were created by the users themselves were preserved.

This cleaned corpus contains 200,399 distinct e-mail messages, distributed over 158 users. It should be noted that this is just about one-third the size of the original corpus. In other words, approximately 62% of the original corpus is made up of duplicate e-mails. The average number of e-mail messages per user is 757, as can be deduced from this version of the corpus. However, this number is not at all indicative of the number of messages sent by each user. The distributions of e-mail messages sent or received per user is not uniform, but exponential. This means that a small number of users have a large number of e-mail messages, and vice-versa.

2.2.3 Shetty and Adibi corpus

This version of the corpus was created by Jitesh Shetty and Jafar Adibi, from the University of Southern California [4]. Their version is interesting because they have used it to study and analyze social networks. A social network is defined as the set of personal and professional relationships between people and the strength of those relationships. In this context, it refers to the types of professional relationships between the employees of the Enron Corporation. Shetty and Adibi aimed to understand the types of inter-personal relationships between Enron employees; who corresponded with whom, the level of communication between top management and other employees, etc.

Shetty and Adibi used the corpus distributed by William Cohen and created a MySQL database for the entire corpus. Shetty and Adibi have also cleaned the corpus, by eliminating blank or duplicate e-mails and e-mails that contained junk data or were returned to the sender due to some transaction failure. This corpus contains 252,759 e-mail messages exchanged between 151 users. These e-mail messages are present in around 3000 user-defined folders.

They then created four relation tables – namely, Employee List, Message, Recipient Info and Reference Info. Though self-explanatory, given below is a brief description of each relation table.

1. **Employee List:** This table contains information about every employee whose e-mail messages are present in the corpus.
2. **Message:** This table contains information about the e-mail message, its sender, the subject, the body of the e-mail and other details.
3. **Recipient Info:** This table contains information about the recipient/s of the e-mail messages. Also, this table tells us whether the message was sent directly *To* the recipient, or whether it was *CCd* or *BCCd* to the recipient.
4. **Reference Info:** This table contains information about e-mail messages that have been used as a reference in other e-mails. This also includes messages that have been forwarded or replied to.

They have used these relation tables to study and analyze social networks contained in the Enron organization. They believe that such a network will help us understand the goings-on in the Enron Corporation better and provide a framework for further analysis of the data contained in the e-mail messages. We shall look at their findings in detail in the Literature Review section. More detailed information about their corpus and the various reports that they generated are available at http://www.isi.edu/~adibi/Enron/Enron.htm

2.2.4 Bekkerman corpus

This corpus was created by Ron Bekkerman from the University of Massachusetts, Amherst [3]. He used a subset of the original Cohen corpus. Instead of using the entire corpus, Bekkerman only used e-mail messages of seven top management personnel of the Enron Corporation. The seven people were selected based on the number of e-mails present in the user folders.

Bekkerman removed all the non-topical folders from the e-mail database of these seven people. When we say non-topical folders, what we mean are folders in which e-mail messages are randomly stored, with no distinction on the basis of content. This includes computer-generated folders like "Inbox" and "Sent Items", as well as common folders like "All Documents." He also eliminated user-specific archiving folders, which were created due to certain circumstances like lack of time, or some user-specific strategy, rather than content.

Bekkerman also removed folders that contained less than three e-mail messages, since they were very small and would not help either in training or testing. Another unique approach that Bekkerman used was flattening folder hierarchies. He reduced the depth of the folders to just two – The first level contained individual folders for each of the seven users, while the second level contained actual top-level directories created by the users themselves. All messages contained in any further sub-directories were brought under this level.

Hence, Bekkerman's corpus now contains a total of 273 folders. There are 20,581 e-mail messages. The smallest folders contain 3 e-mail messages, whereas the largest folder contains 1398 e-mails. The seven preprocessed datasets can be downloaded from Ron Bekkerman's web page on Enron at http://www.cs.umass.edu/~ronb/enron_dataset.html

2.2.5 CALO DARPA/SRI corpus

This is a comparatively smaller corpus of real-world, foldered e-mail messages. This corpus was created as a part of the CALO DARPA/SRI research project [3]. This corpus contains snapshots of the email folders of 196 users, containing approximately 22,000 messages. It was created from the February 2, 2004 snapshot of the SRI CALO directories. Ron Bekkerman selected the seven users with the largest number of e-mail messages, in order to have a dataset with which to compare the results obtained with his version of the Enron corpus. As with his corpus, he also flattened the folder hierarchies, removed all non-topical folders and deleted all directories containing less than three e-mail messages, for the SRI CALO corpus. To learn more about this corpus, kindly visit http://www.cs.umass.edu/~ronb/papers/email.pdf

2.2.6 Corrada-Emmanuel corpus

The Corrada-Emmanuel corpus was derived from the original Cohen corpus also. This corpus was created by Andres Corrada-Emmanuel from the University of Massachusetts, Amherst [9]. He created various mappings between e-mails within the Enron corpus. These include mappings of e-mails to relative paths, authors and recipients. He studied the relationship between user folders and e-mail addresses of various users in detail and concludes that actually, there are only 149 users in the Enron corpus. He has created a mapping between the top folders in the corpus and his normalization for an authors email address. Corrada-Emmanuel also wrote Python scripts to extract word lists from the Enron corpus. The various MD5 files, his mapping files and Python scripts can be downloaded from http://ciir.cs.umass.edu/~corrada/enron/

2.2.7 Fiore and Hearst corpus

Andrew Fiore and Marti Hearst, from the University of California at Berkeley, have created a powerful search interface for the Enron e-mail database [8]. This web-based interface searches the corpus, stored in a MySQL database of unique e-mails from the Enron corpus, for e-mail messages containing a given term. The results obtained can be sorted according to the date, sender, recipient, subject, e-mail address, etc. They have made use of Lucene in order to process text queries. The Advanced Natural Language Processing class of Fall 2004, taught by Marti Hearst also created a subset of the Enron corpus as part of their class project. This subset contained around 1700 labeled Enron e-mails. They created a social network for this dataset and studied the obtained results, as part of the assignment. To know more about this corpus, kindly visit Marti Hearst's web page http://www2.sims.berkeley.edu/courses/is290-2/f04/assignments/assignment4.html

2.2.8 Padhye and Pedersen corpus

The Padhye and Pedersen corpus is a subset of the Bekkerman corpus. We have manually annotated 3021 e-mail messages, using a special annotation tool called the Coder. This tool has been developed by Michael O'Donnell at Wagsoft. We have used version 4.67 of the Coder, which was released in February 2005.

The manually labeled e-mail messages were then separated into topical folders like "Business," "Human Resources," "Personal," etc. We also wrote certain programs to change this data into the Senseval-2 format, which is required for further experiments. Later on, we carried out various experiments based on supervised and unsupervised learning methods to see whether automatic classification matches with our manual distribution. We have also created an e-mail specific stoplist, the details of which will follow in further chapters. We shall see what experiments were carried out and their results in further sections.

Let us see the distribution of e-mail messages in our corpus in a little more detail. Our corpus contains six top-level directories, each of which contains numerous sub-directories. The top-level folders have been named as follows – Business, Personal, Human_Resources, General_Announcements, EnronOnline and Chain_Mail. Of these, the "Business" directory is the largest, containing 1367 e-mail messages. This is followed by "Personal", which contains 792 messages. "Human Resources" is the third largest and contains 429 messages, whereas "General Announcements" with 327 messages comes fourth. "EnronOnline" and "Chain Mail" are comparatively smaller folders with 90 and 16 messages respectively. This data is freely available for download at http://www.d.umn.edu/~tpederse/enron.html

Figure 1 on the next page is a diagram that shows the different variations of the Enron corpus that are currently available, and where our corpus fits into the whole picture.

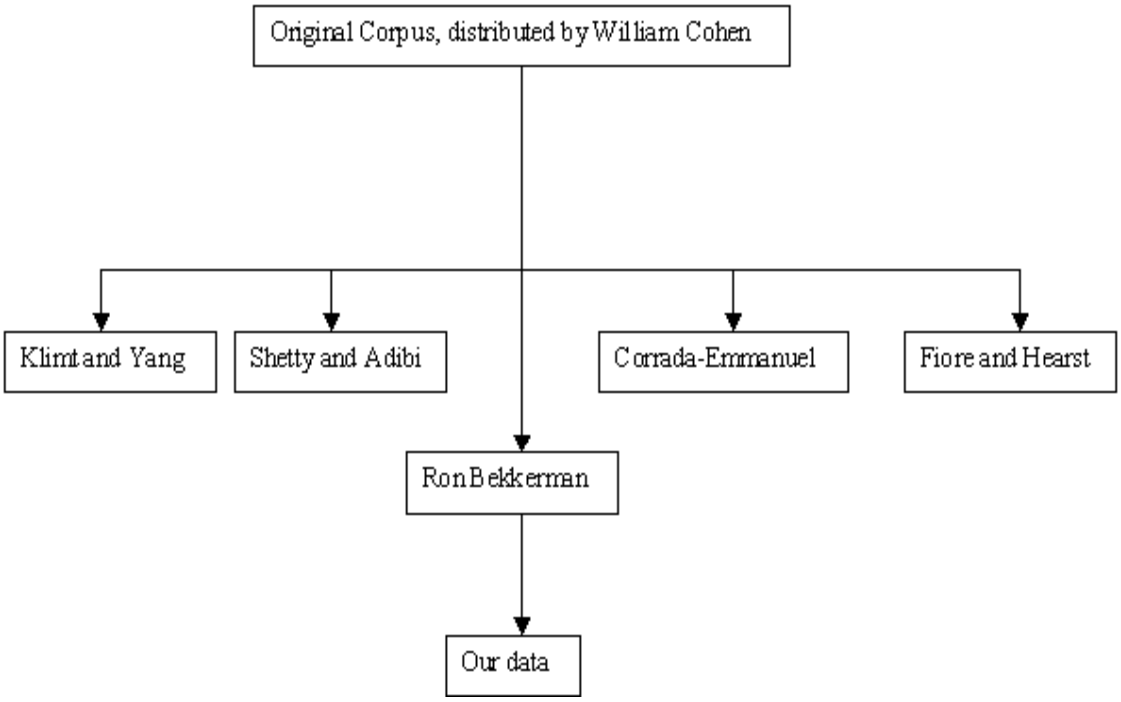

Figure 1: Figure showing the different versions of the Enron corpus that are currently available, and how our corpus fits into the whole picture

3 Description of the Experimental Data

The data that was used for the experiments is a subset of the Bekkerman corpus, which in turn is a subset of the original Enron corpus, as distributed by William Cohen. We have created this corpus by manually annotating a certain number of e-mail messages from Bekkerman's corpus. Ron Bekkerman's corpus contains 20,581 e-mail messages, belonging to seven executives from Enron's top management. These have been sorted based on user.

We have manually annotated 3,021 of those messages and distributed them into folders based on context or topic, rather than user. Each topical folder contains e-mail messages belonging to all users, and not just one user. Hence, we have altered the folder structure for these 3,000 odd e-mail messages from user-based to topic-based. The annotation has been done using a special annotation tool developed by Michael O'Donnell at Wagsoft. The tool is called the "Systemic Coder". We have used version 4.67 of the Coder, which was released in February 2005. The Coder is available for free download at http://www.wagsoft.com

The annotation was done over a period of 12 months from February 2005 to February 2006. The messages to be annotated were manually picked by us at random from among the messages present in Ron Bekkerman's corpus. It should be pointed out here that once an e-mail message was opened, it was always categorized into a folder. No message was ever discarded, or left aside without assigning it to a category once it was opened. Also, every message has been assigned to only one category. In those cases wherein it was felt that a message may belong to more than one category, a judgment was made based on our understanding of the content of the e-mail and the created directory structure; and it was assigned to only one folder that it related most with. Hence, the annotated e-mail messages present in this corpus belong to one single class only, and not multiple classes.

These decisions were made according to our perspective, and thus it is likely that the assigned categories may differ if someone else re-did the annotations.

The messages in the Padhye and Pedersen corpus have been categorized into six broad topics and put into the respective folders. These broad categories (upper level directories) are Business, Personal, Human Resources, General Announcements, EnronOnline and Chain Mails. Each of these top-level directories contains numerous sub-directories. The maximum depth of the directory structure is five.

The distribution of the 3,021 annotated e-mail messages in the six directories is extremely unequal. The Business folder contains a large number of e-mail messages, which is 1367. The second largest directory is Personal, which contains 792 messages. Human Resources and General Announcements are third and fourth with 429 and 327 e-mail messages respectively. After these larger folders, we have the two smallest folders which are EnronOnline, which contains 90 messages, and Chain Mails, which has just 16 of the 3021 e-mail messages. We shall see each of these folders in detail in the following sub-sections. The Padhye and Pedersen corpus is available for downloading at http://www.d.umn.edu/~tpederse/enron.html

Figure 2 shows a diagram that shows the six upper-level directories and the number of sub-directories present under each, if any, and their names.

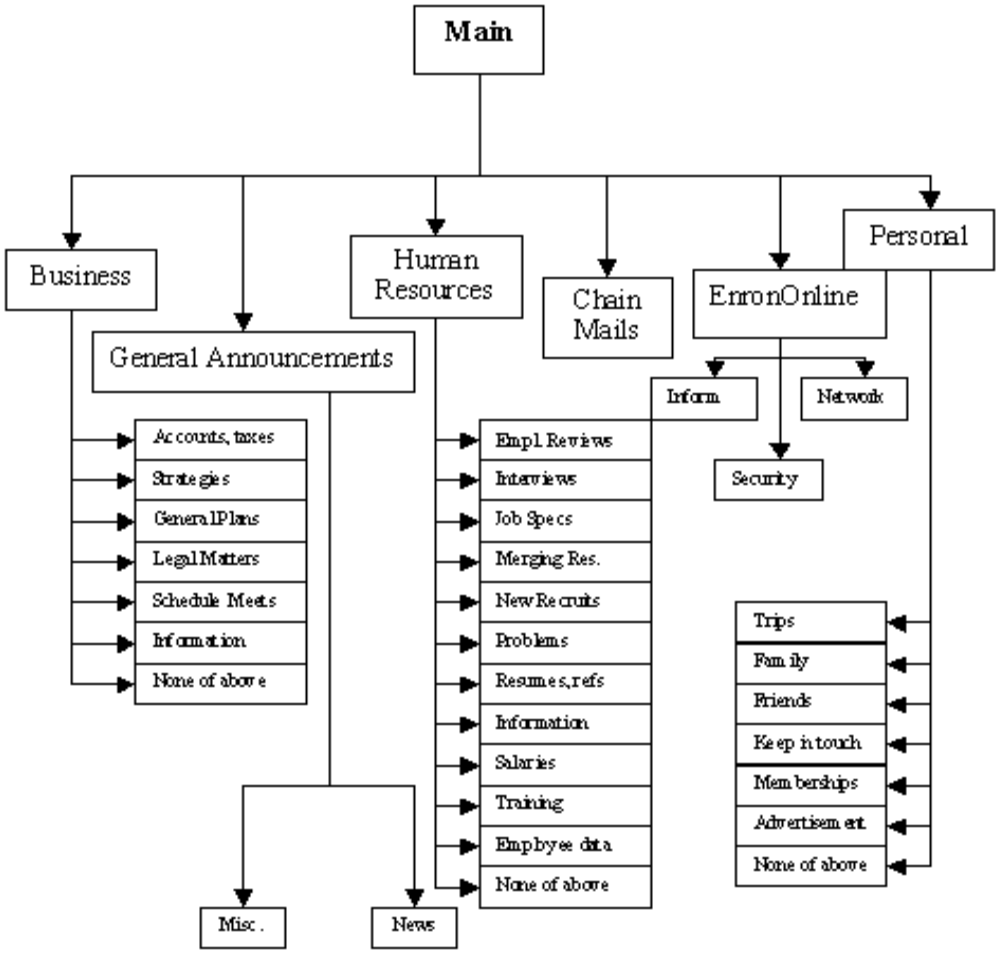

Figure 2: Diagram showing the upper-level directory structure for the Padhye and Pedersen corpus. Shows topmost directories and their sub-directories.

3.1 Business Directory:

The Business directory is the largest category in the Padhye and Pedersen corpus. Almost half of the e-mail messages in the corpus fall in the Business directory. This agrees with our intuitive expectations, since the Enron corpus consists of e-mail messages sent and received by ex-employees of the Enron Corporation via their corporate account. Hence, the Business directory accounts for 45.25% of the Padhye and Pedersen corpus.

There are 1367 e-mail messages in the Business directory. This directory has been further sub-categorized into eight different folders, depending on the topic or context of the messages that they contain. Given below are the names of each of these sub-directories, the number of e-mail messages they contain and a brief description of the kind of e-mail messages they comprise of.

1. Accounts and taxes:

This folder contains e-mail messages pertaining to accounts, taxes and other financial matters related to audit. There are just 16 e-mail messages in this sub-directory. There are no further sub-directories within this folder.

2. Business Strategies:

This folder contains e-mail messages that pertain to general business plans made, discussion of future strategies, implementation of previous plans, results seen after the adoption of certain business tactics, etc. This folder contains 26 e-mail messages. Again, there are no further sub-categories within this directory.

3. Conference Call Information:

This folder contains e-mail messages containing information about conference calls and the proceedings therein. This is the smallest sub-folder within the

Business category with just 4 e-mail messages. There are no sub-folders in this directory.

4. General Plans:

This folder contains e-mail messages related to general business-related plans made by the employees. These plans involve travel, social gatherings, product launches, company meetings, and the like. There are 107 e-mail messages within this directory. The directory has been further sub-categorized into three folders, namely – Events, Scheduling and Travel.

5. Legal Matters:

This folder contains messages that pertain to the legal aspect of the Enron Corporation. These involve e-mails detailing lawsuits, contracts, reports of court proceedings, etc. There are a total of 97 e-mail messages in this folder that have been divided across three sub-directories – Contracts, Legal Documents and News & Information.

6. Schedule Meetings:

This folder contains e-mail messages pertaining to meetings and their scheduling issues. These meetings may either be actual meetings or conference calls. There are 45 e-mail messages in this folder, divided into two sub-directories. These sub-directories have self-descriptive names viz. Actual Meetings and Conference Call Information.

