2010

EMail Data Mining: An Approach to Construct an Organization Position-wise Structure While Performing EMail Analysis

Bhargav Vadher
San Jose State University

EMAIL DATA MINING:

AN APPROACH TO CONSTRUCT AN ORGANIZATION POSITION WISE STRUCTURE WHILE PERFORMING EMAIL ANALYSIS

A Writing Project Presented to

The Faculty of the Department of Computer Science
San Jose State University
California

In Partial Fulfillment of the Requirements for the Degree
Master of Science

By
Bhargav Vadher
Spring 2010

ABSTRACT

In this age of social networking, it is necessary to define the relationships among the members of a social network. Various techniques are already available to define user-to-user relationships across the network. Over time, many algorithms and machine learning techniques were applied to find relationships over social networks, yet very few techniques and information are available to define a relation directly over raw email data. Few educational societies have developed a way to mine the email log files and have found the inter-relation between the users by means of clusters. Again, there is no solid technique available that can accurately predict the ranking of each user within an organization by mining through their email transaction logs. The author in this report presents a technique to mine the email data log files in order to figure out the position wise structure of an organization. The author also discusses send-receive analysis, statistical analysis, semantic analysis and temporal analysis over the data, and has applied them to test cases. Throughout the research the author has used the Enron employees email log files, which was made public on 2001.

ACKNOWLEDGEMENTS

I thank my advisor, Dr. Robert Chun, whose guidance, support, and dedication is priceless. Dr. Chun is an educator with the truest sense of the knowledge and very kind personality suitable for students. I sincerely appreciate Dr. Mark Stamp's and Dr. Chris Pollett's participation as project committee members. The committee has provided me an enlightening insight, guiding and polishing the details presented in this report. All of them, committee members and advisor, have been proved as precious assets to computer science department.

I would especially like to thank the IEEE (Institute of Electrical and Electronics Engineers, Inc.) and to SJSU research articles database [27] from Sjlibrary for providing recent information about my topic.

For sure it has been a challenging, yet rewarding and fruitful journey which I could not have completed alone and I am thankful for all your support.

Thank you.

Table of contents

List of figures

1.0 INTRODUCTION

This is an age of the internet, particularly social networking. Mining data from already existing sources of information can be valuable in order to figure out an innovative output from the data. Advanced techniques in social network data mining has made it possible to convert the raw data into a useful piece of information. We can see many examples of integration and compaction of social networking as a result of data mining. There are many forms of data mining that can be used to analyze a social network, and one of those is data mining of email log files.

1.1 Overview

Email data mining techniques are useful for figuring out how the sender/receiver of email is associated with each other. Because this kind of data mining is a relatively new field of research, there is not much progress involved in the field of email data mining. Some educational societies have discovered a few techniques to find a cluster of users that belong to the same group/team. A few educational groups have started these kinds of projects to find out hidden functionality and past structure of the employees of an organization.

Throughout this research work, the author has used and applied the basic algorithms on the Enron employees database. According to Wikipedia,

> "*Enron Corporation was an American energy company based in Houston, Texas. Before its bankruptcy in late 2001, Enron employed approximately 22,000 and was one of the world's leading electricity, natural gas, pulp and paper, and*

communications companies, with claimed revenues of nearly $101 billion in 2000" [25].

The bankruptcy of Enron occurred from institutional, systematic and planned accounting fraud and was given the name of *Enron scandal.* This dataset was originally made public and was posted to the web by the *Federal Energy Regulatory Commission* while an investigation was going on regarding the accounting fraud [26]. The data is in the form of email log files extracted directly from the pop server. It mostly contains the email transaction of 150 Enron employees during the time span 1998-2002.

1.2 Email Log Mining

The main purpose of email data mining is to present social network relationships and newly emerging parts of a social network. Due to the increasing threats to national security, people have started to use the results of email data mining to figure out terrorist threats. As of now, no one has tried to figure out future relations among the employees or even trace the behavior of an employee. Hence using email data mining and finding out future and behavioral relationships among users could be extremely useful for an organization where the employee survey is often taking place. In some organizations, it is already in use to collect email statistical data and the progress of an employee compared to an employee sitting beside him. Many reputed companies are already in the process of using email log analysis to improvise spam detection, employee personalization and automated filling. The manager of a particular team/group would be able to analyze the behavioral and social relationships between the users under his/her team, by just mining an email archive. As more and more research is being conducted in this area, some

fruitful results have been produced in forensic analysis and national intelligence services. Last but not the least, in the area of decision making systems, researchers are more focused on email mining to achieve future decision making results for an organization.

In recent years, email has become a necessary part of any organization or group of the similar kind of users who share their information with each other on the network. Since email has become a necessary tool to communicate and co-ordinate with each other within an organization, the mining of email is sure to give some future decision and inter-relation based information. Let us consider an example of a software team within an organization. The users of the software team are constantly in touch by means of emails. Here, the email analysis and data mining will be helpful in determining the flow of work within the team and for a particular user from his/her email transactions. It is also possible to figure out the closeness among the users of this software team by finding answers to the questions like, who is communicating more with whom in terms of emails? After tracing an email and pattern matching, managers will be able to determine a unique chain/thread for a particular employee, and that will help the manager/super-user to estimate inter-team relations and in making future decisions.

Figure1 below shows the network of email users within an organization. From the figure we can clearly see that some groups have very dense node and edges between them, while some have fewer nodes and very few edges between them. This explains the inter-relation between users and groups of users within the software team.

Figure 1: A typical network graph of an email user within an organization [25]

The very dense group on the extreme left may be the core group of software developers, and most of the developers in that group send and receive daily updates from each other via email for check-in and check-outs. Whereas, the separated and sparse groups on the right side of the figure might be the group of technical writers or HR who usually don't communicate with the core development team frequently.

Recent use of email analysis and data mining of email contents has proven to be useful in some sensitive places like national security agency to detect threats and fraud determination from terrorists. Moreover, it has been proved to be helpful for decision making, future team co-ordination, fraud detection and tracing the behavior of an employee.

2.0 RELATED WORK

The field of email data mining is relatively new to researchers around the world. Because of this fact, there is not much material available for email log data mining and email analysis to review. After reviewing a number of IEEE journals and conference papers from ACM digital library and SJPL library [27], I had gained access to a few journals that proved to be useful for my research project. Apart from the ACM library journals, I got some latest information in email data analysis from the computer science department of a few universities including UC Berkeley, US Davis and MIT. Below is the list of a few helpful journals that I have used in my initial research.

1. The paper "*A Mining Algorithm for Email's Relationships Based on Neural Networks*" [1] from IEEE computer society, gave me the basic understanding of email transaction flow model, which would be my backbone throughout the research project.
2. The paper "*Mining Email Social Networks*" [2] from ACM digital library, gave me the understanding of how to remove junk data and how to unmask the aliases. The paper also shows few methods of data extraction and content gathering.
3. The journal "*Email Mining Toolkit Technical Manual*" [28] from Department of Computer Science, Columbia University, helped me to understand the backend architecture behind the email analysis and the Clique algorithm implementation for email data mining to find relations between users.

The main and tedious task involved was to find the dataset that can be useful as a test data for the research work, and to remove junk data and redundant data that are not required in test dataset.

2.1 Related Research

Many researchers have been doing similar work on email analysis. Shlomo Hershkop et al worked on a visualization of email relationships for particular organization [4]. Similarly Nimish Pathak, Sandeep Mane and Jaideep Srivastava have done some key research on *socio-cognitive analysis of email networks* [10]. Nimish Pathak et al worked on a huge dataset to find out "who thinks who is who?" On the other hand, Giuseppe Carenini et al have data mined emails from large folder for *scalable discovery of hidden emails* [13]. Other researchers group, Rong Qian et al, have worked on an email corpus to detect a community structure based on link ranking [18]. Rong Qian et al focused on link mining techniques to detect community structure. Rong Qian et al found interesting result in the form of different teams and community for a given email corpus. All above listed research work has done using existing data mining techniques.

There was another group of researchers who worked on social network analysis to detect relationships. A group of Computer Science professors from University of California, Davis, worked on analysis of a social network. The group, Christian Bird et al, have chosen emails for their primary test-data to apply different mining techniques. Christian Bird et al used in-degree and out-degree of email sent to find relationship between the sender and the receiver, and they proved that an email social network is a typical network community [2]. R. Agrawal et al have achieved similar kind of result. R. Agrawal et al focused on social network for news readers. They analyzed behavior of news readers and derived news groups that shares similar taste.

Another group of researchers, Bron C et al, directly applied their algorithms on the email corpus [5]. They have found all the cliques of an undirected graph from the

email corpus. On the other hand, O. De Vel et al worked on the email corpus to detect author identification forensics [6]. They were only interested for the author of particular email for forensic purpose. Hence, they have mined contents of each email to find as much as possible information about the author of that email.

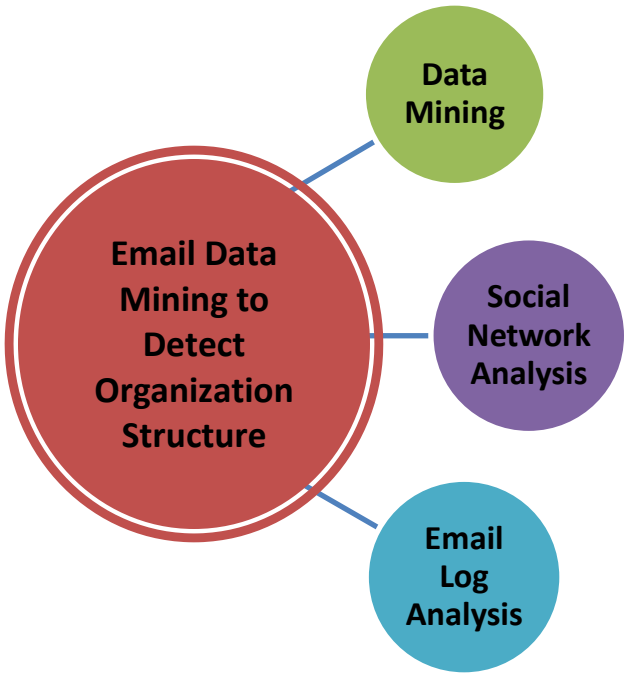

Figure 2: Inter-connection among different research works

Figure 2 above describes the inter-connection between four kinds of research work. Three small circles on the right side describe three different kind of research work done by other researchers. Whereas, the big circle on the left side describes this research work. It is clear from the figure that each of the three research works on the right side were independent and implemented to detect relationship among users. These three research work belongs in Data Mining, Social Network Analysis and Email Log Analysis categories respectively. While exploring literature of above mentioned research works, the author came across an idea of merging all of them into one meaningful entity. The

author utilized basic techniques from each research work and planned to implement them over the Enron Email Corpus made public on October 2001. Since no one has ever tried to construct an organizations hierarchy based on employee ranking and position, the author has started research on it.

