# A FRAMEWORK FOR MINING EVOLUTION OF TOPICS FROM A LARGE EMAIL DATASET

by

Marianne Holland Gibson

A thesis submitted to the Faculty and Board of Trustees of the Colorado School of Mines in partial fulfillment of the requirements for the degree of Master of Science (Mathematical and Computer Sciences).

Golden, Colorado

Date 12 April 2007

Signed: Marianne Holland Gibson

Marianne Holland Gibson

Signed: Jae Young Lee

Dr. Jae Young Lee

Thesis Advisor

Golden, Colorado

Date 4/12/07

Signed: Graeme Fairweather

Dr. Graeme Fairweather

Professor and Head

Department of Mathematical and Computer Sciences

ABSTRACT

In 2002 the Federal Energy Regulatory Commission released the Enron email dataset to the public as part of its investigation. Existing research on this dataset rarely combines text mining techniques with temporal considerations. Such a large, varied dataset with a time stamp for each and every document provides a suitable dataset for studying the combination of these techniques. We present the use of a framework consisting of document clustering combined with time decomposition to mine the *evolution* of email discussion topics, thereby building our unique *Topic Evolution Network*.

Our framework performs document clustering at individual *time slices* with comparison between subsequent *time slices* for modeling topic *evolution*. By representing each document cluster as a topic of email conversation, the unique ideas of *evolution*, *creation*, *destruction*, *spilt*, and *merge* are presented to construct the *Topic Evolution Network*. Changes to the *Topic Evolution Network* are then mined and used to predict external events that might have happened during the time period the dataset captures.

We constructed the *Topic Evolution Network* for a period of 15 months and used it to form a model that predicts events. We found that our framework predicted events over these 15 months with an overall accuracy of 0.769.

TABLE OF CONTENTS


ABSTRACT .......... iii

LIST OF FIGURES AND TABLES .......... vi

CHAPTER 1 INTRODUCTION .......... 1

- Text Mining .......... 1
- Document Clustering .......... 2
- Time Decomposition .......... 2
- The Enron Dataset .......... 2
- Research Overview .......... 4

CHAPTER 2 RELATED WORK .......... 5

- Social Network Analysis .......... 5
- Graph Theory .......... 7
- Time Decomposition .......... 7
- Email Classification .......... 8

CHAPTER 3 RESEARCH APPROACH .......... 10

- Preliminaries .......... 10
- Existing Techniques .......... 11
  - Data Preprocessing .......... 11
  - Vector Space Model .......... 12
  - Feature Selection .......... 15
  - Document Clustering .......... 15
- Unique Contributions .......... 18
  - Topic Evolution Network .......... 18
  - Mining Events from the Topic Evolution Network .......... 20
    - Data Normalization .......... 21
    - Distance Measurements and Distance Threshold .......... 22
- Importance Level of Events .......... 23
    - Level 1 Importance .......... 24
    - Level 2 Importance .......... 24
    - Level 3 Importance .......... 25

CHAPTER 4 EXPERIMENTS AND RESULTS .......... 26

- Closed Universe Emails .......... 26
- Preliminary Results .......... 26
- Complete Framework Results .......... 32
  - Chosen Time Period .......... 32
  - Important Events .......... 32
  - Performance Measurements .......... 35
  - Raw Results for Chosen Time Period .......... 36
  - Normalized Results for Chosen Time Period .......... 38
  - Distance Measurements .......... 41
  - Events Predicted .......... 44

Performance Measurement Results ........ 46
Comparing Two Sets of Time Slices ........ 48
CHAPTER 5 CONCLUSION ........ 53
Applicability of the Framework ........ 54
Issues ........ 55
Future Work ........ 56
REFERENCES ........ 58

## LIST OF FIGURES AND TABLES


Figure 1: Temporal considerations visually depicted as a Topic Evolution Network...... 19

Figure 2: The Topic Evolution Network for five time slices, demonstrating the Network's concepts.......... 29

Figure 3: Topic Evolution Network for time slices $T_{47}$ through $T_{53}$.......... 49

Figure 4: Topic Evolution Network for time slices $T_{25}$ through $T_{28}$.......... 51

Table 1: Top 10 words per cluster for first five weeks of 2002.......... 28

Table 2: Topic Evolution Network measurements for first five weeks of 2002.......... 31

Table 3: Events designated Level 1 importance. .......... 33

Table 4: Events designated Level 2 importance. .......... 34

Table 5: Events designated Level 3 importance. .......... 35

Table 6: Topic Evolution Network counts of creation, evolution, destruction, split, and merge for time period January 1, 2001 through April 1, 2002. .......... 38

Table 7: Topic Evolution Network counts of creation, evolution, destruction, split, and merge for time period January 1, 2001 through April 1, 2002 after Min-Max normalization to [0,1].......... 41

Table 8: 1-norm and 2-norm distances for the Topic Evolution Network for time period January 1, 2001 through April 1, 2002. .......... 43

Table 9: 1-norm distances and their associated weeks where events were defined, noting the level of correctly predicted event, incorrect prediction, or the level of event that was not predicted. .......... 46

Table 10: Performance measurements for the entire time period.......... 46

# CHAPTER 1

# INTRODUCTION

This thesis presents a framework consisting of document clustering combined with time decomposition to mine the evolution of email discussion topics in a large email dataset. We contribute our Topic Evolution Network as well as a unique way to analyze this network for discovery of knowledge from the email dataset. In this chapter, we introduce the domains that our research intersects and motivate the development of our framework.

## Text Mining

Text mining is a data mining task that strives to find useful knowledge from largely unstructured text documents. The Butler Group estimates that 85% of all business data is stored in an unstructured format (White 2005). Such unstructured data makes logic-based programming ineffective for capturing fuzzy and ambiguous relations in text. Text mining attempts to discover hidden information in a text that can simultaneously handle a large number of words and natural language structures, combined with vagueness and uncertainty (Hotho, Nürnberger, and Paaß 2005).

Moreover, print documents may become a rarity in the future. Periodicals and books will be reserved for special purposes, forcing all other documents to be published solely in electronic format. The sheer number of text documents available in electronic format will render human processing impossible. Understanding the information contained in these documents will be vital to capturing the knowledge within each text, a task that text mining strives to achieve (Fan, et al. 2006).

## Document Clustering

Document clustering is a form of text mining suitable for a database of documents with no known structure. This technique can be used to find a concealed structure in the document set with no prior knowledge. Documents initially start as one large group, and then are split into smaller groupings called clusters based on a similarity measure computed between each cluster and document. Ideally, documents in each cluster will be similar because the same words are used frequently. Studying these frequently occurring words can give meaning to the clusters, thus creating a structure based on similar content for an unstructured document database (Weiss, et al. 2005).

## Time Decomposition

Time decomposition refers to partitioning a time period associated with a document set into one or more intervals. Each document is assigned to exactly one partition based on its time stamp that indicates the time of creation or publication. Information can then be mined by examining each partition individually, with comparisons made to other partitions or to the entire time period as a whole (Chundi, Zhang, and Castellanos 2006).

## The Enron Dataset

During its legal investigation in May 2002, the Federal Energy Regulatory Commission (FERC) released the Enron email corpus to the public (FERC 2005). This raw dataset contains 619,446 messages involving 158 users, representing 92% of the Enron staff's emails. Each email in the raw dataset contains information related to the date and time sent, sender, receiver(s), subject, body, as well as the folder the message was saved in. Although this initially appeared to be a boon for researchers, Leslie

Kaelbing from the Massachusetts Institute of Technology purchased the dataset and found it to have a number of integrity issues. Later, Melinda Gervasio from the Space Robotics Institute at Carnegie Mellon University (CMU) collected and prepared the dataset for the CALO (A Cognitive Assistant that Learns and Organizes) project, removing most of its integrity issues. Finally, William Cohen of CMU posted the dataset on the web for researchers to study in March 2004.

While the majority of emails and the associated information are included in this dataset, some had to be removed. All email attachments have been stripped out as well as certain messages. This dataset contains not only official Enron related email; it also contains personal email of the employees. This caused a number of affected employees to request their correspondences not be considered for study and forced a redaction effort. Given these limitations, William Cohen's (2004) posting of the dataset consists of 517,431 emails from 151 users in approximately 3500 folders.

Jitesh Shetty and Jafar Abidi (2004) from the Information Science Institute continued further study of Cohen's dataset and found it to still have integrity issues. These included duplicate, junk, and blank emails. They found certain computer generated folders such as "discussion_threads" and "all documents" contained duplicate emails that were stored elsewhere in the dataset in a user created folder or inbox. Such duplicates were dropped. Messages with junk data that were a remnant of a past attachment were also dropped. Blank messages were removed as well. Any invalid email addresses found were changed to "no.address@enron.com". Certain messages whose recipients were listed as undisclosed were changed to the common format "undisclosed.recepient@enron.com". Any emails with transaction failure notifications were also dropped. Shetty's and Abidi's pruning led to a dataset consisting of 252,759 emails from 151 employees distributed in approximately 3000 user folders. This cleaned dataset is publicly available as a MySQL database, commonly known as the ISI version of the dataset (Shetty and Adidi 2004). All future references made in this thesis to the "Enron email dataset" or "Enron dataset" refer to the ISI version.

**Research Overview**

We present a framework to mine the evolution of topics from a large email corpus of unstructured text and apply it to the ISI version of the Enron email dataset. This is one of the few publicly available corpora of emails, and provides a truly unique body to study. Email provides a snapshot of internal affairs at a given time, something that a large collection of documents without time stamps lacks. The framework we present combines existing techniques in the literature with our novel Topic Evolution Network. We apply our Topic Evolution Network in a unique way to study the changes of topics over time to create a model that predicts the possible presence of an external event that occurred during the time the dataset captures.

Specifically, we present a framework that can be used to detect possible external events by studying changes to an email corpus over time. The rest of this thesis is divided as follows. Chapter 2 contains related work pertinent to this research. Chapter 3 consists of our approach to the framework. Chapter 4 details our experiments and results. Finally, Chapter 5 presents conclusions as well as future work.

# CHAPTER 2

# RELATED WORK

Because of the variety of information captured by the Enron dataset, widely differing domains of research have an interest in discovering what knowledge it contains. Specifically researchers in the areas of social network analysis (SNA), graph theory, time decomposition, and email classification have actively studied the dataset. In this chapter, we present work similar to ours and other work that has been performed on the Enron dataset.

## Social Network Analysis

In addition to releasing a MySQL database containing a scrubbed version of the dataset, Shetty and Abidi (2004) also performed a preliminary study from a social network analysis perspective. A brief quantitative analysis was performed indicating there were more emails exchanged during 2001, the year of the California Energy Crisis. In addition, they derived a bi-directional social network from the dataset. This work proves the dataset's worthiness for further and more in-depth study. Research was continued by Diesner and Carley (2005) in communication networks. Like Shetty and Adidi (2004), their focus was on solely the event when an email was exchanged and who was involved; no consideration was given to the content of the email message. Their study indicates that the network became denser and communication networks more centralized in October 2001 than it had been in October 2000. Both show the dynamic nature of this dataset and that significant changes can be seen during real-world events that happened to the Corporation.

Such dynamics lead to changes in the informal hierarchy of an organization (Frantz and Carley 2006) as well as "who knows what" (Pathak, et al. 2006). In Carley and Frantz (2005), the social network of Enron was studied week by week, with attention given to three distinct time periods: a new CEO was announced, the resignation of the new CEO months later, and bankruptcy was announced coupled with massive layoffs. Their work shows the organization became more hierarchical when the new CEO was appointed, less hierarchical when he resigned, and slightly more hierarchical during the bankruptcy proceedings and layoffs. Hierarchy is measured using Krackhardt's Graph Hierarchy measurement, which indicates the reachability of nodes with regard to the direction of connection. Given such tremendous changes existed within this company, Pathak, et al. (2006) performed sociocentric analysis of the "who knows what" in the company, building a knowledge network. This work specifically studied if people believe "the company image is good/remains good." Utilizing the Dempsher-Shafey theory to model individuals' perceptions about knowledge, they found that overall employees agreed with this statement during the year 2000. In 2001, there was a marked decline in agreement and led to a distinct lack of consensus. Like Shetty and Adidi (2004) and Diesner and Carley (2005), these two works indicate a significant shift in the organization when Enron began its downfall.

