Fourth International Conference on Intelligent Computing and Information Systems
March 19-22, 2009, Cairo, Egypt

Semantic Email using Dominant Meaning Model

Marwa A. Moniem
Information Systems Department
Faculty of Computers and Information
Cairo University, Giza, Egypt
marwaaabdelmoniem@yahoo.com

Mohammad A. Razek
Math. and Computer Science Department
Faculty of Science
Azhar University, Cairo, Egypt
abdelram@azhar.edu.eg

Galal H. Galal-Edeen
Information Systems Department
Faculty of Computers and Information
Cairo University, Giza, Egypt
Galal@acm.org

ABSTRACT
Email is one of the most widely web applications used in communicating between people all over the world. For its great importance new devices appeared like cellular phones and Black-Barry to enable its users to access their mails remotely. The key research question of this paper is how to tackle the complex issues related to building semantic email system that care, ranging from representing knowledge and context to modeling email concepts. This requires linking theory and technology from artificial intelligence, and computer science with theory and practice from linguistics. Semantic email is a challenging search point that has been adapted by many researchers nearly from the start of year 2000. Previous approach introduced methods for summarizing email threads and conversations not each email message as a single unit. This paper applies a new approach called Dominant Meaning Model. It can act like a meaning based dictionary to create RDF XML file required to define semantic approach. This approach has been used to handle emails concepts.

Categories and Subject Descriptors
H.2.8 [**Database applications**]: [Data Mining]

General Terms
Algorithms, Experimentation.

Keywords
Dominant Meaning Model, Summarizing Email

1. INTRODUCTION
Today; Email became indispensable as most of the people daily tasks and activity may vitally depends on some information exists in an email content, that is why the new technology comes out with the new devices that can access the email remotely. Providing email summary has been a great promise to help users extract their crucial information at a glance without having to go through the entire email content. Summarizing email is a very challenging area for summarizing a free form short text sent for any purpose involving many users. Emails could be for information spreading, requests for action in a business or social situation. The most famous use for the email is the last type where an initial email is sent about a specific issue and groups of messages respond to that initial email constructing conversations like structure. Almost all the previous attempts for summarizing email adapt the idea for providing a summary for the dialogue threads exists inside users emails. These approaches put the assumption that only those kinds of emails worthy summary from the email users' point of view. In real life this is not often the case, any normal email user could receive a single message that is not included in any email thread and contains crucial information for his daily actions.

Recently, many researchers adapted the semantic web idea to store the web contents in a hierarchical structure based on the relationships between concepts for better data storage and exchange. Dominant Meaning Model (DMM) is relying on the same idea except we are trying to represent the information in a hierarchal structure depending on the meaning relationships of concepts exist in a domain.

In this paper, we propose our approach for summarizing each coming email message using Dominant Meaning Model. This Model is based on creating a dictionary like system but it stores the words in a specific hierarchal structure relying on the meaning relations between the words. By training this dictionary for specific domain or topic, we can use it to extract the most dominant meaning sentences from an email classified under that topic and consider those sentences as the email summary.

This paper is organized as follows: section 2, introduces the related work for summarizing email dialogues. Section 3, introduces in more detail DMM approach. Section 4, presents our over all system architecture. Section 5, presents system implementation, and results using Enron email corpus. Section 6, concludes our work and suggest future work

2. RELATED WORK
Since the fifteenths, numerous methods have been developed for summarizing documents. Those methods vary from statistical scoring methods, artificial intelligent methods and semantic

methods. Nearly from the start of year 2001, approaches appeared trying to apply the same concept on email messages. Almost all of these approaches focus on summarizing email conversation where there exist a start original message about specific issue and subsequent.

Smaranda Muresan [1] used Machine Learning-*ML* techniques based on linguistic features to train a system for finding salient noun phrases that can represent the summary of an email message.

Derk Lam [2] improved an existing system that designed to summarizing single text documents software. To summarize email threads, they changed the existing system through exploiting two aspects of email, *thread reply chains,* and *commonly-found features*. The system pre-processes email messages using heuristics to remove e-mail signatures, header fields, and quoted text from parent messages. Also the system presents a heuristics-based approach to identify and report names, dates, and companies found in e-mail messages.