7. Information:

This is the largest sub-directory within the Business folder. It contains e-mail messages that are meant to inform one or more people about a particular event, ask questions, file reports, document information, etc. There are a total of 908 e-mail messages in this directory that have been divided into seventeen sub-folders. These folders are as follows – Away from job, Bankruptcy, Business

Advertisements, Contacts, Courtesies, Complaints, Conferences, Delegate Work, Follow-up, For the Record, Presentations, Press-related Matters, Publications, Queries, Replies, Reports and Upcoming Things.

Of these seventeen folders, only For the Record has been further divided into three different categories, which are Automatic, Manual and For Your Information.

8. None of the above:

As the name suggests, this folder contains e-mail messages that we know are business-related, but are difficult to classify into any particular sub-folder. There are a total of 164 such messages, divided into two folders – E-mail messages with only attachments, no text (since all attachments have been removed when the data was cleaned, such e-mail messages are not useful for classification purposes), and E-mails that contain material that is difficult to understand, and hence classify.

Figure 3 shows the distribution of messages within the sub-folders of the Business directory and the number of e-mail messages contained in each.

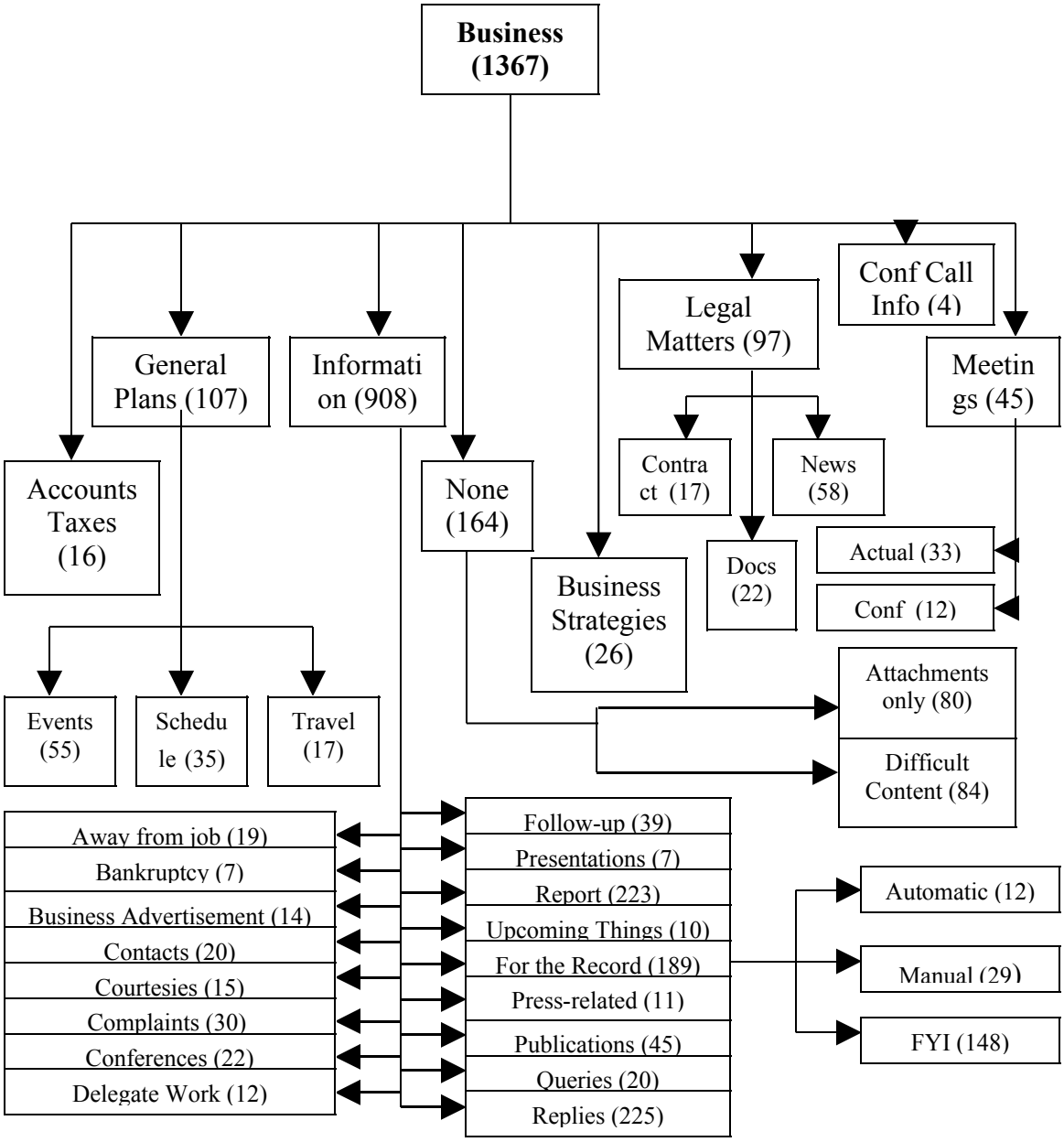

Figure 3: Diagram showing the distribution of e-mail messages in the Business folder

3.2 Personal Directory:

The second largest directory in the Padhye and Pedersen corpus is the Personal directory. This folder contains personal messages sent and received by the ex-employees of Enron via their official e-mail accounts. Being much smaller in size than the Business directory, the Personal directory does not have as complicated a directory structure as the Business directory. The Personal folder makes up only 26.22% of the e-mail messages in the Padhye and Pedersen corpus.

There are 792 e-mail messages in the Personal directory. This directory has been further sub-categorized into seven different folders, depending on the topic or context of the messages that they contain. Given below are the names of each of these sub-directories, the number of e-mail messages they contain and a brief description of the kind of e-mail messages they comprise.

1. **Entertainment:**

   This folder contains e-mail messages that pertain to plans made by the people on the personal entertainment front. These may involve birthday parties, plans for catching a movie, visiting friends, visits to sporting events, operas, etc. There are 16 e-mail messages in this sub-folder. The folder has not been further divided into any other sub-categories.

2. **Family:**

   This folder contains e-mail messages that relate to all family issues and plans made by the employees via e-mails on their official accounts. These may involve messages pertaining to family get-togethers and emails from spouses, parents, siblings and children. There are 71 e-mail messages in this folder, and it has not been further sub-divided into any other categories.

3. **Keeping in Touch:**

   This folder contains e-mail messages pertaining to all personal, social contacts maintained by the Enron employees with each other, or people outside the company. These are social contacts made and maintained for personal reasons, and not official. There are 448 e-mail messages in this folder, which have been distributed into six different sub-directories. These six sub-directories are – Enquiries and Replies, Forwards, Friends, Greetings, Plans to Meet and Thank You.

4. **Memberships:**

   This folder contains e-mail messages about the membership-related information of the users. These involve alumni associations, church and charity memberships, club and golf memberships, memberships of professional organizations not related to their work at Enron, and the like. There are 123 e-mail messages in this folder that have been distributed into three sub-folders. These sub-folders are – Advertisements, Institution Membership and Group mails from an Association.

5. **Personal Advertisements:**

   This folder contains e-mails that are essentially advertisements for a particular product or service, like credit cards, fitness clubs, online diets, online forums, etc. There are 60 e-mail messages in this folder. There are no further sub-divisions within the Personal Advertisements directory.

6. **Trips:**

   This directory contains e-mail messages that pertain to trips or vacations planned or taken by the Enron employees that are not part of their official duties. There are e-mails about trips to Las Vegas, cruises, hiking trips, picnics, camping details, etc. There are 19 total e-mail messages in this folder. The directory does not contain any other sub-directories.

7. **None of the above:**

   The name of this directory is self-descriptive. It contains e-mail messages that are clearly personal, but cannot be classified into any particular sub-folder within the personal directory. This may be because the content of the e-mails is difficult to understand (to a third party, but may have been pertinent to the sender and receiver), the e-mails consist of only pictures (attachments have been removed during the cleaning of the data), or because the e-mail is in a foreign language. There are 55 e-mail messages in this folder. The directory has not been further categorized into sub-folders.

Figure 4 shows the distribution of messages within the sub-folders of the Personal directory and the number of e-mail messages contained in each of these sub-folders.

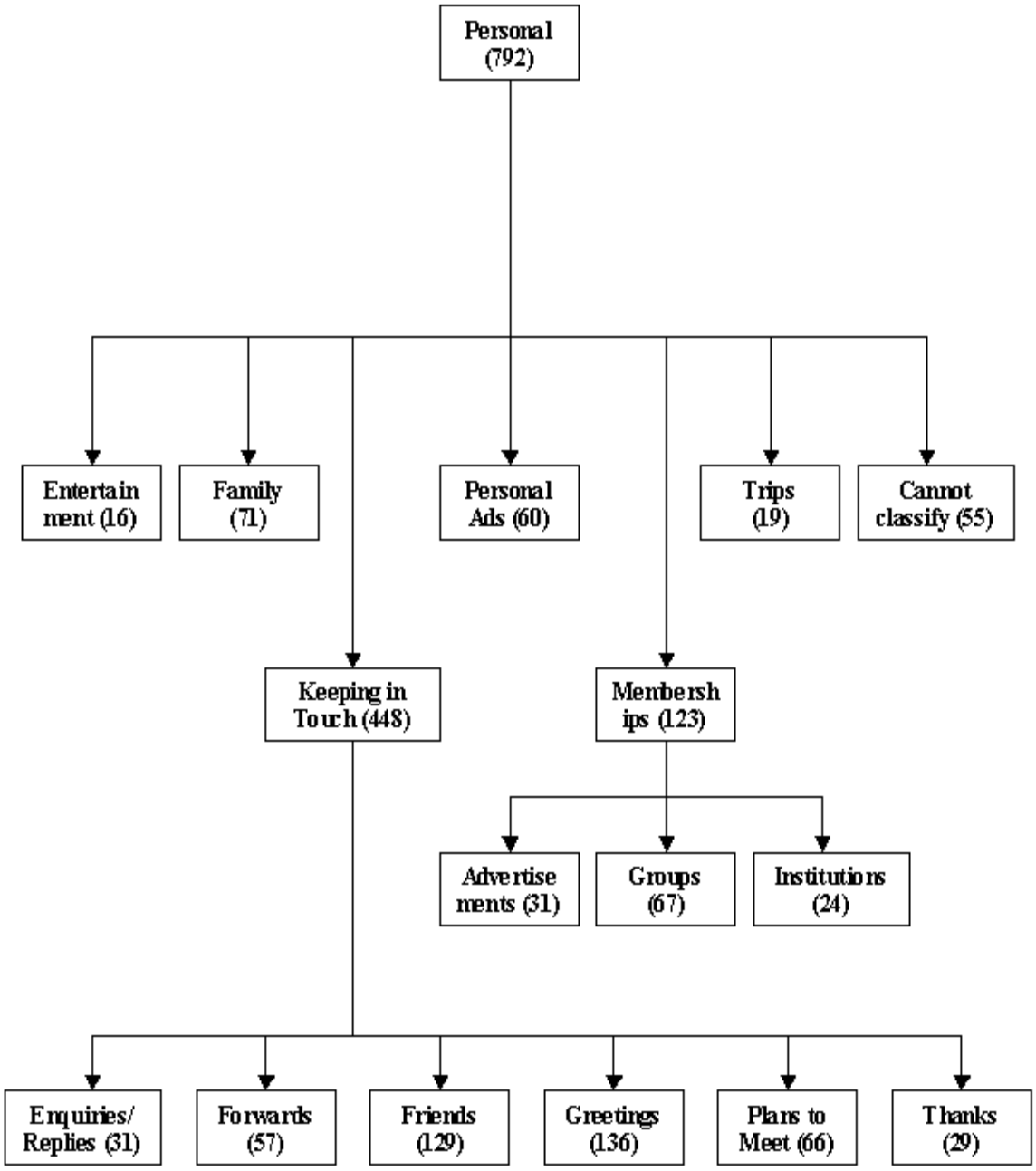

Figure 4: Diagram showing the distribution of e-mail messages in the Personal folder

3.3 Human Resources Directory:

The third largest directory in the Padhye and Pedersen corpus is the directory for Human Resources. Though it can be argued that this directory can be a part of the Business folder itself, two things justified the creation of a separate directory for Human Resources. First was obviously its size, while the second was the varied nature of human resources-related e-mail messages present in the corpus; making it a sub-directory would have further complicated the directory structure of the Business folder. In spite of being smaller in size than the Personal directory, the Human Resources directory has a more complicated directory structure than the Personal folder. The Human Resources directory makes up about 14.20 % of the e-mail messages in the Padhye and Pedersen corpus.

There are 429 e-mail messages in the Human Resources directory. This directory has been further sub-categorized into fourteen different folders, depending on the topic or context of the messages that they contain. Given below are the names of each of these sub-directories, the number of e-mail messages they contain and a brief description of the kind of e-mail messages they comprise of.

1. **Employee Information:**

   This sub-directory contains e-mail messages that pertain to certain information about employees like their addresses, educational qualifications, specific job skills, etc. This also includes e-mails exchanged between managers discussing which employees would be suitable to take up a new position either temporarily or on a permanent basis. There are 22 e-mail messages in this folder. The directory has not been divided into any sub-directories.

2. **Employee Performance Reviews:**

   This folder contains e-mail messages that describe matters related to the performance reviews of Enron employees. These may involve schedules for

performance reviews, assignment of people to conduct the reviews, results of employee reviews, types of reviews to be done, improvements to be made, etc. There are a total of 56 e-mail messages in this folder. The directory has not been sub-divided into any further categories.

3. **Information:**

   This directory contains e-mail messages that relate to information to be given to, or asked from, employees that fall within the domain of Human Resources. This may involve e-mails that ask employees to participate in surveys, take annual health check-ups, participate in some social events, ask about plans made by employees in the near future, etc. There are 29 e-mail messages in this sub-directory of Human Resources. The directory has not been divided any further into sub-directories.

4. **Interviews:**

   This folder contains e-mail messages that pertain to upcoming interviews. These may mean e-mails that are sent to schedule interviews, or follow-up on interviews held previously. The e-mail messages within this folder are either those sent out to the interviewing candidates, or those sent within the company to managers and other hiring staff. There are 76 e-mail messages in this directory. The directory has been divided into three sub-directories, namely – Follow-up, Lunch-Dinner and Schedule Interviews.

5. **Job Specifications:**

   This folder contains e-mail messages that have been sent out as advertisements for jobs or job listings within the company. They specify the available jobs and what qualities are they looking for in a potential employee. This is a fairly small folder that contains only 16 e-mail messages. The directory has not been further sub-categorized into any more folders.

6. **Merging Resources:**

   This folder contains e-mail messages that relate to ventures that Enron undertook with other companies or schools and universities to merge their respective human resources. This includes funding for ongoing research at universities, company takeovers, and other such messages. This is also a small folder and it contains just 4 e-mail messages. The directory does not contain any sub-directories.

7. **New Recruitments:**

   This folder contains e-mail messages that pertain to the hiring and recruitment part of Human Resource duties. They contain information about any new employees hired by the company, internal promotions of existing employees, internship candidates hired, and the resulting paperwork involved in the process. There are 42 e-mail messages in this folder. These messages have been distributed into three sub-folders. These sub-directories are Internships, New Hires and Internal Transfers.

8. **Problems:**

   This directory contains Human Resource-related problems that employees at Enron have and that they have reported to the HR people. This may involve problems regarding treatment by seniors and colleagues, problems related to health insurance issues, etc. This is a relatively small folder containing only 2 e-mail messages. There are no sub-directories within this folder.

9. **Settlements:**

   This directory contains e-mail messages pertaining to settlements offered to employees at Enron. These settlements may be due to termination of employment, retirement, accident settlements, due to temporary inability to work caused because of work-related injuries, etc. This directory is again small in size and contains only 8 e-mail messages. There are no sub-directories within this folder.

10. Resumes:

This folder contains e-mail messages that pertain to resumes. These resumes are either sent to the company for permanent employment, for internships, or for just short-term projects. There are also some messages that offer information on references. Some e-mail messages are those sent within the corporation when a certain resume was found particularly interesting and the input of others was required to take things further. There are 10 e-mail messages in this folder. The directory does not contain any sub-directories.

11. Referrals:

This folder is somewhat related to the folder containing Resumes. The difference with this folder is that these are resumes forwarded to the Human Resource department with referrals from people currently working at Enron. These may target specific jobs that employees know are open, or just general e-mail sent to the HR department with a note to keep a particular candidate in mind, should a position become available. There are 68 e-mail messages found in this folder. There are no sub-directories within this folder.

12. Salary Matters:

This folder contains e-mail messages that deal with salary matters of the employees. Most often these are messages sent to selected candidates informing them of the package offered by Enron and asking them for their acceptance before sending them a formal letter of appointment. Also, there are some e-mail messages that let employees and their supervisors know about annual increments, perks and bonuses. There are a total of 33 e-mail messages in this folder. There are no sub-folders within this directory.

13. Training:

This folder contains e-mail messages pertaining to training offered to employees for things such as increasing their skills, time management, presentation of projects, knowledge about the Enron Corporation that falls out of their field of work, etc. There are a total of 53 e-mail messages in this directory. These messages have been distributed into two sub-directories – Information about upcoming training workshops and Grades obtained in training workshops that employees might have attended.

14. None of the above:

As the name suggests, this folder contains e-mail messages that we know are related to Human Resources, but are difficult to classify into any particular sub-folder. There are a total of 10 such messages in this folder, and no sub-directories.

Figure 5 shows the distribution of messages within the sub-folders of the Human Resources directory and the number of e-mail messages contained in each.

