Condensing biomedical journal texts through paragraph ranking

Jung-Hsien Chiang*, Heng-Hui Liu and Yi-Ting Huang

Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan, Taiwan

ABSTRACT

Motivation: The growing availability of full-text scientific articles raises the important issue of how to most efficiently digest full-text content. Although article titles and abstracts provide accurate and concise information on an article’s contents, their brevity inevitably entails the loss of detail. Full-text articles provide those details, but require more time to read. The primary goal of this study is to combine the advantages of concise abstracts and detail-rich full-texts to ease the burden of reading.

Results: We retrieved abstract-related paragraphs from full-text articles through shared keywords between the abstract and paragraphs from the main text. Significant paragraphs were then recommended by applying a proposed paragraph ranking approach. Finally, the user was provided with a condensed text consisting of these significant paragraphs, allowing the user to save time from perusing the whole article. We compared the performance of the proposed approach with a keyword counting approach and a PageRank-like approach. Evaluation was conducted in two aspects: the importance of each retrieved paragraph and the information coverage of a set of retrieved paragraphs. In both evaluations, the proposed approach outperformed the other approaches.

Contact: jchiang@mail.ncku.edu.tw

Received on December 5, 2010; revised on January 31, 2011; accepted on February 1, 2011

1 INTRODUCTION

In recent years, the development of bioinformatics and information technology has coincided with the rapid growth of biomedical literature databases, and the rate at which more full-text scientific literature becomes available on the web is accelerating. The U.S. National Center for Biotechnology Information (NCBI, under the U.S. National Library of Medicine) has collected more than 20 million biomedical study abstracts in its free-access literature database, PubMed, which acts as a portal site for various biomedical journals and provides a literature search service for browsing and retrieving abstracts. The service helps researchers keep up-to-date on biomedical progress and discoveries, and also reduces the risk of redundant experiments.

This overwhelming volume of information obviously defies manually exploration and, given time limitations, the ability of individuals and organizations to digest this flood of data is an issue of growing concern. Investigations into the automatic extraction of important information begun decades ago (Luhn, 1958). Recently, many text mining applications have been developed to raise efficiency and save time in literature reviews (Goetz and von der Lieth, 2005; Lin and Wilbur, 2007). GoPubMed (Doms and Schroeder, 2005) categorized PubMed search results into various topics to assist users in looking for articles relevant to their specific interests; Litispector (Frisch et al., 2009) enhanced search results by highlighting keywords with various colors according to predefined keyword categories. GeneLibrarian (Chiang et al., 2006) extracted specific information from PubMed abstracts to facilitate awareness of critical relationships between genes of interest. All of these applications use various methods to compile abstracts and present concise information.

In the past, obtaining full-text articles could be very inconvenient. However, these days more and more full-text literature is available online via digital literature databases. PubMed Central (PMC) and PubMed (both established by the NCBI), differ mainly in that PMC is dedicated to collecting full-text articles in the life science, while PubMed collects abstracts only. Both services offer free access to their collections and the PMC database currently consists of nearly 900 life science journals.

Following the increased availability of free full-text articles, text mining applications have allowed readers to peer into the main texts, providing full article searches which extract more complete and detailed information than would otherwise be available in the abstracts. However, directly applying tools designed for searching abstracts on a full-text corpus entails unacceptably high computation and performance costs. In recent years, new research has been proposed on full-text utilities. Lin (2009) compared search engine performance at various levels: abstract, whole article and sections of an article; Shah et al. (2003) studied keyword distribution and difference cross sections; Gay et al. (2005) investigated the performance of semi-automatic indexing on abstracts and full-text articles. All of these studies make clear that the development of tools for full-text use is a matter of great urgency.

The advantage of applications which search full-text articles is that the main text is where the details of the research are addressed. However, this data inevitably is accompanied by redundant or trivial description material. From the perspective of text mining, the variety of elements (e.g., text, equations, figures, tables, etc.) is a significant challenge for full-text information extraction (Laskowski, 2007), and information noise from redundant and trivial description may impair system performance. Abstracts contain much less information than the full-text articles, and they are well-structured and helpful for readers to quickly ascertain a paper’s purpose. An abstract briefly describes the research focus, methods, findings and main conclusion in a brief text.

In this study, we propose an approach to assist readers in digesting full-text more effectively and efficiently. Effective and
As mentioned above, a full-text article not only contains detailed research information but also text that is used to make the essay more complete and readable, such as transitional paragraphs connecting different subjects. This text usually does not provide crucial information, so the objective of our application is to eliminate useless information and to compile the full-text article in the required format.

