CCSE Computer and Information Science
www.ccsenet.org/journal.html
Vol. 2, No. 1
February 2009

A Summary Sentence Decomposition Algorithm for Summarizing Strategies Identification

Norisma Idris (Corresponding author)
Department of Artificial Intelligence
Faculty of Computer Science and Information Technology
University of Malaya
50603, Kuala Lumpur, Malaysia
Tel: 60-3-7967-6395 E-mail: norisma@um.edu.my

Sapiyan Baba
Department of Artificial Intelligence
Faculty of Computer Science and Information Technology
University of Malaya,
50603, Kuala Lumpur, Malaysia
Tel: 60-3-7967-6301 E-mail: pian@um.edu.my

Rukaini Abdullah
Department of Artificial Intelligence
Faculty of Computer Science and Information Technology
University of Malaya
50603, Kuala Lumpur, Malaysia
Tel: 60-3-7967-6378 E-mail: rukaini@um.edu.my

The research is financed by Ministry of Science, Technology and Innovation.(E-Science Project No. 11-02-03-1017)

Abstract

Expert summarizers employ a number of strategies to produce summaries. Teachers need to identify which strategies are used by students to help them improve their summary writing. However, the task is time consuming. This paper reports on our effort to develop an algorithm to identify the summarizing strategies employed by students using summary sentence decomposition. The summarizing strategies used by experts are identified and translated into a set of heuristic rules. A summary sentence decomposition algorithm is then developed based on the heuristic rules. A preliminary test was carried out and the results are discussed.

Keywords: Summary writing, Summarizing strategies, Summary sentence decomposition, Heuristic rules

1. Introduction

Summary writing is an important skill which involves multiple cognitive activities such as reading and understanding of text, identifying relevant theme, and generating a shorter version of it. As opposed to other types of writing such as writing a story or a report, the production of a summary is dependent on existing texts (Hidi & Anderson, 1986). In addition, the skill required some strategies to determine what to include and eliminate, how to reorganize information extracted and how to ensure that the final version captures the key messages of the original text without simply doing a cut and paste job on the original text. Thus, students needs to employ the summary writing skill not only for navigating learning successfully at college and university level but also enables them to work independently.

The task of summary writing can be useful to evaluate students' comprehension about the text (Zipitria, Arruarte, Elorriaga, & Díaz de Ilarraza, 2004). Hence, automated summarization assessment has drawn a lot of interests in recent years. There are a few systems developed for this purpose, e.g. *Summary Street*® (Franzke, Kintsch, Caccamise, Johnson, & Dooley, 2005; Wade-Stein & Kintsch, 2004) *Laburpen Ebaluaka Automatikoa* or *LEA* (Zipitria et al., 2004) and Summarization Assessment Strategies Model (Lemaire, Mandin, Dessus, & Denhière, 2005).

Summary Street® is a computer-based assessment system that provides an environment where students can get feedback about the content of their written summary. It employs *Latent Semantic Analysis (LSA),* a machine learning method to construct the semantic representations that mirrors the structure of human knowledge. The system compares the student's summary and the original text to determine how similar they are. It gives immediate visual feedback based on measures such as content knowledge, writing mechanics, length, redundancy, and plagiarism.

LEA is an automatic summary evaluation environment which takes evaluation decision based on human expertise modeling, to train students in summarization skills and also to assess human summary evaluation. It gives feedback on the coherence, content coverage and cohesion, the use of language and adequacy of the summary. Like Summary Street, *LEA* also employs LSA as a tool to measure domain knowledge and summarization skills.

Another example is modeling summarization assessment strategies using LSA, which models the way teachers assess students' summaries. The model is based on automatic detection of five macrorules in summarization which are implemented in the LSA framework. In this model, each summary sentence is compared with every sentence in the original text.

Although previous works (e.g. as in (Franzke et al., 2005; Wade-Stein & Kintsch, 2004; Zipitria et al., 2004)) have contributed towards the development of the summarization assessment system, their focus is only on the quality of the summary whereby the summary has to be concise, include only main ideas and avoid redundancy,. Few if any, automated summarization assessment systems have been developed for identifying and assessing students' use of a systematic strategy for accomplishing the summary writing task. Since summary writing required some strategies and currently there is no tool for identifying the strategies employed by students, we proposed a computer-based summarization assessment system that assesses a summary by looking into the strategies used. This tool can be used to give information to teachers about the specific strategies used by students to produce their summaries and consequently, help students to hone their summarizing strategies in summary writing.

2. Problem in Summary Writing

Summary writing has been part of the English language syllabus for many years (Zurina, 2003). It is to equip students with the summarization skills needed during tertiary education of after leaving schools. Although the skills can be developed naturally, most students require much help to produce a reasonable summary. In addition, previous research has shown that proper teaching of summary writing will positively influence student's use of summarization strategies and the quality of their summaries (Kaur, 1997). Students are taught how to summarize using a sequence of instructional rules as shown in Figure 1 (a) which are similar to the basic summarization rules shown in Figure 1 (b).