Ani Nekova [3] introduced efficient email threads representation that allows a user to decide which threads to read without browsing the actual content of the thread. They call this a thread overview and it consists of an extractive summary for the documents at the first two levels of the discussion thread tree. The overviews are relatively short and the user can skip through them in order to find threads of interest.

Owen Rambow [4] introduced an approach named sentence extraction where important sentences are extracted from the thread and are composed into a summary. They suppose that certain email-specific features can help in identifying relevant sentences for extraction.

Stephan Wan and Kathy McKeown [5] targeted the emails of the ongoing discussions which appear in a consensus in a decision-making process. The proposed summary provides a snapshot of the current state-of-affairs of the discussion and facilitates a speedy response from the user, who might be crucial in some matter being resolved. They introduce a method which uses the structure of the thread and word vector techniques to determine which sentence in the thread should be extracted as the main issue.

Jen-Yaun Yeh and Aaron Harnly [6] studied how to use quotation matching to reassemble email threads

Giuseppe Carenini [7] introduced a fragment quotation graph to try to capture an email conversation including the hidden emails. They used clue words scores to measure the importance of sentences in email conversation and produce a summary of any length as requested by the user

3. DOMINANT MEANING MODEL

The dominant meaning is "The set of keywords that best fit an intended meaning of a target word" [8], where the target meaning is called the master word along with slave words that clarify the target meaning. For example, suppose that the word “Java.” The word “Java” has three well-known meanings: Java (computer program language), Java (coffee), and Java (Island). Because the target meaning is “computer program language” we look for slave words in the document that best fit this specific meaning – words such as “computer”, “program”, “awt”, “application”, and “swing”. [9]

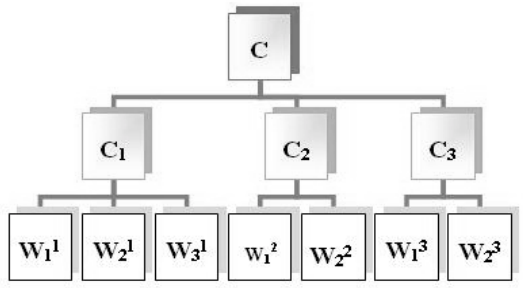


Figure 1: Dominant Meaning Graph (DMG) [10]

As shown in Figure 1, the hierarchical structure for the dominant dictionary is composed of main master word C and its related C_1, C_2, C_3 concepts-*salves* which in turn have their own sub concepts W_1^1, W_2^1, W_3^1 and so on. For building the dominant meaning dictionary, consider the following algorithm [10]:

Suppose having main topic or concept C and we want to extract its slave words from a set of documents D_v where each document is represented by a finite set of words W_{jv}

1. Calculate the frequency values of each word W_{jv} occurs in document D_v

2. Calculate the maximum frequency of main Concept C
$$F_w = \text{Max}\{C_v\}$$

3. Choose the maximum frequency of each word W_j appeared in document D_v
$$F_w = \text{Max}\{W_{jv}\}$$

4. Choose F_c, which satisfies $0 \leq F_w \leq F_c$

5. Now, consider the dominant meaning probability:

$$P_j = P(W_j \mid C) = \frac{1}{v}\left[\sum_{v=1}^{v} \frac{w_{jv}}{F_c}\right], \quad j = 1,\ldots, v$$

So we have from the probability equation

$$0 \leq P_j(w_j \mid C) \leq 1 .$$

Finally we rank the words collection probabilities $\{P_1, P_2, P_{3,\ldots}\}$ in decreasing order. As a result the dominant meaning of the concept C can be represented by the set of words that is corresponds to the set $\{P_1, P_2, P_3, \ldots, P_n\}$ That is $C = \{W_1, W_2, W_3, \ldots, W_j\}$

