# A Novel Automatic Summarization Method from Chinese Document

**Xinglin LIU[1,2,a,*], Qilun ZHENG[1,b], Qianli MA[1,c], Guli LIN[1,d]**

**[1]School of Computer Science and Engineering, South China Univ. of Tech., Guangzhou, China**

**[1]School of Computer Science and Engineering, South China Univ. of Tech., Guangzhou, China**

**[1]School of Computer Science and Engineering, South China Univ. of Tech., Guangzhou, China**

**[2]School of Computer Science, Wuyi University, Jiangmen, China**

**Abstract**

With the rapid development of the Web, automatic summarization has become more and more important for handling the huge amount of text information in the Web. This paper proposes an automatic summarization method based on compound-word recognition and keyword extraction, termed CASKE. CASKE firstly recognizes the compound-words in a document, labels P.O.S. and revises word segmentation. Then, it extracts keywords, and calculates sentence weights by keyword weights. Finally it selects the proportion of the sentences with large weights to construct summary. The generated summary has good continuity and is readable. Experiment results show that the generated summaries are similar with manual reference summaries, achieving 68.31% Precision and 66.72% Recall in average.

***Keywords:*** *Automatic Summarization; Compound-word; Keyword Extraction; Sentence Weight; Natural Languange Processing*

## 1. Introduction

The Web has become the largest resource base. Automatic summarization can help people search for their desired information from the large scale text. Automatic summarization, which was first introduced by Luhn [1], is to extract key sentences to represent the original document.

The existing automatic summarization methods can be divided into two categories: one is to extract summarization based on statistics; the other is to abstract the summarization. Extract summarization means directly extracts sentences from original documents to construct summary. This kind of approach doesn't need parsing and semantic analysis, and the generated summaries are easy to be understood. Abstract summarization generates summaries based on the understanding of documents by using parsing and semantic analysis. It can generate high quality summaries, but is impractical.

This paper proposes a novel automatic summarization method based on compound-word recognition and keyword extraction. The method firstly recognizes the compound-words in a document, then extracts the keywords with weights computed, and computes the weights of sentences, finally selects a number of sentences with proportion as summary.

We conducted experiments on HIT IR-lab Text Summarization Corpus, and the results show that our method achieves good performance on automatic summarization. The remainder of this paper is organized as follows. We first give a brief survey on previous work in Section 2. Then, we detail our proposed automatic summarization method in Section 3. Following that, we present the experiments in Section 4. Finally, conclusions are given in Section 5.

## 2. Related Work

Automatic summarization combines Natural Language Understanding technique and Generation technique [2]. With the rapid development of the Web, many summarization methods were proposed, and some of them have been turned into practical application.

Tao et al. [3] proposed an automatic summarization based on textual unit association networks, which builds association network according to the co-occurrence of

textual units, computes the information quantity of textual unit, and extracts as summary the sentences with large weights, computed by using the weights of textual units.

Wang et al. [4] proposed a Chinese automatic summarization method based on thematic sentence discovery. They utilized terminology rather than traditional word as the minimal semantic unit, computed terminology weight with its length and frequency to extract keywords, and discovered thematic sentences using an improved k-means clustering method.

Wang et al. [5] utilized an MRP based iteration algorithm to simulate the recursive weighted relationship between sentences and words. It firstly calculated the weight of words in sentences. The weight of sentence and the weight of word depend on each other. They utilized iteration algorithm to calculate the weight of sentence, and then extracted the summary sentences.

Chen et al. [6] proposed a Chinese automatic summarization method based on LSI and sentence clustering. They firstly calculated the similarity between sentences using LSI, and then clustered the sentences using Hierarchical clustering algorithm and K-means clustering algorithm, to improve the precision of sentence similarity and topic division. Experiment results showed that the generated summary well covered document topic with less redundancy.

Ai et al. [7] proposed an automatic summarization method based on LSI. They represented documents using VSM, and then calculated the similarity among sentences by semantic index, to get sentence weights. Finally, they selected the summary sentence by weights.

Sun Park et al. [8] proposed a novel summarization method that uses nonnegative matrix factorization (NMF) and the clustering method is introduced to extract meaningful sentences relevant to a given query. The proposed method decomposes a sentence into the linear combination of sparse nonnegative semantic features so that it can represent a sentence as the sum of a few semantic features that are comprehensible intuitively. It can improve the quality of document summaries because it can avoid extracting those sentences whose similarities with the query are high but meaningless by using the similarity between the query and the semantic features. In addition, the proposed approach uses the clustering method to remove noise and avoid the biased inherent semantics of the documents are reflected in summaries.

Wei et al. [9] proposed a multi-document automatic summarization method based on document sensitive graph model, which considers the relevance between sentences in the whole corpus when computing sentence weight.