SNA is combined with natural language processing (NLP) by McCallum, Corrada-Emmanuel, and Wang (2005) as well as by Zhang, et al. (2006). The Author-Recipient-Topic (ART) model discovers topic distribution through the direction-sensitive messages, utilizing a static Bayesian network (McCallum, Corrada-Emmanuel, and Wang 2005). Manual interpretation of the largest conditional probabilities for the presented topics indicate that the ART model does in fact discover unique topics, grouping Enron employees with similar positions or interests together as appropriate. Zhang, et al. (2006) combined probabilistic latent semantic analysis and the influence model into a framework that models pairwise interactions using the semantic content of the emails. Their model shows strong pairwise interaction between employees who likely worked together, such as two individuals who both share "Regulatory Affairs" in their job titles. Combining SNA with NLP shows promise as a way to discover interesting information about the Enron dataset.

**Graph Theory**

Social network analysis itself utilizes graph theory for much of its research since the network built can be modeled as a graph. Chapanond and Yener (2005) constructed an adjacency matrix and modeled it as a graph, capturing the vertices as employees and the edges as undirected email communication. This work studied a survey of graph metrics and performs graph clustering. It was found that their graph obeys the power law for the log of the degree of distribution versus the log of vertices. Also, there is a giant component that contains 62% of the vertices, meaning that this percentage of the vertices in the graph are connected, either directly or indirectly. Graph theory is an intrinsic component of much of the existing Enron research, including time decomposition.

**Time Decomposition**

O'Madadhain, Hutchins, and Smyth (2005) combined SNA with time decomposition to produce a time-varying EventRank. This rank measures the importance of an individual based on the number of incoming emails, here applied at a certain time slice. Utilizing the Enron dataset, they showed results of how rank changes for five individuals over 90 weeks. Credibility of the results is proven by describing how each individual's job role changed over time, information learned only via manual inspection. Chundi, Zhang, and Castellanos (2006) applied entropy-based functions to time stamped data, with results partially supported by the first weeks of April 2002 from the Enron dataset. One such function is the interval entropy measure function that discovers the frequently occurring words in a time interval. The other function is the keyword entropy measure function that finds terms that occur infrequently over time. Using only one week's worth of data decomposed by day, their results found the frequent keywords per day but on a dataset that only contains 146 emails. Nevertheless, their work does utilize entropy measures in a novel way to gain information from time stamped data such as the Enron dataset.

**Email Classification**

Preliminary work using the Enron dataset for email classification was done by Klimt and Yang (2004). Here folder classification is explored, finding that most users do employ folders to classify their emails, a critical requirement in this domain. They also found 62% of emails belong to a thread in the corpus, providing convincing evidence that this dataset is also suitable for thread classification. Further work by Berry and Browne (2005) uses non-negative matrix factorization to detect concepts in emails, with study done in the context of Enron business practices and activities. No fraudulent activity was found, but many concepts related to the business practices were seen as prominent in the dataset. Keila and Skillicorn (2005) classifies email slightly differently, by detecting structure in the dataset via singular value decomposition and semidiscrete decomposition.

They found that emails fall into two distinct categories: short messages using rare words and long messages using common words.

# CHAPTER 3

# RESEARCH APPROACH

Our framework consists of a number of individual steps that take an email dataset from the original raw, unstructured text into a model that can predict external events. This chapter details these steps, distinguishing our unique contributions from existing techniques.

**Preliminaries**

A document set **D** contains a finite set of $n$ documents such that $\mathbf{D} = \{d_1, d_2, ..., d_n\}$. Each document $d$ has a *time stamp* $t_d$ denoting its time of creation. In addition, each document also contains a set of words $\mathbf{W}_d$, treated here as a bag-of-words, that represent the contents of the document. Each $\mathbf{W}_d$ is a subset of **W**, all words found in the dataset **D**.

Time can be defined by its *unit of measure*, such as year, month, week, day, etc. A *time period* **T** captures the entire instance of time for a given time stamped dataset. **T** can be decomposed into *n time slices*, $T_1, T_2, \ldots T_n$, such that $T_{i+1}$ is always chronologically after $T_i$ and $1 \leq i < n$. Each time slice $T_i$ has a *time range* $R_i$ that defines the *start time* $t_s^i$ and *end time* $t_e^i$, inclusively, where $t_s^i < t_e^i$. Thus, all time stamped documents belong to one and only one time slice because each document must have a time stamp that is within a time range, $t_s^i \leq t_d \leq t_e^i$ and where $t_s^i > t_e^{i-1}$ and where $t_e^i < t_s^{i+1}$. All time stamped documents therefore belong to a subset of **D** based on the time slice $T_i$. All documents that have a time stamp within range $R_i$ belong to the subset $\mathbf{D}_{Ti}$, where $\mathbf{D} = \{ \mathbf{D}_{T1}, \mathbf{D}_{T2}, \ldots \mathbf{D}_{Tn}\}$.

**Existing Techniques**

A number of existing techniques are applied to the dataset to carry out this research. These include data preprocessing, converting the dataset into a vector space model, feature selection, and document clustering. Because **T** is decomposed into *n* time slices, these steps are performed *n* times, once for each time slice.

**Data Preprocessing**

In order to obtain all words $\mathbf{W}_{\mathrm{T}i}$ in the document set $\mathbf{D}_{\mathrm{T}i}$, a limited amount of preprocessing must be performed on $\mathbf{D}_{\mathrm{T}i}$. Each *d* in $\mathbf{D}_{\mathrm{T}i}$ is *tokenized* to extract individual words by converting each document into a stream of words, removing punctuation and replacing varying lengths of white space with a single white space. The tokenized representation of *d* is then used for further processing. All tokenized representations of each *d* are merged to create a single dictionary for the document collection. Let $\mathbf{W}_{\mathrm{T}i}$ represent the dictionary for time slice $\mathrm{T}_i$, where $\mathbf{W}_{\mathrm{T}i} = \{ w_1, w_2, \ldots, w_m\}$ contains *m* unique words occurring in $\mathbf{D}_{\mathrm{T}i}$.

This dictionary $\mathbf{W}_{\mathrm{T}i}$ is very large when all *m* unique words are included, thus the size of the dictionary is reduced to create more meaningful entries. One technique to decrease the size of the dictionary is *filtering*. One set of words that is filtered out are so-called stop words. These include articles, conjunctions, prepositions, etc, that yield no meaning when examined on their own. Another technique is *stemming*, which builds the basic forms of words. Porter (1980) introduced a widely used iterative stemming algorithm in 1980 for the English language. For example, Porter's algorithm stems plural forms of nouns into the root by removing the trailing "s" and stem verbs to their root by removing "-ing". With only the roots left intact, many words that have the same meaning but in different contexts are reduced to one single stem that itself conveys the meaning of the root (Hotho, Nürnberger, and Paaß 2005).

In addition to these standard preprocessing techniques, artifacts unique to the Enron dataset must be removed. A careful study of the message contents found that a large number of emails were either email forwards or email replies. The Corporation used Microsoft Outlook, which has a standard format of header information that is populated in the message body of both forwards and replies. These artifacts are also filtered out as they yield no meaning on their own. In addition, most employees sign their emails with either their first name or first and last name. Because this also does not result in meaningful information on its own, all employee names are filtered out of the dictionary as well. Manual inspection also yielded a number of meaningless words when taken individually, such as "thanks", "sincerely", etc. These are also removed where appropriate.

**Vector Space Model**

After preprocessing, another step to improve efficiency is employed. Utilizing the vector space model to represent documents as a compact vector of terms assists with efficient analysis of large document datasets. Each document $d$ is represented as a numerical feature vector in the $m$ dimensional space. Each element in the feature vector is a word $w$ of $\mathbf{W}_{\mathrm{T}i}$, with the size of the vector being equal to $|\mathbf{W}_{\mathrm{T}i}|$. Such a representation allows for the use of simple vector operations for document comparison (Han and Kamber 2006).

The values stored within each document vector represent either the presence of the word in $\mathbf{W}_{\mathrm{T}i}$ or the absence of the word in $\mathbf{W}_{\mathrm{T}i}$ in the specific document. For the simple case, a boolean value can represent this information. Only using a boolean loses information if the word is used more than once. Thus, another method is to capture the frequency of occurrences of each word $w$ in the document $d$, $freq(d,w)$. This method does not give importance to a word, or term $t$, based on the frequency it occurs throughout all of $\mathbf{D}_{Ti}$. Thus another method is defined, the $TF(d,t)$ measure. This is defined as 0 if the document does not contain the word and non-zero otherwise. The Cornell SMART system defines $TF(d,t)$ as:

$$TF(d,t) = \begin{cases} 0 \\ 1 + \log(1 + \log(freq(d,t))) \end{cases} \quad (1)$$

This TF measure accounts for documents that have a very high $freq(d,t)$ by adjusting the TF value as the frequency gets larger by the use of the logarithm. Simply because a document contains one word repeatedly does not necessarily mean the document should have a significantly higher frequency of a particular term. For example, a document might simply be a list of information that has the same word repeated many times as a heading for columns in a list; for example "supplier 1", "supplier 2", etc. A document with paragraphs of text involving supplier discussions will also include the word "supplier" but in a more meaningful usage. This discrepancy is accounted for by scaling the frequency down through the use of the logarithm. The 1 adder is included so that if the document $freq(d,t) = 1$, the TF measure does not become zero since the log(1) = 0.

In addition to term frequency, another measure called the inverse document frequency (IDF) measures the importance of a term $t$. This scales down the importance due to its reduced descriminative power if a term appears in many documents. The Cornell SMART system defines *IDF(t)* as:

$$IDF(t) = \log \frac{1 + |D_{T_i}|}{|d_t|} \tag{2}$$

where $d_t$ is the set of documents containing term $t$. The 1 adder is again included to prevent the value of the IDF from becoming 0 should there be only one document in the document subset $D_{T_i}$ and subsequently one document containing the term in $d_t$.

These two measures are combined in a complete vector-space model to form the *TF-IDF* measure for each term in each document:

$$TF - IDF(d,t) = TF(d,t) \times IDF(t) \tag{3}$$

The *TF-IDF* measure provides a suitable method to convert the simple frequency of a word occurring in a document to a weighted measure that realizes the importance of the term in the entire document dataset. By combining each document vector, the $|\mathbf{W}_{Ti}|$ X $|\mathbf{D}_{Ti}|$ document term matrix is then formed to represent the entire document set.

**Feature Selection**

Since $\mathbf{W}_{Ti}$ can be very large over a large document corpus, the dimensionality of the terms must be reduced to include terms of high quality. Each word $w$ in $\mathbf{W}_{Ti}$ is ranked by its relative importance using the following formula (Berry 2003):

$$q(w_j) = \sum_{i=1}^{|D|} w_{i,j}^2 - \frac{1}{|D|}\left[\sum_{i=1}^{|D|} w_{i,j}\right]^2 \tag{4}$$

where $w_{ij}$ is the *TF-IDF* measure for word *j* in document *i*. Note that the quantity $q(w_j)$ is proportional to the TF-IDF variances. This quality rating gives words that are not uniformly distributed in the document set higher quality (Tang and Vemuri 2005).

Tang and Vemuri (2005) use this method to choose the top 15% of words as features. In the Enron dataset, even the top 15% of words yields too many features. For example, during the week of October 22, 2001, the total possible features after all preprocessing was 4009. Selecting 15% of this would yield 601 features. This was simply too large a feature set to work with. Thus, only the top 5% of words are discovered using this formula, replacing $\mathbf{W}_{Ti}$ with $\mathbf{W}_{Ti}$', making $\mathbf{W}_{Ti}$' << $\mathbf{W}_{Ti}$. Each document *d* represented in vector format is itself reduced to only include the *TF-IDF* values for these top 5% of terms, greatly reducing dimensionality of the dataset.