4. SYSTEM ARCHITECTURE

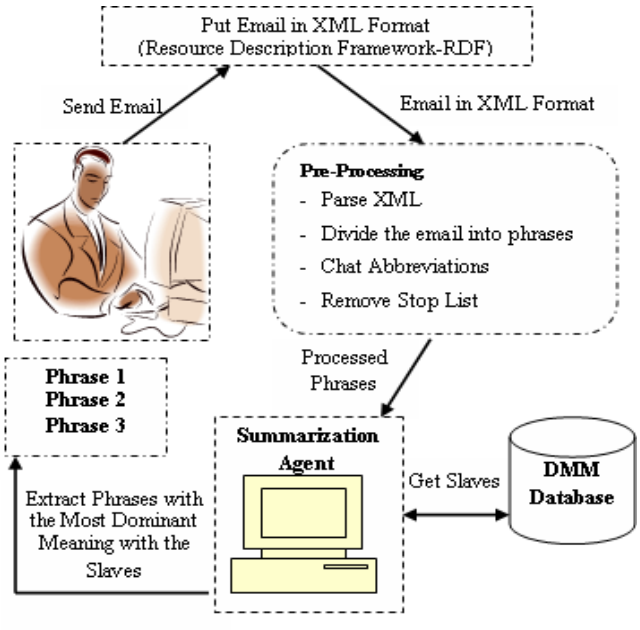


Figure 2: Email System Architecture

After building the dominant meaning dictionary or database that holds our specific domain and its slaves, our system is ready for any incoming email from the user to get its summary.

Each incoming email passes through three phases

1. *Pre-processing*: Each incoming email is preprocessed
    a. Parse XML to extract the email content
    b. Dividing the email content into phrases using regular expressions.
    c. Convert any existing abbreviation to its original using InternetLingoDictionary [11]
    d. Remove the stop words list
2. *Calculate email phrases probability:* Each processed phrases is calculated its probability with the top *n* slaves C = {W_1, W_2, W_3, …, Wn} that can most represent our domain. The probability equation is

$$P_i = P(W_j \mid C) = \frac{1}{n}\left[\sum_{n=1}^{n} \frac{w_j}{F_c}\right], \quad j = 1,\ldots, n \text{ [12]}$$

Where P is the probability of phrase i, F_c is the master word C frequency calculated in the building dictionary stage and W_j is the frequency for the slave W_j in that phrase

3. Extracting the summary phrases
    a. If number of phrases in the email less than or equal 2 return them
    b. Else If all the phrases probabilities equal zero then this email is not related to the specified domain then return first 3 phrases (baseline where n=3)
    c. Else return the 30% of phrases that have the highest probabilities. If two sentences having the same probability, then return the phrase with the least index.

5. SYSTEM IMPLEMENTATAION

In our implementation we used Microsoft Visual Studio Dot Net 2005, SQL Server 2005 and, XML technologies.

For our data, we used the Enron email corpus which is a collection of hundreds of thousands of email messages from the infamous Enron Corporation that researchers have been using to improve and evaluate techniques for analyzing email [13]. This corpus was first presented in March 2004. Many researchers modified the original data and released it. In 2007, Ted Pederson released the annotated release which seemed to be a better choice for our system as it is classified more by topics. [14]

Since almost all the of the Enron emails are classified under general topic that describes the emails' content and does not have any existence in the email content itself, we found that the Enron announcement category is the most suitable category for our experimentation and we selected the master word Enron as the most suitable word describing them. We used 226 email messages under 'main-noheader/general_announcements/misc' classification as a training data to build the dominant dictionary database for the master word Enron. Only 131 messages were accepted with maximum frequency for the master word 55 and 95 messages were rejected by our system for not containing the master word in it. For testing we used 40 messages under 'noheader/general_announcements/news' for testing.

In the following steps we will describe how our system works step by step from building the dictionary up to generating the summary for the tested messages:

1. Putting the email messages for both the training and the testing data in XML (RDF) Format, where each email contains from *Message-ID*, *Date*, *From*, *To*, *CC*, *Bcc*, *Subject*, *Content* tags as shown in the figure below