2.2 Data Gathering

As mentioned above, due to relatively new field of research, there are very few existing resources available. Moreover, concerning the privacy and security of an organization and its employees, most organizations never disclose their email transactions or server email logs to the public. After doing a lot of research to find a test dataset which can be used as a backbone for my research, I found relatively huge and dense dataset of email transaction log files of an organization.

After declaring its bankruptcy in late 2001, Enron made its 150 employee's email transaction log files publicly available for research purpose. The department of computer science at CMU [26] helped me to find the dataset. The dataset contains more than 500,000 email log files shared among 150 Enron employees over a period of 5 years starting from 1998. The dataset contains all the email transactions from each and every personal folder of an employee, i.e. it contains emails from the folders like inbox, sent, all mails etc. It also contains automatically system generated folders like *discussion_threads* and personal folders like *home_mail, from_allen* and many more.

3.0 DESIGN

The design part mainly consists of database table/matrix creation so that those can be used in the future to save I/O time. This was a very crucial phase of the research, as all assignments in the future would be dependent on these matrices. The database contains a total of 21 tables. All of these contain unique information about different parts of raw files. Moreover, some tables are created to support the inter-relationships between the tables e.g. *child_parent*, *sub_response*, timeline and many more.

Later, after creating these tables, four different analyses were done. These are summed up as send-receive analysis, statistical analysis, semantic analysis and temporal analysis. All of these analyses solely depend on the 21 tables.

3.1 Cleaning the Junk Data

As mentioned above, the dataset contains lots of emails for a particular employee from his/her mailbox folders. Hence, it can also contain lots of junk or redundant emails that need to be removed. Each log file has the same format; it contains header data, timestamp of transaction, sender and receiver, subject and content of the email itself. All the attachments and signatures were removed for the security and privacy of an employee. Using the power of PHP's regular expression, all the redundant mails and all emails that did not have either the sender or the receiver, were removed.

The following figure 2 shows a system diagram of the research work. There are mainly four kinds of analysis that will be performed over the email database corpus of Enron Inc. The final results will be compared with the fact sheet that came together with the database.

System diagram

Besides junk mails, there were lots of employees who used to have more than one email ID to contact in different groups inside and outside the organization. One example in this case could be the CEO himself, who used to have three email IDs like Ken.lay@enron.com, chair.enron@enron.com and k.lay@enron.com. So determining alias from the database was an important task. "*Mining Email Social Networks*" [2] from ACM helped the author in that context. Another potential problem was that many emails did not have either a sender or a receiver, and that could become reason for chain-break, so *no sender/receiver* has been placed on those emails sender/receiver field. Moreover, due to system generated emails folders, there were lots of redundant emails that needed to be eliminated. After eliminating all such redundant emails, the size of emails in the dataset reached slightly more than 362,000 from 517,000 emails.

3.2 Database

After cleaning all the redundant mails, it was time to create the MYSQL database of the raw mail logs to arrange in meaningful tables. User could have used the log files directly to program the authors work, but then it would have cost the user a lot of CPU power and memory for I/O operation while reading files in PHP. As it was required to access many records of dataset at a time, it would have almost occupied user's main memory if the user would have used log files directly. Definitely it would have reduced the performance dramatically, but since all log files were converted in 21 different meaningful MYSQL tables. Using those 21 tables the user can query large number of data at a given time without degrading the performance.

All 362,000 log files were modeled into meaningful MYSQL tables and each of them represents different parts of a log file. Some table/matrix represents a relationship between other tables. Tables also represent relation among the employees. Below are some of the examples of such tables.

1. Mailgraph – relates each employee's send/receive statistics with every other employee in an organization.
2. Clique – relation between employees who share at least 5 emails interactively.
3. Sub_response_send/receive – a matrix that contains relative weighted percent of each employee with other employee.
4. Response – a matrix that contains overall weighted percentage of particular employee.
5. Timeline – a matrix shows email flow of every employee, sorted by month for entire 5 year communication.
6. Child-parent – a matrix shows possible child node and parent node for all 150 employees, for which we have original ranking in Enron factsheet.
7. Lawyer_relation – a matrix contains relation between the sender_id and message_id, especially for messages sent to lawyers.
8. que_datails – a matrix contains senders and receivers information for messages for which sender asked questions to receiver.

Creating a database for entire dataset has proven very time-efficient and easy to use throughout this research work. The relational diagram of all the matrices in the database is shown in figure 4. The matrices are divided into three different levels of

hierarchy according to their generation method, and the level showing hierarchy is shown in figure 5.

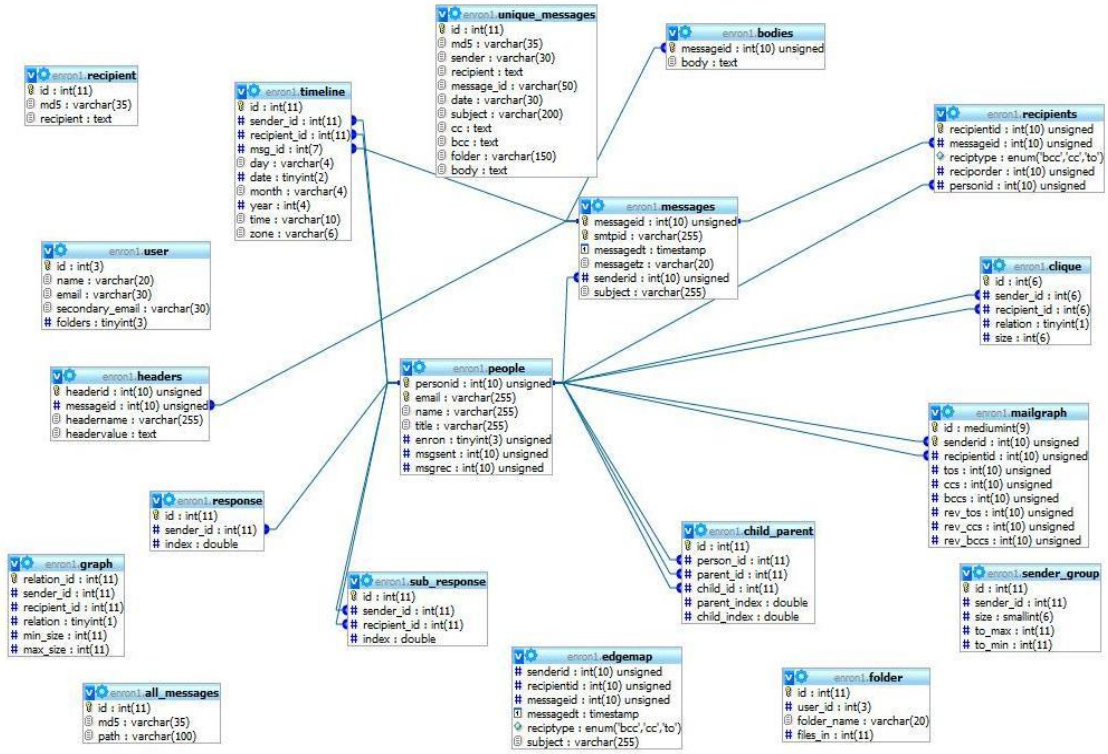

Figure 4: Relational diagram of all database matrices

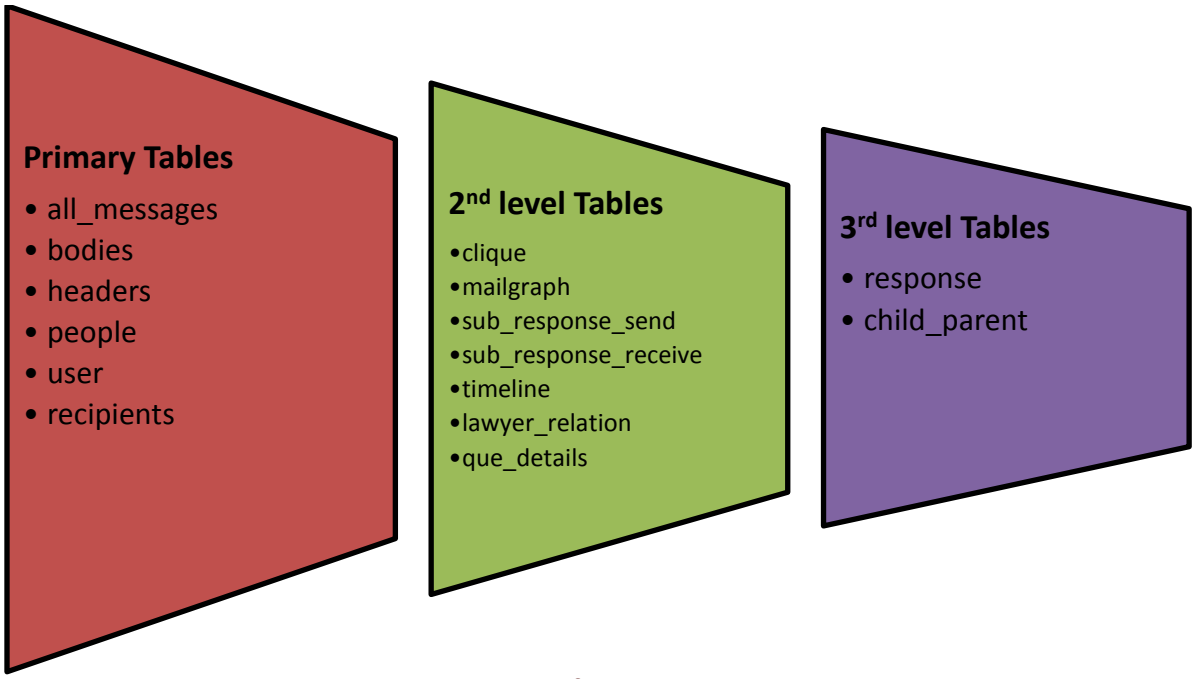

Figure 5: Levels of database matrices

4. ANALYSIS

The analysis part is the core of this research work where the output is going to be exposed in terms of database tables and graphs. Four kind of analyses were done to support the thesis, these are

a. Send-receive analysis (s-r analysis)
b. Statistical analysis
c. Semantic analysis
d. Temporal analysis

The s-r analysis was directly based on primary tables in the database whereas; the statistical analysis was based on outcome of s-r analysis. Hence, they were dependent on other source, i.e. database tables. The semantic analysis and the temporal analysis were done to find inter employee relationship.