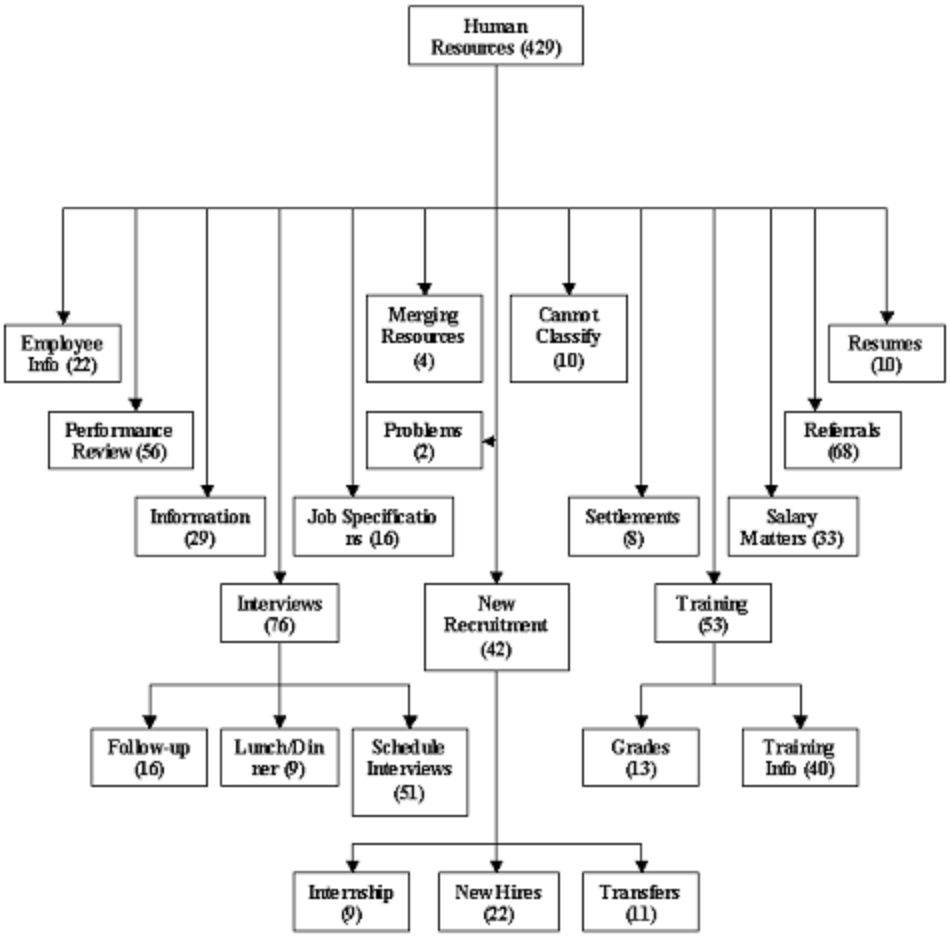

Figure 5: Diagram showing the distribution of e-mail messages in the Human Resources directory

3.4 General Announcements Directory:

The fourth directory, in terms of size, in the Padhye and Pedersen corpus is the directory for General Announcements. This folder contains messages that are intended to be announcements to a large number of Enron employees. Though comparable in size to the Human Resources directory, the General Announcements folder has a very simple directory structure. The General Announcements folder makes up about 10.82% of the e-mail messages in the Padhye and Pedersen corpus.

There are 327 e-mail messages in the General Announcements directory. This directory has been further sub-categorized into only two different folders, depending on the topic or context of the messages that they contain. Given below are the names of each of these sub-directories, the number of e-mail messages they contain and a brief description of the kind of e-mail messages they comprise of.

1. **Miscellaneous:**

   This folder contains announcements that do not fall into any particular category, and are miscellaneous in nature. They comprise of all kinds of announcements ranging from the victory of an Enron-sponsored sporting team to announcements of network outages, power outages, routine system check-ups, etc. There are 218 e-mail messages in this directory. There are no further sub-divisions within this folder.

2. **News:**

   This folder contains announcements that are meant to be news to the employees of Enron. These include e-mail messages sent out from the Chairman's office on account of happy events like a good end to a financial year, new year's, the announcement of a new company chairman, etc.; or not so happy events like the proceedings in court during Enron's trial for fraud. There are also messages sent

out within particular groups meant to convey some news to the other members of that group. Hence, the recipients of these messages may be the entire set of Enron employees, or a subset thereof. There are 109 e-mail messages in this folder. There are no sub-directories within this directory.

Figure 6 shows the distribution of messages within the sub-folders of the General Announcements directory and the number of e-mail messages contained in each.

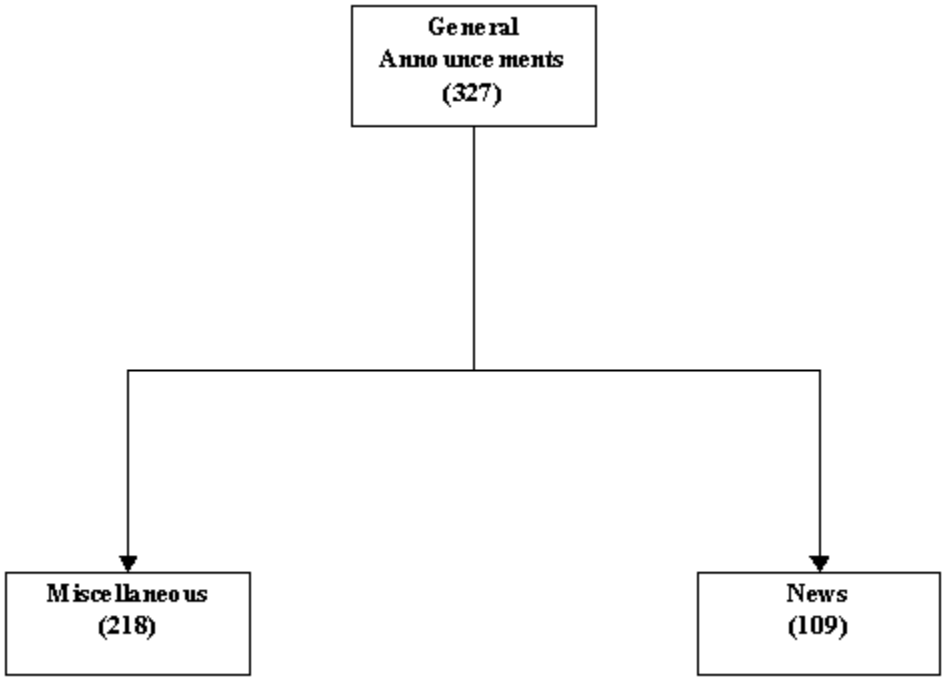

Figure 6: Diagram showing the distribution of e-mail messages in the General Announcements directory

3.5 EnronOnline Directory:

The fifth directory, in terms of size, in the Padhye and Pedersen corpus is the EnronOnline directory. This folder contains messages that are exchanged by the employees of Enron and pertain to a pioneering (at that time) venture of the Enron Corporation, known as EnronOnline. EnronOnline was an e-commerce website in the commodities market. It allowed its users to buy, sell and trade commodities, like natural gas and electricity, online. Louise Kitchen, who is one of the seven users whose e-mail messages we have annotated, was one of the architects of the website. Hence, this corpus contains a fair number of e-mail messages related to EnronOnline. The EnronOnline folder makes up only a small percentage (about 2.98%) of the e-mail messages in the Padhye and Pedersen corpus.

There are 90 e-mail messages in the EnronOnline directory. This directory has been further sub-categorized into just three different folders, depending on the topic or context of the messages that they contain. Given below are the names of each of these sub-directories, the number of e-mail messages they contain and a brief description of the kind of e-mail messages they comprise of.

1. **Information:**

   This directory contains e-mail messages that are related to general information exchanged between the employees of EnronOnline regarding the website. These include routine daily and weekly reports about network traffic, amount of trade done within a particular period, messages about spikes or dips in the expected traffic, etc. There are a total of 64 e-mail messages in this folder. It has been divided into two sub-categories – Announcements and Questions.

2. **Network:**

   This directory contains e-mail messages pertaining to network-related information, like network outages (both scheduled and unscheduled), website maintenance, network upgrade information and granting of passwords to registered users of EnronOnline. There are 12 e-mail messages in this folder. The directory has not been divided into any sub-directories.

3. **Security:**

   This folder contains all e-mail messages that are related to the security issues of EnronOnline. These involve e-mail messages that deal with user authentication, information about security threats, etc. There are a total of 14 e-mail messages in this folder. This folder has not been sub-divided into any further sub-directories.

Figure 7 shows the distribution of messages within the sub-folders of the EnronOnline directory and the number of e-mail messages contained in each.

Figure 7: Diagram showing the distribution of e-mail messages in the EnronOnline directory

3.6 Chain E-Mail Directory: -

The smallest directory in the Padhye and Pedersen corpus is the directory for Chain E-Mail. This folder contains messages that fall under the general category of chain e-mail messages or spam. These include messages sent out en masse by Enron employees or to Enron employees meant to propagate a chain.

These are e-mail messages that play on the sympathies of people like,

> "'X' hospital has a patient who needs blood of this particular group. Please send this e-mail to everyone in your mailing list in order to help the person;" or
>
> "This baby needs an expensive operation in order to survive. AOL has promised 10 cents for every address to which this e-mail is forwarded. Please forward this e-mail to as many people as you can."

Others play on peoples' sense of fear. They are messages like,

> "This is a sacred message. Forward it to 'X' number of people and something good will happen. If you do not, you will be cursed for the rest of your life."

The Chain E-Mail folder makes up a very small portion of the e-mail messages in the Padhye and Pedersen corpus. It contains just 16 e-mail messages. There are no sub-folders within this directory.

This concludes our discussion of the Padhye and Pedersen corpus. This corpus was created in order to verify the effectiveness of supervised and unsupervised learning methods in the categorization of context-specific e-mail messages.

We have also created an e-mail specific stoplist for this purpose. On carrying out some preliminary experiments, it was seen that the usual English stoplist is not adequate for automatic categorization of e-mail messages. Hence, we have added some more terms to

the English stoplist to make it more apposite for e-mail classification. We shall see more about this stoplist in the following section.

3.7 Our E-Mail Specific Stoplist:

Before we begin to discuss the e-mail specific stoplist that we have created, let us first see what a *stoplist* is. A **Stoplist** holds a set of index terms that are to be stripped away from a text (that is, "stopped," or removed) before it is further processed. A stoplist is generally specified in the form of a text file. The file containing the list of stop words should have one stop word per line [21]. These terms are letters, words or phrases that are so commonly used in the English language that they are virtually irrelevant to the categorization process. These include conjunctions, prepositions, and other common words that aid in the easier understanding of the language, rather than any content description.

The e-mail specific stoplist that we have created is nothing more than an extension of the commonly used English stop word list. We have added certain words to the existing stoplist. These include words like Original, Forward, Message, Reply, etc. that do not belong to the usual stop word category. However, any e-mail user will know at first glance that these words are commonly found in a majority of e-mail messages and hence they become insignificant to the task of automatic e-mail classification.

Our stoplist also includes terms that are not words, but are commonly found in e-mail messages. They are terms like cc, bcc, fwd, com, edu, quot, gov, org, etc. We have also excluded HTML tags that get included in e-mail messages like br, gt, lt, hr, apos, amp, etc.

The process of arriving at this stoplist was two-pronged, and done over a period of two months, by trying several experiments for various stoplists. On carrying out some initial experiments on the data it was seen that certain features were created that did not help in the categorization of the e-mail messages. These features contained words or terms that were stop words, in the sense that they misled the classifier by creating ineffective features. Hence, we filtered out those words. Simultaneously, it became obvious that certain terms and words like forward, message, reply, original, yours, and the like were prone to be common to most e-mail messages. Therefore, these words were also added to the stoplist. This two-sided process continued for several iterations until we arrived at a stoplist that optimized the results for a part of the e-mail corpus created (around 1000 e-mail messages.)

There are 469 stop words in this stoplist. Of these, 425 words belong to the original English stoplist while we have added 44 words in order to make the stoplist more suitable for automatic e-mail classification. The table on the following page shows the terms that we have added to the original stoplist.

Table 1: List of terms appended to original English stoplist

gt	forward	truly	bmp
lt	fwd	regards	gif
apos	amp	Hi	txt
CC	Please	September	gov
BCC	Thanks	jpeg	org
quot	re	Love	2000
Reply	edu	Hey	2001
MIME	br	href	1999
nbsp	hr	ul	9/11
Original	Dear	ol	9
message	Yours	com	11

4 Classification Methods

This section briefly describes the various classification methods used in order to categorize the email messages into various folders. We have made use of three supervised and one unsupervised method. The supervised methods used are the Naïve Bayes classifier, J48 Decision Trees and Support Vector Machines, whereas the unsupervised method is an adaptation of the Repeated Bisections clustering method. Let us now see how each method works.

4.1 Supervised Learning Methods:

The supervised methods used are Naïve Bayes classifier, J48 Decision trees and Support Vector Machines. Of these, Naïve Bayes and Support Vector Machines have been selected because they have been used by many other researchers for text classification (including Ron Bekkerman) and this would give us an idea of how well the methods perform on our corpus and whether the results are similar to those obtained by others. J48 Decision trees were selected because it was a different kind of classification algorithm, as compared to the other two. Also, decision trees are extremely popular classification methods.

Before looking at the methods though, let us understand what the terms *Dependent* and *Independent* variables mean. The attribute that is to be predicted is known as the dependent variable, since its value depends upon, or is decided by, the values of all the other attributes. The other attributes, which help in predicting the value of the dependent variable, are known as the independent variables in the dataset.

4.1.1 Naïve Bayes Classifier:

The Naïve Bayes classifier works on a simple, but intuitive concept. The Naïve Bayes classifier is based on the Bayes rule of conditional probability. It makes use of all the attributes contained in the data, and analyzes them individually as though they are equally important and independent of each other.

For example, consider that the captain of a cricket team has to decide whether to bat or field first in the event that they win the toss. He decides to collect the statistics of the last ten matches when the winning captain has decided to bat first, and compare them in order to decide what to do, so as to make conditions most favorable for a win. The table on the following page represents the data that he has collected, in order to help him make a decision. In the following table, Outlook, Humidity and the number of regular batsmen in the team are the independent variables, whereas the dependent variable is the final outcome of the game.

Often, the data available is too little and instances with a particular combination of attributes may not be available, or if they are, they are very few. Hence, it becomes difficult to predict the classification of a new instance using Bayes rule of conditional probability, which needs instances with a combination of features. To overcome this difficulty, the Naïve Bayes classifier will consider each of these attributes separately when classifying a new instance.

So, in our earlier example, when checking to see what the outcome of a match is most likely to be, the Naïve Bayes classifier will not check whether the day is sunny, and the humidity is high and whether there are more than six batsmen in the team (combination of attributes). Rather, it will separately check whether the conditions (new instance) are sunny; whether the humidity is high; and whether there are more than six batsmen in the team (each attribute is considered independently). It works under the assumption that one attribute works independently of the other attributes contained by the sample.

Table 2: Various values of the attributes for the last ten games when the winning captain decided to bat first and the final outcome of the game

Independent Variables			Dependent Variable
Outlook	**Humidity**	**Number of batsmen in team > 6**	**Final Outcome**
Sunny	High	Yes	Won
Overcast	High	No	Lost
Sunny	Low	No	Lost
Sunny	High	No	Won
Overcast	Low	Yes	Lost
Sunny	Low	Yes	Won
Sunny	Low	No	Lost
Sunny	High	No	Won
Sunny	Low	Yes	Won
Sunny	Low	Yes	Won

In our experiments, it is seen that the Naïve Bayes classifier performs almost on par with the other classifiers in most of the cases. Of the 26 different experiments carried out on various datasets, the Naïve Bayes classifier shows a drop in performance in only 3-4 cases, when compared with J48 and Support Vector Machines. This confirms our belief that though simple in concept, the Naïve Bayes classifier works well in many data classification problems.

The possible reason behind this may be the fact that when the data present is small in size, the assumption that all attributes are independent of each other, tends to provide the classifier with more information than that obtained by taking all the attributes together.

The probability of getting instances with the various attribute values taken individually is higher than that of getting instances wherein a particular combination of attribute values occurs.

4.1.2 J48 Decision Trees:

A decision tree is a predictive machine-learning model that decides the target value (dependent variable) of a new sample based on various attribute values of the available data. The internal nodes of a decision tree denote the different attributes, the branches between the nodes tell us the possible values that these attributes can have in the observed samples, while the terminal nodes tell us the final value (classification) of the dependent variable.

The J48 Decision tree classifier uses a simple algorithm. In order to classify a new item, it first needs to create a decision tree based on the attribute values of the available training data. So, whenever it encounters a set of items (training set) it identifies the attribute that discriminates the various instances most clearly. This feature that is able to tell us most about the data instances so that we can classify them the best is said to have the highest information gain. Now, among the possible values of this feature, if there is any value for which there is no ambiguity, that is, for which the data instances falling within its category have the same value for the target variable, then we terminate that branch and assign to it the target value that we have obtained. Figure 8 shows how this process is done with the help of the cricket example.

For the other cases, we then look for another attribute that gives us the highest information gain. Hence we continue in this manner until we either get a clear decision of what combination of attributes gives us a particular target value, or we run out of attributes. In the event that we run out of attributes, or if we cannot get an unambiguous

result from the available information, we assign this branch a target value that the majority of the items under this branch possess.

Now that we have the decision tree, we follow the order of attribute selection as we have obtained for the tree. By checking all the respective attributes and their values with those seen in the decision tree model, we can assign or predict the target value of this new instance. The above description will be more clear and easier to understand with the help of an example. Hence, let us see how J48 decision trees will go about classifying the result in the example of the cricket captain that we saw previously for the Naïve Bayes classifier. As we know, the captain of the team has to decide whether to bat or field first in the event that they win the toss. He decides to collect the statistics of the last ten matches when the winning captain has decided to bat first, and compare them in order to decide what to do, so as to make conditions most favorable for a win.

Though the data is apparently confusing, it looks as though a decision tree model may help the captain get a clearer picture of the underlying situation. Hence, let us build a decision tree model for the available data. As we can see from the table, it is obvious that whenever the winning captain decides to bat first on a day that is not sunny, the team loses the match. As such, this attribute gives us the most information. Let us make this our first attribute for splitting.

Then, we look at the branch where some ambiguity still exists, that is it still has a number of instances with both values of the dependent variable. We then realize that the attribute Humidity will give us more clear information about what happens when the day is sunny. We can see that the batting side wins when the day is sunny and the humidity is high. Thus, Humidity is ideally found to be the next attribute based on which the instances should be split.