**Document Clustering**

Given that each document *d* is now represented by a greatly reduced vector, document clustering is performed to group documents of similar content into clusters. Each cluster *c* contains a set of documents such that each document *d* is assigned to one and only one cluster. A clustering result is considered better when documents assigned to a cluster are more similar to those in the same cluster and more dissimilar to those assigned to other clusters. Clustering results are evaluated using the sum of squared error (SSE), chosen because the dimensions and cluster number are held constant (Hotho, Nürnberger, and Paaß 2005). The SSE is defined as

$$\sum_{a=1}^{c}\sum_{b=1}^{|D_{T_i}^a|} dist(d_b, centroid_a)^2 \tag{5}$$

where $c$ is the number of clusters, $|D_{T_i}^a|$ is the size of the document set assigned to cluster $a$ at time slice $T_i$, $d_b$ is the $b^{th}$ document in $D_{T_i}^a$, and dist is the distance calculated between the document and the centroid for cluster $a$. The clustering result that minimizes the sum of squared error is chosen to be the optimal result.

The $k$-means algorithm was chosen to perform clustering because of efficiency and quality of results (Steinbach, Karypis, and Kumar 2000). The algorithm attempts to place each document into a cluster that it is the nearest to, selecting the cluster that minimizes the distance formula that includes the cosine similarity measure

$$dist(d_i, d_j) = 1 - \cos(d_i, d_j) \tag{6}$$

where

$$\cos(d_i, d_j) = \frac{d_i \bullet d_j}{\|d_i\| \times \|d_j\|} \tag{7}$$

$d_i \bullet d_j$ indicates the vector dot product and $\|d\|$ is the length of the vector, $\sqrt{d \bullet d}$ (Tan, Steinbach, and Kumar 2006). Since the cosine measure presented here is a similarity measure, the value is subtracted from 1 so that it can be used as a distance measure instead.

*k*-means initially selects *k* random documents as initial centroids, designating one document per cluster. Each document is then examined and assigned to the cluster that minimizes the distance to the centroid. The centroid is recomputed after all documents have been assigned. This process repeats until there is no change in the centroid or until a maximum number of iterations is reached. Here, this maximum number of iterations is set to 25. From experimentation, most clusters converged well before 25. One special case that can arise from *k*-means clustering is when no documents are assigned to a cluster. To overcome this problem, the document that has the largest distance from the cluster's centroid is reassigned to the empty cluster and is set as the new centroid (Tan, Steinbach, and Kumar 2006). The entire *k*-means algorithm is run 10 times and the result that generates the smallest SSE is chosen as the optimal clustering result. Running the algorithm 10 times attempts to ensure that little bias is created by the randomly assigned initial centroids.

The number of clusters in our work is set to five at each individual time slice. Five was found to be a suitable number of clusters based on experimentation. Three was too few, causing documents that were not related to be clustered together. Ten was too many, causing documents that were related to be clustered into separate partitions. Five clusters captured unique discussion in most clustering results, as described in the Preliminary Results in Chapter 4.

**Unique Contributions**

Existing work on the Enron dataset has never considered document clustering techniques applied at different time intervals, here defined as time slices. Our research considers time to mine useful information from the email dataset. Specifically, five temporal consideration measures are computed for this purpose. Combined, these measures form a Topic Evolution Network.

**Topic Evolution Network**

Let each cluster $c_{T_i}^{j}$ represent the $j^{\text{th}}$ topic of discussion at an individual time slice $T_i$, where $1 \leq j \leq k$. A topic is said to have *evolved* from $T_{i-1}$ to $T_i$ if the distance measured between the centroid for $T_{i-1}$ and $T_i$ is less than the set threshold $\zeta_{\text{evolution}}$. Thus, the parent for cluster $c_{T_i}^{j}$ is $c_{T_{i-1}}^{j}$. A topic is said to have been *created* at $T_i$ if no parent exists from $T_{i-1}$. A topic is said to have been *destroyed* at $T_{i-1}$ if it has no children in $T_i$. A topic is said to have *split* in $T_{i-1}$ if $c_{T_{i-1}}^{j}$ has multiple children in $T_i$. A topic is said to have *merged* in $T_i$ if $c_{T_i}^{j}$ has multiple parents in $T_{i-1}$. A visual interpretation of these temporal consideration concepts, the Topic Evolution Network, is shown in Figure 1.

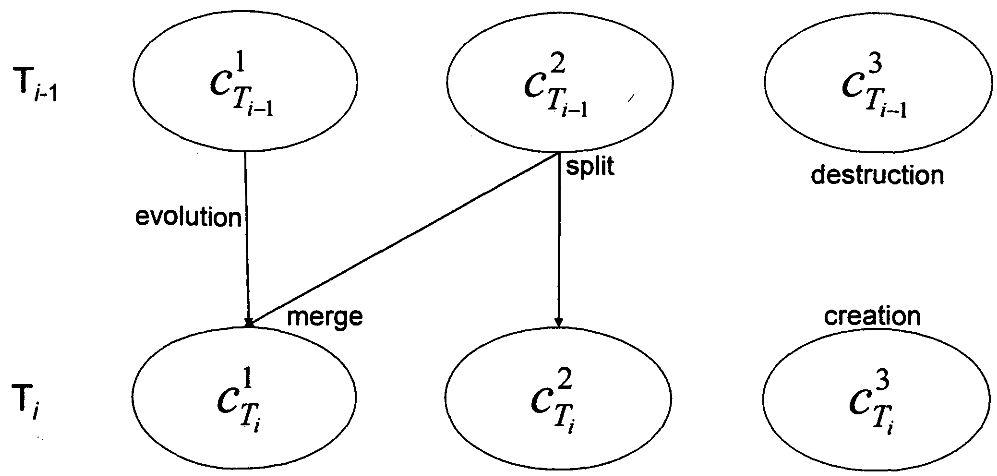


Figure 1: Temporal considerations visually depicted as a Topic Evolution Network.

These temporal considerations can be more clearly understood in terms of email content. An evolution occurs when a topic of conversation carries over from one time slice to the next. An example of this is the discussion of a working group in the human resources department. These individuals will likely discuss similar topics between each time slice, such as keeping track of employee's vacation requests. A destruction occurs when the topic ceases to be discussed. This might happen during the Christmas season when colleagues send greetings to one another prior to the holiday but not after. A creation occurs when a new topic that has not been previous discussed emerges. An emergency meeting may be necessary in human resources that was unknown in the previous time slice, hence employees may be discussing this meeting during the current time slice.

We define a split to cause multiple evolutions. As seen in Figure 1, cluster $c^2_{T_{i-1}}$ splits into $c^1_{T_i}$ and $c^2_{T_i}$ which causes two evolutions. In terms of emails, a split can be understood by considering the planning involved for a company's quarterly meeting. One group of people might first converse about finding a suitable day and considering who might be worthwhile speakers or required guests. This larger conversation will likely split into two groups as the event draws closer: one discussing meeting logistics such as room reservations and the other discussing the featured speaker and guests. Once all the details are nailed down, those involved in the meeting planning will likely converse again about the highlights of the upcoming meetings but not go into as much depth as was earlier required. We define this joining of topics as a merge. The merge is a result of multiple topics evolving into the same topic in the next time slice.

We selected the threshold $\zeta_{\text{evolution}}$ to infer an evolution occurred between time slice $T_{i-1}$ and $T_i$ to be 0.5. 0.5 was selected because it is the midpoint of the distance [0,1] when the difference between 1 and the cosine similarity measure is used as the distance metric. A distance of 0 means the two centroids being compared are the same and a distance of 1 means the two are completely different. Choosing the midpoint implies all documents contained in that cluster are more similar than dissimilar, hence we say an evolution has occurred.

**Mining Events from the Topic Evolution Network**

The creation of a Topic Evolution Network on its own simply details how topics change from one time slice to the next time slice. In order to mine useful information from the Network, the count of each value of evolution, creation, destruction, merge, and split is obtained for each time slice. The counts then are further processed. These further steps include data normalization and distance calculations, both leading to a heuristic for external event prediction.

### Data Normalization

At each time slice, the count of all values in the Topic Evolution network is obtained in the form of a vector. However, these values do not all have the same scale. The count of creation, destruction, merge, and split all represent a number of clusters from 0 to $k$, inclusively. The count of evolution represents the number of conversations that have continued into the next time slice. Graphically, the count of evolutions is the number of directed edges between time slice $T_{i-1}$ and $T_i$. When a topic splits, two or more clusters in the next time slice are continuations of the topic. When a topic merges, two or more topics in the previous time slice evolved into only one topic cluster. Thus the count of evolution can be from 0 to $k*k$, inclusively. Since these are two different scales of values, these must be normalized before any comparisons can be made.

Data is normalized to be on a scale of [0,1] using Min-Max Normalization. Min-Max Normalization transforms each data point $v$ that is currently within the range $[minA, maxA]$ into a new value $v'$ that is within the desired range $[new\_minA, new\_maxA]$ (Han and Kamber 2006). The new value $v'$ is defined as:

$$v' = \frac{v - minA}{maxA - minA}(new\_maxA - new\_minA) + new\_minA \qquad (8)$$

**Distance Measurements and Distance Threshold**

Once all values for evolution, creation, destruction, merge, and split have been normalized to be with the range [0,1], the distance is measured between two time slices. The distance is calculated between the current time slice $T_i$ and the previous time slice $T_{i-1}$ by assuming all values calculated per time slice in the Topic Evolution Network are represented in a vector. Since each value was previously normalized using Min-Max Normalization, all values have the same scale; this makes a vector representation of the time slice suitable for use. Two distance measurements are calculated: the $L_1$ norm and the $L_2$ norm. The $L_1$ norm, more colorfully known as the Manhattan distance, is defined as:

$$d(T_i, T_{i-1}) = \sum_{n=1}^{5} \left| T_{i_n} - T_{i-1_n} \right| \tag{9}$$

where $T_i$ and $T_{i-1}$ represent a vector of all five Topic Evolution Network considerations at time slice *i* and *i*-1.

The $L_2$ norm, also known as the Euclidian distance, is defined as:

$$d(T_i, T_{i-1}) = \sqrt{\sum_{n=1}^{5} (T_{i_n} - T_{i-1_n})^2} \tag{10}$$

Distance is a technique to measure dissimilarity between two objects, here defined in terms of the vector of normalized Topic Evolution Network counts. By computing the distance between the vectors at $T_i$ and $T_{i-1}$, the change in topic conversation can be found. If this change is above a threshold, then it can be said that an external event has possibly occurred in time slice $T_{i-1}$. The external event is said to possibly occur in time slice $T_{i-1}$ because the change is observed between the previous and the current time slice, implying conversation changed between these two time slices. The *event threshold* to predict possible events is defined as:

$$\zeta_{event} = \alpha \times \max Dist \tag{11}$$

where $\alpha$ is a set parameter and *maxDist* is the maximum possible distance between two time slices. Given that this distance utilizes normalized values of the five metrics in the Topic Evolution Network, then *maxDist* for $L_1$ norm is 5 and *maxDist* for $L_2$ norm is $\sqrt{5}$. If the distance between two time slices is less than the distance threshold, then we say that an event has possibly occurred during the $T_{i-1}$ time slice.

We set $\alpha$ to be 0.25. This then forces $\zeta_{event}$ to be 1.25 when $L_1$ norm distance is used and 0.55 when $L_2$ norm distance is used. When any the distance between time slice $T_i$ and $T_{i-1}$ is greater than $\zeta_{event}$, we predict a possible external event occurred during time slice $T_{i-1}$. This possible event predicted can be of any so-called importance level that we have defined in the following section.

**Importance Level of Events**

A number of events can happen at company that have significant differences in importance. For example, the importance of a CEO resigning amid allegations of improprieties is significantly more important than when executives divest their shares of stock. While said sell-offs likely contributed to the resignation of a CEO, they do not make the top news headlines of the day like resignations do. Because of these significant differences in importance of events, we defined three importance levels for event prediction.

### Level 1 Importance

A *Level 1* event is a very public event that happened to the company. One such Level 1 event is an executive change during a time of turmoil. This is defined during a time of turmoil because executive changes during such a time indicate things might be amiss at the highest levels of the company. An event that would not be considered Level 1 would be the promotion of an employee in a Vice President role to that of a C-level executive when this executive has retired. This would likely result in a smooth transition and one that was expected as the former executive was reaching retirement age. Another possible Level 1 event is a public announcement that affects the entire company and every employee, such as declaring bankruptcy. These major Level 1 events truly affect the entire company and each and every employee, thus only a small number of events are considered Level 1.