In Section 2, we describe the proposed ranking framework and a metric named paragraph relevance--inverse sentence relevance (PR–ISR), which is used to evaluate the importance of paragraphs. Section 3 describes the materials and experimental settings used in performance evaluation. Section 4 presents the results of two perspective evaluations and conclusions are drawn in Section 5.

2 METHODS

First, paragraphs related to the abstract are retrieved, with relatedness determined based on the degree of information overlap. This study proposes a metric called PR–ISR to score the importance of paragraphs. By ranking these retrieved paragraphs, we can obtain a set of important abstract-related paragraphs which complement the topic mentioned in the abstract. Rather than reading the entire article, users can first read the summary information from the abstract, and then derive complementary information from the set of paragraphs. Figure 1 illustrates the workflow of this concept.

2.1 Preprocessing

As mentioned above, a full-text article not only contains detailed research information but also text that is used to make the essay more complete and readable, such as transitional paragraphs connecting different subjects. This text usually does not provide crucial information, so the objective of preprocessing was to eliminate useless information and to compile the full-text article in the required format.

2.1.1 Keyword extraction

In our application, keywords are a set of terms that can represent more specific ideas. In terms of the referential value of retrieved sentences to related paragraphs in the full text, therefore the selection of keywords will affect the referential value of retrieved paragraphs. For instance, general nouns usually represent general ideas and proper nouns represent more specific ideas. In terms of the referential value of retrieved paragraphs, proper nouns usually outperform general nouns. To raise the importance of paragraphs, proper nouns usually outperform general nouns. To raise the

2.1.2 Weight assignment

Keywords play the role of linking abstract sentences to related paragraphs in the full text, therefore the selection of keywords will affect the referential value of retrieved paragraphs. For instance, general nouns usually represent general ideas and proper nouns represent more specific ideas. In terms of the referential value of retrieved paragraphs, proper nouns usually outperform general nouns. To raise the

Table 1. Syntactic rules and examples

<table>
<thead>
<tr>
<th>Syntactic rules</th>
<th>Examples</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>/NNP</td>
<td>SWISS-PROT/NNP</td>
<td>2</td>
</tr>
<tr>
<td>/NNP + /NNP (NNP, NNS)</td>
<td>Sequence</td>
<td>2</td>
</tr>
<tr>
<td>/NNP + /NNP (NNP, NNS)</td>
<td>Microarray/NN + analysis/NN</td>
<td>2</td>
</tr>
<tr>
<td>/CD + /NNP (NNP, NNS)</td>
<td>121000/CD + oligonucleotide/NN</td>
<td>2</td>
</tr>
<tr>
<td>/NN (NN, N)</td>
<td>Literature/NN</td>
<td>1</td>
</tr>
</tbody>
</table>

Rules are used to extract keywords and each keyword is assigned an initial weight by rules' weight.

a document. However, this method was generally found to give poor results, and interest turned to other methods. In recent years, several supervised machine learning algorithms have been proposed for classifying candidate terms as keywords (Hulth, 2003; Turney, 2000). However, the training phase of machine learning approaches requires extra training corpora and computational resources, rendering them less economic than rule based approaches.

Using a combination of lexical and syntactic features, Hulth (2003) significantly improved keyword extraction performance over previously published results, suggesting that syntactic information may play an important role in keyword extraction. In our observations, informative keywords usually appear as nouns or noun phrases. We thus designed a syntactic rule-based keyword extractor.

The proposed keyword extractor identified keywords by part-of-speech (POS) tags and a set of syntactic rules. Our study used the Stanford Log-linear Part-Of-Speech Tagger (Toutanova et al., 2003) to tag the parts of speech of words in abstracts at the sentence level. After tagging, every abstract sentence would be turned into a sequence of POS tags. Terms or phrases which matched the patterns of our syntactic rules would be extracted as candidate keywords/key-phrases. In this study we considered only noun or noun phrases as candidates, and the syntactic rules were listed in Table 1. MeSH vocabulary also was included as keywords.