Now we realize that the number of regular batsmen in the team plays an important role at this stage. If a team had more than six batsmen, it won the game. If not, it lost. Therefore,

we now have a decision tree that looks like the diagram given on the following page. From this, the captain knows that statistically, it is better to field first if the day is overcast. If the day is sunny and the humidity is high, batting is advisable. Also, if the day is sunny, the humidity is low, and the team contains at least six regular batsmen, it is safe to bat first; otherwise they should field. Hence, the captain can now make an informed decision after winning the toss.

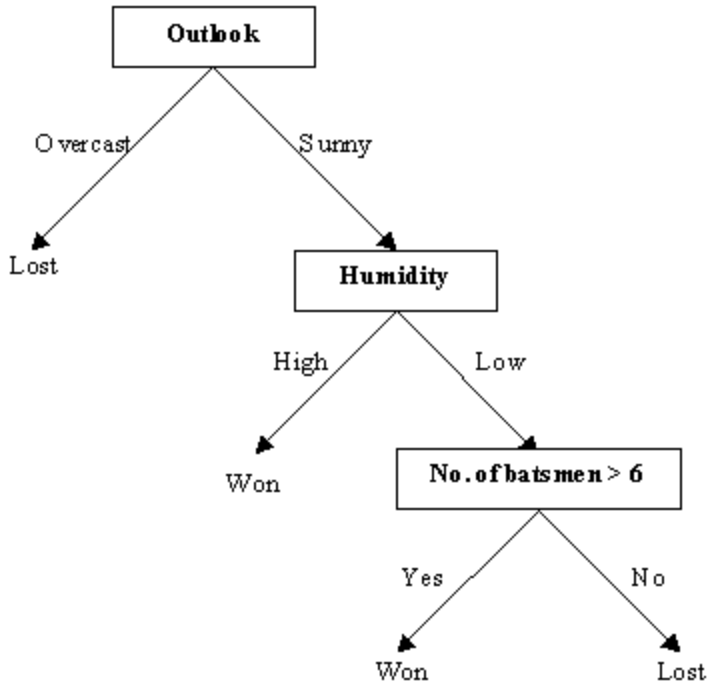

Figure 8: Diagram showing the decision tree created from available data.

The dependent Variable: Is game won or lost

Thus, whenever he comes across a new instance, the captain will just have to compare the attribute values for outlook, humidity and number of batsmen to arrive at the best decision in the event that he wins the toss.

In our experiments it was seen that J48 Decision trees performed almost at par with that of Support Vector Machines. In fact in several cases, it was seen that J48 Decision Trees had a higher accuracy than either Naïve Bayes, or Support Vector Machines.

4.1.3 Support Vector Machines:

Support Vector Machines are supervised learning methods used for classification, as well as regression. When the output of the function is a continuous value, the learning method is said to perform regression; and when the learning method can predict a class label of the input object, it is called classification [24]. The independent variables may or may not be quantitative.

Kernel equations are functions that transform linearly non-separable data in one domain into another domain where the instances become linearly separable. Kernel equations may be linear, quadratic, Gaussian, or anything else that achieves this particular purpose. A linear classification technique is a classifier that uses a linear function of its inputs to base its decision on. Applying the kernel equations arranges the data instances in such a way within the multi-dimensional space, that there is a hyper-plane that separates data instances of one kind from those of another. The advantage of Support Vector Machines is that they can make use of certain kernels in order to transform the problem, such that we can apply linear classification techniques to non-linear data.

Once we manage to divide the data into two distinct categories, our aim is to get the best hyper-plane to separate the two types of instances. This hyper-plane is important because it decides the target variable value for future predictions. We should decide upon a hyper-plane that maximizes the margin between the support vectors on either side of the plane. Support vectors are those instances that are either on the separating planes on each side, or a little on the wrong side. The explanatory diagrams that follow will make these ideas a little more clear.

One important thing to note about Support Vector Machines is that the data to be separated needs to be binary. Even if the data is not binary, Support Vector Machines reduces the multi-class problem to a collection of two-class problems, and completes the analysis through a series of binary assessments on the data. Basically, this involves looking at a particular class present in the data, and predicting the value of an instance for that class alone in a Yes/No manner. This is done for every class and the obtained results are then combined.

Let us now see an example of how Support Vector Machines work. Since it is easier to understand the concept visually, we shall see the data instances in their original form on a graph, and then we shall see how the data instances are separated upon the application of kernel functions on them, and how the best hyper-plane is found. The original space in which the instances are present is called the **Input space.** The new space obtained after applying the kernel function is called the **Feature space.** We shall also see what support vectors exactly are and how does a margin look.

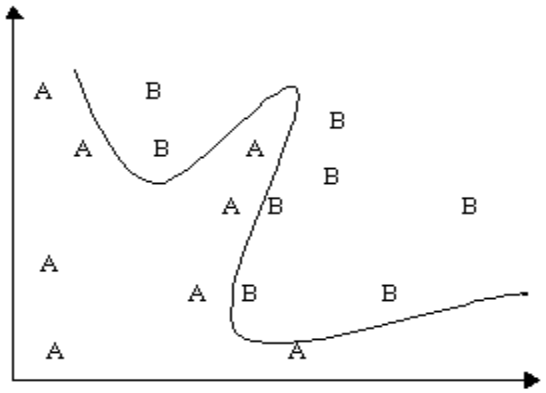

Figure 9: The data instances as seen in the Original Space (Input space). As we can see, the instances are not linearly separable

Idea for diagram taken from class notes of Dr. Rich Maclin (Advanced Machine Learning and Knowledge Discovery Databases)

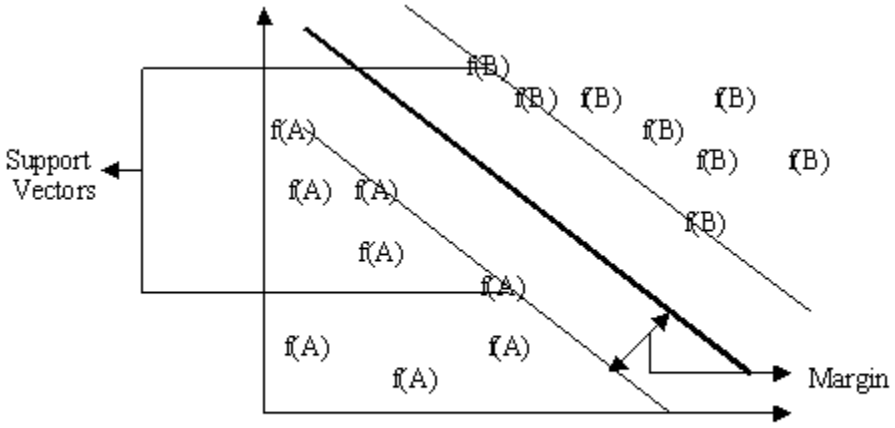

Figure 10: The data instances as seen in the New Space (Feature space). The instances can now be linearly separated

Idea for diagram taken from class notes of Dr. Rich Maclin (Advanced Machine Learning and Knowledge Discovery Databases)

As can be seen from the diagrams on the previous page, the data instances which were not linearly separable in the original domain have become linearly separable in the new domain, due to the application of a function (kernel) that transforms the position of the data points from one domain to another. This is the basic idea behind Support Vector Machines and their kernel techniques. Whenever a new instance is encountered in the original domain, the same kernel function is applied to this instance too, and its position in the new domain is found out. This position determines the binary target value to which the new instance belongs.

In many cases, it is often seen that Support Vector Machines perform the best among all machine-learning methods. It may be interesting to recall that Ron Bekkerman also came to the conclusion that Support Vector machines achieve a higher accuracy than Naïve Bayes, Maximum Entropy or Wide Margin Winnow. Though Wide Margin Winnow does perform faster and in some cases better than Support Vector Machines, on the whole Support Vector Machines outperform any other classifier for the task of email classification.

In our experiments too, it is seen that Support Vector Machines usually have the highest accuracy among any of the other classification methods. Of the 26 experiments that we carried out, it is seen that Support Vector Machines have the highest accuracy in 16 cases, while in most others it is a close second.

4.2 Unsupervised Learning Method: Repeated Bisections clustering algorithm

In this section we will try to understand the Repeated Bisections clustering algorithm [23] that has been used in SenseClusters. Clustering is the process in which we divide the available data instances into a given number of sub-groups, based on the level of

similarity between the instances in a certain group. These sub-groups are called clusters, and hence the name “Clustering”.

The Repeated Bisections clustering algorithm is one way of carrying out the K-means clustering method. So, let us first see what we mean by K-means clustering. To put it simply, the K-means algorithm outlines a method to cluster a particular set of instances into K different clusters, where K is a positive integer. It should be noted here that the K-means clustering algorithm requires knowing the number of clusters from the user. It cannot identify the number of clusters by itself.

The K-means clustering algorithm starts by placing K centroids as far away from each other as possible within the available space. Then each of the available data instances is assigned a particular centroid, depending on a metric like Euclidian distance, Manhattan distance, Minkowski distance, etc. The position of the centroid is recalculated every time an instance is added to the cluster and this continues until all the instances are grouped into the final required number of clusters. Since recalculating the cluster centroids may alter the cluster membership, the cluster memberships are also verified once the position of the centroid changes. This process continues until there is no further change in the cluster membership, and there is as little change in the positions of the centroids as possible.

The initial position of the centroids is thus very important since this position affects the future steps in the K-means clustering algorithm. Hence, it is always advisable to keep the cluster centers as far away from each other as possible. If there are too many clusters, then clusters that closely resemble each other and are in the vicinity of each other are clubbed together. If there are too few clusters then clusters that are too big and may contain two or more sub-groups of different data instances are divided. The K-means clustering algorithm is thus a simple to understand, fairly intuitive method by which we can divide the available data into sub-categories.

Now we come to the Repeated Bisections method. A K-way partitioning via repeated bisections is obtained by recursively applying the above algorithm to compute two-way clustering (i.e. bisections). Initially, the objects are partitioned into two clusters, then one of these clusters is selected and is further bisected, and so on. This process continues (K – 1) times, leading to K clusters. Each of these bisections is performed so that the resulting two-way clustering solution optimizes the selected criterion function [23].

In our experiments, it was seen that in many cases the accuracy of the unsupervised method was not far behind that of the supervised methods. In fact, in a couple of cases, the unsupervised method outperformed any of the supervised methods. This is very encouraging when we consider the fact that we do not require a training set for unsupervised learning methods. In many cases, it may be possible that when faced with the trade-off between making training data available and sacrificing the accuracy by 5-6%, people would prefer the latter option and hence move towards unsupervised methods of learning, as against supervised learning methods. As such, the results we have obtained look pretty encouraging, though unsupervised learning methods may not perform as well as supervised learning methods.

5 Experimental Results

In this chapter we shall be looking at the results obtained by applying supervised and unsupervised learning methods on our data. There are numerous reasons behind this. Firstly, the results of these experiments will indicate the accuracy of our topic-wise categorization of the data. Secondly, these experiments will help us understand compare the results obtained by supervised and unsupervised learning methods. Also, the experiments will tell us which supervised method is better amongst the three that we have selected.

Before looking at the results though, let us see what settings were decided for all the learning methods. We shall begin with the supervised learning methods – Support Vector Machines, J48 Decision Trees and Naïve Bayes, and then move on to the unsupervised learning method (SenseClusters). The selection of features remained constant for all the experiments conducted on various data subsets. We shall see each of these subsets in detail as we see their individual results.

One important decision that we have taken when carrying out the experiments is that we have filtered out all the headers from the e-mail messages. We only look at the body of the e-mail messages when carrying out the classifications. This is because it is commonly seen that the actual message may only be a couple of lines long, while the headers included may be 5-6 lines long or more. In these cases, the headers tend to mislead the classifiers, as they dominate over the actual content of the e-mail messages. Hence, we filter out all the headers from our corpus, before running any experiments.

The results are given in the form of the F-measure values obtained by the individual learning methods. The benchmark is the percentage value of the Majority Sense. The majority sense is chosen as the benchmark because that is the highest accuracy that we can achieve if we do not carry out any classification at all, but just place all the instances

in the cluster that has the maximum number of instances. Thus, this benchmark allows us to understand the improvement achieved by making use of a particular algorithm. Any F-measure value above the majority sense is thus, the performance enhancement achieved by making use of the algorithm.

5.1 Feature Selection for Supervised Methods:

We have made use of a package called WSDShell to carry out the supervised learning experiments. WSDShell is a set of programs and wrapper scripts for running supervised word sense disambiguation (WSD) experiments on any WSD collection adhering to a specific format and directory structure [22]. The WSDShell package makes use of the WEKA Data Mining suite in order to carry out the supervised learning experiments. WEKA is a collection of numerous machine-learning algorithms that can be used for data mining tasks [25]. To learn more about WSDShell, kindly visit its official web page, which is http://www.d.umn.edu/~tpederse/wsdshell.html

Now we come to the feature selections. The *types of features created* are bigrams. Bigrams are word pairs that occur together, and this co-occurrence happens more frequently than by mere chance. This is decided based on the score for tests known as goodness-of-fit tests. These tests of association tell us the amount of certainty with which we can say that a word pair is a bigram, based on the score. In our experiments, we have specified the *score* option as 3.841. We make use of the Log-Likelihood test. The reason for selecting 3.841 is that a score of 3.841 assures us a probability of at least 95% that the co-occurrence of two features is not merely by chance.

Bigrams are not necessarily consecutive words. They may be separated by a 'window' of words. The window size has been kept 5 for all our experiments. This means that two words can become a bigram if they are separated by upto 5 words. The *remove* option has

been selected as 5. This option is used to specify the features to ignore, if their frequency is below a particular number. There are no separate train and test sets. Instead, we have used the ten-fold cross validation method. In this method, we divide the entire data into 'n' sets, which in our case are 10. Then, one set is used for training, while the remaining sets are used as test data. This process is repeated for all the other sets. The final results are an average of the individual results obtained for the 'n' different combinations of train and test datasets. Hence the name, n-fold cross validation.

The last parameter that we have specified is the *stopfile*. A stopfile is a file that contains all of the stopwords that we want our algorithm to ignore when it goes about building features. We have included the e-mail specific stoplist that we have generated, and have described in the previous chapter.

5.2 Feature Selection for Unsupervised Method (SenseClusters):

We have made use of a package called SenseClusters for conducting the unsupervised learning experiments. The parameter selection process for SenseClusters requires a bit more input from the user. This is because SenseClusters is a package for unsupervised learning, and it allows a lot of options when conducting experiments. To learn more about SenseClusters, kindly visit http://www.d.umn.edu/~tpederse/senseclusters.html

We have made use of the unigram option when selecting the *type of features* to build. This is because initial experiments revealed that unigram features give us the best results for automatic e-mail classification. The *remove* option has been selected as 5, which is its default value. The *stopfile* has been specified as our e-mail specific stoplist.

Since we have made use of unigram features, the *context representation* was first order context vectors. The *clustering space* was selected to be vector space. The *clustering*

method that was selected was Repeated Bisections. The *number of clusters* into which classification is to be done varied for each data subset and was thus selected accordingly. We have not made use of the automatic cluster identification feature of SenseClusters.

Let us now look at the various experiments carried out, and the results that were obtained for each.

5.3 Experiments and Results:

The results are shown as an F-measure value for all the data subsets. The *F-measure,* for SenseClusters, is an average of the Precision and Recall values obtained. *Precision* is defined as the number of instances correctly classified, divided by the number of classification instances attempted. *Recall* is defined as the ratio of the number of instances correctly classified and the total number of instances present. Since SenseClusters gives us all three results, we have tabulated all of them. WEKA, on the other hand, attempts to classify all the data instances, and hence just gives us the accuracy of every algorithm. Thus, we have just listed the accuracy of the supervised methods as the value of their F-measure.

The first few experiments were conducted on upper level directories. These showed only a broad categorization of the e-mail messages into one of the six general categories of Personal, Business, Human Resources, General Announcements, EnronOnline and Chain Mails.

5.3.1 Business/Personal (2 classes):

These are the two biggest directories in the Padhye and Pedersen corpus. The Business directory is the dominant directory in the corpus, and in order to get somewhat similar cluster sizes, the best directory that can be paired with Business is the Personal directory. Also, the Business and Personal directories are fairly easy to categorize, as compared to any of the other directories. Hence, our first data subset for the experiments was Business and Personal. The results obtained for these directories follow.

The F-measure value for the above directories for Naïve Bayes classifier was 67.39, while that for J48 Decision trees is 72.46. The highest F-measure value for supervised learning methods is that for Support Vector Machines, which is 73.85. The F-measure value for unsupervised learning is 56.66. The majority sense value is 63.86. Hence in this case, we see that all supervised learning experiments achieve a performance enhancement over the majority sense. However, the unsupervised method achieves a lower accuracy as compared to the majority sense.

5.3.2 Business/Human Resources (2 classes):

The next data subset that we conduct experiments on is the Business and Human Resources directories. The results of this subset are especially remarkable because Human Resources can be considered to be a subset of the Business directory. Hence, it becomes interesting to see whether our learning methods are able to distinguish between the two, thereby validating the creation of a separate directory for Human Resources.

The value of the F-measure for the above directories for Naïve Bayes classifier was 80.10, which is the same as that for Support Vector Machines. The F-measure value was highest i.e., 80.88, for J48 Decision trees, albeit by a very small margin. The value for obtained by SenseClusters was 65.50, while the majority sense was 75.56. Here again, we

see that supervised learning methods have performed better than our benchmark, while the unsupervised method fell short of the majority sense.

5.3.3 Human Resources/Personal (2 classes):

The Human Resources directory is comparable in size with the Personal directory. Hence, this gives us a fairly balanced data subset on which we can test the accuracies of each of our classifiers. Also, the directories are fairly diverse, so we would expect the results to be better than the percentage value of our benchmark.

In this case, the majority sense has a percentage value of 63.58. Surprisingly, the Naïve Bayes classifier does not perform well on this data subset. Its accuracy is just 41.91, while J48 Decision trees, with 74.86 and Support Vector Machines, with an F-measure value of 77.07, perform much better. Even the unsupervised learning method achieves a performance enhancement over the majority sense with the value of the F-measure being 65.76.

The fact that even the unsupervised method scores better than the majority sense, makes the result obtained by the Naïve Bayes classifier even more surprising. We shall later try to deduce the reasons behind such results.

