Towards an Iterative Reinforcement Approach for Simultaneous Document Summarization and Keyword Extraction

Xiaojun Wan **Jianwu Yang** **Jianguo Xiao**
Institute of Computer Science and Technology
Peking University, Beijing 100871, China
{wanxiaojun,yangjianwu,xiaojianguo}@icst.pku.edu.cn

Abstract

Though both document summarization and keyword extraction aim to extract concise representations from documents, these two tasks have usually been investigated independently. This paper proposes a novel iterative reinforcement approach to simultaneously extracting summary and keywords from single document under the assumption that the summary and keywords of a document can be mutually boosted. The approach can naturally make full use of the reinforcement between sentences and keywords by fusing three kinds of relationships between sentences and words, either homogeneous or heterogeneous. Experimental results show the effectiveness of the proposed approach for both tasks. The corpus-based approach is validated to work almost as well as the knowledge-based approach for computing word semantics.

1 Introduction

Text summarization is the process of creating a compressed version of a given document that delivers the main topic of the document. Keyword extraction is the process of extracting a few salient words (or phrases) from a given text and using the words to represent the text. The two tasks are similar in essence because they both aim to extract concise representations for documents. Automatic text summarization and keyword extraction have drawn much attention for a long time because they both are very important for many text applications, including document retrieval, document clustering, etc. For example, keywords of a document can be used for document indexing and thus benefit to improve the performance of document retrieval, and document summary can help to facilitate users to browse the search results and improve users' search experience.

Text summaries and keywords can be either query-relevant or generic. Generic summary and keyword should reflect the main topics of the document without any additional clues and prior knowledge. In this paper, we focus on generic document summarization and keyword extraction for single documents.

Document summarization and keyword extraction have been widely explored in the natural language processing and information retrieval communities. A series of workshops and conferences on automatic text summarization (e.g. SUMMAC, DUC and NTCIR) have advanced the technology and produced a couple of experimental online systems. In recent years, graph-based ranking algorithms have been successfully used for document summarization (Mihalcea and Tarau, 2004, 2005; ErKan and Radev, 2004) and keyword extraction (Mihalcea and Tarau, 2004). Such algorithms make use of "voting" or "recommendations" between sentences (or words) to extract sentences (or keywords). Though the two tasks essentially share much in common, most algorithms have been developed particularly for either document summarization or keyword extraction.

Zha (2002) proposes a method for simultaneous keyphrase extraction and text summarization by using only the heterogeneous sentence-to-word relationships. Inspired by this, we aim to take into account all the three kinds of relationships among sentences and words (i.e. the homogeneous relationships between words, the homogeneous relationships between sentences, and the heterogeneous relationships between words and sentences) in

Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pages 552–559,
Prague, Czech Republic, June 2007.

a unified framework for both document summarization and keyword extraction. The importance of a sentence (word) is determined by both the importance of related sentences (words) and the importance of related words (sentences). The proposed approach can be considered as a generalized form of previous graph-based ranking algorithms and Zha's work (Zha, 2002).

In this study, we propose an iterative reinforcement approach to realize the above idea. The proposed approach is evaluated on the DUC2002 dataset and the results demonstrate its effectiveness for both document summarization and keyword extraction. Both knowledge-based approach and corpus-based approach have been investigated to compute word semantics and they both perform very well.

The rest of this paper is organized as follows: Section 2 introduces related works. The details of the proposed approach are described in Section 3. Section 4 presents and discusses the evaluation results. Lastly we conclude our paper in Section 5.

2 Related Works

2.1 Document Summarization

Generally speaking, single document summarization methods can be either extraction-based or abstraction-based and we focus on extraction-based methods in this study.

Extraction-based methods usually assign a saliency score to each sentence and then rank the sentences in the document. The scores are usually computed based on a combination of statistical and linguistic features, including term frequency, sentence position, cue words, stigma words, topic signature (Hovy and Lin, 1997; Lin and Hovy, 2000), etc. Machine learning methods have also been employed to extract sentences, including unsupervised methods (Nomoto and Matsumoto, 2001) and supervised methods (Kupiec et al., 1995; Conroy and O'Leary, 2001; Amini and Gallinari, 2002; Shen et al., 2007). Other methods include maximal marginal relevance (MMR) (Carbonell and Goldstein, 1998), latent semantic analysis (LSA) (Gong and Liu, 2001). In Zha (2002), the mutual reinforcement principle is employed to iteratively extract key phrases and sentences from a document.

