International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 3 Issue: 1 136 - 139

Auto-Summarization of Email Document Using Statistical Approach

Swati Thakare
Computer Department
BVCOE & RI
Nashik, India.
thakareswati@gmail.com

Shital Kunzarkar
Computer Department
BVCOE & RI
Nashik, India.
kunzarkarshital@yahoo.com

Sonal Chaudhari
Computer Department
BVCOE & RI
Nashik, India.
chaudhari.sonal89@gmail.com

Pranali Bhadane
Computer Department
BVCOE & RI
Nashik, India.
pranali20.patil@gmail.com

***Abstract*—** In today's world email communication is growing with fast rate. Thousands emails are coming into user inbox and he is not able to read each and every email documents. so users need such system by which they can classify and summarize emails. This paper discusses an email auto summarization system using statistical approach which uses vector space algorithm. It generates a summary for an email document from mail Inbox of email server. Our system is based on identification and extraction of important sentences from the input email document file. We listed a set of features that we collect as part of summary generation process. These features were stored using vector representation model. We defined a ranking function which ranks each sentence as a linear combination of the sentence features. We also discussed about techniques to achieve coherent and readable summaries of email message documents. The proposed system showed that the extraction based and position based email document summary generated is coherent the selected features are really helpful in extracting the important information from email documents. That generated summary will be displayed at client application interface. On that summary base we can classify emails.

Keywords- *automatic email summarization, summary extraction, position based algorithm, format based algorithm..*

*****

I. INTRODUCTION

Usage of emails are very high and is still growing fast. While email servers were developed to deal with this huge volume of documents, but users not able to get relevant document or information what actually he needs. So this became very difficult for the user to find important email document from inbox or the document he actually needs, because most of the naive users are reluctant to make the cumbersome effort of going through each of the documents. Therefore systems that can automatically summarize each email available in inbox are becoming increasingly desirable.

A summary can be defined as a short version of text that is produced from one or more texts. Automatic email document summarization is to use automatic mechanism to produce a finer version for an email documents from mail server [12] discussed several ways to classify summaries.

The following factors are important for email document summarization.
Input factors: Email content and attachments, size of email body, genre.
Purpose factors: User groups, purpose of summarization.
Output factors: Summary text will be display after subject line.

Summarizes can be classified into different types based on dimensions, genre, and context.

Dimensions: Single vs. Multi-document summarization
Genre: outlines, minutes, Headlines etc.
Context: Generic, Query specific summaries

As pointed out in [8][9] summaries can be classified in to extracts (most relevant sentences are selected from the text), and abstracts (text is analyzed, a conceptual representation is provided which in turn is used to generate sentences that form summary).

We are using email document for processing, and given email document will be summarized according to generic context.

II. LITERATURE SURVEY

Most of the papers discussed extraction based summaries from the original document. The sentence extraction techniques compute score for each sentence based on features such as position of sentence in the document [1], word or phrase frequency, and key phrases [5]. There were some attempts to use machine learning (to identify important features), use natural language processing (to identify key passages or to use relationship between words rather than bag of words). The application of machine learning to summarization was pioneered by [7], who developed a summarizer for scientific articles using a Bayesian classifier [2]. For the generation of a coherent and readable summary, one has to do significant amount of text analysis to generating good feature vector, handling discourse connectors, and refining the sentences. This system is an attempt in that direction

Indicative and Informative summarization:

In general, email summarization can be classified into two classes: informative and indicative summarization, based on the purposes of the summary. For informative summarization, a summary contains the major content of the original emails, and the readers do not need to refer to the original emails after reading the summary. In contrast, for indicative summarization, a summary only provides clues of the content without details.

Thus, readers still need to go through the original emails for the details. So far, most of the email summarization systems, including mine, focus on the informative summarization, while indicative summarization is neglected. However, indicative summarization is very useful as well, especially for

 ISSN: 2321-8169
136 - 139

summarization based on a query or for mobile email users where the screen size is small. One challenge of indicative email summarization is the effectiveness of a summary within a given short length. Complete sentences may be too long, but key words may not carry enough information to discriminate the content.

An informative summary is meant to represent (and often replace) the original document. Therefore it must contain all the pertinent information necessary to convey the core information and omit ancillary information. An indicative summary's main purpose is to suggest the contents of the article without giving detail on the content. It may serve to entice the user into fetching the full form. Book, card catalog entries are examples of indicative summaries.

