Light-Weight Multi-Document Summarization based on Two-pass re-ranking

Yu-Chieh Wu, Kun-Chang Tsai
Dept. of Computer Science and Information Engineering
National Central University
Taoyuan, Taiwan
{bcbb,turn-into}@db.csie.ncu.edu.tw

Yue-Shi Lee
Dept. of Computer Science and Information Engineering
Ming Chuan University
Taoyuan, Taiwan
lees@mcu.edu.tw

Jie-Chi Yang
Graduate Institute of Network Learning Technology
National Central University
Taoyuan, Taiwan
yang@cl.ncu.edu.tw

Abstract

The goal of topic-oriented text summarization is to produce informative short description according to the given topic or query. This is somewhat similar to the target of question answering which retrieves exact answers from large text collections. In this paper, we present a light-weight and rule-free summarization technique. Our method relies on a two-pass re-ranking framework. The first pass is to order the concepts which were clustered via conventional top-down clustering algorithm. The second pass generates the representative sentences from the top *N* concepts. The main advantage of our work is that we do not need to build external knowledge or pre-defined rules. This is our first time to participate in DUC. Although the result of our system is not comparable with most top-performed methods, the light-weight and rule free techniques still encourage us to further improve via integrating rich sources.

1 Introduction

In recent years, there has been an accumulation of vast amounts electronic texts and web pages. To effectively and efficiently acquire important information, there have been several on-going research domains of natural language processing for this task such as information retrieval (IR), information extraction (IE) and automatic text summarization (ATS).

This year's document understanding conference (DUC-2006) task is the same as past year (Dang, 2005). The target is to generate 250 words summaries from multi-documents according to the given subject or question, i.e. question-focused text summarization. This task is quite different from traditional summarization tasks that only focus on extracting important sentences without regarding the main relevance to users. The question-focused text summarization is very similar to the traditional question answering (Q/A) task (Voorhees, 2001) that aims to find exact answers from huge document collections. But the difference relies on the granularity of questions and returned answers. Traditional Q/A put emphasis on asking the factoid questions, however, answers should be short and exact to answer the question. In contrast, in the question-focused summarization task, the question describes an event, a comparison, or changes whereas the returned summary is like a story to response the requirement. For example, the question of topic 614 in DUC-2006 is "Describe developments in the movement for the independence of Quebec from Canada.".

In this paper, we describe the overview of our light-weight and fully automatic text summarizer at DUC this year. Unlike previous studies (D'Avanzo, and Magnini, 2005; Ye et al., 2005; Li et al., 2005),

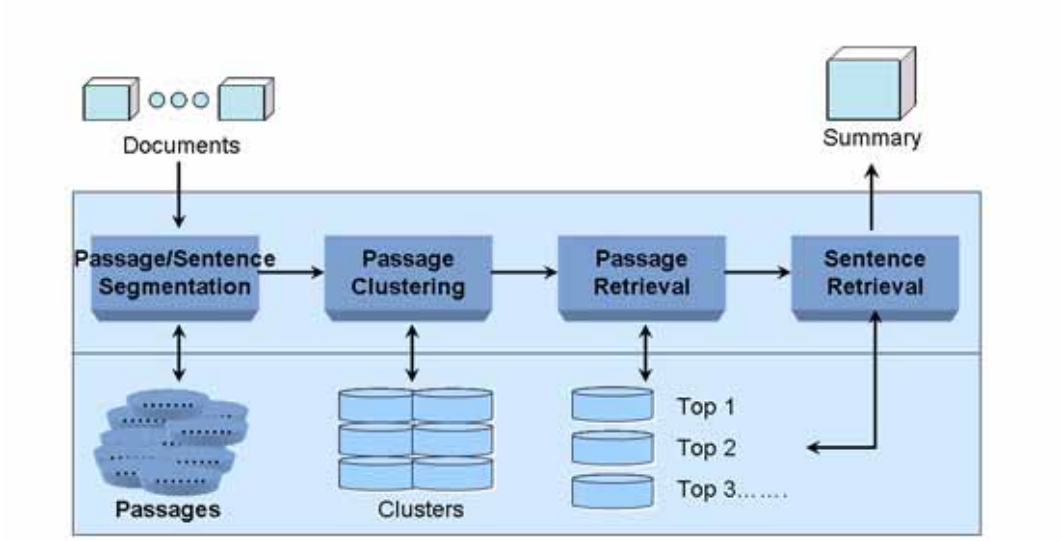

Figure1: System architecture

we show and indicate how effect can the human-free system perform. Our method is built on the two-pass re-ranking process and a density-based scoring function. Both the two pass ranking were employed the density-based scoring function. Before first pass ranking, we adopt an efficient top-down clustering algorithm, then pass one ranking model retrieves several important paragraphs via the scorer. For each retrieved passage, the second pass ranking model selects the most important sentence and adds it to the summary via the same scoring function. Finally, we re-order the retrieved sentences according to their time-stamps.