Most recently, graph-based ranking methods, including TextRank ((Mihalcea and Tarau, 2004, 2005) and LexPageRank (ErKan and Radev, 2004) have been proposed for document summarization. Similar to Kleinberg's HITS algorithm (Kleinberg, 1999) or Google's PageRank (Brin and Page, 1998), these methods first build a graph based on the similarity between sentences in a document and then the importance of a sentence is determined by taking into account global information on the graph recursively, rather than relying only on local sentence-specific information.

2.2 Keyword Extraction

Keyword (or keyphrase) extraction usually involves assigning a saliency score to each candidate keyword by considering various features. Krulwich and Burkey (1996) use heuristics to extract keyphrases from a document. The heuristics are based on syntactic clues, such as the use of italics, the presence of phrases in section headers, and the use of acronyms. Muñoz (1996) uses an unsupervised learning algorithm to discover two-word keyphrases. The algorithm is based on Adaptive Resonance Theory (ART) neural networks. Steier and Belew (1993) use the mutual information statistics to discover two-word keyphrases.

Supervised machine learning algorithms have been proposed to classify a candidate phrase into either keyphrase or not. GenEx (Turney, 2000) and Kea (Frank et al., 1999; Witten et al., 1999) are two typical systems, and the most important features for classifying a candidate phrase are the frequency and location of the phrase in the document. More linguistic knowledge (such as syntactic features) has been explored by Hulth (2003). More recently, Mihalcea and Tarau (2004) propose the TextRank model to rank keywords based on the co-occurrence links between words.

3 Iterative Reinforcement Approach

3.1 Overview

The proposed approach is intuitively based on the following assumptions:

Assumption 1: A sentence should be salient if it is heavily linked with other salient sentences, and a word should be salient if it is heavily linked with other salient words.

Assumption 2: A sentence should be salient if it contains many salient words, and a word should be salient if it appears in many salient sentences.

The first assumption is similar to PageRank which makes use of mutual "recommendations" between homogeneous objects to rank objects. The second assumption is similar to HITS if words and sentences are considered as authorities and hubs respectively. In other words, the proposed approach aims to fuse the ideas of PageRank and HITS in a unified framework.

In more detail, given the heterogeneous data points of sentences and words, the following three kinds of relationships are fused in the proposed approach:

SS-Relationship: It reflects the homogeneous relationships between sentences, usually computed by their content similarity.

WW-Relationship: It reflects the homogeneous relationships between words, usually computed by knowledge-based approach or corpus-based approach.

SW-Relationship: It reflects the heterogeneous relationships between sentences and words, usually computed as the relative importance of a word in a sentence.

Figure 1 gives an illustration of the relationships.

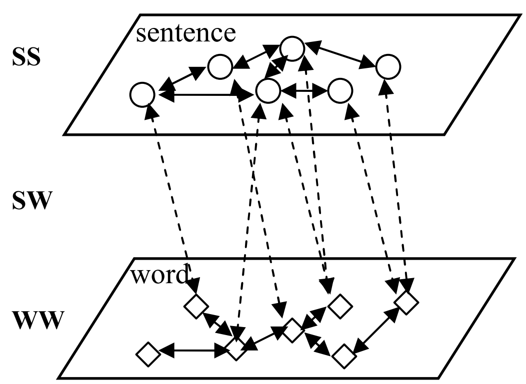

Figure 1. Illustration of the Relationships

The proposed approach first builds three graphs to reflect the above relationships respectively, and then iteratively computes the saliency scores of the sentences and words based on the graphs. Finally, the algorithm converges and each sentence or word gets its saliency score. The sentences with high saliency scores are chosen into the summary, and the words with high saliency scores are combined to produce the keywords.