5.3.4 EnronOnline/General Announcements (2 classes):

The next data subset consists of two folders – EnronOnline and General Announcements. After the three big directories – Business, Personal and Human Resources – these two folders can be considered to be of a comparable size. Hence, at the topmost level, this dataset becomes our next choice for conducting experiments.

The F-measure value for the above directories for J48 Decision trees was 83.45, while that for Support Vector Machines is 83.69. The highest F-measure value for supervised learning methods is that for the Naïve Bayes classifier, which is 84.89. The F-measure value for unsupervised learning is 61.22. The majority sense value is 78.16. Hence in this case, we see that all supervised learning experiments achieve a performance enhancement over the majority sense. However, the unsupervised method achieves a lower accuracy as compared to the majority sense.

5.3.5 Business/Personal/Human Resources (3 classes):

Our next data subset is larger in size, as it consists of the three largest directories – Business, Personal and Human Resources. This is the first subset with more than two clusters (folders) contained in the data subset. This helps us to understand whether our classifiers can handle data that is not binary.

For this data subset, we see that J48 Decision trees have an accuracy of 58.32, while Naïve Bayes has an accuracy of 63.28. Support Vector Machines have an accuracy of 64.33. The majority sense has a percentage value of 52.92. Here again, unsupervised methods do not perform better than the majority sense, with an F-measure value of 42.52 only.

5.3.6 All (6 classes):

After considering all possible subsets at the topmost directory level, we shall now see what the results are for all the directories in the Padhye and Pedersen corpus, that is, the whole dataset. As we know, there are six main directories in the corpus. Let us see what results were obtained for the entire corpus.

Here again, we see that the supervised methods achieve a considerable performance enhancement over the majority sense, whereas the unsupervised method performs poorly for this data. The majority sense is 45.22, while the Value of the F-measure of the unsupervised method is 28.04. Amongst the supervised methods, we see that Support Vector Machines perform the best with an accuracy of 58.34, with J48 Decision trees, having an F-measure value of 57.25, following close behind. The Naïve Bayes classifier has an F-measure value of 51.33.

Table 3: Table showing the results for the top-level directories

Data Subset	Unsupervised	Supervised
Business/Personal	Precision = 58.86 Recall = 54.62 F-Measure = 56.66 Majority sense = 63.86	**F-measure** Naïve Bayes = 67.39 SVM = 73.85 J48 Decision Trees = 72.46
Business/Human Resources	Precision = 67.51 Recall = 63.60 F-Measure = 65.50 Majority sense = 75.56	**F-measure** Naïve Bayes = 80.10 SVM = 80.10 J48 Decision Trees = 80.88
Human Resources/Personal	Precision = 68.17 Recall = 63.52 F-Measure = 65.76 Majority sense = 63.58	**F-measure** Naïve Bayes = 41.91 SVM = 77.07 J48 Decision Trees = 74.86
EnronOnline/General Announcements	Precision = 62.28 Recall = 60.19 F-Measure = 61.22 Majority sense = 78.16	**F-measure** Naïve Bayes = 84.89 SVM = 83.69 J48 Decision Trees = 83.45
Business/Personal/Human Resources	Precision = 44.01 Recall = 41.13 F-Measure = 42.52 Majority sense = 52.92	**F-measure** Naïve Bayes = 63.28 SVM = 64.33 J48 Decision Trees = 58.32
All	Precision = 28.95 Recall = 27.19 F-Measure = 28.04 Majority sense = 45.22	**F-measure** Naïve Bayes = 51.33 SVM = 58.34 J48 Decision Trees = 57.25

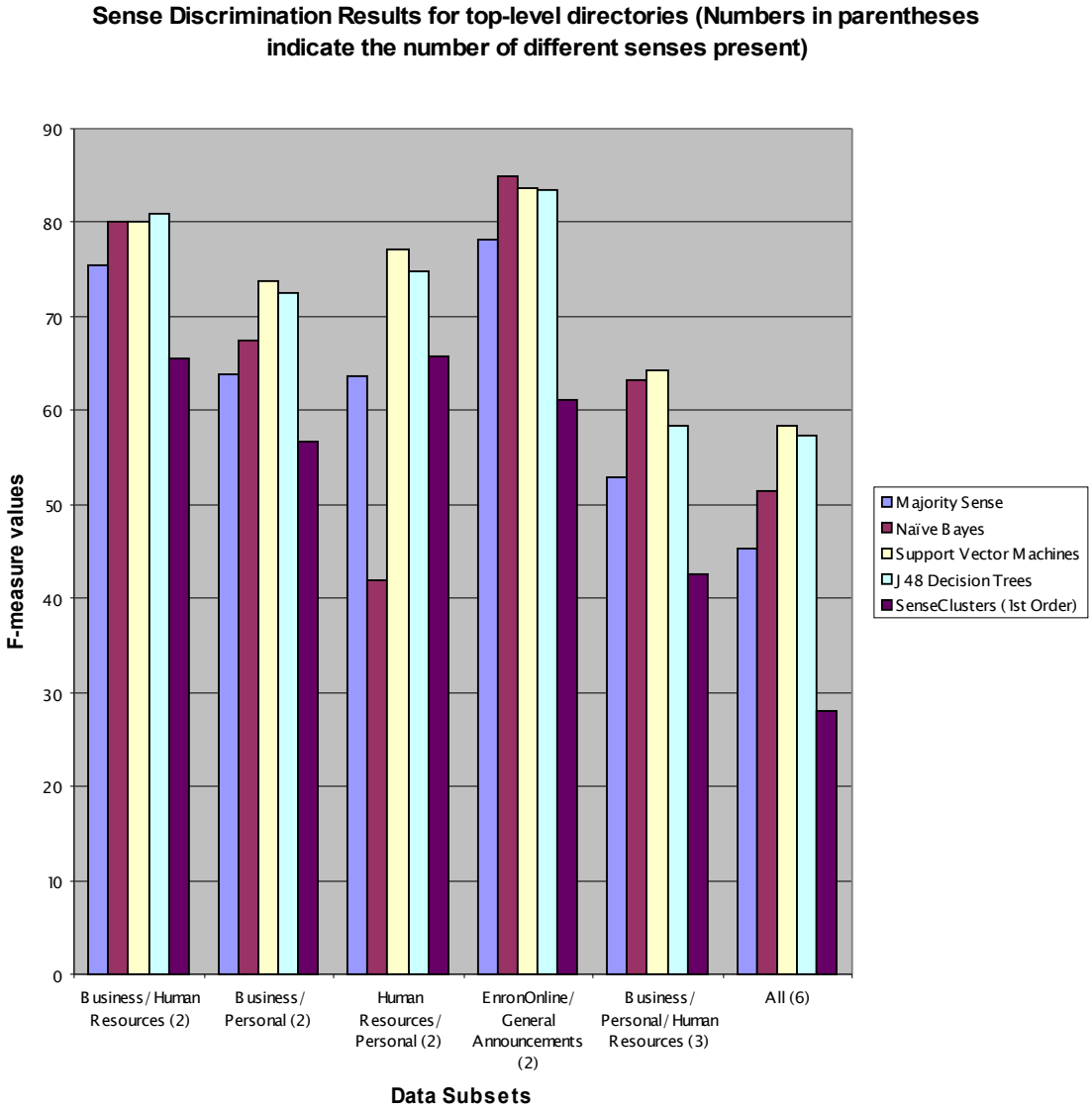

Figure 11: Graph showing the sense discrimination results for top-level directories. This is a coarse-grained result, since we only consider the upper-level directories.

After looking at the various directories at the upper-level, that is, after seeing the coarse-grained results, let us now go into each directory and see whether the classifiers can identify the finer differences present within each directory and correctly classify the data into its individual sub-directories. We shall only be looking at the Business, Personal and Human resources directories, since the other directories are not very large and do not have many sub-directories, or any further levels of distinction.

Sub-directories contained within the Business directory:

5.3.7 Accounts and Taxes/Business Strategies (2 classes):

Within the Business directory, we first take the data subset consisting of the Accounts and Taxes sub-folder and the Business Strategies sub-folder. In this case, we see that none of the algorithms perform as well as the majority sense. The value of the majority sense is 65.00. All the supervised learning methods – Naïve Bayes, Support Vector Machines and J48 Decision trees – have the same accuracy of 61.90. The F-measure value for the unsupervised learning method is 58.54.

5.3.8 Accounts and Taxes/Conference Call Info (2 classes):

Next, we look at another binary data subset, which contains e-mail messages from the Accounts and Taxes, and Conference Call Info sub-directories. This is a classic example of a skewed dataset wherein the majority sense has an extremely high percentage value. This makes it difficult for the algorithms, since they have to perform really well in order to better the majority value. In this instance, we see that all the supervised methods, with an F-measure value of 88.89 do achieve a performance enhancement over the majority sense. However, the unsupervised method has an F-measure value of 60.61 only.

5.3.9 Accounts and Taxes/Legal Matters (2 classes):

Now we see another example of a skewed dataset; but in this case, none of the algorithms manage to beat the majority sense. The data subset contains e-mail messages from the Accounts and Taxes and Legal Matters sub-directories. The majority sense has a percentage value of 87.27. The Naïve Bayes classifier has an accuracy of 76.99; Support Vector Machines come second with 83.19, while the highest accuracy is 85.84 for J48 Decision trees. Here again, the unsupervised method, with an F-measure value of 60.99, does not perform as well as the supervised methods.

5.3.10 Business Strategies/Legal Matters (2 classes):

The next data subset consists of the sub-directories for Business Strategies and Legal Matters. This is another example wherein none of the algorithms manage to perform better than the majority sense. The F-measure value is 63.92 for the Naïve Bayes classifier, while it is 64.95 for both, Support Vector Machines and J48 Decision trees. The F-measure value obtained by using SenseClusters is 55.51. All these values are much below the percentage value for the majority sense, which is 78.69.

5.3.11 Accounts and Taxes/Business Strategies/Conference Call Info (3 classes):

As we did at the upper level, at the sub-directory level too, we shall look at a couple of data subsets that are not binary. This dataset consists of three subdirectories – Accounts and Taxes, Business Strategies and Conference Call Info. Here again, none of the classifiers manage to outperform the majority sense, which has a percentage value of 61.90. The F-measure value is 59.09 for J48 Decision trees, while it is 56.82 for both,

Support Vector Machines and the Naïve Bayes classifier. The F-measure value obtained by using SenseClusters is 37.21

5.3.12 All (8 classes):

Now, we look at the entire Business directory as a data subset, with all its eight sub-directories. This is a complex dataset, since some sub-folders are extremely small, while others are large. This makes the data skewed, in addition to the fact that there are eight classes contained in it. The results for this subset are slightly different than all the results obtained so far. The percentage value of the majority sense for this dataset is 68.30. So far, we'd seen that all the supervised methods performed either better than or worse than the majority sense. Here, we see that two methods – J48 Decision trees, with an accuracy of 57.25 and the Naïve Bayes classifier with an accuracy of 65.86 – perform worse than the majority sense, while Support Vector Machines, with an accuracy of 68.87, performs slightly better than the majority sense. The performance of the unsupervised method, with an F-measure value of 26.43, lags far behind the performances of the other methods.

Table 4: Table showing the results for the sub-directories within the Business directory

Data Subset	Unsupervised	Supervised
Business Accounts and Taxes/ Business Strategies	Precision = 60.00 Recall = 57.14 F-Measure = 58.54 Majority sense = 65.00	**F-measure** Naïve Bayes = 61.90 SVM = 61.90 J48 Decision Trees = 61.90
Business Accounts and Taxes/ Conference Call Info	Precision = 66.67 Recall = 55.56 F-Measure = 60.61 Majority sense = 86.67	**F-measure** Naïve Bayes = 88.89 SVM = 88.89 J48 Decision Trees = 88.89
Business Accounts and Taxes/ Legal Matters	Precision = 61.82 Recall = 60.18 F-Measure = 60.99 Majority sense = 87.27	**F-measure** Naïve Bayes = 76.99 SVM = 83.19 J48 Decision Trees = 85.84
Business Business Strategies/ Legal matters	Precision = 55.74 Recall = 55.28 F-Measure = 55.51 Majority sense = 78.69	**F-measure** Naïve Bayes = 63.92 SVM = 64.95 J48 Decision Trees = 64.95
Business Accounts & Taxes/Business Strategies/Conferenc e Call Info	Precision = 38.10 Recall = 36.36 F-Measure = 37.21 Majority sense = 61.90	**F-measure** Naïve Bayes = 56.82 SVM = 56.82 J48 Decision Trees = 59.09
Business All	Precision = 27.44 Recall = 25.49 F-Measure = 26.43 Majority sense = 68.30	**F-measure** Naïve Bayes = 65.86 SVM = 68.87 J48 Decision Trees = 57.25

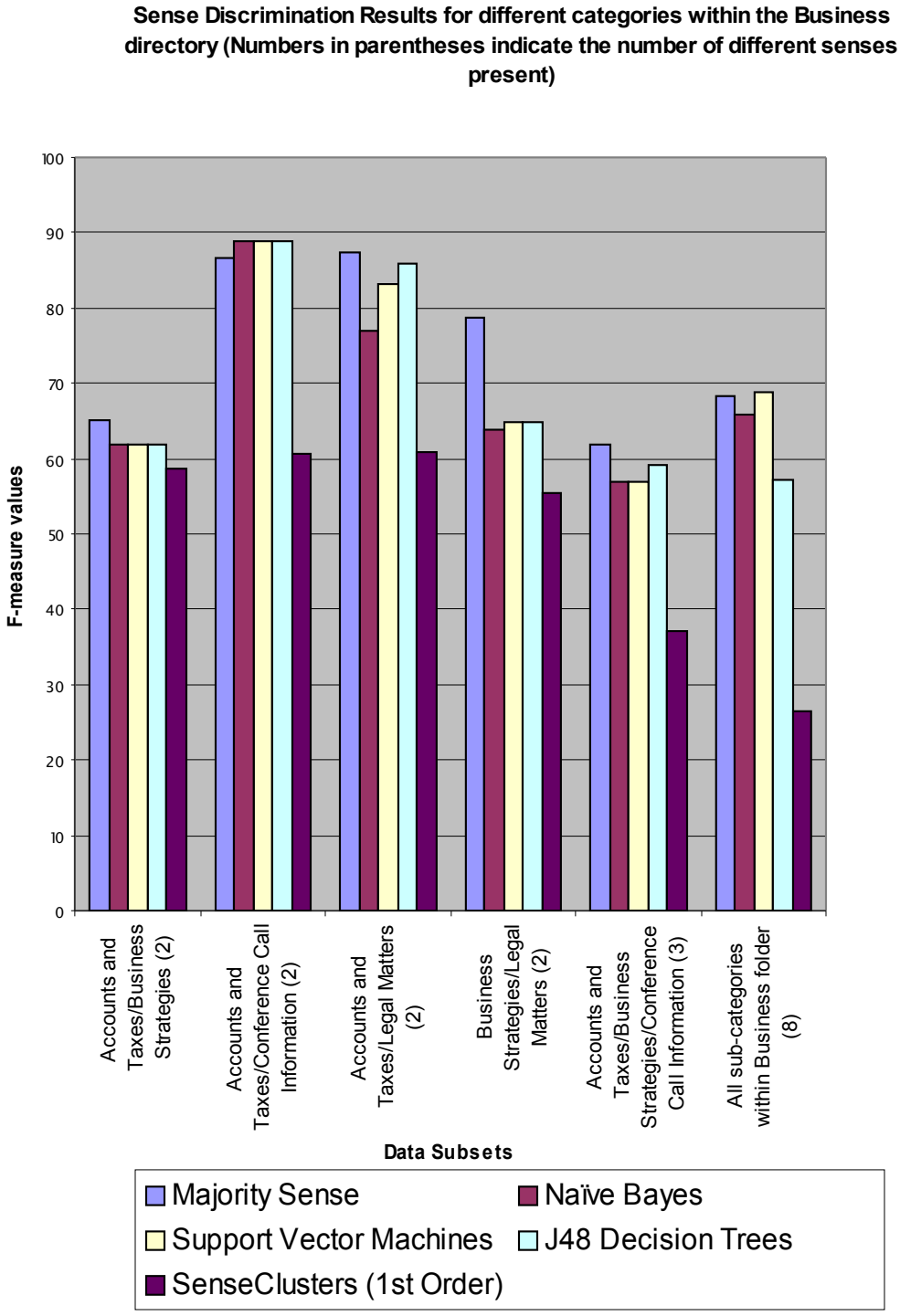

Figure 12: Graph showing the sense discrimination results for different categories within the Business directory.

Sub-directories contained within the sub-directories of the Business folder:

5.3.13 Difficult to classify – Difficult content/Only Attachments (2 classes):

In this data subset, we try to classify the different categories within the Difficult to classify sub-folder of the Business directory. Here, we see that two methods perform better than the majority sense, while two methods perform worse than the majority sense. The percentage value for the majority sense is 52.08. The unsupervised method – with an F-measure value of 48.65 and the Naïve Bayes classifier with an F-measure value of 48.78 – perform worse than the majority sense. However, Support Vector Machines, with an accuracy of 53.66 and J48 Decision trees, with an accuracy of 56.10 achieve an enhancement in performance, as compared to the majority sense.

5.3.14 Schedule Meetings – Actual Meetings/Conference Calls (2 classes):

This data subset consists of the sub-folders contained within the Schedule Meetings sub-directory of the Business folder. In this dataset, the majority sense has a value of 76.19. The performance of the supervised methods is better than the majority sense. The Naïve Bayes classifier has an accuracy of 77.78, J48 Decision trees have an accuracy of 80.00 and Support Vector Machines have an accuracy of 82.22. The F-measure value of the unsupervised method is 68.97, and not as exciting.

5.3.15 General Plans – Events/Scheduling/Travel (3 classes):

This data subset contains three classes, and consists of the sub-folders contained within the General plans directory, which in turn is present in the Business folder. Here, we see that the supervised methods perform much better than the majority sense. The majority sense has a percentage value of 51.96. The accuracies of both Support Vector Machines

and J48 Decision trees are 61.68, while the accuracy of the Naïve Bayes classifier is 64.49. The F-measure value of the unsupervised method is 49.76.

