Faculty of Computing and Informatics
Multimedia University, Malaysia.

Research Proposal

Using Latent Semantic Analysis to improve Fragment Quotation Graph

NG CHIN ANN 1142701684
KHOR KIA KIN 1142701883

Lecturer:
Dr. Bhawani

May 13, 2016

Executive Summary of Research Proposal

Careinini et al. proposed Clue Word Summariser (CWS) algorithm in their paper on summarising email conversations effectively. The algorithm ranks sentences based on reoccurrence of words within both parent and child nodes in fragment quotation graph (FQG). However, the ClueScore used in the algorithm is computed only on words with same stem, thus words with same meaning but different stem are ignored.

In this research, we aim to improve the accuracy of FQG by using latent semantic analysis (LSA) to deal with word pairs with high semantic relatedness or those with looser semantic link.

We will perform an experiment to replace words with same meaning but different stem from a child node into words with same stem from parent nodes. For the experiment, we will use LSA and Infomap to detect words with similar meaning.

Since ClueScore is computed using Porters stemming algorithm, we expect the accuracy of CWS can be improved because the replaced words in a child node will have the same stem as those in its parent node.

If this research is a success, the optimised CWS will summarise email conversations with a higher accuracy and can be applied to other areas such as identifying the main points in an article.

Introduction

Usage of email as a communication tool has become prevalent in our daily life especially in the industry. An analysis predicts that the number of email accounts will be increased from 3.1 billion in 2011 to nearly 4.1 billion by 2015. (Radicati, 2011) However, the increasing usage of email has led to email overload which causes email users to spend a lot of time in organising their email conversations.

Email summarisation can be an effective solution to the problem. Although several consistent email summarisation approaches exist in the literature, they can be further improved.

Sequence of an email conversation is important to generate accurate information in the summary in email summarisation. Carenini et al. introduced fragment quotation graph (FQG) which represents an email conversation in a directed graph form. The graph is able to identify the hidden emails - quoted emails but not present inside the users folders - where previously there is no approach to handle this issue. (Carenini, Ng, Zhou, & Zwart, 2005)

Similarity between words like synonym and polysemy is also an important criteria in text summarisation. For example, the words "talk" and "discuss" carry the same meaning inside a given context. Carenini et al. also introduced ClueWordSummariser (CWS) algorithm to summarise email conversations based on the words with same stem in an email and its reply email. (Carenini, Ng, & Zhou, 2007) However, the algorithm lacks of

some consistency in which a word with same meaning but with different stem is ignored in the summary.

Latent semantic analysis (LSA) is a widely used technique in natural language processing (NLP). It can identify the similarity between words using a term-document matrix. With the help of LSA, we believe FQG can be further improved to become a better method in representing email conversations for email summarization. So, we would like to propose an experiment to apply LSA on FQG to identify words with very similar meaning and then replace the word in child node with the similar word in its parent node before applying FQG in CWS. We believe that this method can increase the accuracy of the CWS which uses Porters stemming algorithm as their scoring standard.

Justification/Motivation of the Research Problem

Although more people are shifting towards social media such as Facebook or Twitter, email still retains its value as the method of communication with commercial entities or organizations. There are plenty of researches on text summarisation, but not many of them are focused on email summarisation. Conducting LSA on the email summarisation presents several new challenges such as:

1. Identifying the structure of a given email conversation since not every email is quoted by the user during a reply. (Here, we consider most of the emails will be quoted before sending out.)
2. Identifying the type of grammar to be included in the term-document matrix when performing latent semantic analysis.
3. Identifying a new algorithm that replaces the word in child node with another word in its parent node if the word pair has a high similarity value.

The result of this research can also be applied into other text summarisers such as MEAD or RIPPER as well.

In the industry, many companies use emails to promote products, send confirmation emails or surveys. This is a good opportunity to help the email users to understand the information contained in the long passage quickly with identified sequence.

Research Objectives

1. To apply LSA on FQG to identify the similarity between the words in a parent node and its child node.
2. To evaluate the efficiency of CWS using FQG with LSA, Infomap and the original FQG.