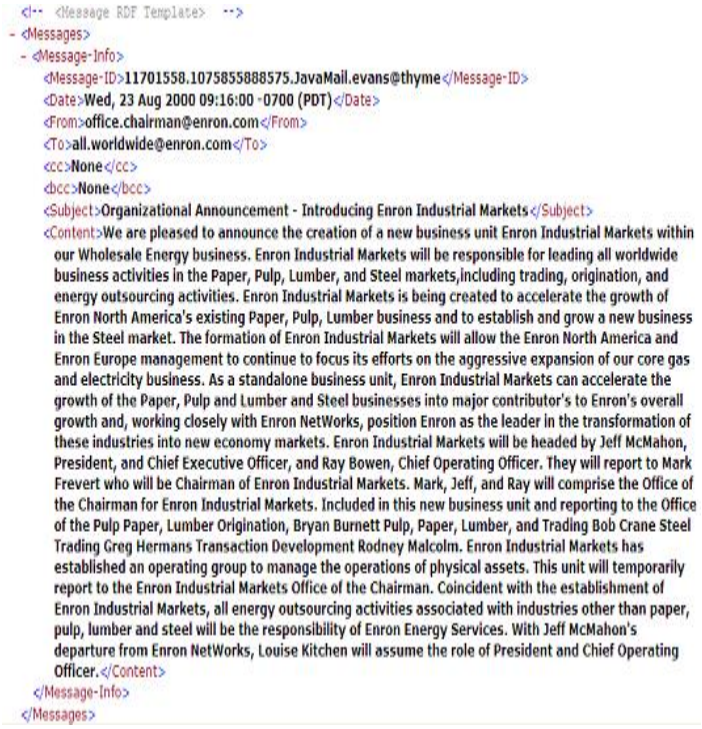


Figure 3: Email XML Format (RDF)

2. In the Building Dictionary step, for each email message in the training collection our system takes the above XML file and parses it to extract its content tag and calculate the Enron existence along with all other words after removing the stop words and replacing any existing abbreviations to its original source as shown in the figure below.

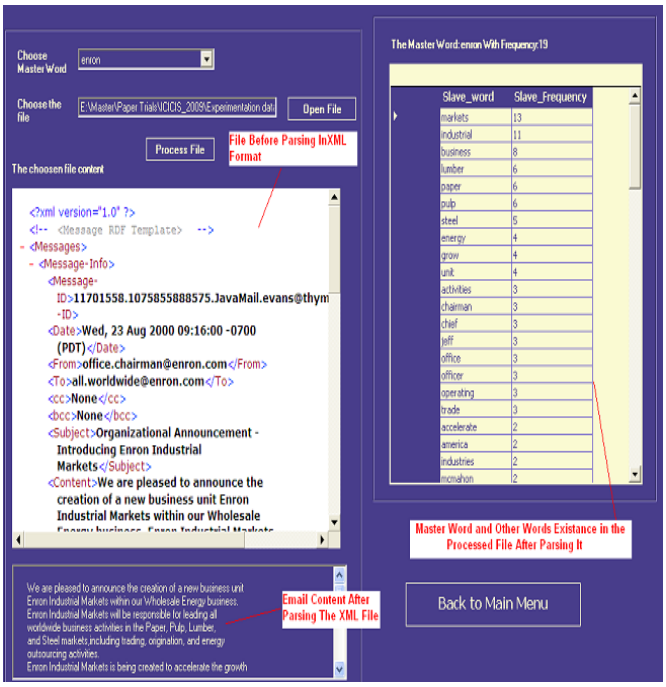
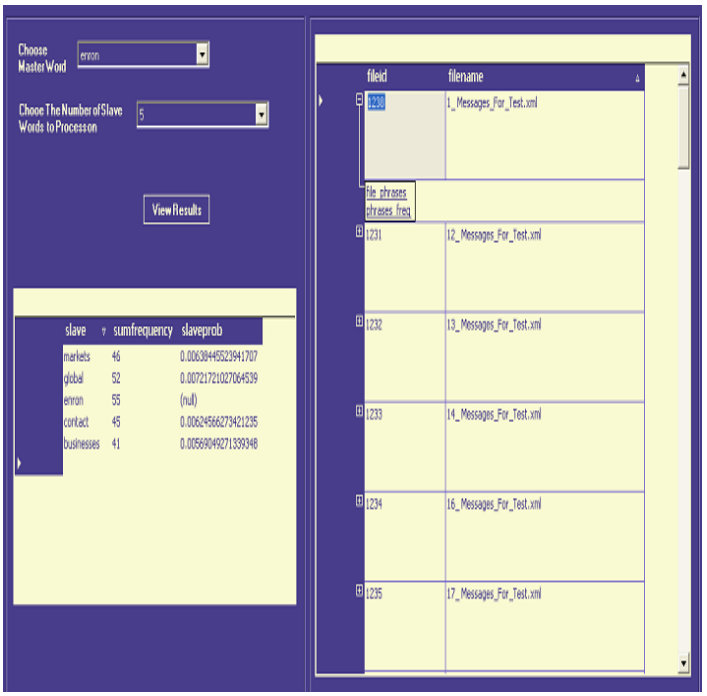