Although our nave user model only accounts for two tasks - browsing and searching - we believe that informative and indicative summaries differ in their power to aid each task. Multi document informative summaries capture broad similarities which are good for browsing, and multi document indicative summaries capture salient differences which are good for searching.

Extractive and generative summarization:

Extractive summary is to extract sentences from the original emails and use them as a summary. The advantage is that each sentence is a meaningful unit from the original emails. However, extractive summaries may not be coherent, and readers need to organize it. In contrast, generative summarization generates coherent sentences to summarize the original emails, which needs techniques in natural language generation.

III. PROPOSE SYSTEM

This email document summarization tool focuses on extraction methods from a single email document. Original email file is preprocessed and plain text is given to phase 1 which divides the text into sentences based on the rules (discussed next).The sentences are again divided in to words. From these words the stop words are eliminated.

In the second phase the score of each word is calculated. From the score of each word the score of the sentence is calculated. Based on the top score of N words the sentences are extracted from the text. Figure 1. Showing the proposed system, which is discussed in following sub topics

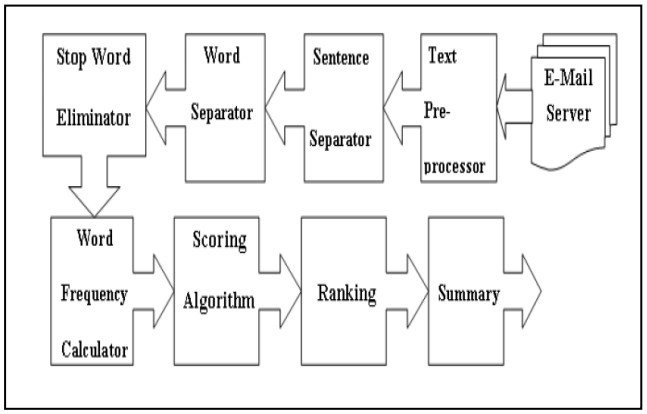

Figure 1. Propose System Architecture.

A. *Text Pre-processor*

This will work on email document. In this functional module email file which has .eml extension is preprocessed for summarization. Email file is structured file which has sender, receiver ids, subject and body content. So we have to convert it into plain text file for further processing. Here we have to separate out sender, title, email body content and mime rules, so we can use it as feature of that documents, because the key word coming into subject has high weight so we are considering it during scoring phase.

B. *Sentence Separator*

This goes through the email document and separates the sentences based on some rules [6] (like a sentence ending is determined by a dot or question mark and a space etc.). Any other appropriate criteria might also be added to separate the sentences.

Rules for separating sentences from text:

1. The end of the sentence must be punctuation (.!?), possibly with closing parenthesis and/or double-quote after it.
2. The next chunk of text has to start with an upper-case character or number, possibly with an opening parenthesis and/or double-quote preceding it.
3. The sentence can't end with "Mr." or titles like it, or an initial. This is to keep the previous rules from splitting sentences like "Hello Mr. John Q. Public!" incorrectly in the middle.
4. The sentence needs to have balanced parenthesis and quotes. This assures that sentences breaks won't be identified in quoted material.

C. *Word Separator*

This separates words based on criteria like a space denotes the end of a word etc.

D. *Stop Word Eliminator*

This eliminates the regular English words (stop words) like 'of, from, a, an, the,' etc. for further processing. These words are known as stop words in document. A list of applicable stop-words for English is available on the Internet. Here in email document you is written as 'u', so we have created separate stop word list for email document by survey. We are eliminating stop words from email document because its weight is zero.

E. *Word Frequency Calculator*

This calculates the number of times a word appears in the document (stop-words have been eliminated earlier itself and will not figure in this calculation) and also the number of sentences that word appears in the document. For example, the word 'Bramha' may appear a total of 100 times in a email document, and in 80 sentences. Some minimum and maximum thresholds can be set for the frequencies (the thresholds to be determined by trial-and-error)

Here we have discussed algorithm for word frequency calculation, which is giving best output.

Algorithm for word frequency Calculation:

A. Calculate Term Frequency in Document : **f (term)**
B. Calculate Inverse Log Frequency in Corpus :

$$if(term) = \log(\frac{n(corpus)}{n(term, corpus)})$$

C. Words with high **f(term) if(term)** are indicative.
D. Keywords Cluster are found (accord. to maximal width) and Weighted.

Figure 2. Word Frequency Calculation Algorithm.