3.2 Graph Building

3.2.1 Sentence-to-Sentence Graph (SS-Graph)

Given the sentence collection $S=\{s_i \mid 1 \le i \le m\}$ of a document, if each sentence is considered as a node, the sentence collection can be modeled as an undirected graph by generating an edge between two sentences if their content similarity exceeds 0, i.e. an undirected link between s_i and s_j ($i \neq j$) is constructed and the associated weight is their content similarity. Thus, we construct an undirected graph G_{SS} to reflect the homogeneous relationship between sentences. The content similarity between two sentences is computed with the cosine measure. We use an adjacency matrix $\boldsymbol{U}$ to describe G_{SS} with each entry corresponding to the weight of a link in the graph. $\boldsymbol{U}=[U_{ij}]_{m \times m}$ is defined as follows:

$$U_{ij} = \begin{cases} \dfrac{\vec{s}_i \cdot \vec{s}_j}{\|\vec{s}_i\| \times \|\vec{s}_j\|}, & if\ i \neq j \\ 0, & otherwise \end{cases} \quad (1)$$

where $\vec{s}_i$ and $\vec{s}_j$ are the corresponding term vectors of sentences s_i and s_j respectively. The weight associated with term t is calculated with $tf_t.isf_t$, where tf_t is the frequency of term t in the sentence and isf_t is the inverse sentence frequency of term t, i.e. $1+\log(N/n_t)$, where N is the total number of sentences and n_t is the number of sentences containing term t in a background corpus. Note that other measures (e.g. Jaccard, Dice, Overlap, etc.) can also be explored to compute the content similarity between sentences, and we simply choose the cosine measure in this study.

Then $\boldsymbol{U}$ is normalized to $\widetilde{\boldsymbol{U}}$ as follows to make the sum of each row equal to 1:

$$\widetilde{U}_{ij} = \begin{cases} U_{ij} \Big/ \sum_{j=1}^{m} U_{ij}, & if\ \sum_{j=1}^{m} U_{ij} \neq 0 \\ 0, & otherwise \end{cases} \quad (2)$$

3.2.2 Word-to-Word Graph (WW-Graph)

Given the word collection $T=\{t_j \mid 1 \le j \le n\}$ of a document[1], the semantic similarity between any two words t_i and t_j can be computed using approaches that are either knowledge-based or corpus-based (Mihalcea et al., 2006).

Knowledge-based measures of word semantic similarity try to quantify the degree to which two words are semantically related using information drawn from semantic networks. WordNet (Fellbaum, 1998) is a lexical database where each

[1] The stopwords defined in the Smart system have been removed from the collection.

unique meaning of a word is represented by a synonym set or *synset*. Each synset has a gloss that defines the concept that it represents. Synsets are connected to each other through explicit semantic relations that are defined in WordNet. Many approaches have been proposed to measure semantic relatedness based on WordNet. The measures vary from simple edge-counting to attempt to factor in peculiarities of the network structure by considering link direction, relative path, and density, such as *vector*, *lesk*, *hso, lch*, *wup*, *path*, *res, lin* and *jcn* (Pedersen et al., 2004). For example, “cat” and “dog” has higher semantic similarity than “cat” and “computer”. In this study, we implement the *vector* measure to efficiently evaluate the similarities of a large number of word pairs. The *vector* measure (Patwardhan, 2003) creates a co–occurrence matrix from a corpus made up of the WordNet glosses. Each content word used in a WordNet gloss has an associated context vector. Each gloss is represented by a gloss vector that is the average of all the context vectors of the words found in the gloss. Relatedness between concepts is measured by finding the cosine between a pair of gloss vectors.

Corpus-based measures of word semantic similarity try to identify the degree of similarity between words using information exclusively derived from large corpora. Such measures as mutual information (Turney 2001), latent semantic analysis (Landauer et al., 1998), log-likelihood ratio (Dunning, 1993) have been proposed to evaluate word semantic similarity based on the co-occurrence information on a large corpus. In this study, we simply choose the mutual information to compute the semantic similarity between word t_i and t_j as follows:

$$sim(t_i,t_j) \quad \log\frac{N \quad p(t_i,t_j)}{p(t_i) \quad p(t_j)} \tag{3}$$

which indicates the degree of statistical dependence between t_i and t_j. Here, N is the total number of words in the corpus and $p(t_i)$ and $p(t_j)$ are respectively the probabilities of the occurrences of t_i and t_j, i.e. $count(t_i)/N$ and $count(t_j)/N$, where $count(t_i)$ and $count(t_j)$ are the frequencies of t_i and t_j. $p(t_i, t_j)$ is the probability of the co-occurrence of t_i and t_j within a window with a predefined size k, i.e. $count(t_i, t_j)/N$, where $count(t_i, t_j)$ is the number of the times t_i and t_j co-occur within the window.