### Level 2 Importance

A *Level 2* event is a public event that would have received coverage from the business press. This is defined to include when political leaders have meetings with executives from the company. Another event is major changes related to company stock, such as declaring a large quarterly profit or loss. A third type of Level 2 event is when the company receives some sort of award, such as being a "great place to work". These are still very significant events, just not as monumental as a Level 1 event.

### Level 3 Importance

A *Level 3* event is one that might not have gotten as much press coverage. These include stock sales by employees of the company or by major stockholders. Such sales are recorded by the company, and only individuals with a keen interest will look up such sales. Thus these are considered more trivial than a Level 1 or Level 2 event, but still are events nonetheless.

# CHAPTER 4

# EXPERIMENTS AND RESULTS

We implemented the framework described in Chapter 3 and applied it to the Enron email dataset. This chapter details our experiments with the framework on the dataset. First, our initial results are presented that show the dataset is suitable for study over time. Following this is our detailed analysis of how well the framework performs at predicting events as well as analysis of the results.

## Closed Universe Emails

The unique corporate nature of this dataset provides challenges for data mining. Only 151 Enron employee email addresses are seen in the dataset, yet there are 252,759 total emails. Many of these emails have either senders or receivers outside of Enron, including spam and personal messages to family members, just to name a few categories. In order to mine information related to the company itself, only emails sent between Enron employees are considered. By reducing the dataset to only consider Enron-to-Enron emails in a closed universe, only 40,489 emails remain.

## Preliminary Results

Presented here are the initial results utilizing a subset of the framework described in Chapter 3. No work is presented beyond the construction of the Topic Evolution Network. This section motivates the use of the Topic Evolution Network for mining information from the email dataset.

**T** is defined to be a subset of the entire dataset, from 1 January 2002 through 2 February 2002 analyzed by week, with $n$=5. These dates capture the first five weeks of the year 2002. The top 10 words with the largest *TF-IDF* values are extracted from each cluster at each time slice and presented here. A graph is constructed to demonstrate how topics are evolved, created, destroyed, split, or merged. Table 1 displays the top 10 words per cluster for each $T_i$.

Stemming clearly has an effect on how words are represented. The original words in each document have been stemmed to create the root of the word, requiring some manual interpretation of the results. A trailing letter "y" is removed from all words and converted to "i", hence days of the week are displayed in a similar to fashion as Friday with "fridai". "ub" is the result of stemming performed on the company name UBS because the algorithm assumes "ubs" is plural and removes the trailing "s". If manual interpretation is used on the resulting top terms, trends can be seen in the results when combined with a graph depicting all temporal considerations.

Figure 2 is a graph representation of the five time slices presented here. Each directed edge indicates a topic evolution. Each edge has a weight representing the distance measure calculated between centroids from each time slice, $dist(c^j_{T_{i-1}}, c^j_{T_i})$.

| Time Slice | Start Date | Cluster 1 | Cluster 2 | Cluster 3 | Cluster 4 | Cluster 5 |
|---|---|---|---|---|---|---|
| $T_1$ | 2002-01-01 | meet | inform | book | power | file |
| | | week | time | matrix | agreement | curv |
| | | discuss | deliveri | group | contract | tomorrow |
| | | offic | number | question | execut | send |
| | | talk | plan | access | provid | chang |
| | | todai | check | add | trade | peopl |
| | | hall | deal | start | pool | chart |
| | | org | oper | daili | mail | problem |
| | | look | question | financi | transmiss | list |
| | | chart | purchas | avail | list | map |
| $T_2$ | 2002-01-08 | master | schedul | trader | energi | new |
| | | list | fridai | eol | review | file |
| | | tomorrow | attend | product | year | us |
| | | identifi | tw | book | contact | right |
| | | curv | meet | file | employe | daili |
| | | financi | continu | physic | trade | copi |
| | | agreement | nng | chang | counterparti | todai |
| | | problem | begin | todai | dai | place |
| | | includ | morn | matrix | deal | contract |
| | | mondai | work | us | provid | dai |
| $T_3$ | 2002-01-15 | fridai | book | issu | employe | contract |
| | | tw | deal | look | execut | discuss |
| | | time | list | curv | chang | meet |
| | | number | check | question | question | todai |
| | | work | question | file | individu | plan |
| | | ga | januari | contact | time | schedul |
| | | netco | start | chang | work | oper |
| | | market | daili | offic | inform | talk |
| | | daili | week | list | compani | peopl |
| | | list | ga | contract | contract | offic |
| $T_4$ | 2002-01-22 | meet | tw | contact | deal | issu |
| | | fridai | file | inform | book | group |
| | | morn | daili | week | list | help |
| | | discuss | januari | current | counterparti | week |
| | | schedul | time | power | new | busi |
| | | start | work | discuss | product | manag |
| | | review | discuss | ga | financi | talk |
| | | januari | question | work | todai | plan |
| | | offic | ga | send | trade | work |
| | | trade | look | person | question | contract |
| $T_5$ | 2002-01-29 | mail | posit | pl | ub | smith |
| | | copi | region | darron | mondai | doug |
| | | requir | close | fw | dai | gilbert |
| | | messag | offer | move | meet | ercot |
| | | secur | ub | gui | check | dai |
| | | transmiss | credit | book | schedul | sign |
| | | hart | forward | price | energi | addit |
| | | power | curv | deal | come | form |
| | | east | termin | new | time | ub |
| | | master | compani | ena | work | legal |

Table 1: Top 10 words per cluster for first five weeks of 2002.

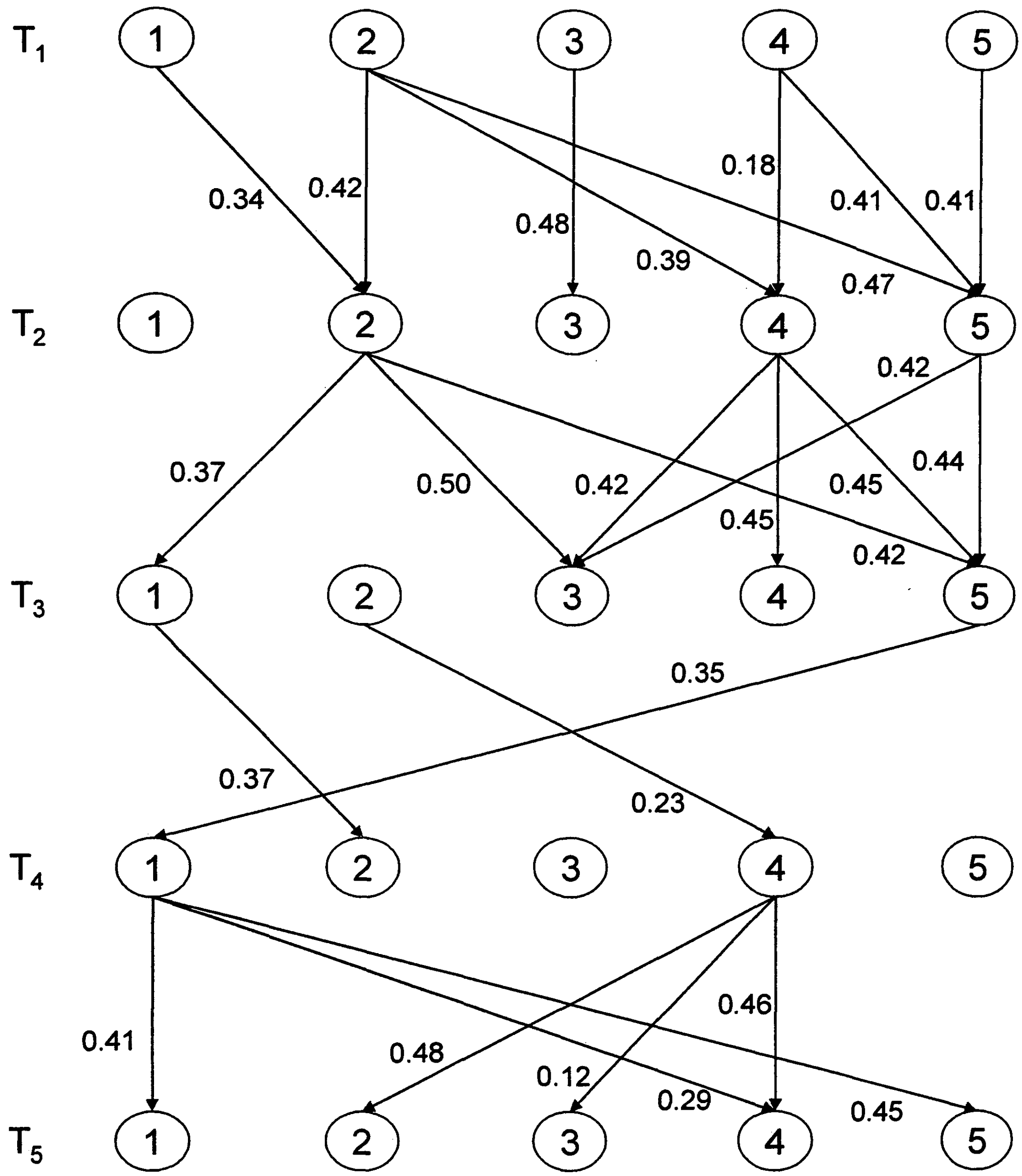


Figure 2: The Topic Evolution Network for five time slices, demonstrating the Network's concepts.

In Figure 2, only edges with weights less than the threshold $\zeta_{\text{evolution}}$, here 0.5, are presented since this represents a topic evolution. $c_{T_1}^1$ to $c_{T_2}^2$ is an example of an evolution. A topic in $T_{i-1}$ with more than one child in $T_i$ represents a split, as seen in $c_{T_1}^2$ because its children are $c_{T_2}^2$, $c_{T_2}^4$, and $c_{T_2}^5$. A topic in $T_i$ with more than one parent in $T_{i-1}$ represents a merge, as seen in $c_{T_2}^2$ because its parents are $c_{T_1}^1$ and $c_{T_1}^2$. A creation in $T_i$ is due to the fact that the topic has no parents in $T_{i-1}$. In $T_3$, $c_{T_3}^2$ demonstrates a creation. A destruction in $T_{i-1}$ is caused by the topic having no children in $T_i$. $c_{T_2}^3$ is a destruction.

Manually examining emails gives more meaning as well as credibility to these results. Starting with $c_{T_1}^1$, we see that it evolves into $c_{T_2}^2$. Most of the emails in $c_{T_1}^1$ discuss an upcoming meeting with Northern Natural Gas (NNG), including planning and logistics. $c_{T_2}^2$ has emails discussing Enron's Transwestern (TW) division as well as NNG, a likely conversation since TW has operations in California. These two acronyms are found to be two of the 10 most frequently occurring words in that topic cluster, giving strong evidence for the concept of evolution. $c_{T_2}^2$ is also a merge of $c_{T_1}^1$ and $c_{T_1}^2$. $c_{T_1}^2$ contains many emails relating to the delivery of power by Enron's Transwestern division, indicating a logical merger of topics from $T_1$. $c_{T_1}^2$ is a split in $T_2$ into $c_{T_2}^2$, $c_{T_2}^4$, *and* $c_{T_2}^5$. $c_{T_2}^4$ also contains emails relating to Transwestern as well as legal emails. $c_{T_2}^5$ contains details of operations in certain parts of California. Clustering emails and extracting the top words per cluster assists in understanding the content of most messages assigned to that cluster. While manual inspection of the emails does yield more meaningful results, the general topics can be gleaned from the top terms and seen continuing in later time slices.

$c_{T_3}^2$ is considered a creation and a destruction as well. Manual inspection of the emails yields conversation about a "master termination list" from January 8th. $T_2$ represents the week of January 8th. This list would only be applicable in $T_2$ and would not have been talked about in $T_1$, thus a creation is made. This list also would likely not be discussed in $T_3$ since it contains temporal information for $T_2$. Therefore, it is also a destruction since it is only applicable to $T_2$. Furthermore, the words "master" and "list" never appear together as top words in any other clusters during any time slice.

A summary of the count of temporal considerations is shown in Table 2.

| Week | Evolved | Created | Destroyed | Split | Merged |
|---|---|---|---|---|---|
| 1 | 8 | - | 0 | 2 | - |
| 2 | 8 | 1 | 2 | 3 | 3 |
| 3 | 3 | 1 | 2 | 0 | 2 |
| 4 | 6 | 2 | 3 | 2 | 0 |
| 5 | - | 0 | - | - | 1 |

Table 2: Topic Evolution Network measurements for first five weeks of 2002.