Literature Review

Currently, there are many approaches introduced to summarize the email such as decision-making summarization for email conversation. (Wan & McKeown, 2004) They use email threading where structure of the conversation is considered in their works. There is also quotation matching method to construct the email threads (Yeh, 2006). However, most of the multi-document summarization did not consider the structure of email conversations and none of the proposed approaches deal with hidden emails. (Carenini et al., 2007) To deal with the problem, fragment quotation graph has the ability to represent the email conversations in a structured form where the sequence of the emails is considered. (Carenini et al., 2005) It is also capable of dealing with hidden emails.

However, in our observation, the FQG is still insufficient to provide a good solution to email summarisation. We speculate that LSA could improve its efficiency. Many researchers had used LSA as a method to extend their works and succeed in increasing the accuracy of their works. A LSA method to summarise Turkish text has proved that it is better than other existing approaches. (Ozsoy, Cicekli, & Alpaslan, 2010) LSA is also used to automate and enhance some aspect of researches in historical semantic and other fields whose focus is on the comparative analysis of word meanings within a corpus. (Sagi, Kaufmann, & Clark, 2009)

Measuring the similarity of words is important in accurately representing and comparing documents which will improve the results of many NLP task. (Tian, Lo, & Lawall, 2014) Meanwhile, incorporating temporal information in LSA is valuable for word similarity computation which could improve the search quality of the CGC. (Wang & Agichtein, 2011)

Research Methodology

The experiment will be conducted on the method we proposed. First, we extract all the given email conversations in the data set using FQG. Second, we apply LSA on FQG to identify the semantic relatedness between words. Then, the word pairs with high similarity will be undergo replacement where the word in the child node will be replaced with the word in its parent node. Below we will briefly describe the methods and how we apply them into our experiment.

Fragment Quotation Graph (FQG)

FQG is a direct graph G = (V, E), where each node u V is a text unit in the email folder, and an edge (u, v) refers to node u is in reply to node v. Similarly with the experiment carried out by Carenini et al., FQG is used to extract the email conversations from the given email dataset.

Tokenisation, Stop Words Removal and Parsing

After FQG has extracted the email conversations, every sentence in the email fragments are tokenised. This is because LSA and Infomap are operated using a matrix, thus each row of the matrix must represents a single word. Hence, we need to chop (tokenise) the sentences in email fragments into single words. Then, very common stop words are removed because they do not contribute much meaning to the

email summary. After stop words removal, Stanford parser is used to parse the words. (Chen & Manning, 2014) Only nouns and verbs are taken out because they normally contain important information regarding the context. For LSA, the parsed nouns are input into a matrix and then used to compute dot product. The procedure is then repeated with the parsed verbs. For infomap, the entire procedure in LSA is repeated but it is used to compute cosine similarity instead.

Singular Value Decomposition (SVD)

SVD is a mathematical technique to reduce the number of rows while preserving the similarity structure among the columns in a matrix. The reason why we use SVD is to remove words with lower reoccurrence where they might not carry important information. In our experiment, we use SVDPACKC package which provides implementation of SVD to do row reduction. (Berry, Do, O"Brien, Vijay, & Varadhan, 1993)

Latent Semantic Analysis (LSA)

LSA is a technique used in NLP to analyse the relationship between a set of documents and the terms inside them. LSA is implemented using a term-document matrix (TDM). The matrix contains rows representing unique words and columns representing each paragraph. In our research, the rows represent every single word in the parsed email fragments while the columns represent every email fragment in a conversation folder. SVD and dot product are then computed on the TDM. SVD is used to reduce the rows of the generated matrix while maintaining the similarity structure in the columns, while dot product is calculated to show how similar the word pairs within two rows.

Dot product of two vectors A = $[A_1, A_2, \cdots, A_n]$ and B = $[B_1, B_2, \cdots, B_n]$:

$$A \cdot B = \sum_{i=1}^{n} A_i B_i = A_1 B_1 + A_2 B_2 + \cdots + A_n B_n \tag{1}$$

Infomap

Infomap is an open source software developed in Stanford. (Standford, 2007) Instead of using email fragments, it uses content bearing words to determine how often a word in a text occurs near to these words. It applies a variant of LSA on free-text corpora to learn vectors which represent words meaning in a vector-space model. The word-word semantic similarity is computed by comparing the word vectors using cosine similarity. In our research, we will use the email conversations as our training corpus. We will use the most common 1000 content bearing words provided to compare with the dataset we use.