This paper is organized as following, Section 2 describes overview of our system, and Section 3 describes the density-based scoring function. In Section 4, we present the evaluations and experimental results. At Section 5, we draw the future direction and conclusion.

2 System Description

The target of multi-document text summarization is to extract or refine important sentences from different documents that belong to the same topic. As described above the goal is quite similar to the Q/A task. However, we tries to combine some advanced techniques in Q/A research domain, e.g., the powerful density-based passage retrieval algorithm (Tellex et al., 2003; Lee et al., 2001). Figure 1 shows the overall architecture of our system.

There are four main components within our model, namely sentence/passage segmentation, passage clustering, passage retrieval and sentence retrieval. For the given question, we first segment each sentences and passages in the given document set respective to the question. Some of the passages might describe the same topic. Therefore, we perform a clustering algorithm to group similar passages into the same paragraph. Finally the two-pass re-ranking models are used to retrieve the useful and informative passages at first pass. For each retrieved passage, the second pass, sentence retrieval component, selects the most important sentence for a passage and add it to be the summary. To make the summary more readable, the added sentences are re-ordered according to their time-stamps.

In the following subsections, we will introduce the first component in Section 2.1. For the two retrievers and the clustering method are introduced in Sections 2.2, 2.3,and 2.4.

2.1 Passage/Sentence Segmentation

In this step, the sentences are first segmented. The words are not stemmed and tokenized. We do not perform the word-stemmer to represent the root of words. Instead, this will be done in the clustering and ranking steps. The sentences segmentation is carried out with a tool[1]. This tool can successfully identify boundaries between sentences without tokenizing words.

The documents in the DUC-provided set had been annotated with passage boundaries. Without employing additional passage segmentation tool, we directly use the tag to split paragraphs.

[1] http://l2r.cs.uiuc.edu/~cogcomp/tools.php

2.2 Passage/Sentence Clustering

In multi-document summarization scenario, multiple passages or sentences may describe the same concept meanings. To reduce the redundancy, a conventional clustering technique is applied to group similar passages into the same group. Different from previous studies, we use the top-down bisecting *K*-means algorithm (Zhao and Karypis, 2002) for clustering. The bisecting *K*-means algorithm is a top-down step-by-step clustering method that incrementally performs *K*(=2)-means to split the largest group into two sub-clusters. In the Zhou's study, they have shown that the bisecting *K*-means outperformed the traditional *K*-means in document clustering task. Nevertheless the passage clustering is very similar to the document clustering. Hence, we select the bisecting *K*-means algorithm to avoid the risk of randomly initialization of the traditional *K*-means.

We slightly modify the bisecting *K*-means and set the number of clusters as 300. There are several criteria functions to evaluate the quality of a clustering result. We use the internal criteria functions to measure the similarity inside the cluster. This function was demonstrated as a very effect method to determine the clustering result at each splitting step for bisecting *K*-means. In addition, the settings of the algorithm are almost the same as the literature (Zhao and Karypis, 2002), except for the example representation. In order to capture more accurate meanings in passages, we do not only use the traditional bag of words model (with Porter stemming), but also include the bag of bigrams. Bigrams are meaningful than unigram.

2.3 Passage Retrieval

Both passage retrieval and sentence retrieval components adopt the same ranking model to extract important passages and sentences. As mentioned above, our method is based on the two pass ranking models. We can replace the two pass framework with one-pass sentence ranking. However, in this way, it will cause the sentences too similar to make the reader over-understanding. In section 3, we will discuss the scoring function that devotes on retrieving important words that appear both in passage and the given question.

For the passage retrieval, we simply use the given question as query to the density-based retrieval algorithm to retrieve top 10 clustered passages. We then use the sentence retrieval to extract sentences from the 10 passages.