2.1.1 Syntactic rules and examples

<table>
<thead>
<tr>
<th>Syntactic rules</th>
<th>Examples</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>/NNP</td>
<td>SWISS-PROT/NNP</td>
<td>2</td>
</tr>
<tr>
<td>/NNP + /NNP (NNP, NNS)</td>
<td>Sequence</td>
<td>2</td>
</tr>
<tr>
<td>/NNP + /NNP (NNP, NNS)</td>
<td>Microarray/NN + analysis/NN</td>
<td>2</td>
</tr>
<tr>
<td>/CD + /NNP (NNP, NNS)</td>
<td>121000/CD + oligonucleotide/NN</td>
<td>2</td>
</tr>
<tr>
<td>/NN (NN, N)</td>
<td>Literature/NN</td>
<td>1</td>
</tr>
</tbody>
</table>

Rules are used to extract keywords and each keyword is assigned an initial weight by rules’ weight.

In information retrieval applications, one of the simplest possible approaches is to use a frequency criterion to select ‘important’ keywords in a document. However, this method was generally found to give poor results, and interest turned to other methods. In recent years, several supervised machine learning algorithms have been proposed for classifying candidate terms as keywords (Hulth, 2003; Turney, 2000). However, the training phase of machine learning approaches requires extra training corpora and computational resources, rendering them less economic than rule based approaches.

Using a combination of lexical and syntactic features, Hulth (2003) significantly improved keyword extraction performance over previously published results, suggesting that syntactic information may play an important role in keyword extraction. In our observations, informative keywords usually appear as nouns or noun phrases. We thus designed a syntactic rule-based keyword extractor.

The proposed keyword extractor identified keywords by part-of-speech (POS) tags and a set of syntactic rules. Our study used the Stanford Log-linear Part-Of-Speech Tagger (Toutanova et al., 2003) to tag the parts of speech of words in abstracts at the sentence level. After tagging, every abstract sentence would be turned into a sequence of POS tags. Terms or phrases which matched the patterns of our syntactic rules would be extracted as candidate keywords/key-phrases. In this study we considered only noun or noun phrases as candidates, and the syntactic rules were listed in Table 1. MeSH vocabulary also was included as keywords.

2.1.1 Syntactic rules and examples

<table>
<thead>
<tr>
<th>Syntactic rules</th>
<th>Examples</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>/NNP</td>
<td>SWISS-PROT/NNP</td>
<td>2</td>
</tr>
<tr>
<td>/NNP + /NNP (NNP, NNS)</td>
<td>Sequence</td>
<td>2</td>
</tr>
<tr>
<td>/NNP + /NNP (NNP, NNS)</td>
<td>Microarray/NN + analysis/NN</td>
<td>2</td>
</tr>
<tr>
<td>/CD + /NNP (NNP, NNS)</td>
<td>121000/CD + oligonucleotide/NN</td>
<td>2</td>
</tr>
<tr>
<td>/NN (NN, N)</td>
<td>Literature/NN</td>
<td>1</td>
</tr>
</tbody>
</table>

Rules are used to extract keywords and each keyword is assigned an initial weight by rules’ weight.

In information retrieval applications, one of the simplest possible approaches is to use a frequency criterion to select ‘important’ keywords in a document. However, this method was generally found to give poor results, and interest turned to other methods. In recent years, several supervised machine learning algorithms have been proposed for classifying candidate terms as keywords (Hulth, 2003; Turney, 2000). However, the training phase of machine learning approaches requires extra training corpora and computational resources, rendering them less economic than rule based approaches.

Using a combination of lexical and syntactic features, Hulth (2003) significantly improved keyword extraction performance over previously published results, suggesting that syntactic information may play an important role in keyword extraction. In our observations, informative keywords usually appear as nouns or noun phrases. We thus designed a syntactic rule-based keyword extractor.

The proposed keyword extractor identified keywords by part-of-speech (POS) tags and a set of syntactic rules. Our study used the Stanford Log-linear Part-Of-Speech Tagger (Toutanova et al., 2003) to tag the parts of speech of words in abstracts at the sentence level. After tagging, every abstract sentence would be turned into a sequence of POS tags. Terms or phrases which matched the patterns of our syntactic rules would be extracted as candidate keywords/key-phrases. In this study we considered only noun or noun phrases as candidates, and the syntactic rules were listed in Table 1. MeSH vocabulary also was included as keywords.