## 3. The Chinese Automatic Summarization method

Automatic summarization is based on word segmentation, and contains several key parts: compound word recognition, word segmentation modification, keyword extraction, sentence weight calculation and summary sentence extraction.

### 3.1 Compound-word Recognition：

Compound-word is constructed by several atom words, expressing a completed concept. In our opinion, a word string can be viewed as a compound word, if it satisfies the conditions below:

The word string is constructed by $L(L \geq 2)$ uninterrupted atom words in a sentence.

The word string occurs several times in documents.

The occurrences of the new word strings constructed by adding other atoms before or after the word string significantly decrease.

Keyword extraction is greatly affected by the recognition of compound-word. Our method recognized compound-word based on P.O.S detection and directed graph of word co-occurrence.

1) Atom word string extraction

In order to reduce the amount of atom words, we use P.O.S detection to extract atom word string, i.e. filtering the atom words that can not construct compound-word. Then, output the atom words with their positions and lengths. The position is a tri-tuple, representing the id of sentence where the word string occurs, starting and ending positions in the sentence. Length means the number of atom words constructing the word string.

2) Generating word co-occurrence directed graph

Word co-occurrence directed graph is labeled as: $G:<V,E>$, where represents the atom word set, is the set of word pairs. The head of an edge is the first word of the corresponding word pair, and the tail is the ending word. The weight of a directed edge is a set of positions where word pairs co-occurs. Each element of the set is a tri-tuple

$< sno, start, end >$, representing the sentence id, the start position and ending position in the sentence.

We define a set operation $\cap^s$ for the edge set of word co-occurrence directed graph:

$$X \cap^s Y = \{< sno, start, end > | < sno, start, mid > \in X, < sno, mid, end > \in Y\} \quad (1)$$

Clearly, $X \cap^s Y \neq Y \cap^s X$. Hence, when using this operation, it should be guaranteed that the tail vertex of the edge of the left operand is the head vertex of the edge of the right operand.

3) Compound-word recognition

Based on the idea of Bellman-Ford algorithm, we propose an algorithm to find the path with longest length and largest weight in word co-occurrence directed graph.

The algorithm presents as follows:

- Set $s = aMatrix(1,0)$, $d$ =NULL, $path = s$, $len(path) = 1$, $ps(path) = \Phi$, $weight(path) = 0$;
- Let $s$ be the original vertex, and search the next vertex $d$ of $s$ in word occurrence directed graph. If failed, turn to Step 3;

a) $ps(path) = ps(path) \cap^s ps < s, d >$;
b) Update $weight(path)$;
c) If $weight(path) \geq T$, set $path = path \,\&\, d$, $len(path) += 1$, $s = d$, $d$ =NULL;
d) Turn to Step 2.

- If $len(path) < L$, turn to Step 7;
- Save the extracted compound-words;
- Delete the path information of the extracted compound-words in word occurrence directed graph;
- Reduce dimensions of the word occurrence directed graph;
- If the graph is not null, turn to Step 1;
- Output the extracted compound-words.

After many times of iteration search, it can finally find the longest path that satisfies the three conditions for compound-word recognition, and then gets the compound-words in the document.

4) P.O.S. labeling and word segmentation revision

We utilized head-feature percolation [11] to label P.O.S. for compound-words. It means that the key parts of compound-word affects the parsing attribute of compound-word and revises the word segmentation results. The labeling format of the compound-word P.O.S. is: P.O.S. + cw + Num, where cw means that the word is compound-word, Num represents the length of the compound-word, such as "Humanities and Social Sciences /ncw3", "Human culture /ncw2".

## 3.2 Keyword Extraction

In our opinion, the importance of a word for a document is related to its position and length. We set different weights according to different positions and lengths of a word.

We define three kinds of word positions in a document:

Definition 1: Paragraph Order (PO) represents paragraph of a document which a word occurs in. PO={First Paragraph (FP), Ending Paragraph (EP), Others (O)}.

Definition 2: Sentence Order (SO) represents sentence of a paragraph which a word occurs in. SO= {First Sentence (FS), Ending Sentence (ES), Others (O)}.

Definition 3: Word Order (WO) represents position of a sentence which a word occurs in. WO={First Word (FW), Ending Word (EW), Others (O)}.

Then, a word may occur in |PO|*|SO|*|WO|=27 different positions in a document. And we give different position values to the 27 positions, which are shown in Table 1.

Table 1. Position Values of PO, SO and WO

| Position | FP | FS | FW | O | EP | ES | EW |
|---|---|---|---|---|---|---|---|
| Value | 64 | 32 | 16 | 8 | 4 | 2 | 1 |

The position value $pv_t$ of word $t$ is calculated by:

$$pv_t = po + so + wo \quad (2)$$

The weight $w_{ti}$ of each occurrence of word $t$ is defined as:

$$w_{ti} = \frac{pv_{ti}}{\sum_{i=1}^{27} pv_i} \cdot \sqrt{n} \quad (3)$$

where $\sum_{i=1}^{27} pv_i$ represent the sum of 27 position values, $n$ represent the length of compound-word, $n = 1$ for non compound-word.