5.3.16 Legal Matters – Contracts/Legal Documents/News and Info (3 classes):

This dataset consists of the Legal matters sub-directory of the Business folder, and the three classes are the directories contained within it. Here again, we see that none of the learning methods performs better than the majority sense. The F-measure value for the unsupervised method is 47.67. The accuracy for the Naïve Bayes classifier is 63.92, while it is 64.95 for both, Support Vector Machines and J48 Decision trees. The majority sense has a percentage value of 66.67.

5.3.17 Information – All (17 classes):

This is the most complex data subset that we have conducted experiments on. It focuses on the Information sub-folder of the Business directory, which has seventeen sub-folders, resulting in seventeen classes for this dataset. The majority sense for this dataset has a percentage value of 25.29. All the supervised methods perform better than the majority sense. The Naïve Bayes classifier, J48 Decision trees and Support Vector Machines have accuracies of 28.77, 29.66 and 30.87 respectively. The F-measure value for the unsupervised method is 16.02.

The table and bar graph on the following pages give us a clearer idea of the results in a tabular and graphical representation.

Table 5: Table showing the results for the sub-sub-directories within the Business directory

Data Subset	Unsupervised	Supervised
Business/Difficult to classify Difficult or no content/ Only attachments	Precision = 63.29 Recall = 34.01 F-Measure = 48.65 Majority sense = 47.92	**F-measure** Naïve Bayes = 48.78 SVM = 53.66 J48 Decision Trees = 56.10
Business/Schedule Meetings Actual meetings/ Conference calls	Precision = 71.43 Recall = 66.67 F-Measure = 68.97 Majority sense = 76.19	**F-measure** Naïve Bayes = 77.78 SVM = 82.22 J48 Decision Trees = 80
Business/General Plans Events/Scheduling/ Travel	Precision = 50.98 Recall = 48.60 F-Measure = 49.76 Majority sense = 51.96	**F-measure** Naïve Bayes = 64.49 SVM = 61.68 J48 Decision Trees = 61.68
Business/Legal Matter Contracts/Legal Docs/News and Info	Precision = 47.92 Recall = 47.42 F-Measure = 47.67 Majority sense = 66.67	**F-measure** Naïve Bayes = 63.92 SVM = 64.95 J48 Decision Trees = 64.95
Business/Information All	Precision = 16.40 Recall = 15.66 F-Measure = 16.02 Majority sense = 25.29	**F-measure** Naïve Bayes = 28.77 SVM = 30.87 J48 Decision Trees = 29.66

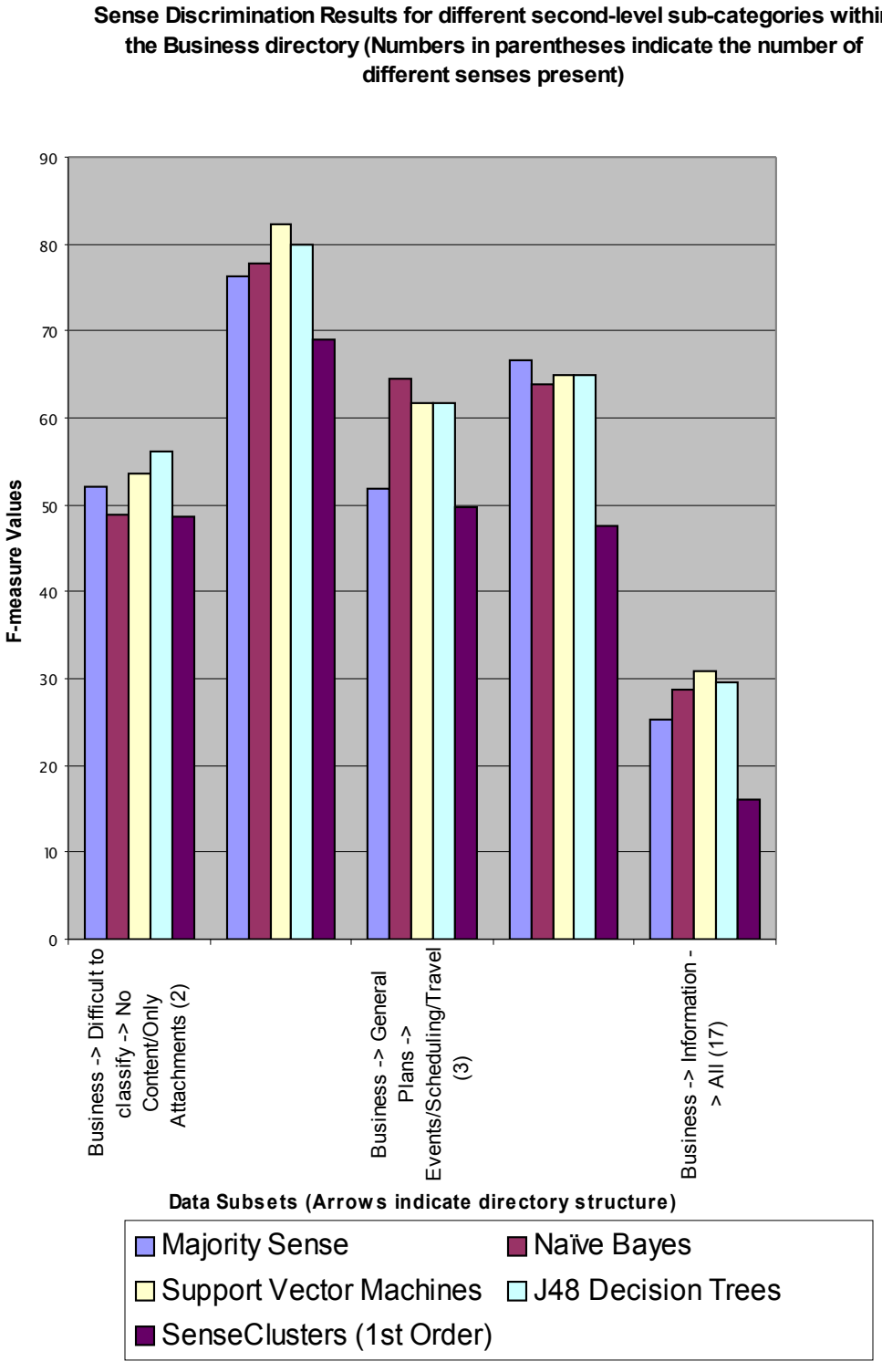

Figure 13: Graph showing sense discrimination results for different sub-directories within the business directory.

Sub-directories within General Announcements and EnronOnline directories

5.3.18 General Announcements – Miscellaneous/News (2 classes):

This folder consists of the sub-directories contained within the General Announcements directory. This is another example wherein none of the algorithms manage to perform better than the majority sense. The F-measure value is 67.18 for the Naïve Bayes classifier, while it is 66.26 for Support Vector Machines and it is 66.87 for J48 Decision trees. The F-measure value obtained by using SenseClusters is 53.67. All these values are much below the percentage value for the majority sense, which is 68.25.

5.3.19 EnronOnline – Inform – Announce/Ask Questions (2 classes):

This data subset focuses on the Inform sub-directory of the EnronOnline folder, which has two classes – Announce and Ask Questions. Here, we see that two methods perform better than the majority sense, while two methods perform worse than the majority sense. The percentage value for the majority sense is 78.31. The unsupervised method – with an F-measure value of 63.10 and the Naïve Bayes classifier with an F-measure value of 70.59 – perform worse than the majority sense. However, Support Vector Machines and J48 Decision trees, both with accuracies of 80.00 achieve an enhancement in performance, as compared to the majority sense.

5.3.20 EnronOnline – Inform/Network/Security (3 classes):

This data subset consists of the sub-directories within the EnronOnline directory. It contains three classes. This dataset is extremely skewed with the percentage value of the majority sense being 94.32. Here again, we see that two methods perform better than the majority sense, while the other two methods perform worse than the majority sense. The unsupervised method – with an F-measure value of 35.96 and the Naïve Bayes classifier with an F-measure value of 90.00 – perform worse than the majority sense. However, Support Vector Machines and J48 Decision trees, both with accuracies of 56.10 achieve an enhancement in performance, as compared to the majority sense.

The table and bar graph on the following pages give us a clearer idea of the results in a tabular and graphical representation.

Table 6: Table showing the results for the sub-directories within EnronOnline and General Announcements directories

Data Subset	Unsupervised	Supervised
General	Precision = 54.60	**F-measure**
Announcements	Recall = 52.76	Naïve Bayes = 67.18
Miscellaneous/News	F-Measure = 53.67	SVM = 66.26
	Majority sense = 68.25	J48 Decision Trees = 66.87
EnronOnline/ Inform	Precision = 63.86	**F-measure**
Announce/Ask Questions	Recall = 62.35	Naïve Bayes = 70.59
	F-Measure = 63.10	SVM = 80
	Majority sense = 78.31	J48 Decision Trees = 80
EnronOnline	Precision = 36.36	**F-measure**
Inform/Network/Security	Recall = 35.56	Naïve Bayes = 90
	F-Measure = 35.96	SVM = 94.44
	Majority sense = 94.32	J48 Decision Trees = 94.44

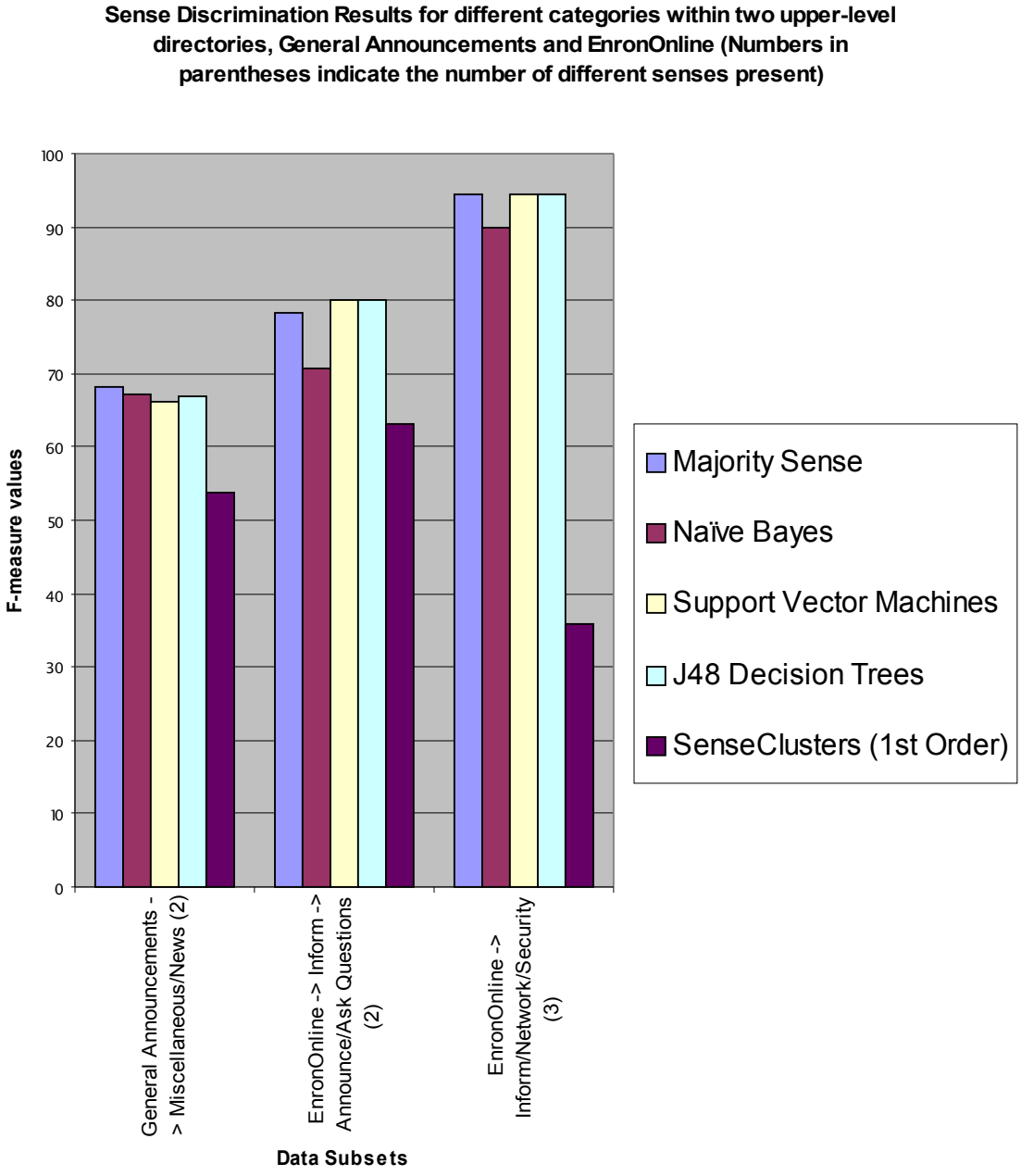

Figure 14: Graph showing the sense discrimination results for different sub-directories within General Announcements and EnronOnline directories.

Sub-directories within Personal and Human Resources folders:

5.3.21 Human Resources – Interviews – Follow-up/Lunch-Dinner/Schedule Interviews (3 classes):

This data subset consists of the Interviews sub-folder contained within the Human Resources directory. It has three sub-folders, thereby creating three classes. Here, we see that two methods perform better than the majority sense, while two methods perform worse than the majority sense. The percentage value for the majority sense is 67.12. The unsupervised method – with an F-measure value of 38.93 and the Naïve Bayes classifier with an F-measure value of 64.47 – perform worse than the majority sense. However, J48 Decision trees with an accuracy of 71.05, and Support Vector Machines with an accuracy of 72.37 achieve an enhancement in performance, as compared to the majority sense.

5.3.22 Human Resources – New Recruitment – Internships/New Hires/Transfers (3 classes):

This data subset consists of the New Recruitments sub-folder of the Human Resources directory. It contains three sub-classes, namely – Internships, New Hires and Transfers. This is another example wherein none of the algorithms manage to perform better than the majority sense. The F-measure values for all the three supervised learning algorithms are 59.52. The F-measure value obtained by using SenseClusters is 58.54. All these values are slightly below the percentage value for the majority sense, which is 60.00.

5.3.23 Personal – Memberships – Advertisements/Groups/Institutions (3 classes):

This data subset consists of the Memberships folder contained in the Personal directory. It contains three categories, giving us data that can be divided into three classes. Here, we see that two methods perform better than the majority sense, while two methods perform worse than the majority sense. The percentage value for the majority sense is 55.93. The unsupervised method – with an F-measure value of 53.94 and the Naïve Bayes classifier with an F-measure value of 54.47 – perform worse than the majority sense. However, Support Vector Machines, with an accuracy of 58.54, and J48 Decision trees, with an accuracy of 60.98 achieve an enhancement in performance, as compared to the majority sense.

5.3.24 Personal – Keep in Touch – All (6 classes):

This data subset consists of the whole Keep in Touch sub-directory of the Personal folder, with its six sub-folders. The results obtained for this dataset are again very different from all the results that we have seen so far. For this dataset, not only does the unsupervised method achieve better performance than the majority sense, but it also gives better results than the Naïve Bayes classifier and J48 Decision trees, two of the supervised learning methods. The percentage value of the majority sense is 30.25, whereas the accuracy of the Naïve Bayes classifier is just 23.43 and that of J48 Decision trees is 43.97. The F-measure value of the unsupervised method is 44.58. The accuracy of Support Vector machines is highest at 46.65.

5.3.25 Personal – All (7 classes):

This data subset consists of the entire Personal directory, with all seven of its sub-folders. Here again, we see that two methods perform better than the majority sense, while the

other two methods perform worse than the majority sense. The unsupervised method, with an F-measure value of 38.67 and the Naïve Bayes classifier with an F-measure value of 20.17 – perform worse than the majority sense that has a percentage value of 56.07. However, J48 Decision trees, with an accuracy of 65.99 and Support Vector Machines, with an accuracy of 66.62 achieve an enhancement in performance, as compared to the majority sense.

5.3.26 Human Resources – All (12 classes):

This data subset consists of the entire Human Resources directory with all twelve of its sub-directories. Hence, we have twelve classes within this dataset. The results of this dataset are extremely encouraging for supporters of unsupervised learning methods. The unsupervised method has achieved the highest accuracy for this dataset amongst all the algorithms. The majority sense, here has a percentage value of 18.73. This indicates that the data is well balanced and no single class dominates over the others. The accuracies of both, the Naïve Bayes classifier and J48 Decision trees are 34.73, while the accuracy of Support Vector Machines is 38.69. The highest F-measure value is 40.95, that of the unsupervised method.

The table and bar graph on the following pages give us a clearer idea of the results in a tabular and graphical representation.