4.1 S-R Analysis and Statistical Analysis

The goal behind these two analyses was to construct an organization employee's hierarchy, so that it can show position/roll of each employee within the organization. Once having a hierarchy, one can apply temporal analysis together with semantic analysis to define a relationship between employees. An excel sheet included with the database has a list of employees and their actual ranking before the bankruptcy of the Enron empire. The author called that excel sheet a factsheet. Hence, one can compare his/her research results with this fact sheet to map two difference hierarchies. Overlapping part of proposed hierarchy will be success and the part of hierarchy that will not overlap with the original hierarchy will be mismatch or exception of noise value.

The construction of a hierarchy was divided into three different parts. These three parts were finding

a. Root
b. Second level employees, and
c. Lower level employees.

4.1.1 Finding the root:

The CEO or the President of Enron organization was considered as the root of a hierarchy. All the VPs, directors, managing directors and managers were considered as second level employees. All the traders, normal employees and employees whose position was not available in the fact sheet were considered as lower level employees.

4.1.1.1 Send-receive analysis

The s-r analysis was purely based on the data stored in the 1st level tables. The s-r analysis was made in top-down pattern, but it is easier to understand it with bottom-up pattern. Mentioned in figure 6 are steps in bottom-up creation and analysis of hierarchy.

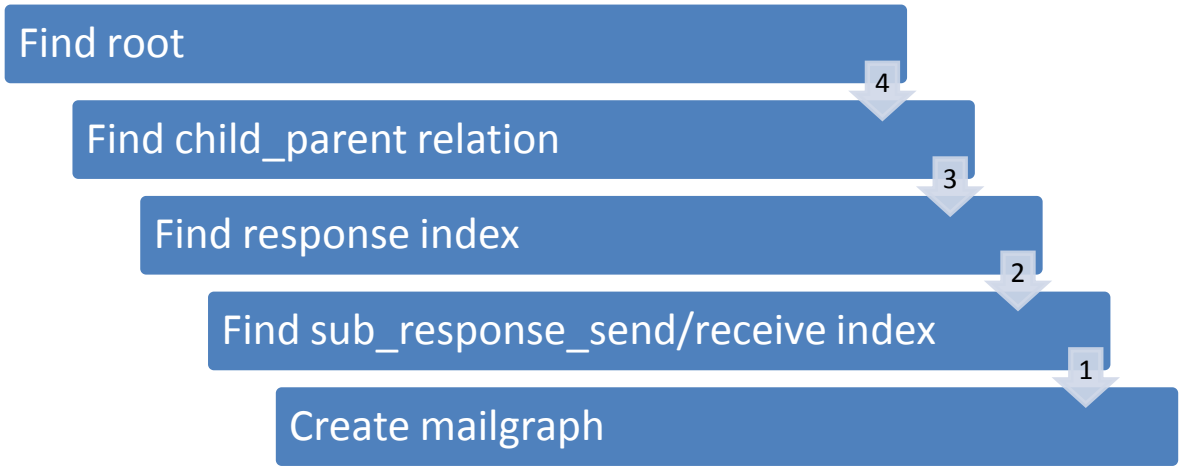

Figure 6: Bottom-up approach of finding root node

4.1.1.1.1 Bottom-up Explanation:

It is natural for creating a hierarchy or a tree structure first thing anyone need is the root node or the base node on which other nodes relies. Hence, first step involved was to find the root node or the CEO/President of the organization, but from above figure it is clear that there were 4 steps need to be completed to reach the root node. The discussion given below is about top-down creation of each table/matrix that leads to creation of the root node of the hierarchy.

Step 1:

The very first step was creation of the 1st level matrices. As mentioned previously in a design section, six 1st level matrices/tables were created; and those will be used in every calculation needs to be done ahead in the research work. Those were very basic tables created directly from raw server log files by using the power of regular expressions in PHP. The tedious part was to create a *people* table. After analyzing all 362,000 emails in dataset, 87,475 unique email IDs were found. From those 87,475 emails, majority of contacts did not belong to the Enron but the outside industries and collaborative firms. Out of 87,475 contacts, 34,000 belonged to the Enron and the author was only interested in particular 150 contacts out of 34,000. The fact sheet contains information regarding 150 employees those were believed to be actively involved in the *'Enron Scandal'*. Different scripts were made for creating 1st level tables. Time complexity for each of those tables was calculated to be O(n) , as one has to check each and every email to store particular detail in the table.

Step 2:

After getting the 1st level matrices, one needed to have a graph or a matrix that shows send-receive details of each user. The *mailgraph* table contains send-receive and reverse send-receive history of each user over time period of 5 years. The matrix *mailgraph* acts as a base for each second level matrix creation process.

Step 3:

The third step in creation of the root node was to create the sub*_response_send* and *sub_response_receive* matrix. Both *sub_response* matrixes highlight the weighted percentage of each employee with other employees with whom the employee has communicated at least once. In other words, if employee A and B have communicated at least once, then there should be some kind of equation/formulae that shows their weighted relationship in terms of send-receive.

Both tables, *sub_response_send* and *sub_response_receive* contain an index for each person_id for send and receive respectively. The number of email sent by the sender will become the send-index for the sender. In a similar way the number of email received by the receiver will become receiver-index for the receiver.

Let us consider another example where employee A sends 12 mails to B, and employee B replies back only 5 times. In this particular example send-index for A will be 12 and a receive-index for A from B will be 5. Similarly a send-index for B will be 5 and receive-index for B will be 12 as opposite to A.

Here an assumption was made with the fact that, if an employee is receiving more replies than number of emails he/she is sending to other employees, then he/she will be considered as an irresponsive employee. Whereas, if an employee is receiving fewer

replies than the number of emails he/she is sending, then an employee will be considered more responsive. Irresponsive means an employee might have busy and was not able to send reply back to the sender, and responsive means an employee might have sent quick reply. If we think logically and apply the situation for a real organization, then only leaders would not reply back quickly to the followers, which mean higher level employees tend to reply less to the lower level employees. Hence, in above example employee B might be the leader of employee A and an employee A might be the follower of employee B.

In short, the matrix *sub_response_send* and *sub_response_receive* contains send and receive index value for each employee as compared to every other employee in the organization, therefore they have a complexity of $O(n^2)$.

Step 4:

From step 3, we have got an index value for each and every employee with his/her communicator. As discussed above in statistical analysis, finding only an index values was not sufficient, it was also required having a general average index value for each employee. For this reason, another matrix was created with the name *response*, to store the general average value of each employee contacting every employee in the time period of 5 years. Here the work was easy; one just needs to calculate the average index value of every record attached with a particular employee as a sender from *sub_response_send* table. Let us consider in *sub_response_send* matrix sender 36 has contacted 5 employees with index value of 2, 4, 3, 6 and 1. All that one needs is to compute average index value by applying standard average formulae, which comes out to be = 16/5 = 3.2; the result

shows that an employee is more likely to fall in follower category and not in the leader category. Hence this employee tends to be a lower level employee in the final hierarchy.

Step 5:

As per our earlier discussion we know that, the aim of s-r analysis was to find out the root and its descendent child, so finally output results in a complete organization structure based on each employee's position. Up to this point, i.e. step 4, only relative index of each employee compared to all other employees was found, but still no sense of parent-child relationship for any two employees has been found. Hence, the next step involved was to find out possible immediate parent and child of each and every employee.

4.1.1.1.2 Filtering process:

When the user applies the algorithm to find out possible child and parent relationship for each employee, he/she needed to eliminate all the employees for which there were no solid proof to compare with the *fact sheet*. In other words, only consider those employees who were listed in the *fact sheet* provided by the dataset. The filtering process is shown below in figure 7.

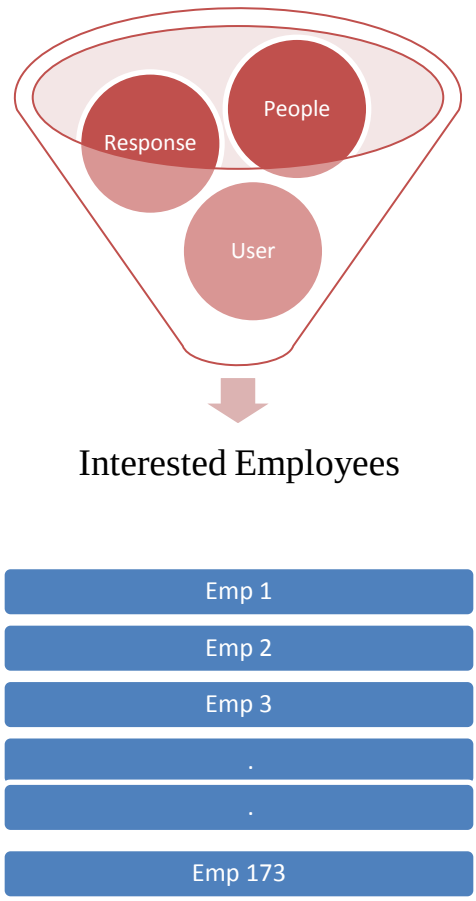

Figure 7: Filtering of Database Matrices

The figure 7 above illustrates the filtering process to find out interested employees those might be in the *fact sheet* to compare with. An idea started with the 150 folders that came with original dataset, i.e. the dataset was mostly about the communication among these 150 employees. The list of 150 employees was obtained by applying regular expression on sender's field of the log itself and results were stored as email ids in the *user* table. Here a problem faced was, some users had more than one email ID, which needs to be taken into account too, and due to this reason there were 173 email IDs for total of 150 employees.