The overall weight $w_t$ of word $t$ is defined as:

$$w_t = \sum_{i=i}^{|t|} t_{wi} \quad (4)$$

where $|t|$ represents the times of occurrence of word $t$.

Based on the above calculations, our method ranks the candidate keywords by their overall weights in descent order. To handle synonym in the candidate keywords, we merge the synonym using HIT IR-Lab Tongyici Cilin (Extended). For automatic extraction's convinence, we output keywords with overall weights and sentence id of each thematic word.

### 3.3 Calculate Sentence Weight

Sentence weights are calculated according to rules below:

Rule 1: If sentence $s_j$ contains $m$ distinct keywords, its sentence weight equals to the weight sum of the $m$ keywords multiplied by $\sqrt{m}$.

Rule 1 is based on a fact that a sentence with many low weight keywords are more possible to be a summary sentence than a sentence with only one high weight keyword.

Let $w_{t_i} (1 \le i \le N)$ represent the weight of the i-th keyword, and $w_{s_j} (1 \le j \le |S|)$ represent the weight of the j-th sentence, where $|S|$ represents the count of distinct sentences that contain the keyword. The weight of sentence is calculated by :

$$w_{s_j} = \sum w_{t_i} \times \sqrt{|w_{t_i}|} \quad (5)$$

where $\sum w_{t_i}$ represents the sum of the weights of all different keywords in sentence $s_j$, and $|w_{t_i}|$ represents the number of different keywords in sentence $s_j$.

### 3.4 Selection of Summarization Sentence

There are two ways to determine the summary length: one is fixing length, as [5] and [9] do; the other is setting with proportion, used in [3], [4], [6] and [7]. We utilized the latter one in our experiments.

Let $L_{sys}$ represents the length of generated summary, and $L_{ref}$ represents the length of reference summary. The summary sentences are selected as follows:

1) Order the sentences containing thematic terms by their weights decreasingly;
2) If $L_{sys} < L_{ref} \times 90\%$, repeat Step 3. Otherwise jump to Step 4;
3) Add the j-th ( $j (1 \le j \le |S|)$ ) into summary, $j += 1$;
4) If $L_{sys} > L_{ref} \times 110\%$, remove the over part;
5) Output the summary sentences with original order.

## 4. Experiments

We conducted experiments on HIT IR-lab Text Summarization Corpus (Corpus HIT hereafter). The corpus has 211 documents in all, containing 57 documents about Olympic Games, 40 Narration, 46 Argumentation, 18 practical writing and 10 863-Evaluation-Corpus-Documents.

Evaluation measures for automatic summarization are generally divided into two categories: inside evaluation and outside evaluation [10]. Inside evaluation directly evaluates the summary quality by two ways: manual evaluation and automatic evaluation. In manual evaluation, experts give scores on the summaries. By contrast, automatic evaluation is done by algorithms, which compare the generated summary with reference summary upon Recall, Precision and F-measure.

In our experiments, we utilized automatic inside evaluation and selected 20% sentence from original document as summary. Recall, Precision and F-measure are defined as:

$$R = \frac{L_{rs}}{L_{ref}} \quad (6)$$

$$P = \frac{L_{rs}}{L_{sys}} \quad (7)$$

$$F = \frac{2 \times R \times P}{R + P} \quad (8)$$

where $L_{rs}$ represents the length of correct sentence in generated summary, $L_{sys}$ represents the length of generated summary, and $L_{ref}$ represents the length of reference summary.

### 4.1 Experiment Results Evaluation

We utilized five groups of people to extract summaries as reference summaries manually. We compared the five groups of reference summaries on 10 documents selected from the corpus randomly. And we found that the manual summaries from different groups had 78.91% similarity average and were the same in some documents. That means the reference summaries have high degree of confidence for evaluation.

Table 2. Experiment Result

| Corpus | Recall | Precision | F-measure |
|---|---|---|---|
| Group-1 | 0.6954 | 0.7046 | 0.6999 |
| Group-2 | 0.6273 | 0.7175 | 0.6694 |
| Group-3 | 0.6490 | 0.6873 | 0.6676 |
| Group-4 | 0.6912 | 0.6694 | 0.6801 |
| Group-5 | 0.6730 | 0.6365 | 0.6542 |
| Average | 0.6672 | 0.6831 | 0.6743 |

Table 2 shows the experiment results of our proposed method compared with the five-group reference summaries. We can see that, the generated summaries of our method achieve average 66.72% Recall, 68.31% Precision and 67.43% F-measure. Moreover, the performances with the five groups are not significantly different. We further analyzed the generated summaries, and found that they had good quality. What's more, since the extracted sentences were original sentences, they had no syntax problems and were readable.