2.4 Sentence Retrieval

We treat each retrieved passage as separable concept. Our summary is mainly derived from these concepts. Therefore for each passage, we extract an important sentence within it to represent the concept. Unlike most approaches, which estimate the similarity between the centroid and each sentence. Instead, the density-based retrieval algorithm again is used to measure the importance of each sentence inside the cluster. At previous stage, some large clusters may be ranked higher due to they contain more question word. Although the centroid sentence indicates that it shared common words or bigrams as similar sentences. It could not be used to be the summary since it would not sufficient to answer the given question.

Moreover, the sentence that can be used to form the summary should be able to answer the question. Thus, we perform the density-based retrieval algorithm to rank the importance of each sentence within the same cluster. For the retrieved top 10 clusters, we derive 10 most important sentences to form the summary. To make the summary more readable, each sentence is further re-ordered according to its time-stamps in the original document. If the summary contains more than 250 words, we remove the final part of the summary to enable the size no more than the size limitation.

3 Density-based Retrieval Algorithm

Searching answers in a small dataset is more efficient than in the whole corpus. To find out answers in passages is much easier than searching the whole relevant document set. In this section, we will introduce our retrieval model.

3.1 Passage Retrieval

The passage retriever segments each retrieved document into passages and retains the paragraphs that contain at least one of the query terms. We implemented the similar idea of the IBM (Ittycheriah et al., 2001), SiteQ (Lee et al., 2001), and ISI (Hovy et al, 2001) passage retrievers and modified

the ranking functions. There are three common features within our algorithms.

(1) Query expansion

Query expansion is useful when a question contains very few informative terms, for example.

"What is an atom?"

The question asks for a definition of the term "atom" where there is no other content word in the question. In this case, both relevant and irrelevant passages will be retrieved, which makes the passage retriever difficult to rank these passages. In order to acquire more information for this short query, we use WordNet to expand query terms. We derive synonyms, hyponyms, and hypernyms of all of the content words in the question. For the above example, without considering the sense of the query term, all the hypernyms, hyponyms, and synonyms of "atom" will be extracted through WordNet querying.

(2) Keyword weighting

As for the second feature "keyword weighting", terms will be weighted in different level. This technique aims to highlight some important content words, like named entity terms and content words in the retrieved passages. In this paper, we define the seven degree of scoring function,

W_1: Named entity match (1.5)
W_2: Question first noun phrase match (1.2)
W_3: Question term exact match (1)
W_4: Stem match (0.7)
W_5: Synonym match (0.5)
W_6: Hyponym match (0.4)
W_7: Hypernym match (0.3)

If a term in the passage has the same named entity type in the question, then this term will be given W_1 (W_1 = 1.5) weight. The second type gives weight (W_2 = 1.2) when a term appears in the first noun phrase of the question. The third type gives weight (W_3 = 1) for a term which matches any of the question words. The forth type gives weight (W_4 = 0.7) for a term if it matches with one of the question stemming words. The remaining match types give the weight (W_5~ W_7) for a term when the term could be found in the synonym, hypernym, hyponym sets of the question words. If a term was matched more than twice, we select the highest level as its weight. For example, when the term matched with W_1, and W_2, W_1 were chosen.

(3) Density-based ranking

The density-based ranking method is different from traditional similarity scoring function which focuses on the density between matched keywords. Traditional similarity criterion, like Euclid, and Hamming distance, estimates the matched numbers of the two vectors without considering the geometric relations among these matched words. Density-based ranking method calculates the distance between each of the matched keywords. The closer the matched words appear, the higher rank the passage can be ordered. As reported by (Tellex et al., 2003), the density-based ranking method can enhance the passage retrieval performance. In this paper, we use the same density-based scoring function which was defined by (Lee et al., 2001) as following equation.

$$\text{Score}(S_i) = \sum_{j=1}^{k-1} \frac{W(t_j) + W(t_{j+1})}{\text{dist}(j, j+1)^2} \times \frac{k}{k-1}$$

$W(t_j)$ is the weighting score (see the seven matched types) of term j; k is the number of matched terms between question and passage. The equation "dist(j, j+1)" computes the number of words between matched term j and j+1. Figure 2 shows the overall passage retrieval algorithm.