2.1.1 Syntactic rules and examples

<table>
<thead>
<tr>
<th>Syntactic rules</th>
<th>Examples</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>/NNP</td>
<td>SWISS-PROT/NNP</td>
<td>2</td>
</tr>
<tr>
<td>/NNP + /NNP (NNP, NNS)</td>
<td>Sequence</td>
<td>2</td>
</tr>
<tr>
<td>/NNP + /NNP (NNP, NNS)</td>
<td>Microarray/NN + analysis/NN</td>
<td>2</td>
</tr>
<tr>
<td>/CD + /NNP (NNP, NNS)</td>
<td>121000/CD + oligonucleotide/NN</td>
<td>2</td>
</tr>
<tr>
<td>/NN (NN, N)</td>
<td>Literature/NN</td>
<td>1</td>
</tr>
</tbody>
</table>

Rules are used to extract keywords and each keyword is assigned an initial weight by rules’ weight.

In information retrieval applications, one of the simplest possible approaches is to use a frequency criterion to select ‘important’ keywords in a document. However, this method was generally found to give poor results, and interest turned to other methods. In recent years, several supervised machine learning algorithms have been proposed for classifying candidate terms as keywords (Hulth, 2003; Turney, 2000). However, the training phase of machine learning approaches requires extra training corpora and computational resources, rendering them less economic than rule based approaches.

Using a combination of lexical and syntactic features, Hulth (2003) significantly improved keyword extraction performance over previously published results, suggesting that syntactic information may play an important role in keyword extraction. In our observations, informative keywords usually appear as nouns or noun phrases. We thus designed a syntactic rule-based keyword extractor.

The proposed keyword extractor identified keywords by part-of-speech (POS) tags and a set of syntactic rules. Our study used the Stanford Log-linear Part-Of-Speech Tagger (Toutanova et al., 2003) to tag the parts of speech of words in abstracts at the sentence level. After tagging, every abstract sentence would be turned into a sequence of POS tags. Terms or phrases which matched the patterns of our syntactic rules would be extracted as candidate keywords/key-phrases. In this study we considered only noun or noun phrases as candidates, and the syntactic rules were listed in Table 1. MeSH vocabulary also was included as keywords.

2.1.1 Syntactic rules and examples

<table>
<thead>
<tr>
<th>Syntactic rules</th>
<th>Examples</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>/NNP</td>
<td>SWISS-PROT/NNP</td>
<td>2</td>
</tr>
<tr>
<td>/NNP + /NNP (NNP, NNS)</td>
<td>Sequence</td>
<td>2</td>
</tr>
<tr>
<td>/NNP + /NNP (NNP, NNS)</td>
<td>Microarray/NN + analysis/NN</td>
<td>2</td>
</tr>
<tr>
<td>/CD + /NNP (NNP, NNS)</td>
<td>121000/CD + oligonucleotide/NN</td>
<td>2</td>
</tr>
<tr>
<td>/NN (NN, N)</td>
<td>Literature/NN</td>
<td>1</td>
</tr>
</tbody>
</table>

Rules are used to extract keywords and each keyword is assigned an initial weight by rules’ weight.
The first consideration is keyword/information overlapping (IO). For a query sentence \( QS \) and a paragraph \( P \), term coverage was defined as:

\[
TCover(QS, P) = \frac{|\{ t | t \in QS \cap t \in P \}|}{|QS|}
\]  

(1)

TCover indicates the proportion of keywords of \( QS \) appearing in a paragraph. To link query sentences to paragraphs, we can highlight paragraphs which share more keywords with \( QS \); that is, paragraphs with a higher TCover score might convey more information related to a query sentence.

In biomedical articles, abstracts are usually written in complex sentences and it is likely that multiple keywords will be mentioned in a single sentence. In this case, TCover would prefer introductory paragraphs because they tend to mention various keywords in their content. But, TCover was unable to reflect the value of paragraphs which focused only on a few keywords of \( QS \). To resolve the issue, we designed an indicator to quantify a paragraph’s specificity to \( QS \). The specificity of paragraph \( P \) to \( QS \) was defined as:

\[
\text{Specificity}(QS, P) = \frac{1}{|P|} \sum_{i=0}^{N} \sum_{j=0}^{M} \delta_{ij}
\]  

(2)

where \( S_i \) is \( i \)-th sentence of \( P \), and \( |P| \) is the number of sentences in a paragraph. To avoid promoting long paragraphs, the frequency of present keywords is normalized by the number of sentences in each paragraph. However, Equation (2) ignored the referential value of keywords; general keywords and informative ones were treated equivalently. To promote informative keywords, we introduced their weights into Equation (2) and defined weighted specificity as:

\[
\text{WS}(QS, P) = \frac{1}{|P|} \sum_{i=0}^{N} \sum_{j=0}^{M} \delta_{ij} w_j
\]  

(3)

where \( w_j \) is the weight of keyword \( t_j \).