Similar to the SS-Graph, we can build an undirected graph G_{WW} to reflect the homogeneous relationship between words, in which each node corresponds to a word and the weight associated with the edge between any different word t_i and t_j is computed by either the WordNet-based *vector* measure or the corpus-based mutual information measure. We use an adjacency matrix $\boldsymbol{V}$ to describe G_{WW} with each entry corresponding to the weight of a link in the graph. $\boldsymbol{V}= [V_{ij}]_{n\times n}$, where $V_{ij} =sim(t_i, t_j)$ if $i\neq j$ and $V_{ij}=0$ if $i=j$.

Then $\boldsymbol{V}$ is similarly normalized to $\widetilde{\boldsymbol{V}}$ to make the sum of each row equal to 1.

3.2.3 Sentence-to-Word Graph (SW-Graph)

Given the sentence collection $S=\{s_i \mid 1\le i\le m\}$ and the word collection $T=\{t_j|1\le j\le n\}$ of a document, we can build a weighted bipartite graph G_{SW} from S and T in the following way: if word t_j appears in sentence s_i, we then create an edge between s_i and t_j. A nonnegative weight $aff(s_i,t_j)$ is specified on the edge, which is proportional to the importance of word t_j in sentence s_i, computed as follows:

$$aff(s_i,t_j) \quad \frac{tf_{t_j} \quad isf_{t_j}}{\sum_{t \quad s_i} tf_t \quad isf_t} \tag{4}$$

where t represents a unique term in s_i and tf_t, isf_t are respectively the term frequency in the sentence and the inverse sentence frequency.

We use an adjacency (affinity) matrix $\boldsymbol{W}=[W_{ij}]_{m\times n}$ to describe G_{SW} with each entry W_{ij} corresponding to $aff(s_i,t_j)$. Similarly, $\boldsymbol{W}$ is normalized to $\widetilde{\boldsymbol{W}}$ to make the sum of each row equal to 1. In addition, we normalize the transpose of $\boldsymbol{W}$, i.e. $\boldsymbol{W}^{\mathrm{T}}$, to $\hat{\boldsymbol{W}}$ to make the sum of each row in $\boldsymbol{W}^{\mathrm{T}}$ equal to 1.

3.3 Reinforcement Algorithm

We use two column vectors $\boldsymbol{u}=[u(s_i)]_{m\times 1}$ and $\boldsymbol{v}=[v(t_j)]_{n\times 1}$ to denote the saliency scores of the sentences and words in the specified document. The assumptions introduced in Section 3.1 can be rendered as follows:

$$u(s_i) \quad \sum_j \widetilde{U}_{ji} u(s_j) \tag{5}$$

$$v(t_j) \quad \sum_i \widetilde{V}_{ij} v(t_i) \tag{6}$$

$$u(s_i) \quad \sum_j \hat{W}_{ji} v(t_j) \tag{7}$$

$$v(t_j) = \sum_i \widetilde{W}_{ij} u(s_i) \tag{8}$$

After fusing the above equations, we can obtain the following iterative forms:

$$u(s_i) = \alpha \sum_{j=1}^{m} \widetilde{U}_{ji} u(s_j) + \beta \sum_{j=1}^{n} \hat{W}_{ji} v(t_j) \tag{9}$$

$$v(t_j) = \alpha \sum_{i=1}^{n} \widetilde{V}_{ij} v(t_i) + \beta \sum_{i=1}^{m} \widetilde{W}_{ij} u(s_i) \tag{10}$$

And the matrix form is:

$$\boldsymbol{u} = \alpha \widetilde{\boldsymbol{U}}^T \boldsymbol{u} + \beta \hat{\boldsymbol{W}}^T \boldsymbol{v} \tag{11}$$

$$\boldsymbol{v} = \alpha \widetilde{\boldsymbol{V}}^T \boldsymbol{v} + \beta \widetilde{\boldsymbol{W}}^T \boldsymbol{u} \tag{12}$$

where α and β specify the relative contributions to the final saliency scores from the homogeneous nodes and the heterogeneous nodes and we have $\alpha+\beta=1$. In order to guarantee the convergence of the iterative form, $\boldsymbol{u}$ and $\boldsymbol{v}$ are normalized after each iteration.