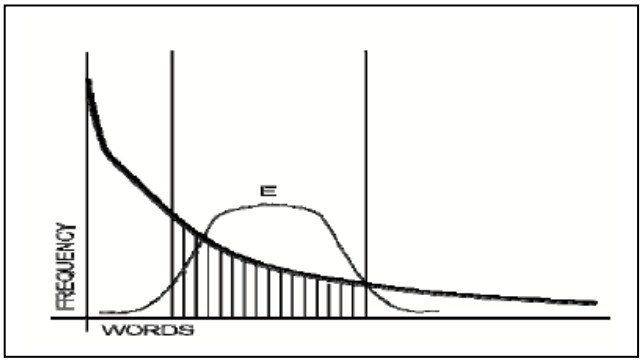

Figure 3. Resolving Power of Significant Words.

Figure 2 has shown the algorithm which we apply on email document for calculating word frequency. Figure .3 is showing resolving power of significant words. So it claims- Important sentences contain words that occur "somewhat" frequently. This method increase sentence score for each repeated word. The straightforward approach empirically shown to be mostly detrimental in auto summarization system. [11]

F. Scoring Algorithms

This algorithm determines the score of every sentences. The score can be made to be proportional to the sum of frequencies of the different words comprising the sentence (i.e., if a sentence has 5 words Ram, Ramesh, Bramha, Sachin and Deva, then score is proportional the sum of how many times Ram, Ramesh, Bramha, Sachin and Deva have occurred in the document). The score can be made to inversely proportional to the number of sentences in which the words in the sentence appear in the email document. Many such heuristic rules can applied to score the sentences in document.

a) Position based scoring:

Position based scoring algorithm considers the sentence location in document. In order to find appropriate weights for keywords for position based scoring, we have investigated how the summaries in the training corpus reflect the first 3 sentences of the original document content, the first sentence after each subtitle and the first 2 sentences of each paragraph of email document.

We have established that the most influential sentences are the sentences following the title – the first sentence of the text document was included in the summary in 100% of the cases, the second and the third sentence in 65% of the cases. The sentences immediately following the subtitles included in the 60% of the case.

TABLE I. POSITION BASED SCORES

Feature	Percentage in Extracts	Given Score
1st Sentence in Article.	100	10
2nd Sentence in Article.	65	7
3rd Sentence in Article.	65	7
1st Sentence after sub header.	60	6
1st sentence in paragraph.	40	4
2nd sentence in paragraph.	20	2
3rd sentence in paragraph.	20	2
Others	6	0

We have also found that the first sentence of the paragraph was included in the summary in 40% of the cases in document, and the second and the third in 20% of the cases. In addition, 20% of the summaries contained the last sentence of the text. The position based scores are given in Table 1.The scores are normalized using following formula (1).

$$n = (p*100) / t \quad (1)$$

Here *n* is normalized score of document, *p* is assigned score of the sentence and *t* is total of all position scores in the article.

b) Format based scoring

Format based scoring algorithm considers the sentence font (default, bold or italic) and punctuation marks. Figure captions and the text author are also detected and given minimum scores. Table 2 depicts the features and scores.

TABLE II. FORMAT BASED SCORES

Feature	Percentage	Given Score
Default Font	32	3
Bold or Italic.	70	10
Exclamation mark/ Question in sentence.	10	0
Quotation mark in sentence.	18	2
Captions, Authors, Sub headers.	0	0

c) Keyword based scoring

Keyword based scoring algorithm uses two techniques for detecting keywords: finding words that are relatively frequent in this article and not very frequent in general word frequency table; extracting words from the text title and all subtitles. When we were training corpus, we got that only 52% of the sentences containing words from the titles were included in summaries. Also, if extra score is assigned to sentences containing most frequent words, then only 25% of the sentences with highest scores are actually available in summaries. When discovering repeated word forms, the summarizer must employ a general word frequency table for a given language and document, in order to estimate whether the word form appears more frequently than it normally does in content written in that language or document. Our keyword

based scoring algorithm also uses a general word frequency table.

TF-IDF DOCUMENT VECTOR

$$w_{ik} = tf_{ik} * \log(N / n_k)$$

T_k = term k in document D_i

tf_{ik} = frequency of term T_k in document D_i

idf_k = inverse document frequency of term T_k in C

N = total number of documents in the collection C

n_k = the number of documents in C that contain T_k

$$idf_k = \log\left(\frac{N}{n_k}\right)$$

Figure 4. TF-IDF Document Vector.

By using above TF-IDF Vector space algorithm, we are calculating word frequency. Here, Wik is word weight which is calculated by equation. And it is giving best weights for word in email documents.