Before starting, we shall remove unimportant passages (clusters) which contain no question words in the passage. In the first step, named entity tagger identifies the proper nouns in the original question. In this paper, we employ the named entity tagger proposed by (Wu et al., 2006). The named entity tagger was trained with MUC-7 training and development set based on SVM learning. The performance of this NER-tagger is 86.40 in $F_{(\beta)}$ rate (Wu et al., 2006). Then, we give more weight to the first noun POS (part-of-speech) tag due to the first appear noun often contains more information. To identify noun POS tags in text we also

Table 1: Overall score of our system in DUC-2006

System	Ling Quality Mean	Responsiveness (Content)	Responsiveness (Overall)	ROURGE2 Score	ROURGE SU4	BE-Score	Pyramid Score
Our Method	3.3	2.4	1.9	0.06	0.11	0.0	0.1374
System (AVG)	3.35	2.56	2.19	0.07	0.13	0.04	0.12
Human (AVG)	4.84	4.75	4.74	0.11	0.17	0.07	0.00
System-baseline	3.38	2.54	2.19	0.07	0.13	0.04	0.12

adopt the Brill-tagger (Brill, 1995) to recognize most noun words.

The second step aims to extend more knowledge from WordNet for content words. The third step evaluates the weight for each term in the passage according to the seven matched types. Step four calculates the density score function for each passage. Here, we select the top N (N=10) passages to be the candidates.

Preprocessing: Removes passages that contain no question terms.

Step1: Identifies Named Entities, question first noun phrase terms, and content words in the given question.
Step2: Extracts the synonym, hypernym, hyponym terms of all content words.
Step3: Weights all terms in each passage according to the match level (W_1~W_7).
Step4: Calculate the density score (see equation (3)) of each passage
Step5: Select top N passages as the answer candidates.

Figure 2: Passage Retrieval Algorithm

4 Evaluation Results

DUC-2006 has evaluated summaries in several ways: human evaluation with pyramid score, responsiveness to the topic and linguistic quality, and automatic ROUGE evaluations. The overall results are shown in Table 1.

4.1 Responsiveness

This evaluation gives the responsiveness score between one (lowest) to five (highest) to each automatic summaries. Responsiveness is a measurement that is supposed to contribute toward satisfying the information need expressed in the topic statement. Our summarization system achieved 2.4 and rank 24 (out of 34) on content response score. The overall responsiveness score was 1.9 and rank 31 (out of 34). We do not surprise the low score since we do not pre-define any templates or rules to pre-assume the summaries. All of the sentences are fully scored from the set of documents.

4.2 Linguistic Quality

This measurement estimates the linguistic quality of the auto-generated summaries. NIST employ several human experts who develop the given topic. They created the following judgments for evaluations.

- Grammatically
- Non-redundancy
- Referential clarity
- Focus
- Structure and coherence

Each summary is judged for each of the above factor and gave it the score from one (lowest) to five (highest). As shown in Table 1, we found our auto-summarizer attended quite satisfactory score (at the middle rank).

4.3 ROUGE

The recall-oriented understudy of gisting evaluation (ROUGE) (Lin, 2004) is a statistical summarization measurement. ROUGE computes the recall-based metrics using N-gram matching between the candidate summary and a reference set of summaries. The longer the N-gram matches, the higher score the summary achieves. Table 2 shows the evaluation result of our method in ROUGE measurement.

Table 2: ROUGE Evaluation

System	ROUGE-2	ROUGE-SU4
Our method	0.065	0.115
Human AVG	0.11	0.17
System AVG	0.07	0.13

Even the ROUGE provides an automatic method to evaluate the auto-generate summaries, in comparison to the human judgment, the ROUGE do not sufficient to measure the system performance. Nenkova and Passonneau (2004) considered that such a surface lexical matching could not absolutely capture the content of the summary.

4.4 Pyramid

As reported by (Nenkova and Passonneau, 2004), the ROUGE method can not be used as the absolute measurement. To fill up this gap they proposed the Pyramid manual evaluation approach with peer to peer annotation. The annotator firstly labels the summarization content units (SCU) with importance order, then assessor assign the sentences or short descript that corresponding to the SCU in the auto-generate summaries. The overall score of pyramid measurement is totally the sum of the score of its SCU score. Note that each of the SCU has its importance, which should be predefined at first.

This is the first year we participate in DUC, and also join the pyramid score. As shown in Table 1, the pyramid score of our method was 0.137 and rank 20 (out of 21).