For numerical computation of the saliency scores, the initial scores of all sentences and words are set to 1 and the following two steps are alternated until convergence,

1. Compute and normalize the scores of sentences:
$$\boldsymbol{u}^{(n)} = \alpha \widetilde{\boldsymbol{U}}^T \boldsymbol{u}^{(n-1)} + \beta \hat{\boldsymbol{W}}^T \boldsymbol{v}^{(n-1)},$$
$$\boldsymbol{u}^{(n)} = \boldsymbol{u}^{(n)} / \left\| \boldsymbol{u}^{(n)} \right\|_1$$
2. Compute and normalize the scores of words:
$$\boldsymbol{v}^{(n)} = \alpha \widetilde{\boldsymbol{V}}^T \boldsymbol{v}^{(n-1)} + \beta \widetilde{\boldsymbol{W}}^T \boldsymbol{u}^{(n-1)},$$
$$\boldsymbol{v}^{(n)} = \boldsymbol{v}^{(n)} / \left\| \boldsymbol{v}^{(n)} \right\|_1$$

where $\boldsymbol{u}^{(n)}$ and $\boldsymbol{v}^{(n)}$ denote the vectors computed at the n-th iteration.

Usually the convergence of the iteration algorithm is achieved when the difference between the scores computed at two successive iterations for any sentences and words falls below a given threshold (0.0001 in this study).

4 Empirical Evaluation

4.1 Summarization Evaluation

4.1.1 Evaluation Setup

We used task 1 of DUC2002 (DUC, 2002) for evaluation. The task aimed to evaluate generic summaries with a length of approximately 100 words or less. DUC2002 provided 567 English news articles collected from TREC-9 for single-document summarization task. The sentences in each article have been separated and the sentence information was stored into files.

In the experiments, the background corpus for using the mutual information measure to compute word semantics simply consisted of all the documents from DUC2001 to DUC2005, which could be easily expanded by adding more documents. The stopwords were removed and the remaining words were converted to the basic forms based on WordNet. Then the semantic similarity values between the words were computed.

We used the ROUGE (Lin and Hovy, 2003) toolkit (i.e.ROUGEeval-1.4.2 in this study) for evaluation, which has been widely adopted by DUC for automatic summarization evaluation. It measured 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 toolkit reported separate scores for 1, 2, 3 and 4-gram, and also for longest common subsequence co-occurrences. Among these different scores, unigram-based ROUGE score (ROUGE-1) has been shown to agree with human judgment most (Lin and Hovy, 2003). We showed three of the ROUGE metrics in the experimental results: ROUGE-1 (unigram-based), ROUGE-2 (bigram-based), and ROUGE-W (based on weighted longest common subsequence, weight=1.2).

In order to truncate summaries longer than the length limit, we used the "-l" option [2] in the ROUGE toolkit.

4.1.2 Evaluation Results

For simplicity, the parameters in the proposed approach are simply set to $\alpha=\beta=0.5$, which means that the contributions from sentences and words are equally important. We adopt the WordNet-based *vector* measure (WN) and the corpus-based mutual information measure (MI) for computing the semantic similarity between words. When using the mutual information measure, we heuristically set the window size k to 2, 5 and 10, respectively.

The proposed approaches with different word similarity measures (WN and MI) are compared

[2] The "-l" option is very important for fair comparison. Some previous works not adopting this option are likely to overestimate the ROUGE scores.

with two solid baselines: SentenceRank and MutualRank. SentenceRank is proposed in Mihalcea and Tarau (2004) to make use of only the sentence-to-sentence relationships to rank sentences, which outperforms most popular summarization methods. MutualRank is proposed in Zha (2002) to make use of only the sentence-to-word relationships to rank sentences and words. For all the summarization methods, after the sentences are ranked by their saliency scores, we can apply a variant form of the MMR algorithm to remove redundancy and choose both the salient and novel sentences to the summary. Table 1 gives the comparison results of the methods before removing redundancy and Table 2 gives the comparison results of the methods after removing redundancy.