Another problem faced was, all information related to those 173 employees were distributed into 3 different tables namely *people*, *user* and *response*. It was necessary to

filter all the three matrices using MYSQL queries and then integrate the results into a single hash table. The idea of filtering information is shown above in figure 7 that gives us a result in form of a hash table with all 173 employees and their response index along with their email IDs and secondary email IDs.

After having the required employee list, it was needed to find possible parent and child who were immediately associated with them. Again, for finding child-parent relationship, the same algorithm process was used that was used for finding an index of each employee from the *sub_response_send* and *sub_response_receive* matrix. This algorithm works in the following manner; first the index of a particular employee out of 173 employees was taken against all other 172 employees, and the maximum of 172 send-indexes from sub_response_send was taken as a possible child and the minimum of 172 receive-indexes from sub_response_receive as a possible parent. After having possible parent and child for all 173 employees they were stored into the matrix called *child_parent*. Hence, up to this point the possible child and parent for every node has been found, but still the final child-parent relationship of the organization hierarchy was unclear.

After completing the 5^{th} step, next task was to find the root node in the hierarchy or in other words, to find the CEO/President of the organization. The process involved in finding the root was pretty simple, we just needed to find out the employee/node that was voted maximum times a parent node of some child node. In fact, the voting algorithm was created which finds the possible root of the hierarchy.

The easiest way to find the root in this case was simple voting algorithm. Voting algorithm looks into the *child_parent* table to search an employee that has been voted

maximum times as a parent. This algorithm would detect an employee who is likely to be the root of the tree. A simple MYSQL query was created that served this algorithm. The algorithm will look for every employees from the list of 173 employees and creates a list of each employees indexed by the number of voted as parent. The figure 8 below shows the proposed root node and figure 9 is the original hierarchy for the root node.

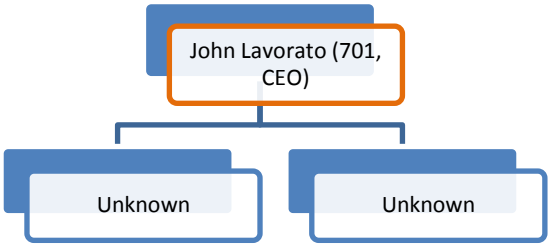

Figure 8: Proposed hierarchy for root node

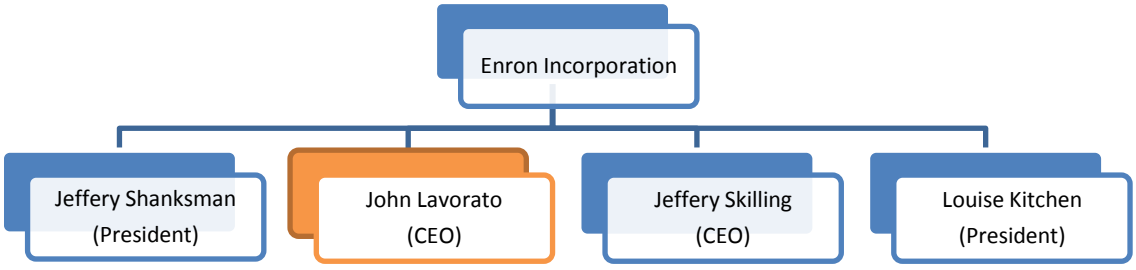

Figure 9: Original hierarchy from fact sheet for root node

Until this point the first out of 3 hierarchy creation part was completed. The algorithm just found the root of the hierarchy, and next steps was to find out 2nd level employees and lower level employees. So the following step was to find the second level node or second level employees in the hierarchy.

4.1.2 Finding 2nd level nodes:

From this point onwards only statistical analysis was used to find nodes in the hierarchy. The basic fundamental behind this phase was to find out all the immediate employees in the organization to whom the root has sent a lot of emails. Why this idea will work? Until now we used the fact that more the send mail lower the ranking, and more the reply back higher the ranking, but this idea was only to find the root node or the president of an organization. In the second phase of hierarchy creation the user will consider the reverse fact, i.e. an employee with high number of sent messages will be placed on top level compared to an employee who received a lot of messages and will be placed on bottom level of the hierarchy. The reason behind this fact was logical. The root node was successfully found, i.e. the president of the company, now if we think logically, the president will only talk more with the higher level employees directly. So here we have to reverse our idea of high send message low position phenomenon.

The president would have only talked with vice presidents, MDs a lot of time directly compared to the lawyers and other regular employees. Moreover, the fact from s*ub_response* matrixes that higher the index of the sender to the receiver, high the number of messages sender might has sent to the receiver. Similarly lower an index of sender to the receiver, less number of messages the sender might have sent to the receiver or in other words, higher the number of replies receiver would have gotten back.

For finding the second level nodes directly from the root node, a function, *getChildren*, was created that takes an input as root node and finds its possible immediate children. From sub_response_receive matrix it was required to find the maximum index value among those 173 employees we are interested in. Thereafter, one will take that

index and finds the sender_id of matching row where parent_id is the input parent. So this method returns the immediately possible child of the root. In successive iteration I have received four nodes; 293, 1637 and 8303 for the second level employees.

So if one looks for node 701, the CEO, in the *sub_respons_receive* matrix, and collect the higher valued index, they are likely to be the vice presidents or the second level employees directly below the root node. The result of above query gives us some high value index nodes such as node 293, 1637 and 8303.

After getting proposed second level nodes from the root node, it was time to compare those nodes with the original hierarchy of Enron. Below figure 10 shows proposed second level nodes and figure 11 shows original hierarchy for second level nodes in the Enron.

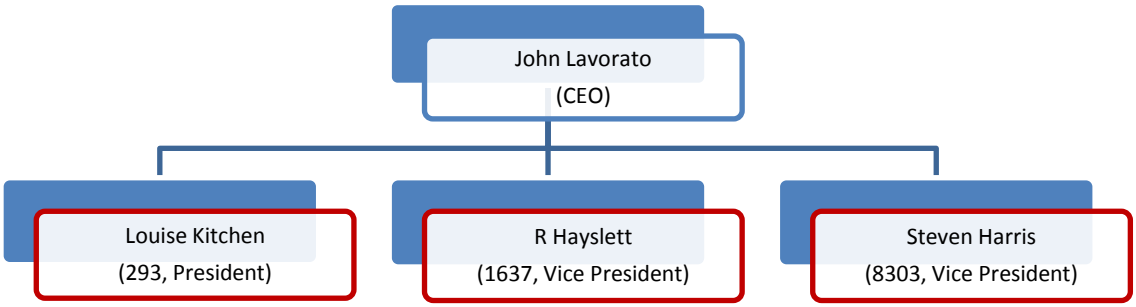

Figure 10: Proposed hierarchy with top 2nd level nodes

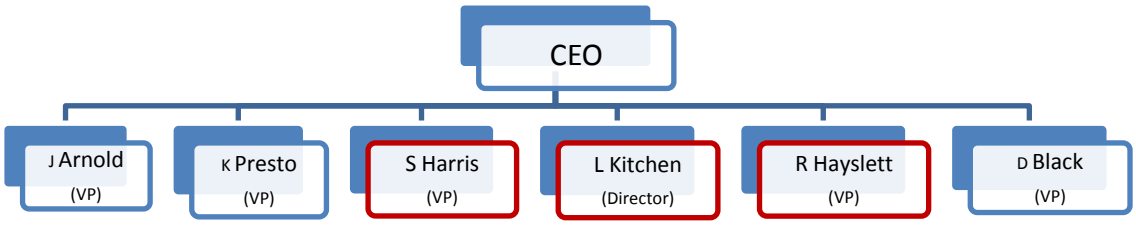

Figure 11: The Enron 2nd level hierarchy with some of the VPs

From the figure 11 above, one can clearly see that a hierarchy successfully matched the President and two vice president nodes with the original 2nd level hierarchy of the fact sheet. So, nodes 293, 1637 and 8303 were successfully classified as the President, VP and VP node respectively in the hierarchy.

Then it was time to go further deep in to the 2nd level hierarchy to find out other nodes like Vice Presidents, director, managing director and managers. Starting with node 293, two top indexed nodes 185 and 1492 were found. Afterwards for node 1637 node 2537 found as top indexed node. Finally for node 8303, nodes 1672 and 1903 were found. After having completed statistical analysis on all of 5 listed nodes against their parents, the hierarchy that was found is shown in figure 12 below.

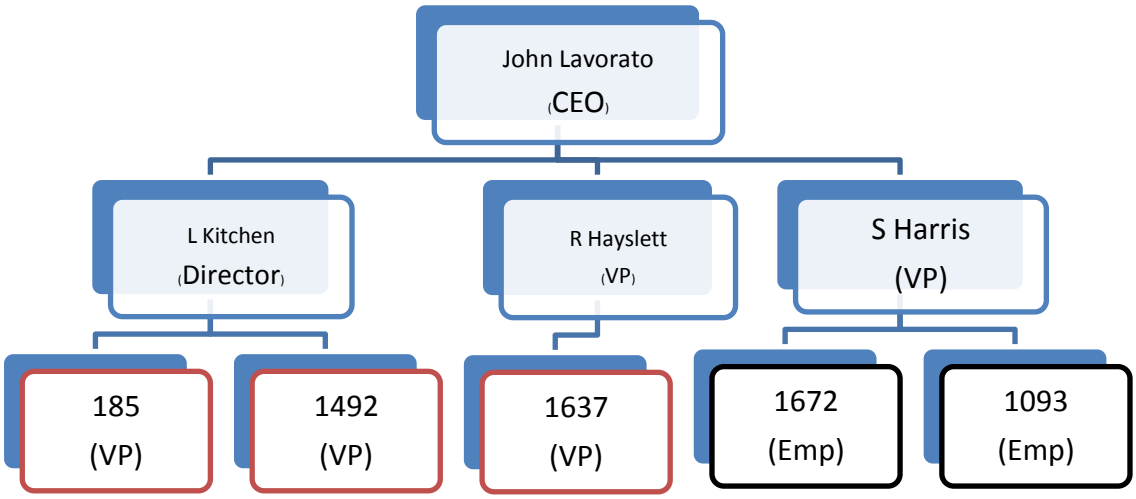

Figure 12: Proposed 2nd level hierarchy for Enron

Up to this point the 2nd level of the hierarchy was almost covered which is shown in the figure 12 above. As seen from the figure the nodes with red outline shows correctly matched nodes with original hierarchy. We can also see some nodes in black outline, e.g.

node 1672 and 1093, which shows that employees have been found, but at wrong position.