5 Conclusions and Future Work

Text summarization is one of the most important issues in information retrieval and natural language processing community. This paper presents the impact of the automatic and rule-free summarization system with minimally human effort. To reach state-of-the-art, our method still need to combine more rich resources as most advanced techniques. The main focus of our work coincides with the original target goal of DUC conference, i.e. to automatic summarize multi-document without human intervene. One of the future work is to integrate more and more resources such as full parsers, human-made rules and thesaurus to refine the text summaries. In addition, we also find that many question answering technologies can be applied to retrieve important concepts in documents. We start to address the issues of combining question answering models for text summarization.

References

E. Brill. 1995. Transformation-based error-driven learning and natural language processing: a case study in part of speech tagging, Computational Linguistics, 21(4):543-565.

E. D'Avanzo, and B. Magnini. 2005. A Keyphrase-based Approach to Summarization: the LAKE System at DUC-2005. In Proceedings of the Document Understanding Conference (DUC).

H. T. Dang. 2005. Overview of DUC 2005. In Proceedings of the Document Understanding Conference (DUC).

J. Goldstein, M. Kantrowitz, V. Mittal, J. Carbonell. 1999. Summarizing Text Documents: Sentence Selection and Evaluation Metrics. In Proceedings of the 22th ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 121-128.

E. Hovy, U. Hermjakob, and C. Y. Lin. 2001. The use of external knowledge in factoid QA. In Proceedings of the 10th Text Retrieval Conference (TREC), pp. 644-652.

K. Ishikawa, S. Ando and A. Okumura. 2001. Hybrid Text Summarization Method based on the TF Method and the Lead Method. In Proceedings of the 2nd National Institute of Informatics Test Collection Information Retrieval (NTCIR) Workshop.

H. Jing. 2000. Sentence Simplification in Automatic Text Summarization. In Proceedings of the 6th Applied Natural Language Processing Conference (ANLP).

G. G. Lee, J. Y. Seo, S. W. Lee, H. M. Jung, B. H. Cho, C. K. Lee, B. K. Kwak, J. W. Cha, D. S. Kim, J. H. An, and H. S. Kim. 2001. SiteQ: Engineering high performance QA system using lexico-semantic pattern matching and shallow NLP. In Proceedings of the 10th Text Retrieval Conference (TREC), pp. 437-446.

W. Li, W. Li, B. Li, Q. Chen, and M. Wu. 2005. The Hong Kong Polytechnic University at DUC2005. In Proceedings of the Document Understanding Conference (DUC).

C. Y. Lin. 2004. Rouge: A Package for Automatic Evaluation of Summaries. In Proceedings of the Association of Computational Linguistics Workshop, pp. 74-81.

K. Ohtake, D. Okamoto, M. Kodama, and S. Masuyama. 2001. Yet another Summarization System with Two Modules Using Empirical Knowledge. In Proceedings of the 2nd National Institute of Informatics Test Collection Information Retrieval (NTCIR) Workshop.

D. Radev. 2000. Text summarization tutorial. In Proceedings of the 23th ACM SIGIR Conference on Research and Development in Information Retrieval.

Y. Seki. 2002. Sentence Extraction by Tf/idf and Position Weighting from Newspaper Articles. In Proceedings of the 3rd National Institute of Informatics Test Collection Information Retrieval (NTCIR) Workshop.

S. Tellex, B. Katz, J. J. Lin, A. Fernandes, and G. Marton. 2003. Quantitative evaluation of passage retrieval algorithms for question answering. In Proceedings of the 26th ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 41-47.

E. Voorhees. 2001. Overview of the TREC 2001 question answering track. In Proceedings of the 10th Text Retrieval Conference (TREC), pp. 42-52.

Wu, Y. C., Chang, C. H. and Lee, Y. S. 2006a. A general and multi-lingual phrase chunking model based on masking method. Lecture Notes in Computer Science (LNCS): Computational Linguistics and Intelligent Text Processing, 3878: 144-155.

Wu, Y. C., Fan, T. K., Lee Y. S. and Yen, S. J. 2006b. Extracting named entities using support vector machines," Lecture Notes in Bioinformatics (LNBI): Knowledge Discovery in Life Science Literature, (3886): 91-103.

S. Ye, L. Qiu, T. Chua, and M. Kan. 2005. NUS at DUC 2005: Understanding Documents via Concept Links. In Proceedings of the Document Understanding Conference (DUC).

Y. Zhao, and G. Karypis. 2002. Evaluation of hierarchical clustering algorithms for document datasets. In Proceedings of Information and Knowledge Management, pp. 515-524.