System	ROUGE-1	ROUGE-2	ROUGE-W
Our Approach (WN)	**0.47100***#	**0.20424***#	**0.16336**#
Our Approach (MI:k=2)	0.46711#	0.20195#	0.16257#
Our Approach (MI:k=5)	0.46803#	0.20259#	0.16310#
Our Approach (MI:k=10)	0.46823#	0.20301#	0.16294#
SentenceRank	0.45591	0.19201	0.15789
MutualRank	0.43743	0.17986	0.15333

Table 1. Summarization Performance before Removing Redundancy (w/o MMR)

System	ROUGE-1	ROUGE-2	ROUGE-W
Our Approach (WN)	**0.47329***#	0.20249#	0.16352#
Our Approach (MI:k=2)	0.47281#	**0.20281**#	**0.16373**#
Our Approach (MI:k=5)	0.47282#	0.20249#	0.16343#
Our Approach (MI:k=10)	0.47223#	0.20225#	0.16308#
SentenceRank	0.46261	0.19457	0.16018
MutualRank	0.43805	0.17253	0.15221

Table 2. Summarization Performance after Removing Redundancy (w/ MMR)

(* indicates that the improvement over SentenceRank is significant and # indicates that the improvement over MutualRank is significant, both by comparing the 95% confidence intervals provided by the ROUGE package.)

Seen from Tables 1 and 2, the proposed approaches always outperform the two baselines over all three metrics with different word semantic measures. Moreover, no matter whether the MMR algorithm is applied or not, almost all performance improvements over MutualRank are significant and the ROUGE-1 performance improvements over SentenceRank are significant when using WordNet-based measure (WN). Word semantics can be naturally incorporated into the computation process, which addresses the problem that SentenceRank cannot take into account word semantics, and thus improves the summarization performance. We also observe that the corpus-based measure (MI) works almost as well as the knowledge-based measure (WN) for computing word semantic similarity.

In order to better understand the relative contributions from the sentence nodes and the word nodes, the parameter α is varied from 0 to 1. The larger α is, the more contribution is given from the sentences through the SS-Graph, while the less contribution is given from the words through the SW-Graph. Figures 2-4 show the curves over three ROUGE scores with respect to α. Without loss of generality, we use the case of k=5 for the MI measure as an illustration. The curves are similar to Figures 2-4 when k=2 and k=10.