These results demonstrate the applicability of the dataset for study over time. Some conversations continue (evolve) over time, yet others do not (destruction). Some only start (creation) at a certain time because of factors that become present only at that time slice. Other topics split or merge because certain subjects become more broad or more precise, respectively. Given all these measures, useful information can be mined from such a time stamped dataset.

### Complete Framework Results

The results presented in this section are the final results after applying the entire framework presented in Chapter 3. This section begins by introducing the time period studied for application of the framework. The following section details the events we attempt to predict at each importance level. Next performance measurements are discussed. Finally, the results of the framework as well as how well we predicted events are discussed.

#### Chosen Time Period

Even though the entire dataset encompasses May 1999 through June 2002, very few notable public events occurred early in the dataset. On January 20, 2001, then CEO Kenneth Lay and president Jeffrey Skilling attended the inauguration of George W. Bush (Washington Post 2004). While this not a major event, it indicates that events external to the company began to be publicly noticed. Since this occurred in the month of January, the starting date for studied emails was chosen to be January 1, 2001. Very few emails exist in the last two and a half months of data. Only 48 closed universe emails are recorded from April 2002 through June 2002, yielding an average of less than one email per day. Since having such sparse data will cause many empty clusters per time slice, thus these emails will not be considered. These restrictions for the start date of January 1, 2001, and the end date of April 1, 2002, result in 32,114 emails belonging to the time period for which results are presented.

#### Important Events

Tables 3, 4, and 5 detail the events we attempted to predict, separated by importance levels. The leftmost column in each table is the date the event occurred. The middle column is the time slice, or week, the event occurred in our subset of the data. That is, week 1 begins on January 1, 2001 and the final week 65 begins on March 25, 2002. The rightmost column is the event (Washington Post 2004; Wikipedia 2007).

| Date | Week | Event |
|---|---|---|
| 08/14/2001 | 33 | Kenneth Lay takes over as CEO after Jeffrey Skilling resigns for personal reasons. |
| 12/01/2001 | 49 | Enron files for Chapter 11 bankruptcy protection |
| 01/23/2002 | 56 | Lay resigns as Enron's CEO |

Table 3: Events designated Level 1 importance.

It is critical to note that multiple events sometimes happen within one time slice. The framework only predicts the presence of an external event during the particular time slice, here defined in terms of weeks. Working with this limitation, this table summarizes the count of the number of time slices (weeks) for which we attempt to predict events.

| Type | Weeks |
|---|---|
| Level 1 | 3 |
| Level 2 | 20 |
| Level 3 | 3 |
| All Levels | 25 |

There is an overlapping event for week 30 in both Level 2 and Level 3. Thus, the total count for all levels is 25 rather than 26.

| Date | Week | Event |
|---|---|---|
| 01/20/2001 | 3 | Kenneth Lay and Jeffrey Skilling attend the inauguration of George W. Bush. |
| 02/06/2001 | 6 | Enron is named "most innovative company in America" by Fortune magazine. |
| 02/12/2001 | 7 | Skilling is named CEO. |
| 02/22/2001 | 8 | Lay and other officials meet with the Dick Cheney energy task force. |
| 03/05/2001 | 10 | Fortune Magazine prints article "Is Enron Overpriced?" |
| 03/07/2001 | 10 | Lay and other officials meet with the Dick Cheney energy task force. |
| 04/17/2001 | 16 | Enron announces a first quarter profit of $536 million |
| 05/05/2001 | 18 | Enron's stock price closes below $59.78, a critical level for a partnership. |
| 05/17/2001 | 20 | The energy task force releases its report, endorsing some of Enron's proposals. |
| 07/23/2001 | 30 | Enron's stock price closes below $47, a critical level for a partnership. |
| 08/21/2001 | 34 | Lay emails Enron employees, claiming one of his highest priorities is to restore investor confidence. |
| 10/10/2001 | 41 | Enron officials discuss energy policy with Cheney. |
| 10/16/2001 | 42 | Enron announces a third quarter loss of $618 million. |
| 10/22/2001 | 43 | Enron announces the U.S. Securities and Exchange commission has begun an inquiry into its accounting practices. |
| 10/24/2001 | 43 | CFO Andrew Fastow is forced to leave Enron. |
| 10/26/2001 | 43 | Lay makes a phone call to chairman of the Federal Reserve Alan Greenspan about Enron. |
| 10/28/2001 | 43 | Lay talks to Paul H. O'Neill, Secretary of the Treasury. |
| 10/29/2001 | 43 | Lay talks to Donald L. Evans, Secretary of Commerce. |
| 10/31/2001 | 44 | Enron announces the SEC inquiry is now a formal investigation. |
| 11/08/2001 | 45 | Enron announces it overstated profits by $586 million over five years. |
| 11/09/2001 | 45 | Dynegy announces it will acquire Enron for $9 billion. |
| 11/28/2001 | 48 | Dynegy retracts its offer. |
| 12/25/2001 | 52 | Christmas / New Years |
| 01/09/2002 | 54 | The U.S. Justice Department confirms that it has begun a criminal investigation into Enron's bankruptcy. |
| 02/27/2002 | 61 | Former Enron employees offer to testify in exchange for immunity from prosecution. |
| 03/01/2002 | 61 | New documents collected by House committee conflict sworn testimony given by Skilling. |
| 03/07/2002 | 62 | President Bush reveals detailed proposal for corporate executives to personally vouch for their companies' financial statements. |

Table 4: Events designated Level 2 importance.

| Date | Week | Event |
|---|---|---|
| 06/05/2001 | 23 | President Bush's senior adviser Karl Rove divested his shares. |
| 06/06/2001 | 23 | Enron general counsel Jim Derrick sells 160,000 Enron shares. |
| 07/13/2001 | 28 | Chief executive of Enron Broadband Services Ken Rice sells 386,000 shares. |
| 07/27/2001 | 30 | Director Robert Belfer sells 100,000 shares. |
| 09/17/2001 | 37 | Jeffrey Skilling sells 500,000 shares. |
| 09/21/2001 | 38 | Robert Belfer sells 109,000 shares. |

Table 5: Events designated Level 3 importance.

### Performance Measurements

In order to assess the performance of the Topic Evolution Network on predicting external events, three performance measurements were calculated: sensitivity, specificity, and accuracy. Sensitivity defines how well the model correctly identifies events, whereas specificity defines how well the model correctly identifies when no event has occurred. Accuracy measures the quality of both the presence and absence of an actual event over the entire predicted group. A *true positive* (TP) corresponds to an actual event present in the timeline that is correctly predicted by the model. A *false negative* (FN) is an actual event in the timeline that is predicted as not present. A *false positive* (FP) is an event that is not present in the timeline but is predicted as such. A *true negative* (TN) is when no event is present in the timeline and this lack of event is predicted by the model. Sensitivity, specificity, and accuracy can be seen in the equations 12, 13, and 14 (Tan, Steinbach, and Kumar 2006).

$$sensitivity = \frac{TP}{TP + FN} \tag{12}$$

$$specificity = \frac{TN}{TN + FP} \tag{13}$$

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{14}$$

Sensitivity is the only measure that can be computed for the results overall and per level. This is not applicable per level because the concept of true negative and false positive do not include the level. Instead, both TN and FP only represent the absence or presence, respectively, of an event at a time slice. Thus, sensitivity is used as a measurement per level and for the overall results. Specificity and accuracy are used for entire dataset.

**Raw Results for Chosen Time Period**

The Topic Evolution Network was constructed for the 65 weeks comprising January 1, 2001, through April 1, 2002. The raw counts of each concept in the Topic Evolution Network, creation, evolution, destruction, split, and merge, are shown in Table 6.

| Week | Evolution | Destruction | Split | Creation | Merge |
|---|---|---|---|---|---|
| 1 | 6 | 1 | 2 | - | - |
| 2 | 7 | 1 | 1 | 1 | 1 |
| 3 | 4 | 1 | 0 | 0 | 2 |
| 4 | 6 | 2 | 1 | 3 | 1 |
| 5 | 5 | 2 | 1 | 0 | 1 |
| 6 | 7 | 1 | 2 | 1 | 1 |
| 7 | 7 | 1 | 2 | 1 | 2 |
| 8 | 7 | 1 | 3 | 1 | 2 |

Table 6: Topic Evolution Network counts of creation, evolution, destruction, split, and merge for time period January 1, 2001 through April 1, 2002

| Week | Evolution | Destruction | Split | Creation | Merge |
|---|---|---|---|---|---|
| 9 | 7 | 0 | 2 | 1 | 2 |
| 10 | 7 | 2 | 2 | 2 | 2 |
| 11 | 8 | 0 | 3 | 0 | 2 |
| 12 | 5 | 1 | 1 | 2 | 3 |
| 13 | 6 | 1 | 1 | 1 | 1 |
| 14 | 7 | 1 | 1 | 1 | 1 |
| 15 | 6 | 1 | 2 | 0 | 1 |
| 16 | 9 | 1 | 3 | 2 | 2 |
| 17 | 8 | 0 | 3 | 0 | 3 |
| 18 | 2 | 4 | 1 | 1 | 2 |
| 19 | 6 | 0 | 1 | 3 | 0 |
| 20 | 5 | 1 | 1 | 1 | 2 |
| 21 | 4 | 0 | 1 | 1 | 1 |
| 22 | 7 | 0 | 2 | 0 | 1 |
| 23 | 3 | 2 | 0 | 1 | 2 |
| 24 | 5 | 2 | 1 | 3 | 1 |
| 25 | 7 | 0 | 1 | 1 | 1 |
| 26 | 7 | 1 | 2 | 1 | 1 |
| 27 | 7 | 1 | 2 | 1 | 2 |
| 28 | 11 | 1 | 3 | 1 | 2 |
| 29 | 9 | 0 | 2 | 0 | 4 |
| 30 | 7 | 1 | 3 | 0 | 3 |
| 31 | 8 | 0 | 2 | 0 | 2 |
| 32 | 6 | 1 | 2 | 1 | 2 |
| 33 | 9 | 2 | 3 | 1 | 2 |
| 34 | 7 | 1 | 2 | 0 | 4 |
| 35 | 8 | 1 | 2 | 1 | 2 |
| 36 | 12 | 1 | 4 | 0 | 2 |
| 37 | 10 | 0 | 3 | 0 | 4 |
| 38 | 13 | 0 | 4 | 0 | 3 |
| 39 | 8 | 1 | 2 | 0 | 5 |
| 40 | 11 | 0 | 4 | 0 | 2 |
| 41 | 5 | 1 | 1 | 1 | 4 |

Table 6: Topic Evolution Network counts of creation, evolution, destruction, split, and merge for time period January 1, 2001 through April 1, 2002

| Week | Evolution | Destruction | Split | Creation | Merge |
|---|---|---|---|---|---|
| 42 | 5 | 2 | 2 | 1 | 1 |
| 43 | 8 | 0 | 2 | 0 | 0 |
| 44 | 8 | 0 | 3 | 0 | 3 |
| 45 | 8 | 2 | 3 | 1 | 2 |
| 46 | 5 | 1 | 1 | 1 | 4 |
| 47 | 6 | 0 | 1 | 1 | 1 |
| 48 | 12 | 0 | 1 | 1 | 1 |
| 49 | 4 | 1 | 0 | 1 | 2 |
| 50 | 6 | 2 | 3 | 2 | 1 |
| 51 | 4 | 1 | 0 | 1 | 2 |
| 52 | 8 | 1 | 3 | 2 | 1 |
| 53 | 9 | 0 | 2 | 0 | 2 |
| 54 | 7 | 1 | 2 | 0 | 3 |
| 55 | 7 | 0 | 2 | 0 | 1 |
| 56 | 7 | 1 | 2 | 1 | 2 |
| 57 | 5 | 1 | 1 | 0 | 2 |
| 58 | 3 | 2 | 0 | 2 | 1 |
| 59 | 6 | 0 | 1 | 2 | 0 |
| 60 | 6 | 1 | 2 | 1 | 2 |
| 61 | 8 | 0 | 3 | 1 | 2 |
| 62 | 5 | 0 | 0 | 0 | 2 |
| 63 | 4 | 2 | 1 | 2 | 2 |
| 64 | 4 | 2 | 0 | 1 | 0 |
| 65 | 0 | 5 | 0 | 2 | 0 |

Table 6: Topic Evolution Network counts of creation, evolution, destruction, split, and merge for time period January 1, 2001 through April 1, 2002.