Table 7: Table showing the results for the sub-directories within Personal and Human Resources directories

Data Subset	Unsupervised	Supervised
Human Resources/ Interviews Follow-up/Lunch Dinner/ Schedule Interview	Precision = 39.73 Recall = 38.16 F-Measure = 38.93 Majority sense = 67.12	**F-measure** Naïve Bayes = 64.47 SVM = 72.37 J48 Decision Trees = 71.05
Human Resources/ New Recruitment Internship/New Hires/ Transfers	Precision = 60.00 Recall = 57.14 F-Measure = 58.54 Majority sense = 60.00	**F-measure** Naïve Bayes = 59.52 SVM = 59.52 J48 Decision Trees = 59.52
Personal/Membership Advertisements/Groups/ Institutions	Precision = 55.08 Recall = 52.85 F-Measure = 53.94 Majority sense = 55.93	**F-measure** Naïve Bayes = 54.47 SVM = 58.54 J48 Decision Trees = 60.98
Personal/Keep in touch (6 clusters) All	Precision = 47.25 Recall = 42.19 F-Measure = 44.58 Majority sense = 30.25	**F-measure** Naïve Bayes = 23.43 SVM = 46.65 J48 Decision Trees = 43.97
Personal (7 clusters) All	Precision = 40.59 Recall = 36.93 F-Measure = 38.67 Majority sense = 56.07	**F-measure** Naïve Bayes = 20.17 SVM = 66.62 J48 Decision Trees = 65.99
Human Resources (12 clusters) All	Precision = 41.85 Recall = 40.09 F-Measure = 40.95 Majority sense = 18.73	**F-measure** Naïve Bayes = 34.73 SVM = 38.69 J48 Decision Trees = 34.73

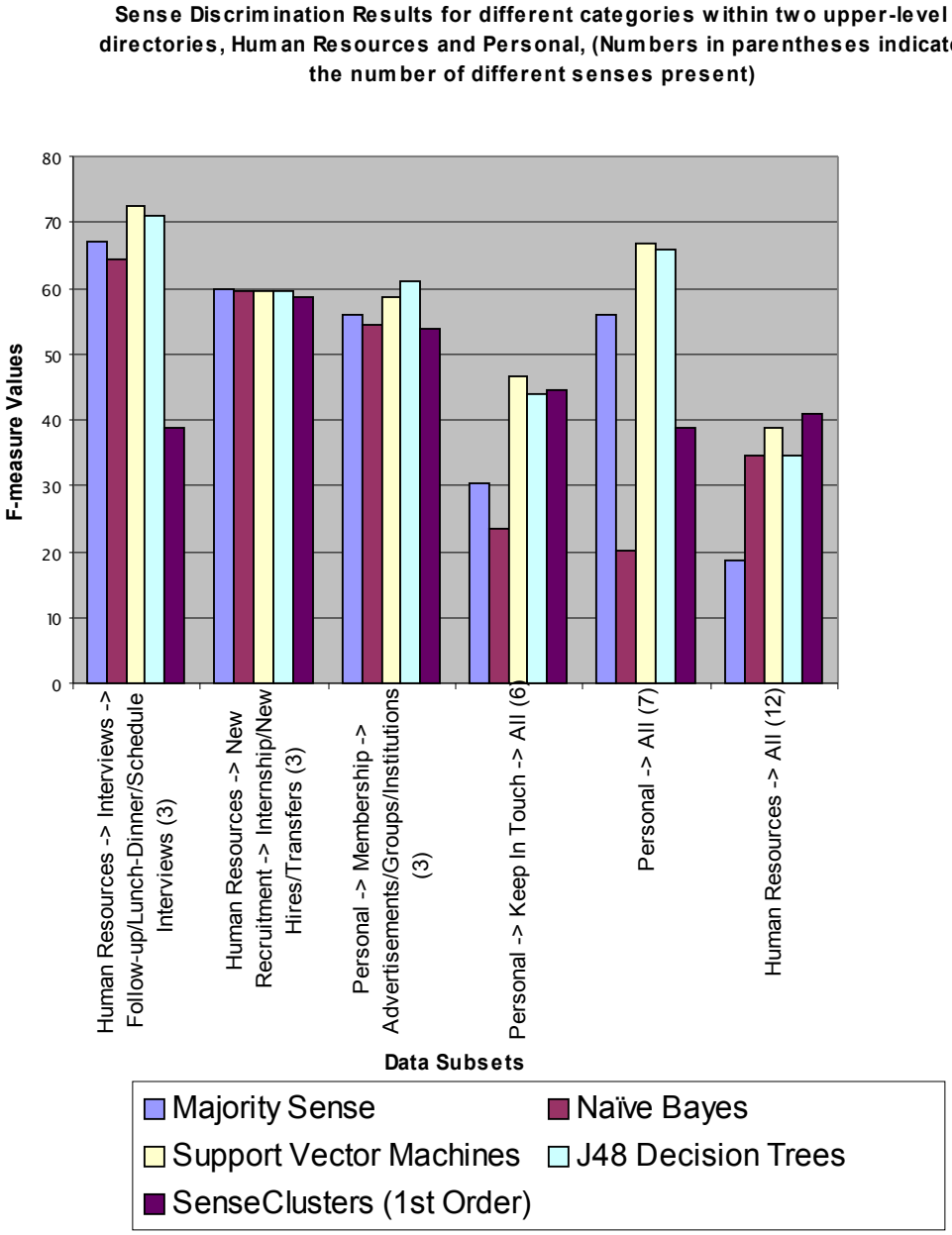

Figure 15: Graph showing sense discrimination results for different categories within Human Resources and Personal directories.

5.4 Analysis of Results:

5.4.1 The results of the topmost category show the best improvement over the majority sense:

This is expected, since the topmost category shows results for experiments between Business-Personal, Business-Human Resources, Personal-Human Resources, etc. Since the sense discrimination is most identifiable at this upper level, this is why the results are better for the uppermost level. All the remaining charts show results for sub-folders within a particular folder, and hence the sense distinction there is not as marked.

5.4.2 Supervised methods score over unsupervised methods in most of the cases:

As we saw in the results, except for a couple of cases, the performance of supervised methods was better than the performance of the unsupervised method. However, considering the fact that supervised methods had the entire corpus available as training data, this result is not surprising.

5.4.3 However, the performance of the unsupervised method was pretty close to that of supervised methods in many cases:

The performance of the unsupervised was really close to that of the supervised methods in a significant number of cases. This is an encouraging observation and can thus be a point in favor of unsupervised methods since we can get similar results (with only a 2-3% drop in accuracy) without having to provide training data.

5.4.4 Among supervised methods, Support Vector Machines shows the best results a majority of the times:

Though J48 Decision trees did have a higher accuracy than Support Vector Machines in a couple of cases, overall, we can easily see that Support Vector Machines outperform all the other classifiers. This is expected and agrees with the outcome of the experiments conducted by Ron Bekkerman, wherein Support vector machines gave the best results.

5.4.5 In some cases, we see that the Naïve Bayes classifier performs miserably, while in most cases its performance is pretty close to that of the other supervised learning algorithms.

The results of experiments 5.3.3, 5.3.24 and 5.3.25 show that the performance of the Naïve Bayes classifier can be really bad in certain cases. On looking at the categories included in the experiments, we see that the Personal directory, or some of its sub-directories, is common in all these experiments. The personal directory is not as complex in structure as the other upper-level directories. Hence, the assumption of variable independence made by the Naïve Bayes classifier seems to fail in this case, as there are many examples with a given combination of attributes present in the Personal directory. This may be the reason why the Naïve Bayes classifier does not perform well in the above-mentioned examples.

5.4.6 The performance of all classifiers was mostly found to better the majority sense in the experiments conducted on the upper-level directories.

There can be two reasons for this result. Firstly, the upper-level directories are broad categorizations and hence, it is easier for the classifiers to discriminate amongst the different senses at this level. As we move lower in the hierarchy, the sense discrimination gets more fine-grained. Hence, subtle sense discriminations that may seem simple enough for humans to make can tend to confuse the classifier. Secondly, as we move lower within any directory, we see that the data becomes more skewed. The majority sense can be very high at the sub-directory level. This makes the benchmark very high, making it difficult for the classifiers to achieve an enhancement in performance over the benchmark.

6 Related Work

The purpose of this thesis is to create a corpus of real-life e-mail messages, in which the messages have been categorized on the topic or context of the e-mail messages; and then compare the results obtained by applying supervised and unsupervised learning methods on this e-mail corpus. Other people have also worked on the Enron corpus, and people have worked on e-mail classification. However, we have combined all of this in our thesis.

In this section, we shall see some of the work done by other people on the Enron corpus. There is a section on papers that introduced the Enron corpus, work done by other people on the Enron corpus, and finally we shall see previous work done on classifying e-mail messages using supervised and unsupervised learning methods.

6.1 Introduction of the Enron corpus:

Brian Klimt and Yiming Yang are amongst the first people to handle the Enron corpus. They also published a paper that briefly introduces and describes the Enron corpus [1]. The most significant finding of Klimt and Yang is the fact that the Enron corpus is representative of general e-mail messages [1]. This one fact is very crucial for our thesis as there is no other source of real-life e-mail messages currently available that is as large, as diverse, or as interesting as the Enron corpus. Previously, all research material used for e-mail classification was limited as it was created from e-mail messages of a small group of people. Due to this limitation, it did not capture the essence of e-mail classification strategies as used by real users. Just the sheer volume of the Enron corpus makes it priceless in terms of research. It is large and diverse enough to help us understand the e-mail foldering strategies of a wide range of e-mail users.

Klimt and Yang [2] report that their version of the (cleaned) corpus now contains a total of 200,399 e-mail messages, belonging to the original 158 users. Thus, on an average, every user has 757 messages. However, the distribution of e-mail messages amongst users is obviously not uniform. Rather, Klimt and Yang say that the e-mails are divided in an exponential manner. What this means is that most of the e-mail messages are limited to a small number of users, whereas a majority of the users have comparatively few messages. This forms the basis of their claim that the Enron dataset captures the e-mailing styles or pattern of all kinds of users, with varying amounts of e-mail messages either sent or received.

Over the course of their research, Klimt and Yang have deduced the following key points. Firstly, they say that the Enron dataset confirms the widely held belief that a majority of e-mail users have some kind of foldering strategy in order to classify e-mail. This strategy is of course, unique to every individual user. That is, the number of folders created and the granularity of the classification will vary from user to user. They also mention that the number of folders created by the user is in no way indicative of the number of e-mail messages stored and/or received. This implies that there is no lower bound for the number of folders created by any user, irrespective of the number of e-mail messages he/she has. However, they do point out the obvious fact that no user has more folders than actual e-mail messages. Also, Klimt and Yang have arrived at the conclusion that the upper bound for the number of folders created by each user tends to be a log (to the base 2) of the number of messages he/she receives.

Klimt and Yang then go on to describe their research in the field of automatic e-mail classification, using supervised learning methods. They have made use of the Enron corpus as well as another, smaller e-mail corpus created at Carnegie Mellon University (CMU), to carry out various experiments using the Support Vector Machines (SVM) classifier. The CMU dataset was created by collecting e-mail messages from students and a faculty member at the Language Technology Institute of CMU. For each of the corpora,

the data was split into the training set and the test set. This was done by simply sorting the e-mail messages of every user according to the date and then dividing them into half. The first half was used as the train set, whereas the second half was the test set.

Then, the SVM classifier was used to categorize the e-mails into folders automatically. The accuracy of the classification was tested by comparing the results of this classification to the actual folder a message actually belonged to as per the user's foldering strategy. This was done in two ways. In one approach, the whole e-mail message was treated as a bag-of-words, wherein no special importance is given to any particular part of the e-mail; rather the entire e-mail is treated like a collection of words. In the second approach, the e-mail was separated into various fields consisting of certain header fields (From, To, Subject) and the body of the e-mail message. Each of these fields had a certain score/weight associated with it, depending on the importance of that particular field. It is important to note that Klimt and Yang did not consider the Date as one of the fields, as this was not text data. Finally, the SVM scores of each of the different fields were linearly combined to reach a final SVM score for the entire e-mail message. The message was then placed in the appropriate folder, depending on this combined score.

On the basis of these experiments, Klimt and Yang deduce that the "From" field of the header and the body of the e-mail message are the fields which best help to classify e-mail messages into user-defined folders. The bag-of-words approach is not as effective as the other approach, wherein we associate a certain weight to every field. The best results were obtained by a method in which ridge regression was used to linearly combine all the header fields. This in fact, reflects our intuitive understanding that a user classifies adopts a more or less holistic approach to e-mail classification, in that, the classification is based on a number of things, and not just one particular field.

Klimt and Yang also mentioned that Support Vector Machines performed better with users who had more folders, since SVM works best on classes with more training data.

Also, they stress the fact that the number of folders created, and not the number of e-mail messages, are more important in automatically classifying a user's e-mail into folders. Also, both the Enron and the CMU datasets gave similar results.

The most important result of their research is the detection of "*E-mail threads.*" Klimt and Yang define a thread to be a set of e-mail messages that belong to a certain group of people and deal with a particular subject [2]. They decided whether or not a message dealt with the same subject from the "Subject" header. Messages whose "Subject" fields were empty were ignored. However, the detection of threads is difficult to evaluate since subjects of e-mail messages tend to vary from user to user. Also, messages may begin with a certain subject, but as the conversation proceeds, they may drift into an entire new topic.

In the Enron corpus, Klimt and Yang thus discovered a total of 30,091 "non-trivial" threads. What we mean by non-trivial is that the threads contained more than one e-mail message. Thus, they conclude that the average thread size is 4.1 messages. Again, larger threads tend to make up a small percentage of the total threads, but they contain the highest number of e-mail messages. The larger threads are more useful for research as they provide more information, but there are very large threads. Klimt and Yang also discovered that on average, messages pertaining to a single thread are distributed among 1.37 folders.

This is an encouraging result as it confirms the widely held belief that a majority of email users also think of the messages in a thread as similar and hence, distribute them among very few folders. Klimt and Yang also suggest that working on the temporal aspect of e-mail will definitely lead to better results, and suggest this as a future field of research [2].

6.2 Work done on the Enron corpus and in e-mail classification:

In this section, we shall see the work done by Ron Bekkerman, whose corpus is the superset of the corpus created by Padhye and Pedersen. Ron Bekkerman has made use of four supervised learning methods – Naïve Bayes, Maximum Entropy, Support Vector Machines and Wide Margin Winnow – in order to classify the e-mail messages from his corpus. We shall also look at the work done by Kulkarni and Pedersen in the field of automatic classification of e-mail messages, using unsupervised methods. Like we have done, Kulkarni and Pedersen have also made use of SenseClusters for this purpose.

6.2.1 Work done by Ron Bekkerman:

Ron Bekkerman has done the work most relevant to our thesis. In their paper, Bekkerman et al., [3] state and describe various experiments carried out on the Enron corpus and the results of those experiments. Bekkerman et al., have made use of a number of Supervised Learning approaches on the e-mail directories of seven ex-employees of the Enron Corporation. These seven people are Sally Beck (Chief Operating Officer), Darren Farmer (Logistics Manager), Vincent Kaminski (Head of Quantitative Modeling Group), Louise Kitchen (President of EnronOnline), Michelle Lokay (Administrative Assistant), Richard Sanders (Assistant General Counsel) and William Williams III (Senior Analyst). Although the original dataset contains messages from over 150 people, these seven have been selected because of the number of e-mail messages and folders that were present in their respective directories. As of now, we are also classifying and studying e-mail messages contained in the directories of these people only.

Bekkerman et al., have also made use of another dataset to aid in comparison of results. This dataset is a subset of the CALO DARPA/SRI corpus and is called the SRI dataset, named after the Stanford Research Institute. As in the case of the Enron dataset, the folder hierarchies have been flattened, all non-topical folders have been deleted and very small folders have been eliminated. In creating the SRI corpus, seven users with the

largest number of e-mail messages from the original CALO DARPA/SRI corpus have been chosen. These people are A. Cheyer, B. Mark, D. Israel, M. Gervasio, M. Gondek, R. Perrault and V. Chaudri.

This relatively small dataset consists of a total of 3559 e-mail messages distributed among 143 folders. Hence, we have approximately 25 messages per folder. The number of e-mails present in the smallest folder is 3, whereas the largest folder contains 205 e-mail messages. Hence, this dataset is around 17.5%, that is, less than $1/5^{th}$ the size of the Bekkerman dataset.

Like Klimt and Yang, Bekkerman et al., have also stressed the temporal aspect of e-mail classification. As stated previously, e-mail classification is very different from general text classification due to the fact that e-mail messages are extremely sensitive to time. Hence, Bekkerman et al., feel that classification of e-mails into folders should also be done in such a way as to keep the time-dependent nature of e-mail in context. To facilitate the realistic, time-based classification of e-mail message they introduce an enhanced version of the Winnow classifier [3]. In addition to this, they have made use of three other popular classifiers – namely, Maximum Entropy, Naïve Bayes and Support Vector Machines. All four classifiers were used to categorize e-mail messages from both the corpora.

Bekkerman et al., adopt a different approach from Klimt and Yang in dividing the data into train and test splits. They say that current e-mail is related to previous e-mail, and hence training the classifier on past e-mail, and using current or recent mail, as a test set is obviously the most realistic approach. This is because in the real world too, a system will be trained on previous data to help classify new, incoming e-mail messages. Hence Bekkerman et al., propose an incremental time-based splitting of training and test data [3]. This will be done after sorting the e-mail messages according to their dates. Their splitting of data differs from the Klimt and Yang approach in that they sort the e-mail messages according to their time stamps. They decided to train classifiers on batches of

'N' messages, where N can be any random value. Bekkerman et al. used N = 100 for the Enron corpus, and N = 50 for the SRI corpus. Then, a classifier is trained on N messages, and tested on the next N messages. Next, you train the classifier on 2N messages and test it on the next N messages. Then you train it on 3N messages, and continue further till you finally train the classifier on (K-1) N messages and test it on the remaining N messages, where 'K' is the total number of sets of N present in the corpus. It should be noted here that though the results of this approach are not as good as the random splitting method, it is important to train a classifier according to the time an e-mail message was sent because this best simulates a real-world scenario.

Bekkerman et al., then go on to describe the various classifiers that they have used, and the method of parameter selection for each. We shall discuss the classifiers briefly, since they are relatively well known. The first classifier used was Maximum Entropy. The principle of maximum entropy states that when one has only partial information about the probabilities of possible outcomes of an experiment, one should choose the probabilities so as to maximize the uncertainty about the missing information. Put another way, since entropy is a measure of randomness, one should choose the most random distribution subject to whatever constraints are imposed on the problem [11]. Bekkerman et al., use a quasi-Newton method, called BFGS which is an acronym for Broyden-Fletcher-Goldfarb-Shanno, who were the inventors of this method [12]. They state that this method suffers from overfitting in case of sparse data.

The second classifier used was Naïve Bayes. A Naive Bayes classifier is a simple probabilistic classifier. Naive Bayes classifiers are based on probability models that incorporate strong independence assumptions which often have no bearing in reality, hence are (deliberately) naïve [13].

The third classifier used was Support Vector Machines. Support Vector Machines are a set of related supervised learning methods, applicable to both classification and regression. They are usually used for two-class problems, but can be adapted for multi-

class problems by decomposing multi-class problems to many binary sub-problems. Bekkerman et al., used a simple linear kernel, which has so far been proven to be very effective for text classifications.