4.1.3 Finding lower level nodes:

The final and third step was to find the lower level employee of the hierarchy. The idea was simple compared to the first two level of hierarchy. To find out the lower level employee one just needed to find the descendents of the last level nodes of 2nd level hierarchy. Below listed are the steps to find out lower level employees from the 2nd level hierarchy.

1. Take 2nd level employee from the last row.
2. Look for this employee in *sub_response_send* matrix to get maximum index
3. Look in to the *sub_response_receive* matrix and *response* matrix

After applying the above mentioned steps for each of the last level nodes of 2nd level hierarchy, all nodes those were normal employee were found. In other words, the 3rd level of hierarchy only contained the nodes those were with the high index value against their parent in the last raw of 2nd level hierarchy. For example, let us consider 185, last raw node in 2nd level hierarchy. When the above mentioned steps were applied, the highest index node found and it was 22786. When the employee 22786 was compared, after querying the people table, it was found as normal employee. Likewise, those 3 steps were applied for all last level nodes in 2nd level hierarchy and the results found was as shown in the figure 13 below.

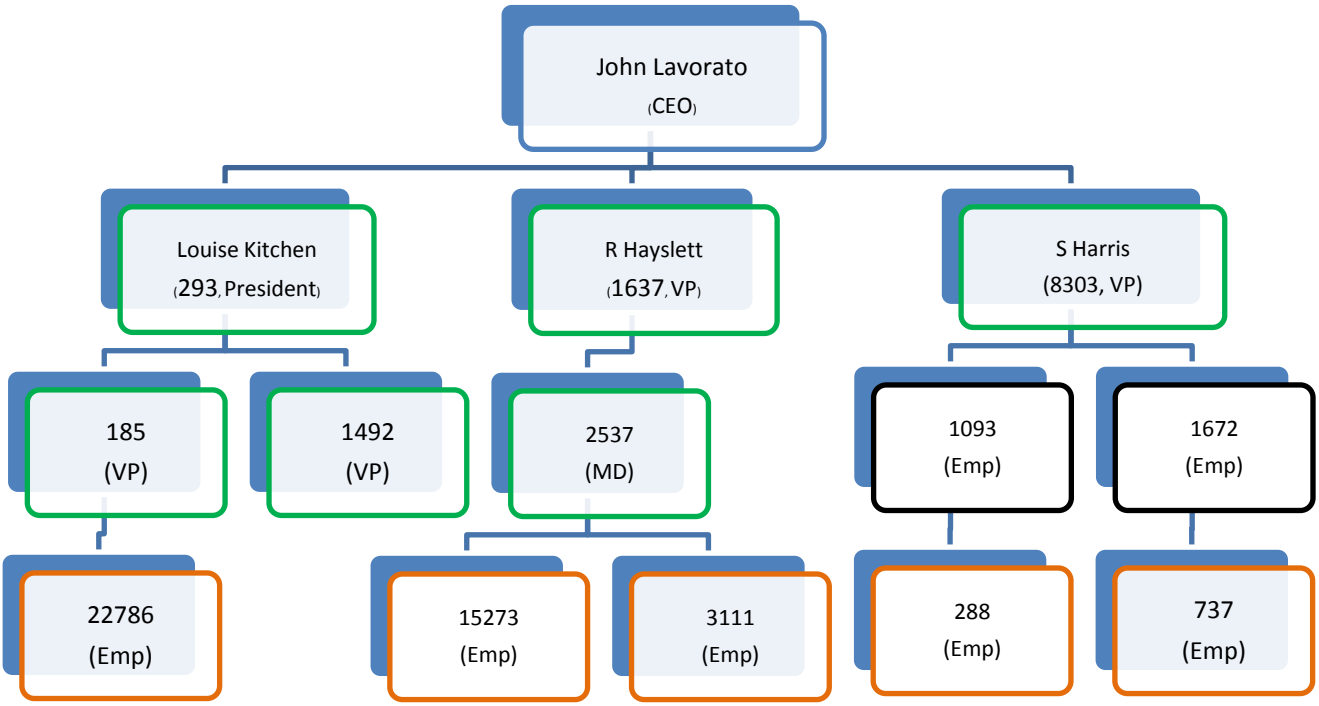

Figure 13: Overall Enron hierarchy with 3rd level nodes

The figure 13 above shows the partially complete hierarchy for the Enron Incorporation which has derived using this research work. After comparing it with the original hierarchy from fact sheet; most of the time proposed nodes in figure 13 above and their level matched with the original hierarchy. There were some exception and noise nodes in the hierarchy. Two noise nodes that were present in the above hierarchy were 1093 and 1672. These noise and exception nodes cases were covered in the experimental results' section

4.2 SEMANTIC ANALYSIS

The meaning of semantic analysis is to find out the results from the meaning of words and sentences in the document. In this paper the semantic analysis was applied to the content of an email itself. The following section discusses reasons for this analysis, how it works and specific case of semantic analysis.

4.2.1 Need for Semantic Analysis

In the previous analysis, the statistical analysis, the partial hierarchy of an organization has found but there were no unyielding proof for that. Because each and every email has its body or content, an idea of applying semantic analysis to the content came into the picture. Semantic analysis could make the results of statistical analysis more solid and acceptable.

4.2.2 Methodology

The basic idea behind semantic analysis was to find important keywords, sentences or even some symbols that might able to reveal some critical information. For example user A and user B were having lot of communication through an email exchange; in this particular communication there might some keywords or symbols available that can reveal information regarding those users. Each email has its content in the corpus of 517,000 emails; hence the semantic analysis has applied on the content itself rather than headings or message lines.

Another essence behind the semantic analysis was to add more sense to the results of statistical analysis in terms of users. So if semantic analysis can have better results, it can make significant impact on the results of statistical analysis while merging both results together. With the fact that there were lots of communication were going on

between the same level of employees and between the different levels of employees, users should have been asking lot of questions to each other. Taking this fact in mind the semantic analysis has applied to find out “who asked questions to whom and in what amount”? The simplest solution was to mine each and every email's content and figure out how many questions were asked in that particular email. Every time it’s not necessary that an email has '?' in its content so it should be considered to mine that email for further information.

So some criteria has applied in order to find out a perfect sentence which contains the '?' plus the necessary information and words that makes the sentence an interrogatory sentence. Those criteria includes some basic words in an interrogatory sentence followed by the '?'. To implement this structure in programming language, two arrays were used so one of them contained important words of the first part of an interrogatory sentence like how, what, can, could etc. Whereas the second array contained some important words of the second part of an interrogatory sentence like I, you, much, many etc. The function that checks for these conditions is described below for the reference.

```
protected function analyzeMailBody($body){

        $arrSentances = explode('?', $body); $cnt = 0;

        if (count($arrSentances) == 1) return false;

        $arrWords = array('what', 'how', 'when', 'whose', 'should', 'would', 'can', 'could',
              'please', 'is', 'are', 'do', 'did', 'have', 'has', 'was', 'were');

        $arrPronoun = array('i', 'we', 'you', 'they', 'do', 'does', 'are', 'there', 'can',
               'much', 'many', 'this', 'that');

        foreach ($arrSentances as $sentance){
                 if(preg_match('/^[a-zA-Z0-9\$_\-@!#%&\. ]*(' . implode('|', $arrWords) . ')+
                 (' . implode('|', $arrPronoun) . ')+
```

```
                [a-zA-Z0-9 ]*/i', $sentance, $match
            ) == true){
                $cnt++;
            }else{
                continue;
            }
            $match = '';
        }
        if ($cnt >= 1){
            return true;
        }else{
            return false;
        }
}
```

Figure 14: code snippet displaying method to find root

The code snippet above in figure 14 determines whether the content of an email has any interrogatory sentence or not? Two arrays *arrWords* and *arrPronoun* contained necessary words to make an interrogatory sentence. If any sentence with the '?' found then PHP regular expression would check for it. With this analysis every sentence in each and every email has checked for potential interrogatory sentence.

The next step was to determine the number of questions asked by each employee to every other employee. That part was easy because all the questions asked were already there in the database. Those questions were found using the PHP regular expressions and applying the logic for simple interrogatory sentence. So it just needed to find who asked whom and how frequently?

4.2.3 Inspection of Questions

All the questions asked were present but it was required to find the sender and the receiver of that particular email sent in which a particular question asked. Again this was an easy part as in the *header* table all the recipient and sender were stored along with

their message id. So after mapping message id in 'question found' email to the message id in header table, the receiver and sender has identified. After this point, it was required to calculate who asked most questions to whom and who replied back in either very number of time or very frequently. After applying a MYSQL query the number of question sent by a particular sender and the number of question received by particular receiver has determined. Now next step involved was to determine who sent highest number of question and who replied back very less.

The result of this step revealed a lot of significant information that can be associated with the results of statistical analysis. Employee who asked high number of questions belonged to the normal employees or workers. Whereas, employee who replied back very less to the question asked compared to the number of question he/she asked to other belonged to high level of employees on the tree. The figure 15 below explains a basic logic of employee's position on a tree according to the number of question asked.