Cosine similarity between two vectors is calculated by:

$$\cos\theta = \frac{\overrightarrow{A} \cdot \overrightarrow{B}}{\| \overrightarrow{A} \| \cdot \| \overrightarrow{B} \|} \tag{2}$$

SemanticReplacer algorithm

After the dot product or cosine similarity is calculated, we apply word replacement

to the word pairs with similarity value greater or equal to 0.8. The reason we choose 0.8 as our scoring standard is because this number is relatively high so it wouldn't affect the meaning of the text while replacing the word. (Note: Dot product is calculated from LSA and cosine similarity is calculated from Infomap. The value which is closer to 1 means that the word pairs are with a high similarity.)

1. If the similarity value is >= 0.8, the word in a lower row inside TDM will always be replaced by the word in a higher row. This is because the word in a lower row is in the child node or within the same fragment but is located after the word in a higher row since we insert the words according to the sequence of email.
2. If the word in the parent node is not the root node and there occurs a replacement requirement of word to its parent node but its child node has the similar word which is also need to be replaced with that word, the word to be replaced in the child node will be replaced with the word in the grandparent node.

For example, Node A is root node follow by Node B and Node B is followed by Node C. There are words "human", "user" and "people" inside the Node A, B, C respectively. If the similarity between "people" and "user" is higher than "people" and "human" but the "user" and "human" similarity is also high, the word "people" will be replaced with "human" which is in its grandparent node. To do this, the graph is traversed to find the replace word until it reached the root node.

After all the word pairs with high similarity are replaced, we can start to apply the FQG into the CWS to do the experiment. Note that we will carry out the experiment in three ways:

1. Test the CWS with original FQG which is introduced by Carenini et al.
2. Test the CWS with FQG where LSA is applied into it.
3. Test the CWS with FQG where Infomap is applied into it.

We will use Enron email dataset as our dataset. This dataset contains emails conversation from 150 users which is enough for us to carry out the experiment effectively. Here, we will not recruit human summarizer as our gold standard since the purpose of this experiment is just to compare the efficiency of FQG before and after extra features are applied into it. We will evaluate the output of FQG using precision, recall and F-measure where the summary length is controlled at variant length.