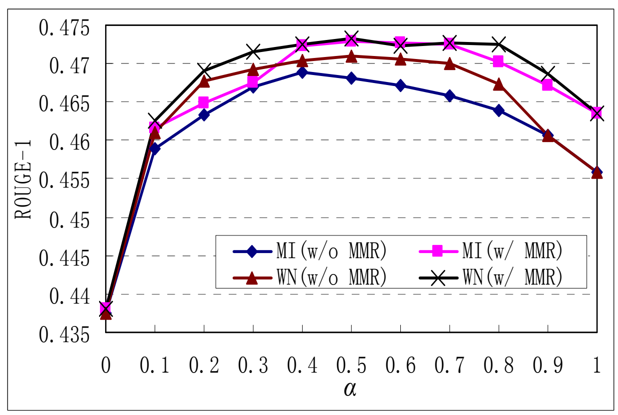

Figure 2. ROUGE-1 vs. α

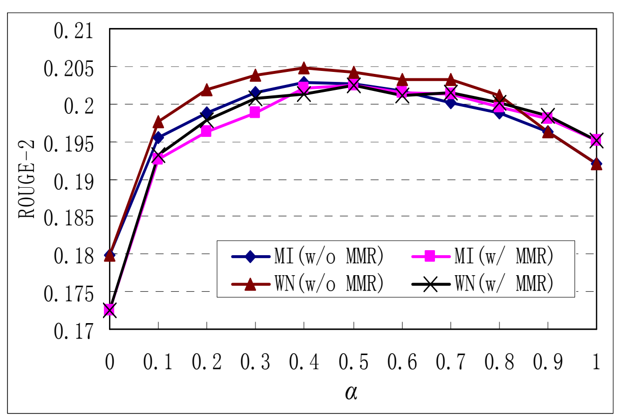

Figure 3. ROUGE-2 vs. α

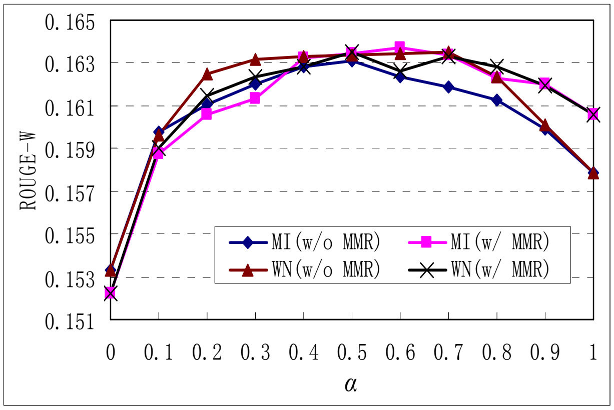

Figure 4. ROUGE-W vs. α

Seen from Figures 2-4, no matter whether the MMR algorithm is applied or not (i.e. w/o MMR or w/ MMR), the ROUGE scores based on either word semantic measure (MI or WN) achieves the peak when α is set between 0.4 and 0.6. The performance values decrease sharply when α is very large (near to 1) or very small (near to 0). The curves demonstrate that both the contribution from the sentences and the contribution from the words are important for ranking sentences; moreover, the contributions are almost equally important. Loss of either contribution will much deteriorate the final performance.

Similar results and observations have been obtained on task 1 of DUC2001 in our study and the details are omitted due to page limit.

4.2 Keyword Evaluation

4.1.1 Evaluation Setup

In this study we performed a preliminary evaluation of keyword extraction. The evaluation was conducted on the single word level instead of the multi-word phrase (n-gram) level, in other words, we compared the automatically extracted unigrams (words) and the manually labeled unigrams (words). The reasons were that: 1) there existed partial matching between phrases and it was not trivial to define an accurate measure to evaluate phrase quality; 2) each phrase was in fact composed of a few words, so the keyphrases could be obtained by combining the consecutive keywords.

We used 34 documents in the first five document clusters in DUC2002 dataset (i.e. d061-d065). At most 10 salient words were manually labeled for each document to represent the document and the average number of manually assigned keywords was 6.8. Each approach returned 10 words with highest saliency scores as the keywords. The extracted 10 words were compared with the manually labeled keywords. The words were converted to their corresponding basic forms based on WordNet before comparison. The precision p, recall r, F-measure (F=2pr/(p+r)) were obtained for each document and then the values were averaged over all documents for evaluation purpose.

4.1.2 Evaluation Results

Table 3 gives the comparison results. The proposed approaches are compared with two baselines: WordRank and MutualRank. WordRank is proposed in Mihalcea and Tarau (2004) to make use of only the co-occurrence relationships between words to rank words, which outperforms traditional keyword extraction methods. The window size k for WordRank is also set to 2, 5 and 10, respectively.

System	Precision	Recall	F-measure
Our Approach (WN)	0.413	**0.504**	0.454
Our Approach (MI:k=2)	**0.428**	0.485	0.455
Our Approach (MI:k=5)	0.425	0.491	**0.456**
Our Approach (MI:k=10)	0.393	0.455	0.422
WordRank (k=2)	0.373	0.412	0.392
WordRank (k=5)	0.368	0.422	0.393
WordRank (k=10)	0.379	0.407	0.393
MutualRank	0.355	0.397	0.375

Table 3. The Performance of Keyword Extraction

Seen from the table, the proposed approaches significantly outperform the baseline approaches. Both the corpus-based measure (MI) and the knowledge-based measure (WN) perform well on the task of keyword extraction.

A running example is given below to demonstrate the results:

Document ID: *D062/AP891018-0301*

Labeled keywords:

insurance earthquake insurer damage california Francisco pay

Extracted keywords:

WN: *insurance earthquake insurer quake california spokesman cost million wednesday damage*

MI(k=5): *insurance insurer earthquake percent benefit california property damage estimate rate*

5 Conclusion and Future Work

In this paper we propose a novel approach to simultaneously document summarization and keyword extraction for single documents by fusing the sentence-to-sentence, word-to-word, sentence-to-word relationships in a unified framework. The semantics between words computed by either corpus-based approach or knowledge-based approach can be incorporated into the framework in a natural way. Evaluation results demonstrate the performance improvement of the proposed approach over the baselines for both tasks.

In this study, only the mutual information measure and the *vector* measure are employed to compute word semantics, and in future work many other measures mentioned earlier will be investigated in the framework in order to show the robustness of the framework. The evaluation of keyword extraction is preliminary in this study, and we will conduct more thorough experiments to make the results more convincing. Furthermore, the proposed approach will be applied to multi-document summarization and keyword extraction, which are considered more difficult than single document summarization and keyword extraction.

Acknowledgements

This work was supported by the National Science Foundation of China (60642001).

References

M. R. Amini and P. Gallinari. 2002. The use of unlabeled data to improve supervised learning for text summarization. In *Proceedings of SIGIR2002*, 105-112.