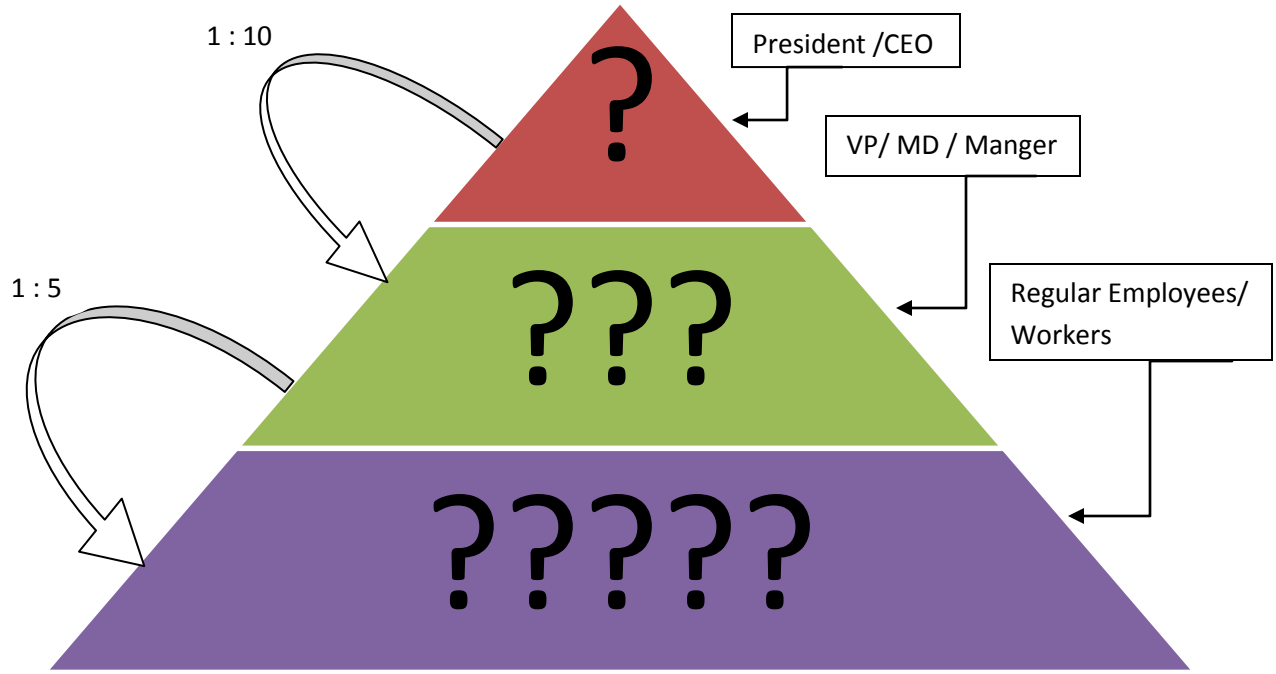

Figure 15: Number of questions asked at each level of hierarchy for the Enron

4.2.4 Integration with the Statistical Analysis

Statistical analysis reveals the partial hierarchy of an organization particularly in three different levels first roots level; second middle level and third the lower level with normal employees and workers. Whereas the semantic analysis reveals the information about "who asked questions to whom"? Having both partial results, the effort of integrating both results has made to gain superior and significant results. The idea was simple, just compare all employees who asked high number of questions with the hierarchy of the statistical analysis and in the same way compare all employees who replied back minimum number of question with different position in the hierarchy.

4.2.5 Result of Semantic Analysis

The results from integration of both types of analysis were very productive. Employees who asked lot of questions were tend to stay in lower part of the tree, that is in lower level, whereas employees who asked relatively very less questions were tend to stay in middle and root level of the tree. Second noteworthy information found after the integration was the ratio of more question asked and less question asked at two different level of the hierarchy. Particularly the ratio found between lower level and middle level was 1:5 whereas the ration between middle level and root level was 1:10. Now what does these numbers in ratio means? A ratio of 1:5 between lower and middle level means, in general, an employee from middle level asked only once in every 5 questions asked from lower level. In another words, from every 5 question asked from lower level employees to the middle level, i.e. their managers and team leaders, only one question asked from middle level employee. Similarly the ratio between middle level and root level was 1:10 means the root employee asked only once of every 10 question asked from the middle level employees. In other words, managers and team leaders asked 10 questions to root level employee and in contrast, only one question asked from root level employees to middle level employees.

The question is why the numbers in the two ratios are important? Those numbers are important because they are directly related to the position/level of employees in the organization hierarchy created from statistical analysis. The figure 15 will clear the picture of ratio and hierarchy. As seen from the figure 15, even though the number of email transferred between the lower level and the middle level was high, and it was high

as 1:5. Whereas the amount of email transferred between the middle level and the root level was less compared to middle and lower level, and the ratio was as less as 1:10. Again these ratio numbers are related to position of employees in the hierarchy. The less ratio of 1:5 itself says that there was heavy question transaction between middle and lower employees whereas the high ratio of 1:10 says there was less question transaction between root and middle employees. This is exactly matches with the results of statistical analysis that higher level employees replied/sent less mails compared to the lower/middle level employees.

4.3 TEMPORAL ANALYSIS

4.3.1 Need for Temporal Analysis

This analysis is independent of previous three analyses. After reading lot of literature of the Enron scandal, the author came across the fact that Enron got bankrupt in December 2001. Aim behind this analysis was to figure out some details that might give some clue for *'Enron Scandal'* or '*Enron Fraud'*.

This was the last analysis of this research work that has performed over the content of an email. The idea behind this analysis was to find out some relationship between employee and how their behavior changes over the time. Main focus was on the lawyer-employee relationship at some particular period of time throughout the time-line of the Enron Inc. In fact there were few lawyers and traders involved in the list of employees. Hence, it might have possible that before the company filed bankruptcy, suddenly some employees have tried to talk to the lawyers with a lot of emails or on other hand some have stopped talking at all. So the idea was to figure out when the first peak occurred in the time-line of email communication with the lawyers.

4.3.2 Methodology

As the email database corpus contains emails from 1998 to 2002, and with the fact that Enron was bankrupted in December 2001, there should be an instance before December 2001 when the graph for communication between lawyer and other employees should have been at peak. In other words, try to find when the first peak occurred in the graph of email communication between the lawyers and other employees. The fact-sheet

that came along with the database contains 3 lawyers in it who served the company. Now the challenge was to figure out when the first peak occurred in graph for communication with lawyers and traders?

The first task to be completed was to find all the lawyers and traders in the organization and check for every other employee who has sent an email to any one of lawyers or traders.

The table *lawyer_relation* has created which kept all the necessary data relates every employee to every lawyer or traders in the form of number of emails sent. So the task of figuring out number of email sent by the lawyers and received by the lawyers in different years has completed first. Definitely this data has proven very useful in further investigation of email communication. The figure 16 below shows such data of two years namely 2000 and 2001. Table contains the number of emails sent/received by the lawyers/traders in year 2000 and 2001.

	2000		**2001**	
	Sent	**Received**	**Sent**	**Received**
Lawyers	72.00%	28.00%	52.00%	48.00%
Traders	62.00%	38.00%	32.00%	68.00%

Figure 16: comparison of sent and received emails with lawyers/traders

The Figure 16 above shows clear distinction between the number of email exchange between employees and lawyers/traders in the year 2000 and 2001. The percentage of emails sent and received has a big difference in both years. In year 2000 lawyers, in general, sent 72% compared to 52% in the year 2001, whereas in year 2000 lawyers received 28% emails compared to 48% in 2001. So there was clearly about 40%

increase of receiving emails to lawyers in year 2001 compared to in year 2000. In other words, employees sent approximately 40% more emails to lawyers in 2001 compared to year 2000.

The same fact is true in case of traders as well. It is clear that number of received email to traders is high in year 2001 compared to year 2000 from the table. In other words, employees have sent more emails (68%) to traders in 2001 compared to year 2000 (38%). With these two facts in mind, it was obvious that something big happened in the year 2001 because more and more employees were talking to lawyers and traders with emails than ever before. So the next step involved was to figure out when particularly in year 2001 that big thing occurred. The chart below in figure 17 shows monthly email transaction of year 2000 and 2001 particularly those were sent to lawyers and traders.

FIGURE 17

The figure 17 describes number of emails sent to either lawyers or traders for particular month of year 2000 and 2001. The two lowest peaks in the chart for each year are January and February that can be counted as vacation period. The highest first peak was in October 2001 and this was the same month when *Securities and Exchange Commission* (SEC) has started an investigation into Enron for possible accounting fraud. So the employees might have started talking with the company lawyers a lot in the month of October 2001 by email exchange. The market share value for Enron went below $1 from 52 weak highest $85 in the month November 2001. In order to compensate the losses of hundreds of millions, Enron eventually filed bankruptcy in December 2001 that can be seen as an email communication with second highest peak after October 2001.

5.0 SOFTWARE AND TOOLS USED

In this research work XAMPP 1.7.2 [29] stack has been used for the development and storage purposes. XAMPP 1.7.2 provides PHP 5.2.9, Apache server 2.2.9 and MYSQL 5.0.67. PHPs regular expression functionality is used in this research for most of the time while creating primary level table and stored them into MYSQL database. An extremely good speed and performance of PHPs regular expression and ease of scripting are the reasons for choosing PHP 5. Apache server provided the required platform to run the application and test cases over localhost on author's PC. Furthermore Net-beans 6.7.1 [30] has used as a scripting platform, which is really good for handling a lot of client libraries and classes in a single package.

For GUI purpose Jpgraph [31] PHP client library has been used which is freely available for commercial and non-commercial usage. In this proposed research work graphs are used to plot the relationship between sender and receiver over the period of 5 years ranging from 1998-2002. The graph shows users email communication frequency separated by each month in 5 years. Last but not the least, Microsoft® word 2007 is used for the report creation and it truly provided the best possible graphics and charts for hierarchy structures.

6.0 EXPERIMENTAL RESULTS AND TEST CASES

6.1 Related experiments:

As mentioned earlier in the report, the topic of email log data mining is relatively new to the research group among researchers and students. There are only a few related experiments that have been done over email log data mining. For example, Ziv Bar-Yossef et al [19] has put some work on email cluster ranking based on a particular organization. Rong Qian, Wei Zhang, Bingru Yang [18] has put their efforts in experiments for community structure detection. Whereas, Anton Timofieiev et al [12] has done similar research over email mining as Rong Qian et al, but Anton Timofieiev has used H-index to determine communities in email corpus. On the other hand Nishith Pathak, Sandeep Mane and Jaideep Srivastava has done socio-gognitive analysis of an email corpus in their paper '*Who thinks Who knows Who?*' [10]. In the paper '*MODELING INTERACTIONS FROM EMAIL COMMUNICATION*' [14] by Dong Zhang et al, the authors have shown the methods to learn the topic based interaction between a pair of email users. Giuseppe Carenini, Raymond T. Ng and Xiaodong Zhou have presented the paper '*Scalable Discovery of Hidden Emails from Large Folders*' [13], and also proposed a method of reconstructing hidden relationships from the communication between the users in an organization.