An earlier study has shown that student's difficulties in summarizing were linked to the students' use of strategic skills (Winograd, 1984). In addition, an analysis of students' summaries shows that the performances of the students do not reflect directly the students' strategies in summarizing (Norisma et al., 2007). In other words, students who received good marks in summary writing do not necessarily possess good summarizing skills. Unfortunately, in assessing a summary writing, only the quality of the summary is evaluated, not the strategies used to produce the summary itself. This is quite a common practice amongst teachers in schools since identifying student's summarizing strategies is a very time consuming process. Thus, teachers do not know for sure about the specific summarizing strategies used by the students to process information. In addition, the students have to learn and apply the strategies of summarization. Only then will they improve their summarizing skills. Because of that, we proposed a sentence decomposition algorithm that can identify the strategies used by students in summary writing. The development of the algorithm involves a set of heuristic rules formulated from a careful analysis of the experts' summaries using summary sentence decomposition.

3. Formulation of Heuristic Rules

To formulate the heuristic rules for identifying students' summarizing strategies, the teachers' strategies in summary writing can be used as rules to identify students' strategies. Thus, the formulation of the heuristic rules is based on the expert summarizing strategies which are acquired mainly by studying the experts' summaries. The study was conducted to identify the experts' summarizing strategies and how they used the strategies to produce their summary sentences. From the study, we identified 8 types of strategies that are commonly used by the experts:

- Deletion - trivial or redundant information are eliminated from the sentence
- Sentence combination – two or more original sentences is combined to construct a summary sentence

- Topic sentence selection - only one important sentence is chosen to represent the main idea of the whole paragraph.
- Paraphrase – similar phrase or word is used to replace the original phrase or word
- Generalization - a list of words or items is replaced by a more general word in the same class.
- Syntactic transformation – the order of the words in a sentence is changed.
- Sentence reordering – the order of the sentences in the original text is changed.
- Invention – new but more concise sentences are used to replace the original one.

To represent the sentences and words in a text, we propose the notation below for our discussion.

If *T* is a text consisting of m sentences, t_i where $i = 1, 2, 3, \ldots m$, then,

$T = \{ t_i \};\ i = 1, 2, 3, \ldots, m$ (1)

Hence, for sentence t_i, comprising a string of n_i words, t_{ij}, *where* $j = 1, 2, 3, \ldots n_i$, then, t_i can be written as,

$$t_i = \{ t_{ij} \} \qquad j = 1, 2, 3, \ldots\ldots, n_i \quad (2)$$

Similarly for summary text, *S*, where every summary sentence, s_i, comprises a string of words as represented as,

$s_i = \{ s_{ij} \},\quad i = 1, 2, 3, \ldots\ldots, m \ and \ \ j = 1, 2, 3, \ldots\ldots, n_i$ (3)

The summarizing strategies of the experts were analyzed and translated into a set of heuristic rules on how to identify the summarizing strategies as shown in Table 1. Not all strategies are presented in the rules since we used *position-based method* in our algorithm which is only applicable to some of the rules which are deletion, sentence combination, syntactic transformation and sentence reordering. Copy-paste, a strategy where a summary sentence is produced by copying the original sentence without making any changes, is included in the rules although it is not part of the summarization rules and the experts' strategies. It is important to identify the usage of this strategy in students' summary writing since it is not a good strategy and it promotes students to plagiarism.

4. Summary sentence decomposition algorithm

4.1 Problem Description

The main focus of our work is to develop a program to identify students' summarizing strategies using summary sentence decomposition which is a process to determine whether a summary sentence is generated from the original text and to identify the position of the words in the original text (Jing & McKeown, 1999). The task of the decomposition program is to analyze the construction of summary sentences in students' summaries. It deals with the following problem:

Given a student's summary sentence,

- *What words in the summary sentence come from the original document?*
- *Where are the words located in the original document?*
- *Which is the best sequence used to represent the words in the summary sentence?*
- *What is the original sentence of the best sequence?*
- *How is the summary sentence constructed from the original sentence?*

The position of a word in the original text is represented by the sentence position and the word position: *(q,r)*. Thus, an input summary sentence, s_i can be represented as a word sequence, $\{s_{ij}\}$. If a word, s_{ij}, is found in T, and the word is t_{qr}, then, $s_{ij} \equiv t_{qr}$.