Figure 4: Building System Dictionary

3. After building the system dictionary, we tested our work for each of the 40 emails in the testing collection. We parse the XML tested email, divide it into phrases, and calculate each phrase probability based on the selected top n slaves that best describe the master word. We experimented our system three times; each time with different number (n) of top slaves (i.e. 5, 10, 15) representing the master word to test the system performance. The below figures shows how the system test the file and test Email example and its resulted summary



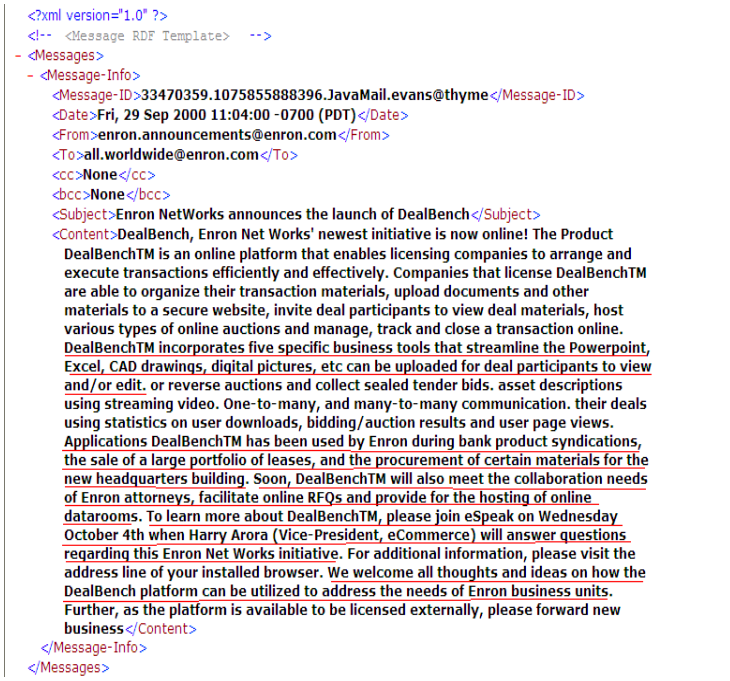


Figure 6: Email Test Sample and its Resulted Summary Showed by Red Underline

6. EVALUATION

Writing summaries differs from one human to another; each will extract from his/her point of view the most crucial information representing the document being summarized. That is why we used the human factor to evaluate our work. Two human summarizers were asked to extract 30 % phrases from the original message that best present the message summary. We also compared our work with the baseline algorithms that consider the first *n* sentences as the email summary. We compared our work with baseline n=1 and n=2.

Table 1. DMM Results Compared with Baseline n=1 and Baseline n=2 Approaches

Method	Human Summarizer	Average Precision	Average Recall
DMM (5 Slaves)	First Summarizer	50.60%	61.15%
	Second Summarizer	46.21%	51.49%
DMM (10 Slaves)	First Summarizer	49.90%	60.08%
	Second Summarizer	45.52%	51.11%
DMM (15 Slaves)	First Summarizer	45.46%	53.52%
	Second Summarizer	48.29%	51.53%
Baseline n=1	First Summarizer	21.93%	26.06%
	Second Summarizer	43.47%	52.55%
Baseline n=2	First Summarizer	17.97%	21.81%
	Second Summarizer	25.70%	29.79%

To evaluate our system we used the Precision and the Recall measures; where the Precision represents the fraction of the phrases retrieved that were correct to the user's information need and the Recall represents the fraction of the phrases that were correct to the total retrieved phrases by the proposed system.

$$\text{Recall of an Email} = \frac{\text{Number of Correct Phrases Retrieved by the Proposed System}}{\text{Total Number of Phrases Retrieved by the summarizer}}$$

$$\text{Precession of an Email} = \frac{\text{Number of Correct Phrases Retrieved by the Proposed System}}{\text{Total Number of Phrases Retrieved by the Proposed System}}$$