Although there were quite a few past projects that have been done on the email data mining field in order to determine community detection, hidden emails, topic based interaction and many more, no one has really tried to propose a position wise or ranking wise structure of an organization.

6.2 Experimental results:

In the research work the goal was to create a partial position wise hierarchy of the Enron employee mainly in three divisions. Another goal was to find some relation between the employees using the semantic analysis and temporal analysis. Concerning the earlier goal, the first part was to find the root, i.e. President/CEO of an organization. The second phase finds immediate second level employees to the root, and those include vice president, director, managing director etc. Final phase finds the lower level employees, in other words a regular employees that might be listed in fact sheet.

Together with the Enron dataset, an excel sheet has received that shows the actual ranking of senior level management employees of Enron before it filed bankruptcy. So in this research work as each phase results were obtained, they were compared with original results and then kept going ahead. Below mentioned figures 18, 19 and 20 which describe author's experimental results of all 3 phase accordingly. Figure 18 shows result of first phase, i.e. finding root node, figure 19 shows the combined result of first phase and second phase to find VPs, MDs and directors. Figure 20 shows the complete hierarchy structure along with phase 3 result involved to find lower level employees.

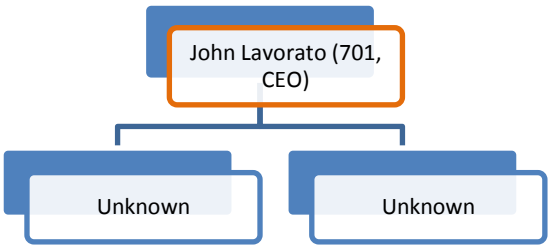

Figure 18: Proposed root node

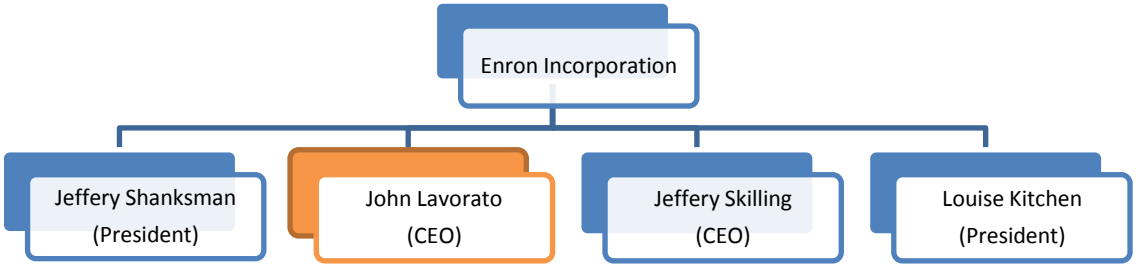

Figure 19: Proposed 2nd level nodes

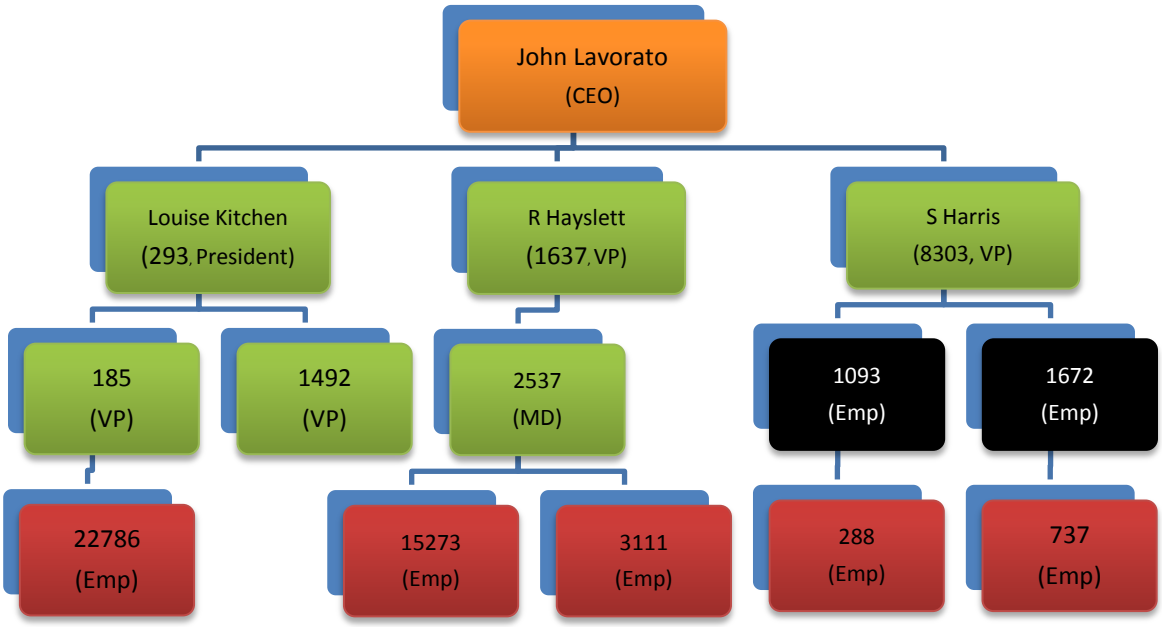

Figure 20: Final hierarchy with exception nodes in black box

All of the above displayed figures show the correct, exceptions and noise node all over the hierarchy. In both phase 1 and phase 2, a success ratio of 100% was obtained;

that means the root and second level nodes were correctly predicted. However, in the third phase few exceptions at node 1093 and node 1672 were obtained. Below is given the explanation of these exceptions in section 6.3.

Semantic analysis was intended to get inter relation among employees. The case of interrogatory sentences proved that the higher-level employees were asked very few questions compared to middle level, and middle level employees asked fewer questions compared to the lower level employees. As mentioned previously in case of S-R analysis and statistical analysis there were two exceptions, those will be mentioned in next section, but with semantic analysis there was no case of exception because the average ratio for questions asked between lower level and middle level employees was 1:5 whereas it was 1:10 for root level and middle level employees. So these ratios were persistent throughout all the employees. To summarize, by merging the result of semantic analysis with the result of S-R analysis more significant and accurate results have been achieved compared to just S-R analysis.

The combined results of S-R analysis and Statistical analysis were 86% accurate. Somehow remaining 14% were not correct results because of the presence of exception nodes. One essence behind semantic analysis was to improve the accuracy of the previous results.

Last but not the least, the result of temporal analysis was also significantly fruitful. By analyzing the email exchange pattern between employees with lawyers and traders, it was possible to find a major event in an organization that lead the Enron to bankruptcy. Using temporal analysis it was possible to find the particular month of

October 2001, in which the news made public for major fraud scandal going on in the organization. The figure 21 below explains ratios as each level.

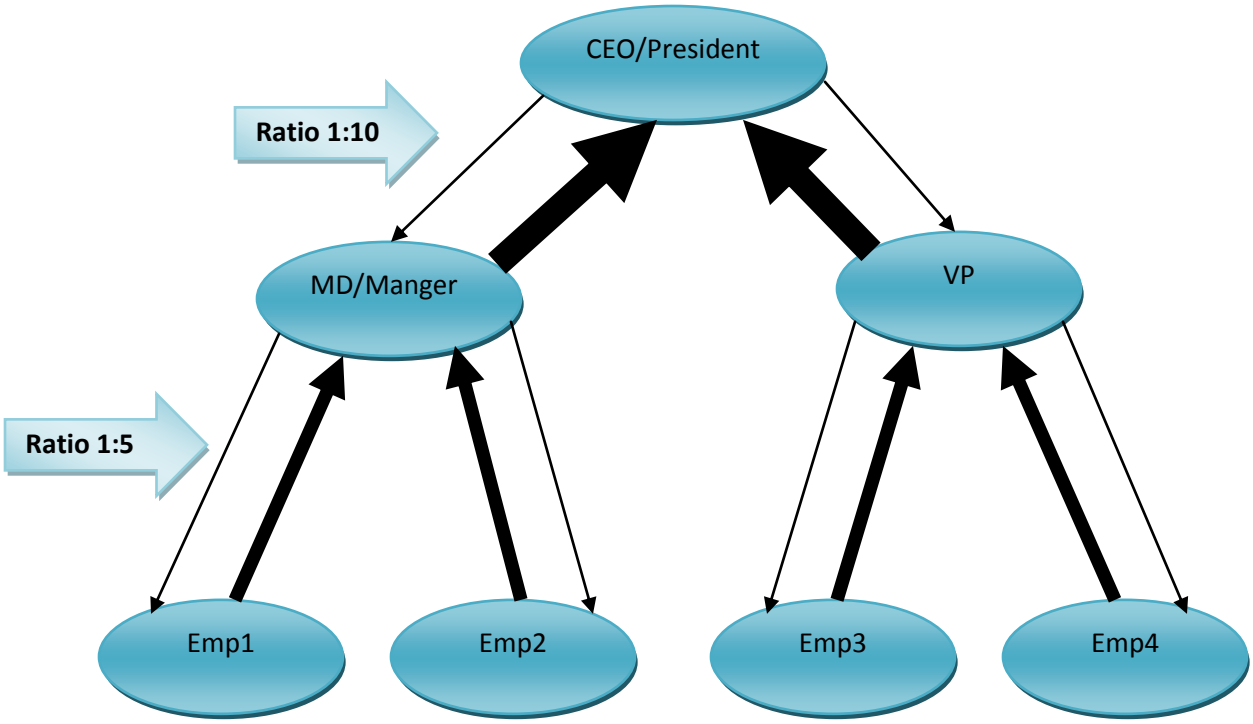

Figure 21: Ratio of question asked at each level

6.3 Test cases:

In the construction of ranking wise hierarchy of an organization some nodes that should not be there at that position were plotted. Most noticeable nodes were node 1093 and node 1672 in the 2nd level. Below is the description and reason behind these two test cases.

6.3.1 Case 1: *node 1093 shouldn't be at top level in 2nd phase*

From the figure 26 above, we can see the node 1093 is directly below the VP node 8303. So what does that mean? What is the relation between these two nodes? Simply that means the node which is a children of VP node 8303 should be VP, MD, director or any second level employee in the hierarchy as per the algorithm, but it is not! In fact from statistical analysis it was clear that node 1093 was a normal employee, and it should resides on lower part of the hierarchy.