Finally, the relevance score of a paragraph to the query sentence is defined as:

\[
\text{PR}(QS, P) = TCover(QS, P) \times \text{WS}(QS, P)
\]  

(4)

Subsequently, we could use relevance score as a measure to evaluate a paragraph’s relevance to a query sentence.

### 2.3 PR–ISR

In information retrieval, term frequency–inversed document frequency (TF–IDF) is a well-known statistical measurement used to evaluate how important a word is to a document corpus. The importance rises proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus. We think that a similar relationship exists between \( QS \) and paragraphs: the relevance of a paragraph to \( QS \) rises proportionally to PR score of the paragraphs and is offset by summation of PR scores of the paragraphs to all \( QS \)s. Therefore, we propose a measure called PR–ISR to assess how relevant a paragraph is to a \( QS \). When a paragraph was considered to be relevant to a sentence, we expect the paragraph to contain complementary information for certain concepts addressed in the sentence. As the PR score was an indicator of the complementariness of a paragraph, a higher PR score means that a paragraph is more relevant to a query sentence. However, when the paragraph is relevant to many query sentences, it implies that the content of the paragraph is general, thus its ranking should be lowered. We define a value called inverse sentence relevance (ISR) to reflect this characteristic, and define the PR–ISR of paragraph \( i \) to sentence \( j \) as:

\[
\text{PRISR}_{ij} = \text{PR}_{ij} \times \text{ISR}_j
\]  

(5)

\[
\text{ISR}_j = \log \left( \frac{|S_j|}{\sum_{k=0}^{N} \text{PR}_{kj} \times |S_k|} \right)
\]  

(6)

where \( |S_j| \) is the number of sentences in the abstract. Through this definition, a paragraph relevant to a greater number of sentences in the abstract results...
in a higher sum of PR scores, thus diluting the influence of its PR score and resulting in a lower ranking. Accordingly, the relevance of paragraph \( i \) to sentence \( j \) increased proportionally to PR score but was offset by the sum of PR scores to all query sentences. By determining the PR-ISR of every paragraph to a target sentence, paragraphs can be ranked by this value, thus identifying the most relevant and specific paragraph for a given sentence.

3 EXPERIMENTS

3.1 Experimental settings

In this study, we conducted a preliminary experiment to evaluate whether IO correlated with the importance of paragraphs. Consequently, we evaluated the performance of the retrieval algorithms in two aspects. One was to evaluate the agreement of results of single paragraph retrieval with human opinion, using a small annotated corpus. The other was to evaluate the qualities of condensed text, using a recall-oriented metric.

3.2 Materials

The source of the full-text corpus for experiments is the NCBI's PMC. We randomly selected 1000 BMC Bioinformatics articles on the PMC ftp site. To compare human opinions against the algorithm retrieval results, 10% of the 1000 articles were annotated by eight human assessors experienced in reading scientific articles in the field of bioinformatics. Each paragraph was annotated with either related or unrelated and labeled with an importance level from 1 to 5, indicating the relative importance of each paragraph. A total of 3610 paragraphs from 100 articles were annotated.

3.3 Algorithms for comparison

3.3.1 Keyword-count approach

We designed a baseline system to retrieve paragraphs that contained the most shared terms of a query sentence. The query sentence was also represented in the bag-of-words model, and terms were selected using the same process as the proposed approach. In the baseline system, however, term weighting was not considered and, unlike in the proposed approach, all terms were treated equally.

3.3.2 PageRank-like approach

PageRank (Brin and Page, 1998) is a graph-based ranking algorithm used by Google’s search engine to assigns numerical scores to web pages through analysis of the hyperlink structure of the World Wide Web. The basic idea implemented in PageRank is that of “voting”. A link to a page counts as a vote of support. A page having more incoming links will gain more support, thus making it more important. In addition, the importance of the page casting the vote determines how important the vote itself is. That is to say, importance of a page is derived from how many pages link to it and how important these pages themselves are. In practice, the importance score (IS) of page \( V_i \) can be derived from recursive calculation through following equation:

\[
S(V_i) = (1-d) + d \sum_{j \in \text{In}(V_i)} \frac{S(V_j)}{\text{Out}(V_j)}
\]

where \( \text{In}(V_i) \) are pages which link to \( V_i \), and \( \text{Out}(V_j) \) are pages to which \( V_j \) links. In the above equation, \( d \) is a damping factor.