Based on the results presented in Table 1, our system proved much better performance than the baseline approaches even in its worst performance when we took 15 top slaves. The main goal for experiment different number (n) of top slaves was for testing how the system will perform when we increase the number of slaves representing the master word. Although we thought the more the number of slaves, the better summary we get, the experimentations proved otherwise based on the results illustrated in the above table. We may explain that by the more slaves we take the more we go far from the main required topic

7. CONCLUSION AND FUTURE WORK

This paper presents a new technique called Dominant Meaning Model to help in semantic Email. This approach creates RDF XML based on a dominant meaning dictionary rather than a bag of word. This contribution could assist new devices such as Pocket PC, Hand Held Computer, cellular phones, and Black-Barry in accessing emails and related tools. This method extracts the most dominant sentences in an email as email brief summary. We compared our approach with the baseline algorithms and the results showed that DMM proved its high performance without using complicated data mining techniques.

This paper presents the best sentences in incoming email which can represent the whole idea of it. For future work, we need to study further the depth for the dominant meaning dictionary, and to study more the bases of threshold for choosing the top slaves to best represent a topic by using more humans in our evaluation. These studies could help in presenting the whole idea of the email in new sentences.

8. REFERENCES

[1] Muresan, S., Tzoukermann, E., and Klavans, J.L. Combining Linguistic and Machine Learning Techniques for Email Summarization. In *Proceeding of the CoNLL Workshop at ACL-EACL* 2001.

[2] Lam, D., Rohall, S., Schmandt C., and Stern, M. Exploiting e-mail structure to improve summarization. In *ACM Conference on Computer Supported Cooperative Work (CSCW2002)*. Interactive Posters, New Orleans, LA, 2002.

[3] Nenkova, A., and Bagga, A. Facilitating email thread access by extractive summary generation. In *Proceedings of RANLP*. Bulgaria, 2003.

[4] Rambow, O., Shrestha, L., Chen, J., and Laurdisen, C. Summarizing Email Threads. In *the Proceedings of HLT-NAACL 2004*: Short Papers, 2004.

[5] Wan, S., and McKeown, K. Generating overview summaries of ongoing email thread discussions. In *Proceedings of COLING'04*, the 20th International Conference on Computational Linguistics, August 23–27, 2004.

[6] Harnly, A., and Yeh, J. Email thread reassembly using similarity matching. In *Third Conference on Email and Anti-Spam (CEAS)*, July 27 - 28 2006.

[7] Carenini, G., Ng, R., and Zhou, X. Summarizing email conversations with clue words. In *Proceedings of the*

Sixteenth International World Wide Web Conference (WWW2007), 2007.

[8] Razek, M., Frasson, C., and Kaltenbatch, M. *Context-Based Information Agent for Supporting Intelligent Distance Learning Environment*. The Twelfth International World Wide Web Conference, 20-24 May, Bundapest, HUNGARY, 2003.

[9] Razek, M., and Frasson, C. *Dominant-Model Approach for Web Information*. International Conference in Artificial Intelligence, Las Vegas, USA, June 2005.

[10] Razek, M., Frasson, C., and Kaltenbatch, M. (2004) *Dominant Meanings towards Individualized Web Search for Learning Environments* In: Magoulas, D., and Sherry, Y. Advances in Web-based Education: Personalized Learning Environments. USA: IDEA Group Publishing. P46-69.

[11] Wasden, L., "Attorney General." November 2006 (Accessed June 21, 2008) <http://www2.state.id.us/ag/protecteens/InternetLingoDictionary.pdf>

[12] Razek, M., Frasson, C., and Kaltenbatch, M. *Dominant Meanings Classification Model for Web Information*. Third International Conference on Hybrid Intelligent Systems (HIS'03), 14 - 17 December 2003b Melbourne, Australia, 2003.

[13] Cohen, W., CALD, C., "Enron Email Dataset." Apr 2005 <http://www-2.cs.cmu.edu/~enron/>

[14] Pedersen, T., Padye, A., "Ted Pedersen - Enron Email Corpus by Topic." March 2006 <http://www.d.umn.edu/~tpederse/enron.html>