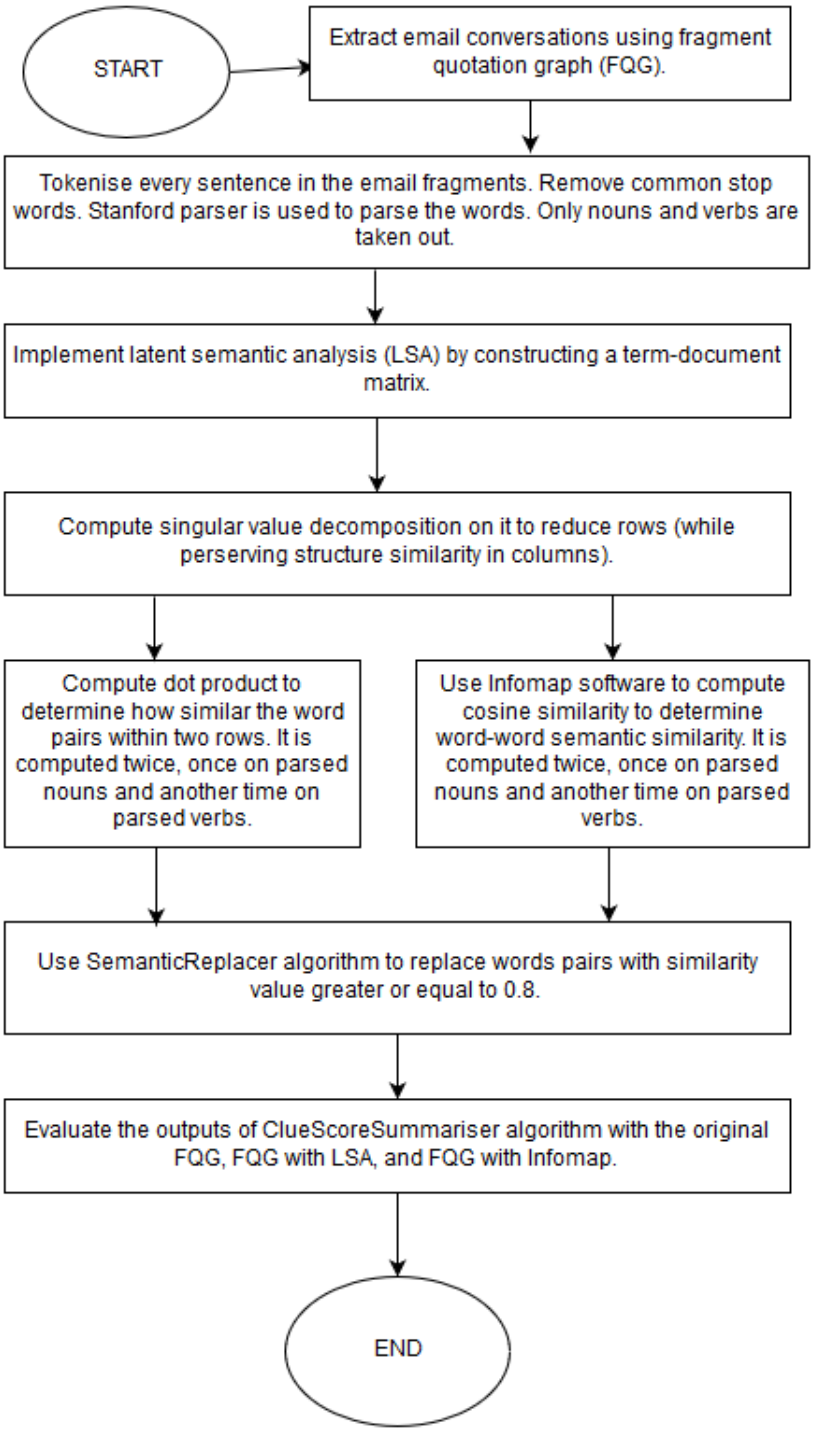

Figure 1: Flow chart of the work.

References

Berry, M., Do, T., O"Brien, G., Vijay, & Varadhan, S. (1993). Svdpackc (version 1.0) user"s guide. *Technical Report. University of Tennessee Knoxville, TN, USA.*.

Carenini, G., Ng, R. T., & Zhou, X. (2007). Summarizing email conversations with clue words. *ACM WWW '07 Proceedings of the 16th international conference on World Wide*, 91-100.

Carenini, G., Ng, R. T., Zhou, X., & Zwart, E. (2005). Discovery and regeneration of hidden emails. *ACM SAC 05: Proceedings of the 2005 ACM Symposium on Applied Computing*, 503-510.

Chen, D., & Manning, C. D. (2014). A fast and accurate dependency parser using neural networks. *Proceedings of EMNLP 2014*.

Ozsoy, M. G., Cicekli, I., & Alpaslan, F. N. (2010). Text summarization of turkish texts using latent semantic analysis. *COLING '10 Proceedings of the 23rd International Conference on Computational Linguistics*, 869-976.

Radicati, G. (2011). *Email statistics report.* Retrieved from `http://www.radicati.com/?p=7269`

Sagi, E., Kaufmann, S., & Clark, B. (2009). Semantic density analysis: Comparing word meaning across time and phonetic space. *GEMS '09 Proceedings of the Workshop on Geomatrical Models of Natural Language Semantics*, 104-111.

Standford, C. (2007). *Infomap [computer software].* Retrieved from `http://infomap-nlp.sourceforge.net/`

Tian, Y., Lo, D., & Lawall, J. (2014). Sewordsiml software-specific word similarity database. *ICSE Companion 2014 Companion Proceedings of the 36th International Conference on Software*, 568-571.

Wan, S., & McKeown, K. (2004). Generating overview summaries of ongoing email thread discussions. *In Proceedings of COLING04, the 20th International Conference on Computational Linguistics*.

Wang, Y., & Agichtein, E. (2011). Temporal latent semantic analysis for collaboratively generated content: Preliminary result. *SIGIR '11 Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, 1145-1146.

Yeh, A. H. J.-Y. (2006). Email thread reassembly using similarity matching. *In Third Conference on Email and Anti-Spam (CEAS)*.