In this study, the first step of applying the PageRank approach was the modeling of a full-text article as an undirected graph. Paragraphs were regarded as vertices, and linkages were defined by the existence of keywords shared between paragraphs. In contrast to Web, we modeled the paragraph network as an undirected graph (Mihalcea and Tarau, 2004), and named this approach PageRank-like approach.

Similar to a search engine, relevant paragraph to a query sentence was determined by the number of shared keywords and the scores of paragraphs. Candidate paragraphs were those sharing shared keywords with the query sentence. Candidate paragraphs were ranked first by the number of shared keywords and then by the paragraph’s score. Therefore, the paragraph containing the most shared keywords and having the highest score would be recommended as the query sentence’s most relevant paragraph.

3.4 IO and paragraph importance

The determination of importance is very subjective and may vary among different readers. A more objective comparison requires a metric which can objectively reflect the opinions of most assessors. We conducted an experiment to evaluate the correlation between IO and the importance of given paragraphs. Here we regard an abstract as a set of keywords, and the IO between a paragraph and the abstract was defined as:

\[
\text{IO}(A, P) = \sum_{i=1}^{|P|} \delta(S_i, A)
\]

where \( \delta(S_i, A) = \begin{cases} 1 & S_i \text{ contains keywords of } A \\ 0 & \text{otherwise} \end{cases} \)

where \( A \) is the abstract, \( P \) is a paragraph of full text and \( S_i \) is the \( i \)-th sentence in \( P \). In this experiment, \( P \) was derived from 3610 paragraphs annotated for importance and \( A \) was the abstract.
The results indicate that IO and a paragraph’s importance are positively correlated.

3.5 Assessing agreement with human opinion

To evaluate the capacity to recommend truly useful paragraphs to readers, we conducted an experiment to assess whether paragraphs recommended by our method agreed with those chosen by human reviewers, based on the 100 manually annotated articles. Each paragraph was annotated with an importance level from 1 to 5. For every query sentence, the algorithm (keyword count or PageRank-like, Section 3.3) recommended the most relevant paragraph. The algorithm then was scored according to importance levels of the paragraphs. For example, the algorithm obtained a score of 1 when it selected a paragraph labeled level 1. The algorithm achieving a higher average score was thought to be better able to recommend important paragraphs.

3.6 Evaluation qualities of condensed text

Our system recommended one most appropriate paragraph for each target sentence, and integrated all these recommended paragraphs into a short text that ideally addressed most of the significant information which authors intended to convey in the article. We ran the experiment on 1000 full-text articles to evaluate the system’s performance.

Recall-Oriented Understudy for Gisting Evaluation (ROUGE) is a set of metrics used for evaluating automatic summarization software in natural language processing (Lin, 2004). ROUGE measures summary quality by counting overlapping units such as the n-gram, word sequences and word pairs between the candidate summary and the reference summary. ROUGE-N is an n-gram recall measure computed as follows:

$$\text{ROUGE-N} = \frac{\sum_{S \in \text{reference}} \sum_{n-grams \in S} \text{Countmatch}(n-gram)}{\sum_{S \in \text{reference}} \sum_{n-grams \in S} \text{Count}(n-gram) \times \text{Countmatch}(n-gram)}$$  (10)

where $n$ stands for the length of the n-gram, and Countmatch(n-gram) is the maximum number of n-grams co-occurring in a candidate summary and a set of reference summaries. Count(n-gram) is the number of n-grams in the reference summaries.

Average IO for various levels of importance is shown in Figure 3. The results indicate that IO and a paragraph’s importance are positively correlated.

Fig. 3. Average IO increasing with importance level.

In addition, the ratio of paragraphs marked as abstract-related correlated positively with importance level (Fig. 4). Abstract-related paragraphs made up >80% of important paragraphs, and 96% of paragraphs labeled with importance level 5 were considered to be abstract-related. These results suggest that taking advantage of information in the abstract could contribute to the retrieval of important paragraphs from full-text articles.

Fig. 4. Annotation results of 3610 paragraphs. The percentage refers to proportion of abstract-related paragraphs of individual importance level. Result shows that abstract-related paragraphs were in the majority among high importance level collections.