### Normalized Results for Chosen Time Period

By studying a distance measure between each week, external events can be predicted. Using the raw counts from the Topic Evolution Network, however, will not yield a meaningful result. These raw counts are dependent on the number of clusters chosen and also do not have the same range of values. To overcome these issues, the raw counts are normalized with Min-Max normalization to be within the range [0,1]. Table 7 details the results of Table 6 after Min-Max normalization, rounding values to three decimal places where appropriate.

| Week | Evolution | Destruction | Split | Creation | Merge |
|---|---|---|---|---|---|
| 1 | 0.462 | 0.2 | 0.5 | 0 | 0 |
| 2 | 0.538 | 0.2 | 0.25 | 0.333 | 0.2 |
| 3 | 0.308 | 0.2 | 0 | 0 | 0.4 |
| 4 | 0.462 | 0.4 | 0.25 | 1 | 0.2 |
| 5 | 0.385 | 0.4 | 0.25 | 0 | 0.2 |
| 6 | 0.538 | 0.2 | 0.5 | 0.333 | 0.2 |
| 7 | 0.538 | 0.2 | 0.5 | 0.333 | 0.4 |
| 8 | 0.535 | 0.2 | 0.75 | 0.333 | 0.4 |
| 9 | 0.535 | 0 | 0.5 | 0.333 | 0.4 |
| 10 | 0.535 | 0.4 | 0.5 | 0.667 | 0.4 |
| 11 | 0.615 | 0 | 0.75 | 0 | 0.4 |
| 12 | 0.385 | 0.2 | 0.25 | 0.667 | 0.6 |
| 13 | 0.462 | 0.2 | 0.25 | 0.333 | 0.2 |
| 14 | 0.538 | 0.2 | 0.25 | 0.333 | 0.2 |
| 15 | 0.462 | 0.2 | 0.5 | 0 | 0.2 |
| 16 | 0.692 | 0.2 | 0.75 | 0.667 | 0.4 |
| 17 | 0.616 | 0 | 0.75 | 0 | 0.6 |
| 18 | 0.154 | 0.8 | 0.25 | 0.333 | 0.4 |
| 19 | 0.462 | 0 | 0.25 | 1 | 0 |
| 20 | 0.385 | 0.2 | 0.25 | 0.333 | 0.4 |
| 21 | 0.308 | 0 | 0.25 | 0.333 | 0.2 |
| 22 | 0.538 | 0 | 0.5 | 0 | 0.2 |
| 23 | 0.231 | 0.4 | 0 | 0.333 | 0.4 |
| 24 | 0.385 | 0.4 | 0.25 | 1 | 0.2 |
| 25 | 0.538 | 0 | 0.25 | 0.333 | 0.2 |

Table 7: Topic Evolution Network counts of creation, evolution, destruction, split, and merge for time period January 1, 2001 through April 1, 2002 after Min-Max normalization to [0,1].

| Week | Evolution | Destruction | Split | Creation | Merge |
|---|---|---|---|---|---|
| 26 | 0.538 | 0.2 | 0.5 | 0.333 | 0.2 |
| 27 | 0.538 | 0.2 | 0.5 | 0.333 | 0.4 |
| 28 | 0.846 | 0.2 | 0.75 | 0.333 | 0.4 |
| 29 | 0.692 | 0 | 0.5 | 0 | 0.8 |
| 30 | 0.538 | 0.2 | 0.75 | 0 | 0.6 |
| 31 | 0.615 | 0 | 0.5 | 0 | 0.4 |
| 32 | 0.462 | 0.2 | 0.5 | 0.333 | 0.4 |
| 33 | 0.692 | 0.4 | 0.75 | 0.333 | 0.4 |
| 34 | 0.538 | 0.2 | 0.5 | 0 | 0.8 |
| 35 | 0.615 | 0.2 | 0.5 | 0.333 | 0.4 |
| 36 | 0.923 | 0.2 | 1 | 0 | 0.4 |
| 37 | 0.769 | 0 | 0.75 | 0 | 0.8 |
| 38 | 1 | 0 | 1 | 0 | 0.6 |
| 39 | 0.615 | 0.2 | 0.5 | 0 | 1 |
| 40 | 0.846 | 0 | 1 | 0 | 0.4 |
| 41 | 0.385 | 0.2 | 0.25 | 0.333 | 0.8 |
| 42 | 0.385 | 0.4 | 0.5 | 0.333 | 0.2 |
| 43 | 0.615 | 0 | 0.5 | 0 | 0 |
| 44 | 0.615 | 0 | 0.75 | 0 | 0.6 |
| 45 | 0.615 | 0.4 | 0.75 | 0.333 | 0.4 |
| 46 | 0.385 | 0.2 | 0.25 | 0.333 | 0.8 |
| 47 | 0.462 | 0 | 0.25 | 0.333 | 0.2 |
| 48 | 0.923 | 0 | 0.25 | 0.333 | 0.2 |
| 49 | 0.3078 | 0.2 | 0 | 0.333 | 0.4 |
| 50 | 0.462 | 0.4 | 0.75 | 0.667 | 0.2 |
| 51 | 0.308 | 0.2 | 0 | 0.333 | 0.4 |
| 52 | 0.615 | 0.2 | 0.75 | 0.667 | 0.2 |
| 53 | 0.692 | 0 | 0.5 | 0 | 0.4 |
| 54 | 0.538 | 0.2 | 0.5 | 0 | 0.6 |
| 55 | 0.538 | 0 | 0.5 | 0 | 0.2 |
| 56 | 0.538 | 0.2 | 0.5 | 0.333 | 0.4 |

Table 7: Topic Evolution Network counts of creation, evolution, destruction, split, and merge for time period January 1, 2001 through April 1, 2002 after Min-Max normalization to [0,1].

| Week | Evolution | Destruction | Split | Creation | Merge |
|---|---|---|---|---|---|
| 57 | 0.385 | 0.2 | 0.25 | 0 | 0.4 |
| 58 | 0.231 | 0.4 | 0 | 0.667 | 0.2 |
| 59 | 0.462 | 0 | 0.25 | 0.667 | 0 |
| 60 | 0.462 | 0.2 | 0.5 | 0.333 | 0.4 |
| 61 | 0.615 | 0 | 0.75 | 0.333 | 0.4 |
| 62 | 0.385 | 0 | 0 | 0 | 0.4 |
| 63 | 0.308 | 0.4 | 0.25 | 0.667 | 0.4 |
| 64 | 0.3078 | 0.4 | 0 | 0.333 | 0 |
| 65 | 0 | 1 | 0 | 0.667 | 0 |

Table 7: Topic Evolution Network counts of creation, evolution, destruction, split, and merge for time period January 1, 2001 through April 1, 2002 after Min-Max normalization to [0,1].

### Distance Measurements

After Min-Max normalization was performed on all counts in the Topic Evolution Network, the distance between the current time slice (week) with the previous was calculated. Both Minkowski distances of order $L_1$ (1-norm) and $L_2$ (2-norm) are shown in Table 8, rounding values to three decimal places.

| Week | $L_1$ | $L_2$ |
|---|---|---|
| 1 | - | - |
| 2 | 0.861 | 0.469 |
| 3 | 1.014 | 0.517 |
| 4 | 1.804 | 1.080 |
| 5 | 1.077 | 1.003 |
| 6 | 0.937 | 0.487 |
| 7 | 0.2 | 0.2 |
| 8 | 0.25 | 0.25 |

Table 8: 1-norm and 2-norm distances for the Topic Evolution Network for time period January 1, 2001, through April 1, 2002.

| Week | $L_1$ | $L_2$ |
|---|---|---|
| 9 | 0.45 | 0.320 |
| 10 | 0.733 | 0.521 |
| 11 | 1.394 | 0.820 |
| 12 | 1.798 | 0.910 |
| 13 | 0.810 | 0.526 |
| 14 | 0.077 | 0.077 |
| 15 | 0.660 | 0.424 |
| 16 | 1.347 | 0.775 |
| 17 | 1.143 | 0.728 |
| 18 | 2.295 | 1.120 |
| 19 | 2.174 | 1.157 |
| 20 | 1.344 | 0.806 |
| 21 | 0.477 | 0.293 |
| 22 | 0.814 | 0.477 |
| 23 | 1.741 | 0.810 |
| 24 | 1.271 | 0.755 |
| 25 | 1.221 | 0.793 |
| 26 | 0.45 | 0.320 |
| 27 | 0.2 | 0.2 |
| 28 | 0.558 | 0.396 |
| 29 | 1.337 | 0.630 |
| 30 | 0.804 | 0.408 |
| 31 | 0.727 | 0.385 |
| 32 | 0.687 | 0.418 |
| 33 | 0.681 | 0.395 |
| 34 | 1.337 | 0.630 |
| 35 | 0.810 | 0.526 |
| 36 | 1.141 | 0.675 |
| 37 | 1.004 | 0.535 |
| 38 | 0.681 | 0.395 |
| 39 | 1.485 | 0.773 |
| 40 | 1.530 | 0.837 |

Table 8: 1-norm and 2-norm distances for the Topic Evolution Network for time period January 1, 2001, through April 1, 2002.

| Week | $L_1$ | $L_2$ |
|---|---|---|
| 41 | 2.145 | 1.042 |
| 42 | 1.05 | 0.680 |
| 43 | 1.164 | 0.604 |
| 44 | 0.85 | 0.65 |
| 45 | 0.933 | 0.558 |
| 46 | 1.331 | 0.709 |
| 47 | 0.877 | 0.637 |
| 48 | 0.462 | 0.462 |
| 49 | 1.265 | 0.723 |
| 50 | 1.637 | 0.882 |
| 51 | 1.637 | 0.882 |
| 52 | 1.591 | 0.899 |
| 53 | 1.394 | 0.770 |
| 54 | 0.554 | 0.322 |
| 55 | 0.6 | 0.447 |
| 56 | 0.733 | 0.437 |
| 57 | 0.737 | 0.444 |
| 58 | 1.471 | 0.781 |
| 59 | 1.081 | 0.562 |
| 60 | 1.183 | 0.611 |
| 61 | 0.604 | 0.355 |
| 62 | 1.314 | 0.853 |
| 63 | 1.394 | 0.820 |
| 64 | 0.983 | 0.578 |
| 65 | 1.241 | 0.752 |

Table 8: 1-norm and 2-norm distances for the Topic Evolution Network for time period January 1, 2001, through April 1, 2002.

The distance between each week and its immediate predecessor represents how much change there was in topic conversation between two weeks. Because all Topic Evolution Network measurements that contributed to the distance are inclusive between 0 and 1, the maximum distance for each of the five measures is 1. Thus the maximum 1-

norm distance possible is 5 and the maximum 2-norm distance is $\sqrt{5}$. Because the values for all Network measurements are less than 1, the 2-norm distance will be artificially small since each term is squared in this distance calculation. Thus, 1-norm distance is only considered for event prediction. As described in Chapter 3, $\zeta_{event}$ is 1.25. Thus all events with $L_1$ distances greater than 1.25 predict the possible presence of at least one event of any importance level in the previous time slice $T_{n-1}$.