The last classifier that Bekkerman et al., used was the Wide Margin Winnow, which is an extension of the Winnow classifier. We shall first see how the Winnow classifier works, and then understand the modifications made in Wide Margin Winnow. The Winnow classifier is a learning algorithm that automatically adjusts weights during training in case of an incorrect labeling. It is very similar to a perceptron, but instead of doing additive updates, the Winnow does multiplicative updates. The Wide Margin Winnow method is an adaptation of the Winnow algorithm. In Wide Margin Winnow, instead of adjusting weight vectors after an incorrect guess only, they are also adjusted when an answer is "barely correct".

The results of the experiments carried out by Bekkerman et al., are definitely interesting, if not very encouraging. The results obtained for the Enron corpus clearly show that Support Vector Machines are the best classifier for the current data. Support Vector Machines are the best overall for the SRI corpus too, but what is encouraging is the fact that Wide Margin Winnow outperforms Support Vector Machines in 3 out of the 7 subsets of the SRI corpus. This is significant because Wide Margin Winnow is known to perform well with sparse, multi-dimensional data. The obtained results also prove this point. Also, Wide Margin Winnow is the fastest classifier in this group after Naïve Bayes. It outputs results in a fraction of the time required for Support Vector Machines to train and test the data.

To conclude, Bekkerman et al., state that the classification results for both datasets are significantly low, when we do a time-based split. Results are slightly better for the Enron dataset, with a much greater size than the SRI dataset. They also state that much better results can be obtained if traditional text classification methods for dividing train and test data are used, but the incremental time-based split is much more realistic [3]. They go on

to say that this displays the complexity of the problem, and the importance of creating an efficient time-based measure. Bekkerman et al., also conclude that the accuracies of the classification are higher for users whose e-mail messages are divided into one or two dominant folders. Also, newly created folders significantly affect the accuracy due to the incremental time-based splitting of data.

6.2.2 Work done on e-mail classification by Kulkarni and Pedersen:

At the outset, it is important to know that Kulkarni and Pedersen do not work on the Enron corpus. They work on the 20 Newsgroups corpus of USENET mailing list articles. This corpus consists of about 19,997 USENET articles that have been pre-classified into twenty different categories, with nearly similar number of articles in each category. Some of the categories found here are *Talk politics miscellaneous, Computer Systems Mac Hardware, Science Space,* etc. Hence, the work done by Kulkarni and Pedersen is unsupervised clustering of these articles according to topics. Unlike us, they do a topic-wise categorization of the articles, rather than actual e-mail clustering.

Kulkarni and Pedersen [6] have made use of SenseClusters to carry out the unsupervised e-mail categorization. They have made use of the bigram feature and the log-likelihood ratio with a cut-off of 3.841 for ranking the associativity between the bigrams. As in our experiments, the value of the *remove* feature for their experiments was 5. They have made use of the standard English stoplist, rather than any e-mail specific stoplist.

Kulkarni and Pedersen report that the results obtained for e-mail classification are not as good as those obtained for Name Discrimination. They suggest that this may be because e-mail tends to create noisy context vectors, and this can affect the results. Also, e-mail messages are not as structured as news articles (which have been used for Name Discrimination), and contain a large amount slang or regional vocabulary. Thirdly, they have not filtered out any of the headers from their data. Also, they have not used any e-

mail specific stoplist. According to Kulkarni and Pedersen, all these factors contribute towards the lower performance of the method on the 20 News Groups data, when used for e-mail classification.

6.3 Work done on the Enron corpus, other than classification of e-mail messages: -

Jitesh Shetty and Jafar Adibi have also worked on the Enron corpus, though their field of research is completely different. Shetty and Adibi do not work on e-mail classification. Their research is focused on creation and analysis of the social networks that can be found in the Enron corpus. In their paper, Shetty and Adibi briefly describe the Enron corpus. What is interesting about the work done by Shetty and Adibi is that they have created a MySQL database for the Entire Enron corpus, in an effort to help the statistical analysis of data. The tables created have been described in the previous section. They have made use of the original Enron corpus, as distributed by William Cohen.

Shetty and Adibi have made relation tables to derive a social network from the Enron corpus. They define a social network as a network of those employees within the Enron Corporation, who had social contacts with each other [4]. A social contact is established if two people have exchanged more than a certain threshold number of e-mail messages; the threshold was 5 in their experiments. It is important to note that there should be an exchange of e-mail involved. Messages sent in one direction only, or announcements sent out to a whole group do not qualify as a social contact. The derived social networks help us to understand the interactions between people at various levels in the Enron Corporation before, during and after the scandal.

Social networks provide a framework for further analysis of the data contained within e-mail messages. Shetty and Adibi suggest that it will help us better understand who sent e-mails to whom, in what capacity and also, whether the e-mails pertained to any known

criminal offenses within the Enron Corporation. What this means is that, it will help us understand whether or not the employees involved in the scandal knew that they were a part of a major scandal.

It is interesting to note that networks do give us an idea of what is happening within an organization. As an example, Shetty and Adibi provide an image of the social networks seen in the Enron Corporation for the same period of time – the months of October, both in 2000 and 2001. It is seen that the social network is very different in October 2001, compared to what it was in October 2000. The network is much more dense in October 2001, right around the time that Enron was going through extremely difficult times. This proves their claim that analyzing the social networks within an organization can help give us an insight into what's happening there.

Shetty and Adibi suggest that the networks also provide information on the statistical aspects of the Enron corpus, like the number of e-mails belonging to each user, and how that number changes over time. They contend that big organizations like Enron can make use of this information to help detect any fraudulent transactions, irregularities or anything out of the ordinary happening within the company. This analysis will help companies become aware of problems within the company in their incipient stage, and they can take preventive action to reduce the effect of these problems.

Though interesting, we do not study the social networks present in the Enron corpus. Rather, our research focuses more on the efficacy of the Enron corpus for the automatic categorization e-mail messages into user-defined folders. We have made use of both, Supervised and Unsupervised methods in classifying e-mails into folders. Since this is similar to the research done by Ron Bekkerman, we have used his dataset to create another subset of the Enron corpus. How this was done has been explained in previous sections. In the following sections, we shall see a bit more about our corpus, how we refined the Bekkerman corpus for our purposes and our methodology for analyzing the newly created, labeled dataset. We shall also see the results of our study, as well as the

limitations of our experiments. This will lead us to the final sections in which we discuss the results in detail, and suggest future areas of research.

7 Conclusions

The main aim of this thesis was to create and introduce a new corpus of manually annotated e-mail messages. Also, we wanted to compare the results of Supervised and Unsupervised techniques in the classification of messages from the Enron E-mail Corpus. Towards that end, we have created a subset of the Enron corpus, called the Padhye and Pedersen corpus. This corpus contains 3,021 e-mail messages, belonging to seven ex-employees of the Enron Corporation. The messages in this corpus have been manually annotated as belonging to one of six directories, depending on their context or topic.

We have also created the complete topic-wise hierarchy into which the e-mail messages are classified. At the upper level, these categories are broad, and get finer-grained as we move lower into a particular directory. In addition to this, we have created an e-mail specific stoplist in order to improve the performance of the classification methods used. We have also developed a package of Perl programs that can be used to convert the e-mail messages from XML format to the Senseval-2 format required by both, WSDShell and SenseClusters. These programs allow the user to either keep the headers, or filter them out of the corpus. In our experiments, we have filtered out the headers from the corpus.

We have then carried out several experiments, using supervised and unsupervised methods, on various subsets of the Padhye and Pedersen corpus. These have helped us understand how well the classifiers work on e-mail data, and what can be done to improve the performance of the classification algorithms.

We can conclude the following things based on the work that we have done in the course of this thesis.

The Enron e-mail corpus is representative of real-life e-mail. The tone of the e-mail messages and the content of these messages (both, Business and Personal) are very similar to the way people usually write e-mails. This makes the corpus an invaluable resource for carrying out research on e-mail.

1. The corpus tends to contain more business related e-mail. This is expected since the corpus contains e-mail messages exchanged between ex-employees of the Enron Corporation on their corporate accounts. Hence, it is but natural that they would exchange more business related messages from these accounts.

2. The results obtained confirm the fact that the categorization structure we have created is pertinent and valid. Some of the results obtained by supervised methods have accuracies in the 90s. This tells us that the hierarchical structure we have created tallies with what the classifiers think should be individual categories too.

3. The introduction of an e-mail specific stoplist greatly improves the performance of all classifiers. This stoplist filters out those terms from the corpus that can lead to the formation of noisy features. Hence, using this stoplist helps in enhancing the performance of the classifiers.

4. Supervised methods work better than unsupervised methods do on the task of automatic e-mail classification. This is expected since supervised methods have the facility of a training set, which gives them an edge over unsupervised methods, which do not have any training data.

5. Among supervised methods, Support Vector Machines work best, as compared to J48 Decision Trees and Naïve Bayes classifiers.

8 Future Work

The work done in this thesis tries to compare the results obtained by supervised and unsupervised learning methods on e-mail specific data. We have learnt some things from what we have done so far. However, there does exist scope for improving the obtained results by making certain refinements, as detailed below.

8.1 Filtering out only certain headers:

In our experiments, we have removed all the headers from the corpus, since the presence of the headers tended to confuse the classifiers, and resulted in noisy features. However, not all header information is useless. Certain headers like the Subject and Date fields can give us valuable information about the content of the e-mail message. Hence, removing only noise-inducing headers and retaining the others can improve the performance of the classifiers.

8.2 Making use of a more comprehensive subset of the corpus:

We have made use of a subset of the Enron corpus that contains e-mail messages exchanged between only seven ex-employees of the Enron Corporation. This can make the data more biased towards a particular category or type of e-mail messages. However, making use of a corpus that has e-mail messages from numerous users can add a little more variety to the available e-mail messages.

8.3 Creating data subsets that are more balanced in size:

The dataset that we have created is very skewed. The Business folder alone makes up for almost half the e-mail messages in the corpus. Though this is expected of business e-mail messages, having a corpus in which the categories are somewhat balanced will lead to better results than those obtained by us.

8.4 Designing better evaluation methods:

As mentioned previously, e-mail is dependent on time and the context of an e-mail message or thread can vary as time passes. Also, a message can belong to more than one folders. Hence, we need to have an evaluation method that takes these factors into consideration. The method has to allow for the time-sensitive nature of e-mail. Also, in cases wherein a message may belong to two or more folders, a foldering strategy that assigns the message to the multiple folders with a certain weighing scheme will prove to be useful.

Appendix

1. Example of an e-mail message from the Business folder: -

Sally,
We have been able to work with Legal and Tax to impact the way our trading has been set up in London and Tokyo to increase our control of the business and simplify the requirements for support.

In London, for Equity Trading we will be trading as ECT Investments Inc. (the same entity as in Houston). We will have mirror books in the UK as in Houston (e.g. Energy and Energy - London). We have put a service agreement in place setting up a Enron Investment Services Ltd. (uk entity) who will trade on our behalf as an agent. This keeps ENA from establishing a presence for Equity Trading in the UK Tax authority. We pay the UK entity, our agent, a service fee and deduct the service fee on ECT Investment's taxes as an expense. Controls are increased because all the books and reporting remains consolidated into one entity.

In Tokyo, we have started a Rate & Currency Trading Desk. We initially were told by tax that we would have to trade in the name of Enron Japan. This would require separate bank accounts, separate counterparty agreements and much more coordination to control. We and the trader worked with Tax so they understood the difficulties that this would present. Also, Japan did not want to assurme the funding requirements for this activity with their office and wanted it segregated. This further understanding caused Tax to push further to gather other advise from outside counsel operating in the Tokyo market.

We are and will be able to trade as ENA. Therefore, the same bank accounts and ISDAs may be utilized. All activity, positions and VAR continues to be netted for ENA. Additionally our counterparties can trade with a trusted, established entity, ENA, and we get to net, limiting exposure.

These are two cases where we can impact the structure of the business to increase controls and our level of support. In both cases, joint and successful cooperation occurred between trading, tax, legal and operations.

2. Example of an e-mail from the Personal folder: -

Hi!

Thought about calling you this weekend but thought since you had been gone most of the week I'd let your family have you all to themselves.

Did want to let you know I am changing jobs week after next. Will be going to viviance new education. They are out of Switzerland. I, of course, will work for their N.A. subsidiary. They are small, with other offices in ITaly, France, UK, Spain, and Germany. My passport is ready!

Will forward new work number to Patti when I am sure of what it will be. Our offices are on 6th Street on the 2nd and 3rd floor of an old building...will ride the bus...

Talk to you soon. Hope Canada was cool and your family is doing great.

3. Example of an e-mail from the Human Resources directory: -

Hiring Managers,

Here are a few tips that will immediately improve the time it takes to make an offer to your candidates.

All applicants must fill out and sign an Application and a Fair Credit Reporting Act form (included with the application) prior to being interviewed. You may either have the applicant fill out a form before you interview them or you can send the applicant to the 36th floor to fill out the forms before you interview them.

Notify Hector McLoughlin or Frank deJesus if you are interviewing an external candidate that is a former Enron employee. All previous Enron employees will be reviewed by HR and Sally Beck prior to being invited to interview.

We hope that this process will eliminate re-hiring poor performers. If you want to make an offer of employment to a candidate for a Management Position, please include Sally Beck in the process for input. As a final step of the selection process, Sally would like to meet with the candidate if time allows or to phone interview the candidate as an alternative. Sally's involvement should occur after you have identified the candidate as a potential Management new hire.

Thanks for your help with this process.

References

[1] Klimt, B., and Yang, Y.: Introducing the Enron Corpus. *First Conference on Email and Anti-Spam (CEAS)*, (2004).
http://www.ceas.cc/papers-2004/168.pdf

[2] Klimt, B., and Yang, Y.: The Enron Corpus: A New Dataset for Email Classification Research. In *Proceedings of ECML '04, 15th European Conference on Machine Learning*, pages 217-226, (2004).
http://springerlink.metapress.com/(deihe0450o3wtnm4yumgky45)/app/home/contribution.asp?referrer=parent&backto=searcharticlesresults,1,3;

[3] Bekkerman, R., McCallum, A., and Huang, G.: Automatic Categorization of Email into Folders: Benchmark Experiments on Enron and SRI Corpora. *Center for Intelligent Information Retrieval, Technical Report IR-418,* (2004)
http://www.cs.umass.edu/~ronb/papers/email.pdf

[4] Shetty, J., and Adibi, J.: The Enron Dataset: Database Schema and Brief Statistical Report. *Information Sciences Institute, University of Southern California, Technical Report,* (2004)
http://www-scf.usc.edu/~jshetty/Enron_EmailDataset_Report.pdf

[5] Diesner, J., and Carley, K.M.: Exploration of Communication Networks from the Enron Email Corpus. In *Proceedings of Workshop on Link Analysis, Counter-terrorism and Security, SIAM International Conference on Data Mining 2005,* pages 3-14, (2005)
http://www.andrew.cmu.edu/user/jdiesner/publications/diesner_carley_siam_enron_03_05.pdf

[6] Kulkarni A. and Pedersen T.: Name Discrimination and Email Clustering using Unsupervised Clustering and Labeling of Similar Contexts, In *Proceedings of the Second Indian International Conference on Artificial Intelligence*, (2005) http://www.d.umn.edu/~kulka020/iicai05-kulkarni.pdf

Websites

[7] William W. Cohen, "Enron Email Dataset," 4 April 2005, <http://www.cs.cmu.edu/~enron/>

[8] Marti Hearst, "UC Berkeley Enron Email Analysis,"*UC Berkeley Enron Email Analysis Project,* December 2004, <http://bailando.sims.berkeley.edu/enron_email.html >

[9] Andres Corrada-Emmanuel, "Enron Email Dataset Research ," January 2005, <http://ciir.cs.umass.edu/~corrada/enron/>

[10] SRI International's Artificial Intelligence Center, *Cognitive Agent that Learns and Organizes Project,* April 2006, <http://www.ai.sri.com/people/gervasio>

[11] Julie Jugdale, "Glossary," *Complexity in Social Science Project,* 11 October 2000, <http://www.irit.fr/COSI/glossary/fulllist.php>

[12] Semichem Inc., "AMPAC 8 User Manual," January 2004, <http://www.semichem.com/ampacmanual/bfgs_kw.html>

[13] Wikipedia, the free encyclopedia, "Naive Bayes Classifier," 29 May 2006, <http://en.wikipedia.org/wiki/Naive_Bayes_classifier>

[14] itmWeb Media Corporation, “Sherron Watkins eMail to Enron Chairman Kenneth Lay,” February 2006, <http://www.itmweb.com/f012002.htm>

[15] Wikipedia, the free encyclopedia, “Enron,” 12 June 2006, <http://www.wikipedia.org/wiki/Enro>

[16] BBC News Online, “The Enron Affair,” 17 February 2003, <http://news.bbc.co.uk/hi/english/static/in_depth/business/2002/enron>

[17] Time.com, “Behind the Enron scandal,” 20 January 2002, <http://www.time.com/time/2002/enron/collapse>

[18] WashingtonPost.com, “Timeline of Enron's collapse,” 30 September 2004, <http://www.washingtonpost.com/wp-dyn/articles/A25624-2002Jan10.html>

[19] Federal Energy Regulatory Commission, “FERC Western Energy Markets – Enron Investigation,” 18 October 2004, <http://www.ferc.gov/industries/electric/indusact/wem/pa02-2/info-release.asp>

[20] Leslie P. Kaelbling, “Homepage,” September 2003, <http://people.csail.mit.edu/lpk/lpk.html>

[21] Martin Hassel, “Class Stoplist,” 23 October 2004, <http://www.nada.kth.se/~xmartin/java/JavaSDM/moj/lang/StopList.html>

[22] Ted Pedersen and Mahesh Joshi, “Supervised Word Sense Disambiguation,” <http://www.d.umn.edu/~tpederse/wsdshell.html>

[23] George Karypis, “wCluto: A Web-Enabled Clustering Toolkit,” 30 June 2003, <http://www.pubmedcentral.gov/articlerender.fcgi?artid=523878>

[24] Chih-Jen Lin, “LIBSVM – A Library for Support Vector Machines,” April 2006, <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

[25] Ian H. Witten and Eibe Frank (2005) “Data Mining: Practical machine learning tools and techniques,” 2nd Edition, Morgan Kaufmann, San Francisco, 2005, <http://www.cs.waikato.ac.nz/ml/weka/>