Generally speaking, ROUGE-1 is a metric of informativeness, while ROUGE-2, which considers bi-gram, and ROUGE-L, which considers the longest common subsequence, are metrics of fluency and grammaticality, respectively. Among these different scores, the uni-gram-based ROUGE score (ROUGE-1) has been shown to best agree with human assessment. In this study we focused on a paragraph’s informativeness, and thus applied ROUGE-1.

Assessing the quality of the condensed text requires a reference text. Here, IO score was adopted as an index to classify paragraphs into notable and non-notable categories. The notable reference was used to assess the systems’ capability to retrieve notable information.

4 RESULTS AND DISCUSSION

4.1 Analysis of annotation of paragraphs

According to the annotation result, only 28% of the 3610 annotated paragraphs were considered to be important (i.e. labeled with 4 or 5), while 40% were considered to be trivial descriptions (i.e. labeled with 1 or 2). Not all paragraphs facilitated understanding of an article’s critical information, and the important paragraphs made up only small parts of the full text articles. That is, a literature review would be more efficient if researchers could focus on important paragraphs. The result addressed a need for an automatic paragraph recommendation system.

In addition, the ratio of paragraphs marked as abstract-related correlated positively with importance level (Fig. 4). Abstract-related paragraphs made up >80% of important paragraphs, and 96% of paragraphs labeled with importance level 5 were considered to be abstract-related. These results suggest that taking advantage of information in the abstract could contribute to the retrieval of important paragraphs from full-text articles.

4.2 Importance of retrieved paragraphs

The results of an experiment to assess the agreement with manual reviews is shown in Figure 5; PR–ISR achieved an average IS of 3.69, compared to 2.98 for the keyword-count approach. 3.26 for PageRank-like and 1.79 for a random selection. Together, Figures 4 and 5 show a higher proportion of sentences containing keywords is in important paragraphs, which explains the better performance
We used ROUGE-1 as the metric for evaluating the quality of words. But, in this evaluation, the keyword-count approach did paragraphs achieve a higher recall rate because they contain more paragraphs was longer in the keyword-count approach results. Long category. In addition, on average the length of the recommended paragraphs which were in condensed texts belonged to the notable category.

The result is shown in Figure 6. All three methods obtained higher ROUGE-1 scores in the notable categories, indicating that most paragraphs which were in condensed texts belonged to the notable category.

Compared with the keyword-count approach, PR–ISR obtained the highest scores in the notable category, but the difference in the non-notable category was negligible, indicating that paragraphs recommended by PR–ISR were more concentrated in the notable category. In addition, on average the length of the recommended paragraphs was longer in the keyword-count approach results. Long paragraphs achieve a higher recall rate because they contain more words. But, in this evaluation, the keyword-count approach did not outperform PR–ISR, most likely due to the effect of ISR. As mentioned previously, ISR was designed to lower a paragraph’s ranking when it is relevant to many sentences, which indicates that the paragraph’s content may be too general.

Thus, PR–ISR reduced the rankings of non-notable paragraphs through ISR and retrieved more notable paragraphs, even though they were shorter in length.

5 CONCLUSION

As the availability of full-text articles grows, how to use the resource effectively and efficiently becomes an important issue. In this article, we proposed a paragraph-ranking approach to produce mid-size texts to assist literature review. The annotation result indicates that not all the selected paragraphs contribute to a good understanding of the literature content, thus a literature review would be more efficient if researchers could focus on important paragraphs. Abstracts and full-text articles each have distinct advantages: abstracts are well-structured short texts expressing the main research focus, while full-text papers contain many details of that research. Readers may thus find it helpful to have a condensed text that combines the advantages of both.

In our evaluation, the outcome of the proposed approach outperformed the other two algorithms. Paragraph-ranking achieves the highest average IS, that is, the approach best agreed with results of manual reviews.

Paragraph-ranking retrieves paragraphs using keywords as query terms. That is, determining the relation between the query sentence and paragraphs requires the existence of shared keywords. However, we found that a small number of paragraphs were labeled with high importance level by curators, but do not mention any keywords. In fact, they do include some valuable information. Deeper semantic analysis is required to produce a semantically enhanced recommendation, and we note this issue for future work.

6 ACKNOWLEDGEMENTS

The authors would like to thank Prof. Tien-Hao Chang for technical assistance and helpful discussions.

Funding: National Science Council, Taiwan (NSC 99-2627-B-006-013 and NSC 96-2628-E-006-132) in part.

Conflict of Interest: none declared.
REFERENCES