**Events Predicted**

Table 9 includes all weeks designated with at least one Level 1, Level 2, or Level 3 importance event. When the $L_1$ distance is greater than 1.25, the TP (true positive) column captures the fact that the event was correctly predicted and notes the given importance level. It is important to note that this level is not predicted by the framework, but is manually included here for summary purposes. The FP (false positive) column indicates whether an event was predicted but one did not exist during that week. Since the model cannot predict levels, an "X" designates that this was a false positive hit. The FN (false negative) column operates in the same manner as the TP column. The number in the column represents that an event should have been predicted during that time slice and it associated level. Week 30 should have captured at least one event out of a possible two levels of events, and is marked as such by the entry "2,3".

| Week | $L_1$ | TP | FP | FN |
|---|---|---|---|---|
| 4 | 1.804 | 2 | | |
| 6 | 0.937 | | | 2 |
| 7 | 0.200 | | | 2 |
| 8 | 0.250 | | | 2 |
| 11 | 1.394 | 2 | | |
| 12 | 1.797 | | X | |
| 16 | 1.347 | | X | |
| 17 | 1.143 | | | 2 |
| 18 | 2.295 | | X | |
| 19 | 2.174 | 2 | | |
| 20 | 1.344 | | X | |
| 21 | 0.477 | | | 2 |
| 23 | 1.741 | | X | |
| 24 | 1.270 | 3 | | |
| 29 | 1.337 | 3 | | |
| 31 | 0.727 | | | 2,3 |
| 34 | 1.337 | 1 | | |
| 39 | 1.485 | 3 | | |
| 40 | 1.530 | | X | |
| 41 | 2.145 | 2 | | |
| 42 | 1.050 | | | 2 |
| 43 | 1.164 | | | 2 |
| 44 | 0.933 | | | 2 |
| 46 | 1.330 | 2 | | |
| 49 | 1.265 | 2 | | |
| 50 | 1.637 | 1 | | |
| 51 | 1.637 | | X | |
| 52 | 1.591 | | X | |
| 53 | 1.394 | 2 | | |
| 55 | 0.600 | | | 2 |

Table 9: 1-norm distances and their associated weeks where events were defined, noting the level of correctly predicted event, incorrect prediction, or the level of event that was not predicted.

| Week | $L_1$ | TP | FP | FN |
|---|---|---|---|---|
| 58 | 1.471 | 1 | | |
| 62 | 1.314 | 2 | | |
| 63 | 1.39 | 2 | | |

Table 9: 1-norm distances and their associated weeks where events were defined, noting the level of correctly predicted event, incorrect prediction, or the level of event that was not predicted.

**Performance Measurement Results**

As described in the Performance Measurements section, sensitivity, specificity, and accuracy are calculated for the entire time period of 65 weeks. Sensitivity is the only measurement applicable on a per level basis. Table 10 summarizes the results of each calculation over the entire time period.

| Measurement | Overall |
|---|---|
| Sensitivity | 0.545 |
| Specificity | 0.762 |
| Accuracy | 0.769 |

Table 10: Performance measurements for the entire time period.

Table 10 measures the performance metrics as a fraction of events properly predicted. Sensitivity shows that 54% of the time the model correctly predicts an event occurs; 46% of the time the model incorrectly predicts an event occurred. Specificity implies hat 76% of the time the model correctly predicts an event did not occur; 24% of the time the model incorrectly predicts an event did not occur when in reality an event did happen. Accuracy demonstrates that the model predicts both the presence and absence of an event correctly 80% of the time; 20% of the time it incorrectly predicts the presence or absence of an event. Specificity and accuracy are both reasonable results, with values above 75% for each. Sensitivity is, however, a lower value.

The overall sensitivity is low due to a number of reasons. It appears that some emails from Enron executives have been removed from this dataset. In fact, only 28 emails sent by Ken Lay exist in the time period studied here. This implies that many emails were removed from this dataset, possibly by FERC during its initial investigation. Moreover, coworkers tend to discuss important events in person rather than over email. Since many companies keep logs of their corporate emails, employees may tend not to discuss very sensitive information over email and prefer face to face dialogue for such discussions. Not having these hot topics in the emails could have contributed to the lack of change between time slices that resulted in false negative predictions.

Sensitivity is the only performance measurement valid on a per level basis. This is because the model has no method of predicting the level of events in the case of a false positive; the model only can predict the presence or absence of events. This table details sensitivity calculations for each level of event importance.

| **Measurement** | **Level 1** | **Level 2** | **Level 3** |
|---|---|---|---|
| Sensitivity | 0.667 | 0.444 | 0.833 |

Clearly, this model has a variety of performance characteristics at each level. This model works well at predicting Level 1 events, with 2 out of 3 being predicted as present. Level 2 is not as good, only predicting events 44% of the time. Level 3 events are predicted much more reliability than Level 2, with 83% of events being correctly predicted by the model. Level 2 is likely not as sensitive because of the variety of events that were defined as Level 2. Rather than capturing one unique type of event, Level 2 captures such things as political meetings and major corporate announcements. These are more varied than what defines a Level 1 or Level 3 event. Thus this variety might be why the value for Level 2 is lower than that of Levels 1 and 3.

**Comparing Two Sets of Time Slices**

The validity of the performance measurements can be better understood when details of the Topic Evolution Network are discussed. Detailed analysis was performed on two sets of time slices, one with valid true negative predictions and the other where many events occurred coupled with true positive model predictions. First, the results detailing the Topic Evolution Network around the time where many events happened and were correctly predicted is presented. Next, the Topic Evolution Network is presented for time slices where external event activity was quiet and when no events were predicted. A comparison of the two is then made to motivate why the results presented here are valid.

Figure 3 is the Topic Evolution Network for time slices $T_{47}$ through $T_{53}$. This set of time slices represents the time when Enron publicly collapsed: a buyout offer fell through, Chapter 11 bankruptcy protection was sought, and massive layoffs began. The distance calculated between $T_{47}$ and $T_{48}$ is less than the $\zeta_{event}$ at 0.46154; it is included here for reference before the major events began. In $T_{48}$, the Dynegy takeover of Enron collapsed (Washington Post 2004). This Level 2 event was predicted as present, with the distance between $T_{48}$ and $T_{49}$ calculated to be 1.27. In $T_{49}$, Enron declared Chapter 11 bankruptcy protection. This Level 1 event was predicted as present, with the calculated distance between $T_{49}$ and $T_{50}$ 1.64. In both $T_{50}$ and $T_{51}$, no event occurred yet the model predicted one present with distances of 1.64 and 1.59 respectively. $T_{52}$ included Christmas and New Year holidays. This external event was correctly predicted with the distance between $T_{52}$ and $T_{53}$ calculated to be 1.39. Figure 3 is a graphical representation of Topic Evolution Network for time slices $T_{47}$ through $T_{53}$.

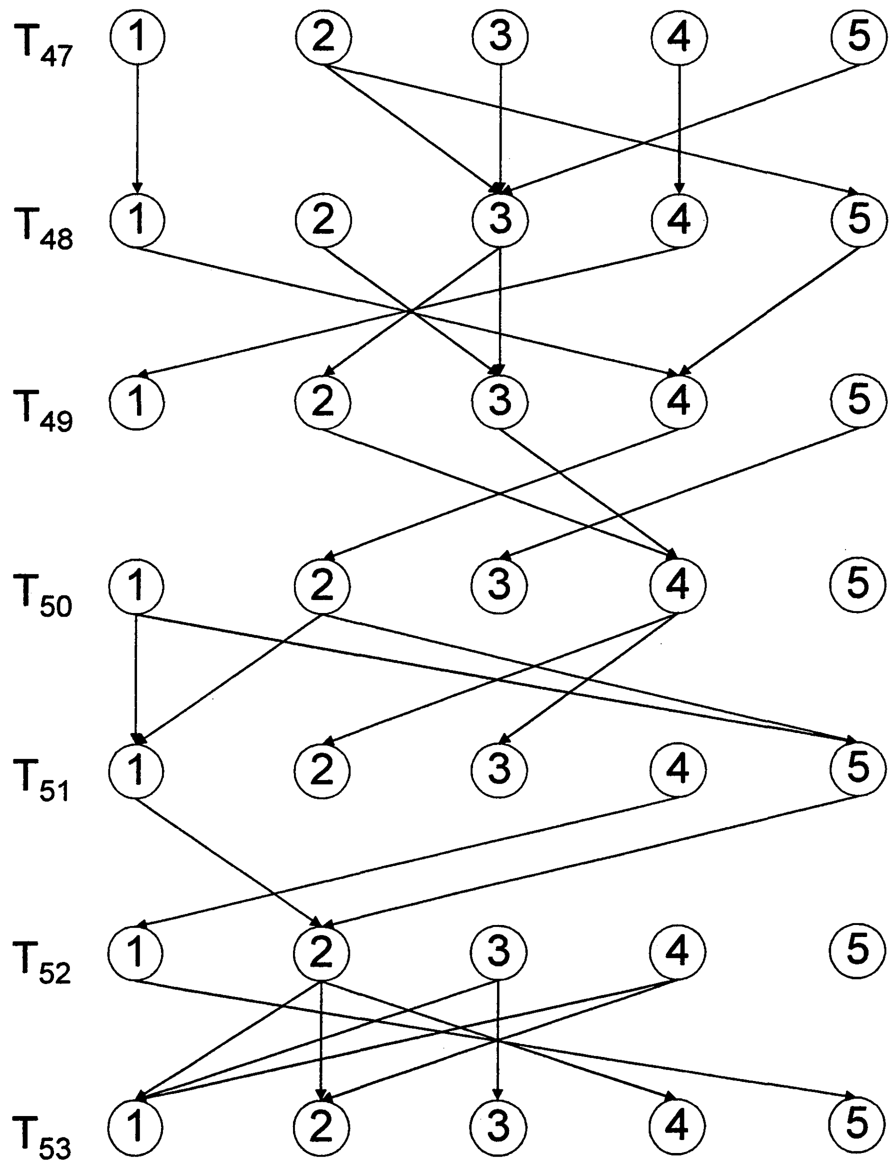


Figure 3: Topic Evolution Network for time slices $T_{47}$ through $T_{53}$..

Examining Figure 3 closely in conjunction with actual emails shows why the model predicted events correctly. Zooming in on $T_{48}$ and calculating its distance from $T_{49}$ yields a number smaller than the event threshold. Only the number of evolutions changed between these two time slices, indicating conversations were carrying on and it was "business as usual". There was more change between $T_{49}$ and $T_{50}$ when the Dynegy buyout offer collapsed. Examining the emails for this time slice in $c^4_{T_{49}}$ finds significant discussion of a number of agreements and payments, which could have possibly been related to the Dynegy deal or precursors to the Chapter 11 filing. The change between $T_{50}$ and $T_{51}$ is also above the event threshold, which correctly predicted Enron's filing for Chapter 11 bankruptcy protection. $c^1_{T_{50}}$, the evolution of $c^4_{T_{49}}$, contains more discussion of payments and how they were to be made. Also in $T_{50}$ is talk of year end bonuses and vacation time in $c^2_{T_{50}}$, even after bankruptcy was announced. This cluster was a creation in $T_{50}$, and is a likely topic for the first week of December given that the end of the year was quickly approaching as was Christmas time. Such large $L_1$ distances between these time slices truly do capture the events occurring at Enron during the tumultuous events of bankruptcy.

Even though many events were captured and predicted as present, Figure 3 also includes two false positive predictions where no events occurred. This is likely due to the fallout caused by such a monumental announcement as bankruptcy. It is one thing for a company to announce it, but the ramifications of such an announcement are felt throughout the entire organization after the fact. The evolution of $c^1_{T_{50}}$ into $c^4_{T_{51}}$ continues discussion of payments but now mentions how bankruptcy will affect them. Changes to the Topic Evolution Network during the aftermath of bankruptcy likely caused the false positive predictions.

Figure 4 represents the set of time slices $T_{25}$ through $T_{28}$. These weeks are when no external events occurred and where the model predicted such. All distances are less than the event threshold $\zeta_{event}$ of 1.25. The distance between $T_{25}$ and $T_{26}$ was found to be 0.45. The next distance was smaller than the first, from $T_{26}$ to $T_{27}$ was calculated to be 0.2. The final distance in this set of time slices from $T_{27}$ to $T_{28}$ was 0.558, the largest of all the distances. Figure 3 is a graphical representation of the Topic Evolution Network for time slices $T_{25}$ through $T_{28}$.