From the table *mailgraph* lookup has made to find the details. It made clear that the number of messages sent by 8303 were very few, in fact just 1, compared to the number of messages he received from node 1093, and that was 33. So what do those numbers suggest? Well in that case the table s*ub_response_send* and *response* were not useful enough, because they were already used to locate those two nodes. Some analysis has to be done over the period of time due to this, or in other words it was required to compare the communication in terms of function of time. For timely inspection a table called *timeline*, was created which stores the monthly communication between any two employees over the period of 5 years. When the lookup was made for transaction from node 8303 to node 1093, just one email sent found! On the other side when the lookup was made for transaction from node 1093 to node 8303 total of 33 emails were found. That was a huge gap between those two nodes.

From the timeline table it was found that node 1093 sent 33 emails to 8303 over a period of 5 years. The interesting fact is the node 1093 was only active for 4 months out of 5 years. Does that make any difference? Yes, because the other calculations for node

8303 says that he only sent 7 messages over the period of 5 months starting from January 2001 to May 2001. This fact can also be obtained from *mailgraph* matrix that was created previously. Further analysis of node 8303 showed that he only sent 7 messages to total of 4 employees and out of those four we already got two node 1093 and node 1672. The other two nodes were not identified in the fact sheet. Moreover when the lookup for node 1093 in *mailgraph* was made, it was found that the node 1093 has sent a lot of emails to a lot of other employee, this shows that node 1093 was working with company for long time compared to the node 8303.

From the *timeline* table, it was clear that the node 1093 worked for Enron from June 2000 to January 2002, which was after bankruptcy. To put in the nutshell, VP node 8303 contacted very few people for little time with few emails, literally just 7. So it might be possible that node 8303 has sent one email directly to node 1093, and node 1093 has replied it. So due to less distribution of email among the user for long time, the node 1093 has been taken as a second level employee but not lower level employee.

6.3.2 Case 2: *node 1672 shouldn't be at top level in 2nd phase*

This case was very shocking as it says that node 1672, is at 2nd level. This is exactly the same problem as case 1. From the final proposed hierarchy it can be seen that both nodes 1672 and 1093 are at the same level below node 8303. Interesting thing found was both were elected as an Employee node at 2nd level. In the case 1, I described why the node 1093 is at 2nd level of hierarchy. Same procedure was followed for node 1672 and the same reason found as of case 1.

So from both of these test cases, it was proven that both exception nodes were misplaced because of node 8303. The node 8303 was VP of the company and he was active for very short time period, literally 4 months. In his position as a Vice President, he had sent only 7 emails in 4 months and out of those 7 emails one email sent to 1903 and 1672 each. And again *timeline* table shows that both node 1903 and 1672 were actively involved in email reply. Hence, the algorithm chooses the node 8303 as a parent of both node 1903 and 1672.

7.0 CONCLUSION

After a threat to an organization in terms of accounting fraud in recent years, the future decision making and study of employee relationships within an organization becomes more focused on an email data mining and analysis [32]. One reason for increasing trends toward an email analysis and mining is that, email becomes a central tool for an information exchange at most of the levels in a professional hierarchy. This paper presents an analysis of email log files that gives position-wise organization hierarchy as an output and highlights behavioral-relations among the employees. From the research work, a conclusion is made that, it is possible to carve out the ranking of employees and an organization structure based just on server email log files. This kind of research can be helpful to figure out a question like “who was who”?, who was an expert within the group, what happened in a given particular span of time, progress difference between two users as a function of time, behavioral study of an employee and even future decision making.

8.0 FUTURE WORK

The research work included four types of analysis: first, send-receive (s-r analysis), second statistical analysis, third semantic analysis over email content and fourth temporal analysis. The current research work has involved mainly statistics and analyzed only the Enron dataset as of now, but future research is intended to work with any kind of email log dataset.

The test case in this research exhibits exceptions and noise nodes in the hierarchy; those should not be there. As part of future work, the project could improve the exception and noise by performing high level semantic analysis together with time series as a function.

As Enron has filed bankruptcy in late 2001 due to systematic and intentional account fraud at higher level management, I would like to find out the cause of that kind of fraud. Who were the employees who may have started some email that contains suspicious keywords or some encoded codename? Moreover, it is also possible to cluster out different teams within Enron, based on email communication. If it is possible to differentiate the teams and their team members, then it is also possible to predict the inter-team and intra-team employee relations that would help management employees to see the future relation among the teams and among the employees.

This research work has found some important statistics including ratio of email sent between two levels of an employee hierarchy. Ratios 1:10 and 1:5 were discovered for top-to-middle level and middle-to-lower level respectively. These ratios were obtained solely from the Enron email database, and only applied on Enron employees.

As a future work in this particular issue, I would like to combine management theory, if available, with the ratios discovered from this research work. Applying these ratios to other organizations' email log would give some fruitful information. If the result of this analysis matches with any available management theory of employee relationships from emails, it would be a great discovery to figure out email communication at each level of hierarchy.

REFERENCES

[1]. Hongjun Li, Jiangang Zhang, Haibo Wang and Shaoming Huang, "*Mining Algorithm for Email's Relationships Based On Neural Networks*". 2008 International Conference on Computer Science and Software Engineering

[2]. Christian Bird, Alex Gourley, Prem Devanbu, Michael Gertz and Anand Swaminathan, "*Mining Email Social Network*s". (May 22–23, 2006). Dept. of Computer Science, University of California, Davis.

[3]. R. Agrawal, S. Rajagopalan, R. Srikant, and Y. Xu. "*Mining newsgroups using networks arising from social Behavior*". (2003).

[4]. Shlomo Hershkop, Ke Wang, Weijen Lee, Olivier Nimeskern, Germán Creamer, and Ryan Rowe, "*Email Mining Toolkit Technical Manual*". Department of Computer Science Columbia University, Version 3.6.8 – June 2006.

[5]. Bron, C. and J. Kerbosch. "*Algorithm 457: Finding all cliques of an undirected graph.*" In *Comm ACM,* vol. 16, pg. 575-577, 1973.

[6]. O. de Vel (Information Technology Division Defence Science and Technology Organisation) and A. Anderson, M. Corney and G. Mohay (Queensland University of Technology), "*Mining Email Content for Author Identification Forensics*".

[7]. A. Anderson, M. Corney, O. de Vel, and G. Mohay."*Identifying the Authors of Suspect E-mail*". Communications of the ACM, 2001.

[8]. C. Apte, F. Damerau, and S. Weiss, "*Text mining with decision rules and decision trees*". In Workshop on Learning from text and the Web, Conference on Automated Learning and Discovery, 1998.

[9]. W. Cohen. "*Learning rules that classify e-mail*". In Proc. Machine Learning in Information Access: AAAI Spring Symposium (SS-96-05), pages 1825, 1996.

[10].Nishith Pathak, Sandeep Mane and Jaideep Srivastava, "*Who thinks who is who? Socio-cognitive analysis of email networks*". Sixth international conference of Data Mining (ICDM '06).

[11].Ding Zhou et al and Ya Zhang, "*Towards Discovering Organizational Structure from Email Corpus"*. Fourth International Conference on Machine Learning and Application (ICMLA '05).

[12].Anton Timofieiev, Vaclav Snasel and Jiri Dvorsky, "*Social communities detection in Enron Corpus using h-Index*". Dept. of Computer Science, VSB – Technical University of Ostrava.

[13].Giuseppe Carenini, Raymond T. Ng and Xiaodong Zhou , "*Scalable Discovery of Hidden Emails from Large Folders*". Department of Computer Science, University of British Columbia, Canada.

[14].Dong Zhang et al, "*MODELING INTERACTIONS FROM EMAIL COMMUNICATION*". ICME 2006.

[15]. Anton Timofieiev, Vaclav Snasel and Jiri Dvorsky, "*H-Index Analysis of Enron Corpus*". Dept. of Computer Science, VSB – Technical University of Ostrava.

[16]. Garnett Wilson and Wolfgang Banzhaf, "*Discovery of Email Communication Networks from the Enron Corpus with a Genetic Algorithm using Social Network Analysis*". 2009 IEEE Congress on Evolutionary Computation (CEC 2009).

[17].Hung-Ching Chen el al, "*Discover The Power of Social and Hidden Curriculum to Decision Making: Experiments with Enron Email and Movie Newsgroups*". Sixth International Conference on Machine Learning and Applications.

[18]. Rong Qian, Wei Zhang ,Bingru Yang, "*Detect community structure from the Enron Email Corpus Based on Link Mining*". Sixth International Conference on Intelligent Systems Design and Applications (ISDA'06).

[19].Ziv Bar-Yossef et al, "*Cluster Ranking with an Application to Mining Mailbox Networks*". Sixth International Conference on Data Mining (ICDM'06).

[20].Deepak P, Dinesh Garg and Virendra K Varshney, "*Analysis of Enron Email Threads and Quantification of Employee Responsiveness*". IBM India Research Lab, Bangalore - 560 071, India.

[21].Hua Li et al, "*Adding Semantics to Email Clustering*". Sixth International Conference on Data Mining (ICDM'06).

[22].Einat Minkov and William W. Cohen. "*Learning to Rank Typed Graph Walks: Local and Global Approaches*". Joint 9th WEBKDD and 1st SNA-KDD Workshop '07 (WebKDD/SNAKDD '07), August 12, 2007, San Jose, California, USA.

[23].Salvatore J Stolfo et al, "*Behavior Profiling of Email*". Columbia University, New York, NY 10027, USA.

[24].Salvatore J Stolfo et al, "*A Behavior-based Approach To Securing Email Systems*". 450 Computer Science Building, Columbia University, USA.

[25].http://en.wikipedia.org/wiki/Main_Page - A Wikipedia reference.

[26]. http://www.cs.cmu.edu/~enron/ - Enron dataset source

[27]. http://libguides.sjsu.edu/a-z - The SJPL library database

[28]. Shlomo Hershkop et al, "*Email Mining Toolkit Technical Manual*". Version 3.6.8 – June 2006

[29]. http://www.apachefriends.org/en/xampp-windows.html, XAMPP source.

[30]. http://netbeans.org/downloads/indexC.html, Netbeans source.

[31]. http://www.aditus.nu/jpgraph/jpdownload.php, JpGraph library for PHP5.

[32]. http://en.wikipedia.org/wiki/Enron_scandal