S. Brin and L. Page. 1998. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1–7).

J. Carbonell and J. Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of SIGIR-1998*, 335-336.

J. M. Conroy and D. P. O'Leary. 2001. Text summarization via Hidden Markov Models. In *Proceedings of SIGIR2001*, 406-407.

DUC. 2002. The Document Understanding Workshop 2002. http://www-nlpir.nist.gov/projects/duc/guidelines/2002.html

T. Dunning. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics* 19, 61–74.

G. ErKan and D. R. Radev. 2004. LexPageRank: Prestige in multi-document text summarization. In *Proceedings of EMNLP2004*.

C. Fellbaum. 1998. WordNet: An Electronic Lexical Database. The MIT Press.

E. Frank, G. W. Paynter, I. H. Witten, C. Gutwin, and C. G. Nevill-Manning. 1999. Domain-specific keyphrase extraction. *Proceedings of IJCAI-99*, pp. 668-673.

Y. H. Gong and X. Liu. 2001. Generic text summarization using Relevance Measure and Latent Semantic Analysis. In *Proceedings of SIGIR2001*, 19-25.

E. Hovy and C. Y. Lin. 1997. Automated text summarization in SUMMARIST. In *Proceeding of ACL'1997/EACL'1997 Worshop on Intelligent Scalable Text Summarization*.

A. Hulth. 2003. Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of EMNLP2003*, Japan, August.

J. M. Kleinberg. 1999. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632.

B. Krulwich and C. Burkey. 1996. Learning user information interests through the extraction of semantically significant phrases. In *AAAI 1996 Spring Symposium on Machine Learning in Information Access*.

J. Kupiec, J. Pedersen, and F. Chen. 1995. A.trainable document summarizer. In Proceedings of SIGIR1995, 68-73.

T. K. Landauer, P. Foltz, and D. Laham. 1998. Introduction to latent semantic analysis. *Discourse Processes* 25.

C. Y. Lin and E. Hovy. 2000. The automated acquisition of topic signatures for text Summarization. In *Proceedings of ACL-2000*, 495-501.

C.Y. Lin and E.H. Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of HLT-NAACL2003*, Edmonton, Canada, May.

R. Mihalcea, C. Corley, and C. Strapparava. 2006. Corpus-based and knowledge-based measures of text semantic similarity. In *Proceedings of AAAI-06*.

R. Mihalcea and P. Tarau. 2004. TextRank: Bringing order into texts. In *Proceedings of EMNLP2004*.

R. Mihalcea and P.Tarau. 2005. A language independent algorithm for single and multiple document summarization. In *Proceedings of IJCNLP2005*.

A. Muñoz. 1996. Compound key word generation from document databases using a hierarchical clustering ART model. *Intelligent Data Analysis,* 1(1).

T. Nomoto and Y. Matsumoto. 2001. A new approach to unsupervised text summarization. In *Proceedings of SIGIR2001*, 26-34.

S. Patwardhan. 2003. Incorporating dictionary and corpus information into a context vector measure of semantic relatedness. *Master's thesis*, Univ. of Minnesota, Duluth.

T. Pedersen, S. Patwardhan, and J. Michelizzi. 2004. WordNet::Similarity – Measuring the relatedness of concepts. In *Proceedings of AAAI-04*.

D. Shen, J.-T. Sun, H. Li, Q. Yang, and Z. Chen. 2007. Document Summarization using Conditional Random Fields. In *Proceedings of IJCAI 07*.

A. M. Steier and R. K. Belew. 1993. Exporting phrases: A statistical analysis of topical language. In *Proceedings of* Second *Symposium on Document Analysis and Information Retrieval*, pp. 179-190.

P. D. Turney. 2000. Learning algorithms for keyphrase extraction. *Information Retrieval*, 2:303-336.

P. Turney. 2001. Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In *Proceedings of ECML-2001*.

I. H. Witten, G. W. Paynter, E. Frank, C. Gutwin, and C. G. Nevill-Manning. 1999. KEA: Practical automatic keyphrase extraction. *Proceedings of Digital Libraries 99 (DL'99),* pp. 254-256.

H. Y. Zha. 2002. Generic summarization and keyphrase extraction using mutual reinforcement principle and sentence clustering. In *Proceedings of SIGIR2002*, pp. 113-120.