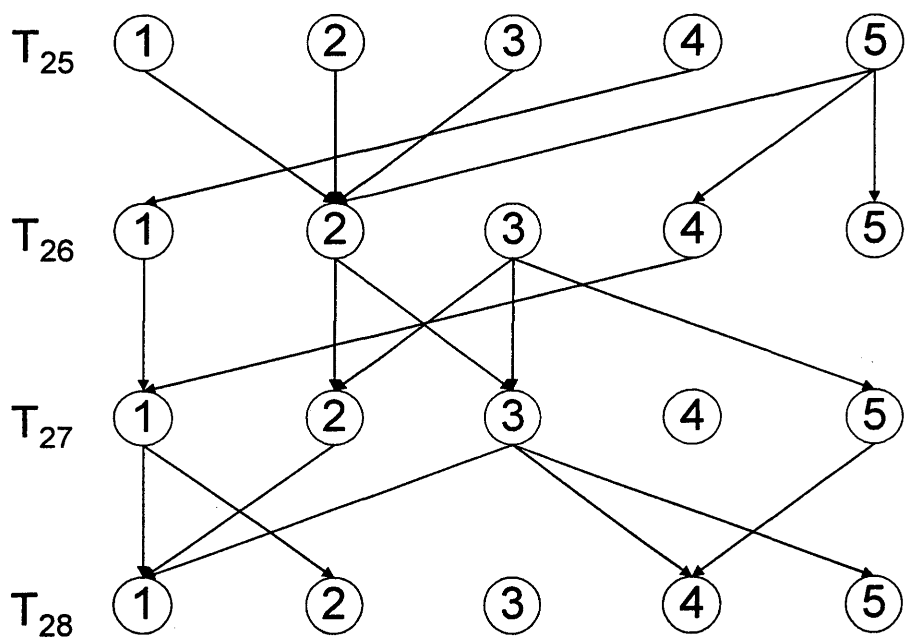


Figure 4: Topic Evolution Network for time slices $T_{25}$ through $T_{28}$.

Time slices $T_{25}$ through $T_{28}$ represent "business as usual" during the booming times of Enron. California was having an energy crisis and Enron was there with solutions for the crisis and to profit. $c^2_{T_{25}}$ contains discussion about the Transwestern pipeline division as well as the current state of affairs in California. This discussion evolves into $c^2_{T_{26}}$ where a market based solution for California is discussed. Cluster $c^4_{T_{25}}$ has mundane emails about reviewing documents that were attached to emails as well as logistics for upcoming meetings. This cluster evolves into $c^1_{T_{26}}$ where more discussion about attached documents is captured. Truly nothing out of the ordinary is happening during these time slices.

Comparing these two sets of time slices from the time period of January 1, 2001 through April 1, 2002, indicates why certain performance measurements are quite high for the overall dataset. The state of flux at Enron immediately before and after bankruptcy is captured by the change in the Topic Evolution Network between the applicable time slices. When the current affairs are ordinary and nothing more than "business as usual", the Topic Evolution Network does not change significantly between time slices. These two example sets of time slices give strong evidence that this framework does reasonably well at predicting external events given an email dataset.

# CHAPTER 5

# CONCLUSION

We present a framework for mining the evolution of topics from a large email dataset. This framework can be used to predict possible external events based on changes to our Topic Evolution Network. This framework consists of a number of steps that transform the original email dataset into our Topic Evolution Network to mine external events. First, time decomposition is performed and the dataset is divided into time slices. Next, all emails go through a preprocessing step, removing unimportant words and stemming those that remain to their root. The third step converts all emails into the vector space model, calculating TF-IDF values for each word in each email. Next the words are ranked and the top words that represent the dataset are chosen; the vectors that now represent emails are reduced to only include these words. The *k*-means clustering algorithm is then used to perform document clustering on all emails. All previous steps are conducted at each time slice. Finally, the best clustering result for each time slice is then utilized to construct our Topic Evolution Network.

Once the Topic Evolution Network is constructed, the counts for our temporal considerations of evolution, creation, destruction, merge, and split are obtained at each time slice. Because these counts have differing scales, all counts are normalized to be between [0,1] using Min-Max normalization. These normalized counts are then treated as vectors at each time slice and the $L_1$ norm distance is computed between the previous and the current time slice. If this distance is above the event threshold, then we predict an external event might have occurred during the previous time slice.

While our framework does consist of some well researched techniques, we apply them in a novel way to an email dataset to predict external events. Our Topic Evolution Network studies document clustering results over different periods of time, yielding the number of topic evolutions, creations, destructions, splits, and merges at each of our so-called time slices. To our knowledge, the literature provides no similar method of document clustering over time in the way that we present. By studying the distance between our temporal considerations from time slice to time slice, we are able to predict external events with an accuracy of 0.769 when the distance between time slices is less than our event threshold. Our technique of building the Topic Evolution Network and using it to construct a model that predicts external events is our unique, novel contribution.

**Applicability of the Framework**

The Enron email dataset was used as the experimental dataset for this framework. The public availability of newsworthy events surrounding the rise and fall of Enron made this an applicable dataset for study. The dataset itself showed clear evolutions of conversations when individual emails were read; details of the California energy crisis are discussed along with what Enron did to create business from the debacle. Periods of great change occurred in the company when Chapter 11 bankruptcy protection was filed in December 2001, as did their email dialogue. Studying changes in the Topic Evolution Network constructed for the Enron dataset showed the applicability of the model for predicting external events when a well defined timeline of events is available.

**Issues**

Issues arose when working with this dataset and applying our framework to predict events. The clustering method selected, here chosen to be *k*-means, will likely affect the clustering result greatly. Considering only *k*-means makes the clustering result dependant on the initial centroid. If an initial centroid chosen is a document not similar to any others in the dataset, then the clustering result will not be representative of the entire dataset. Care was taken to avoid this by requiring 10 runs of the *k*-means algorithm at each time slice, selecting the result that minimized the SSE. However, this still is a possible issue that is addressed here.

Another issue that arose was that the email message bodies themselves contained some integrity issues. A number of emails contained stray, but predictable, characters in the middle of words such as "==90" and other characters that were present between words. Manual inspection of the dataset captured many of these stray characters and found new characters for tokenization beyond the set presented by Weiss, et al. (2005). These were implemented in the data preprocessor. Unfortunately, the dataset also contained a number of message bodies that had lost some white space between individual words that conveyed meaning. These strings such as "OutstandingBelow" were impossible to detect using our preprocessor and thus were considered as one individual term "outstandingbelow". This is an open issue that resides in this research.

The unique nature of an email dataset poses an interesting issue that is only applicable to email documents, that is the content of the email body can have multiple topics of discussion. It is very common for email bodies to not contain just one subject area of discussion. Instead, multiple subjects may be talked about in a single email. For example, one paragraph of an email may request the recipient's presence at an upcoming meeting while another paragraph might include details of work that needs to be completed which is unrelated to the meeting request. It is more convenient for a sender to include all subjects in one email rather than send multiple emails to the same person, thus multiple topics of discussion may be present in one email. This type of issue is not present in documents that reside on a company's server as it is unlikely that, for example, software design documents will also include details on the financials for the company. Multiple topics are not likely to be found on documents that reside in a file system, but are much more likely to be found in email documents.

**Future Work**

A number of components of our framework can be modified for future consideration of this research. This thesis presents the use of the *k*-means algorithm for document clustering but that does not mean it is only limited to this technique. The document clustering method presented here can be interchanged with a number of other types. Bisecting *k*-means or a hierarchical clustering method would be interesting to study per Steinbach, Karypis, and Kumar (2000). Our temporal considerations presented for the Topic Evolution Network can be reconsidered, extended, or added to. Changes to these measures will greatly affect the performance measures of sensitivity, specificity, and accuracy used to analyze the results. These components are suitable for future research and consideration towards extending the work completed in this thesis.

The dataset could also be exchanged with another large email dataset. Enron was chosen because of its public availability, existing research, and well defined timeline of major external events that happened during the time the dataset captures. This framework could be used in a corporate environment to analyze the vast amount of email stored as proprietary information for that company. Studying the changes in the Topic Evolution Network might indicate the presence of events that management might not be aware of, but should be. While only one dataset is used for study of the Topic Evolution Network, we contend that the framework can be used on any similar dataset to that of the ISI version of the Enron email dataset.

## REFERENCES


Michael Berry. Survey of Text Mining: Clustering, Classification, and Retrieval. 2003. Springer.

Berry, Michael W. and Murray Browne,. 2005. Email Surveillance Using Non-negative Matrix Factorization. *Comput. Math. Organ. Theory* 11(3):249-264.

Chapanond, Anurat and Bülent Yener. 2005. Graph Theoretic and Spectral Analysis of Enron Email Data. *SDM 05.*

Chundi, Parvathi, Rui Zhang, and Malu Castellanos. 2006. Entropy Based Measure Functions for Analyzing Time Stamped Documents. *SDM 06.*

Cohen, Robert. 2004. Enron Email Dataset. http://www.cs.cmu.edu/~enron/ (accessed 5 December 2006).

Diesner, Jana and Kathleen M. Carley. 2005. Exploration of Communication Networks from the Enron Email Corpus. *SDM 05.*

Fan, Weiguo, Linda Wallace, Stephanie Rich and Zhongju Zhang. 2006. Tapping the power of text mining. *Commun. ACM,* (49)9:76-82.

Federal Energy Regulatory Commission. 2005. Information Released in Enron Investigation. http://www.ferc.gov/industries/electric/indus-act/wec/enron/info-release.asp (accessed 14 February 2007).

Frantz, Terrill L. and Kathleen M. Carley. 2006. Organizational Response to the Turmoil of Personnel Turnover. *SDM 06.*

Han, Jaiwei and Micheline Kamber. 2006. *Data Mining: Concepts and Techniques.* San Francisco: Morgan Kaufmann.

Hotho, Andreas, Andreas Nürnberger, and Gerhard Paaß. 2005. A Brief Survey of Text Mining. *LDV Forum - GLDV Journal for Computational Linguistics and Language Technology,* 20(1):19-62.

Keila, P.S., and D.B. Skillicorn. 2005. Structure in the Enron Email Dataset. *SDM 05.*

Klimt, Bryan and Yiming Yang. 2004. Introducing the Enron Corpus. *CEAS.*

McCallum, Andrew, Andrés Corrada-Emmanuel, and Xuerui Wang. 2005. The Author-Recipient-Topic Model for Topic and Role Discovery in Social Networks, with Application to Enron and Academic Email. *SDM 05.*

O'Madadhain, Joshua, Jon Hutchins, and Padhraic Smyth. 2005. Prediction and ranking algorithms for event-based network data. *SIGKDD Explor. Newsl.*, 7(2):23-30.

Pathak, Nishith, Sandeep Mane, Jaideep Srivastava, and Noshir S.Contractor. 2006. Knowledge Perception Analysis in a Social Network. *SDM 06.*

Porter, M.F. 1980. An Algorithm for Suffix Stripping. *Program*, 14: 130-137.

Priebe, Carey E., John M. Conroy, David J. Marchette, and Youngser Park. 2005. Scan Statistics on Enron Graphs. *SDM 05.*

Shetty, Jitesh and Jafar Adibi. 2004. The Enron Email Dataset Database Schema and Brief Statistical Report. http://www.isi.edu/~adibi/Enron/Enron.htm (accessed 5 December 2006).

Steinbach M., G. Karypis, and V. Kumar. 2005. A Comparison of Document Clustering Techniques. *KDD Workshop on Text Mining.*

Tan, Pang-Ning, Michael Steinbach, and Vipin Kumar. 2006. *Introduction to Data Mining.* San Francisco: Addison Wesley.

Tang, Na and V. Rao Vemuri. 2005. An Artificial Immune System Approach to Document Clustering. *SAC*, 918-922.

Washington Post. 2004. Timeline of Enron's Collapse. http://www.washingtonpost.com/wp-dyn/articles/A25624-2002Jan10.html (accessed 14 February 2007).

Weiss, Sholom M., Nitin Indurkhya, Tong Zhang, and Fred J. Damerau. 2005. *Text Mining: Predictive Methods for Analyzing Unstructured Information.* New York: Springer.

Wikipedia. 2007. Timeline of the Enron Scandal. http://en.wikipedia.org/wiki/Timeline_of_the_Enron_scandal (accessed 28 March 2007)

White, Colin. 2005. Consolidating, Accessing and Analyzing Unstructured Data. http://www.b-eye-network.com/view/2098 (accessed 14 Februrary 2007).

Zhang, Dong, Daniel Gatica-Perez, Deb Roy, and Samy Bengio. 2005. Modeling Interactions from Email Communication. *Proc. IEEE International Conference on Multimedia & Expo (ICME)*, 51.