### 4.2 Comparing With Other Methods

To compare with other summarization methods, we conducted experiments using 2003 National 863 automatic summarization system evaluation corpus in Corpus HIT. The corpus has overall 10 documents, and in average each document has 3017 words and 56.2 sentences. The performance of our method and other methods [5] are shown in Fig. 1.

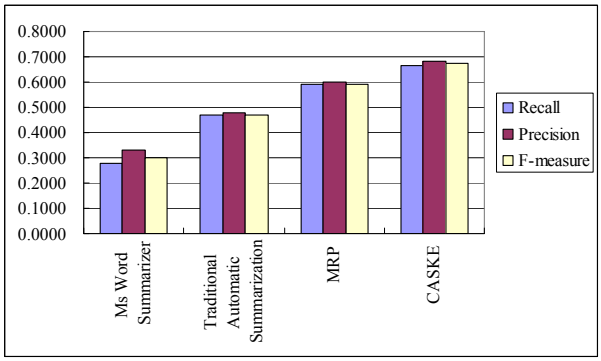


Fig 1. The Result of Compare with other methods

We can see that our method achieves the best performance among these methods upon Recall, Precision and F-measure.

## 5. Conclusion and Future Directions

This paper proposes an automatic summarization method based on compound-word recognition and keyword extraction. Experiment results show that the method can generates good summary. And the performance of our method positively depends on the precision of keyword extraction.

Future works include: 1) improving word segmentation, compound-word recognition and keyword extraction; 2) evaluating the quality of automatic summary with content similarity measure; 3) improving the efficiency of the algorithm.

## Acknowledgment

This work is supported by the Natural Science Foundation of Guangdong Province, China (Grant No. 9451064101003233); Guangdong science and technology plan projects(Grant No. 2010B010600039); the Fundamental Research Funds for the Central Universities, SCUT(Grant No. 2009ZM0125, 2009ZM0189, 2009ZM0255).

## References

[1] Luhn H P. The automatic creation of literature abstract. IBM Journal of Research and Development, 2(2), (1958)159-165.

[2] Dragomir R Radev, Simone Teufel, Horacio Saggion, et al. Evaluation Challenges in large-scale document Summarization. ACL2003, Sapporo, Japan:[s.n.], (2003)375-382.

[3] Yu-hui TAO, Shui-geng ZHOU, Ji-hong GUAN. Automatic Text Summarization Approach Based on Textual Unit Association Networks. Pattern Recognition & Artificial Intelligence, 22(3), (2009) 440-444.
[4] Meng WANG, Chun-gui LI, Pei-he TANG, Xiao-rong WANG. Chinese Automatic Summarization Based on Thematic Sentence Discovery. Computer Engineering, 33(8), (2007)180-181,189.
[5] Zhi-qi WANG, Yong-cheng WANG, Chuan-han LIU. A Sentence Weighting Method for Automatic Text Summarization Based on MRP. Journal of Shanghai Jiao Tong University, 41(8), (2007)1297-1300.
[6] Ge CHEN, Jian-yong DUAN , Ru-zhan RU. Chinese Automatic Text Summarization Based on Latent Semantic Indexing and Sentence CIustering. Computer Simulation, 25(7), (2008)82-85.
[7] Dong-mei AI, Yu-chao ZHENG and De-zheng ZHANG. Automatic text summarization based on latent semantic indexing. Artificial Life and Robotics, 15(1), (2010)25-29.
[8] Sun Park, ByungRea Cha, Dong Un An. Automatic Multi-document Summarization Based on Clustering and Nonnegative Matrix Factorization. IETE Technical Review, 27(2), (2010)167-178.
[9] Fu-ru WEI, Wen-jie LI, Qin LU and Yan-xiang HE. A document-sensitive graph model for multi-document summarization. Knowledge and Information Systems, 22(2), (2010)245-259.
[10] Yin LIU, Bi-cheng LI. The Overview and Prospect of Automatic Summarization Evaluation. Journal of The China Society for Scientific and Technical Information, 27(2), (2008)235-243.
[11] Lieber, R. On the Organization of the Lexicon, Doctoral dissertation, MIT, Cambridge, Massachusetts. Distributed by Indiana University Linguistics Club, Bloomington, Indiana, 1980

**Xinglin LIU** Lecturer, School of Computer Science, Wuyi University.Master of Computer Technology(2005.12), School of Computer, Chongqing University; Ph.D. of Applied Computer Technology(2012.6), School of Computer Science and Engineering, South China Univ. of Tech.. Current research interests: Data Mining, Intelligence Computing, Text Knowledge Acquisition.