G. *Ranking*

The sentences will be ranked according to the scores of sentences. Other criteria's like the position or format of a sentence in the document can be used to control the ranking. For example, even though the scores are very high, we will not put consecutive sentences all together.

H. *Summarazing*

Based on the user input on the size or percentage of the summary, the sentences will be taken from the ranked list and concatenated. The resulting summary will be displayed when mouse pointer hover to subject keyword in mail clients software.

Final Sentence Selection:

Add sentences to the pool so as to avoid dangling discourse relations. For example if a sentence starts with "afterwards" or "but", the preceding sentence was marked as important as well and added to the set of important sentences. Some sentences are removed depending on the length of the desired summary. If a short length summary is requested, than it is good to select many short sentences and remove very long sentences. If the length of summary is comparable with the length of the document than sentences which are less than some threshold are re-moved from the pool. Remove questions, title and subtitles from the set of sentences. Rewrite sentences by deleting marked parenthetical units. Each third person pronoun that referred to an entity that was not mentioned al-ready in the summary was replaced with the complete referring expression, if previously computed. In the final step, we generate the summary by concatenating the remaining sentences.

IV. CONCLUSION

Email-summarization is a technique used to generate summaries of electronic mail. In this an Email auto summarization tool is developed using statistical approach. The techniques involve finding the frequency of words, inverse frequency of words, scoring the sentences, ranking the sentences. The summary is obtained by selecting a particular number of sentences (specified by the user) from the ranked list. It operates on a single document (but can be made to work on multiple documents by choosing proper algorithms for integration) and provides a summary of the document.

This paper proposes a new summarization method based on the vector space algorithm. This method can eliminate the situations that a word has several meanings and several words have the same meaning. It can also solve the problem that the assignment of summarization sentences. It has a certain coverage and completeness. The experimental results indicate that the method we proposed is more efficient than traditional ones, but the problem of this summarization is that it is slower and not better than abstract summarization.

ACKNOWLEDGMENT

We would like to express our special thanks to Prof. V. D. Badgujar and our Head of Department Prof. H. D. Sonawane, who also helped us in doing a lot of Research. We are really thankful to Principal Prof. C. K. Patil.

REFERENCES

[1] Baxendale, P. B. 1958. Man-made index for technical literature--An experiment. IBM Journal of Research and Development, 2(4):354-361

[2] Daume and Marcu 2005 Hal Daume and Daniel Marcu. Bayesian Multi-Document summarization at MSE. In ACL 2005, Workshop on Multilingual Summarization Evaluation.

[3] D. Schuff, O. Turetke, D. Croson, F 2007, 'Managing Email Overload: Solutions and Future Challenges', IEEE Computer Society, vol. 40, No. 2, pp. 31-36.

[4] E. D' Avanzo, B. Magnini, A. Vallin 2004, Keyphrase Extraction for Summarization purposes: The LAKE system at DUC-2004, Document Understanding Conference.

[5] Edmundson, H. P. 1969. New methods in automatic extracting. Journal of the Association for Computing Machinery, 16(2):264-285.

[6] Jagadeesh J1, Vasudeva Varma1, Single Document Summarization Using Natural Language Processing , Language Technologies research Centre International Institute of Information Technology Hyderabad, India.

[7] Jones, S., Lundy, S. and Paynter, G.W. (2002). Interactive document summarisation using automatically extracted keyphrases. Hawai'I International Conference on System Sciences:Digital Documents: Understanding and Communication Track, Hawai'i, USA, January 7-11, 2002, IEEE-CS, pp101.

[8] Mani, Inderjeet and Mark Maybury, editors. 1999. Advances in Automatic Text summari-zation. MIT Press, Cambridge.

[9] Miller 1990 George A Miller. 1990. Nouns in WordNet: a lexical inheritance system. International Journal of Lexicography, 3(4):245-264.

[10] N. Kushmerick, T. Lau, 2005, 'Automated Email Activity Management: An Unsupervised learning Approach', Proceedings of 10th International Conference on Intelligent User Interfaces, ACM Press, pp. 67-74.

[11] Raghu Krishnapuram 2005, Senior Member, IEEE , Bayesian Multi-Document summarization at MSE. In ACL 2005, Workshop on Multilingual Summarization Evaluation.

[12] Spark Jones, Karen. 1999. Automatic summarizing: Factors and directions. In Ad-vances in Automatic Text Summarization. MIT Press, Cambridge, pages 1-13.