For every word in a summary sentence, the location of the same word is found in the original text. When the locations of all words have been identified in the original text, a sequence of word positions is obtained. However, some common words in the summary sentence like *a, an* and *the* may occur more than once in the original text and in different sentences in the original text. Hence, a set of sequence of locations of words in original text corresponding to the different sentences are found. For example, the second summary sentence in a summary text, *"I like the movie",* can be represented as $s_2 = \{(2\ 1)\ (2\ 2)\ (2\ 3)\ (2\ 4)\}$ which indicates that $(2,1), (2,2), (2,3), (2,4)$ are the positions of the summary words respectively. These words are found in the original text, T, and word like "*I*" has multiple occurrences in T. Thus,

$$s_{21} \equiv t_{13} \equiv t_{41} \equiv t_{52} \equiv t_{63} \equiv t_{91}$$

This rule is applied to all summary words and the positions of these words in T, are presented in Figure 2. Each position of a word is link to the position of the next word until the last word to produce a sequence of word positions. By referring to Figure 2, the first possible sequence can be {(1, 3), (4, 2), (4, 3), (4, 4)}, another sequence can be {(1, 3), (8, 7), (4, 3), (4, 4)}, and so on. Since the word *I* occurs 5 times in the original text, *like* occurs twice, *the* occurs once and *movie* was found in 4 locations, then, the total number of possible sequences of positions is 40 (5 x 2 x 1 x 4). Therefore, it is important to determine which is the sequence best used among all the possible sequences before we can determine how this sequence was used to construct the summary sentence.

4.2 Summary sentence decomposition algorithm

Summary sentence decomposition algorithm is developed to:

- determine whether the words in the summary sentence are from the original text,
- locate the locations of the words in the original text using position-based method
- find the best sequence of locations of words used to represent a phrase in the summary sentence,
- identify the strategies used to produced the summary sentence.

This algorithm consists of two sub-algorithms which are:

4.2.1 Best sequence selection algorithm

As described in the previous section, some words in the summary sentence may occur more than once in the original text and in different sentences in the original text. Hence, a set of sequence of locations of words in original text corresponding to the different sentences are found. Therefore, it is important to determine which is the best sequence used to represent a phrase of words in the summary sentence. Best sequence selection is a process to locate the locations of words in the original text and to identify the best sequence which represents phrases in the summary sentence. To reduce the number of possible sequences of words locations found in a text, we used a technique called *sequencing* in our algorithm. The technique sequences the words locations according to their sentence positions since a phrase (a group of words) usually comes from the same sentence in a text. Thus, reducing the number of space search in finding the best sequence to represent a phrase.

The algorithm of the process is as follows:

```
For each $s_i$,
1) Let $m$ be the number of words in $s_i$.
2) Let p be the number of phrase in the $s_i$. For p = 1,
   i. For $s_{ij}$ the $j^{th}$ word in sentence $s_i$, locate the positions $(q_{ij_k}, r_{ij_k})$ of the same word in $T$; where $q_{ij_k}$ is the sentence position and $r_{ij_k}$ is the word position in $T$; $k = 1(1)l$; $x^l$ - the number of times the same word is repeated in $T$.
   ii. From these positions, determine the best sequence of locations of words to represent a phrase in $s_i$:
      a. Sequencing the locations of words into sequences, $seq$, according to their sentence positions.
      b. For each $seq$,
         i. Let $n$ be the number of words in the sequence.
         ii. Compare the number of words, $m$ in $s_i$, with the number of words, $n$ in the sequence and calculate the difference between them where,
                $diff = m - n$
                $(if\ diff = 0\ then\ m = n;\ all\ the\ words\ in\ s_i\ are\ found\ in\ the\ sequence)$
         iii. Calculate the average distance, $d$, among words in the sequence where,
                $d = \sum_{j=1}^{n-1} (r_{ij+1} - r_{ij})/(n-1)$
         iv. Categorize the sequence to word category, $w_{cat}$, according to their distance based on the finding (from the experiments on 29 summary sentences produced using deletion strategy) that a phrase is a group of words that acts as a unit in a sentence where the best distance is between 1.0 and 3.0:
            • $If\ d = 1, then\ w_{cat} = 0$
              (The words are in adjacent positions)
            • $If\ 1 < d \leq 3, then\ w_{cat} = 1$
              (The words are closed and retain their orders as in the original document)
            • $If -3 \leq d < 1, then\ w_{cat} = 2$
              (The words are closed by but the order is reversed)
            • $If\ d > 3\ or\ d < -3, then\ w_{cat} = 3$
              (The words are far from each other)
```

3) Select the sequence with the smallest value for $diff$ *and* w_{cat} *to be the best sequence in* T *used to produce the phrase in the summary sentence.*
4) Using the best sequence, search the sentence, t_i*, in the original text that is used to produce* s_i*. Let l be the number of words in* t_i*.*
5) A summary sentence comprises one or multiple phrases which extracted from different sentences. Thus, check the value of diff. If $diff \neq 0$ *then from the positions of the remain words in the summary sentence, repeat Step 2 to determine the best sequence of words positions for the next phrase.*
6) Calculate the distance, d, between the sentence of the phrase found earlier with the sentence of each sequence to find the next phrase where,

$$d = q_{i+1j} - q_{ij}$$

7) Categorize the distance to sentence category, s_{cat}*, according to the finding (from the experiments on 47 summary sentences produced using sentence combination strategy) that a summary sentence is produced by sentences which are near to each other where the best distance is between 1.0 to 3.0:*

- $If\ d = 1, then\ s_{cat} = 0$

(The sentences are in adjacent positions)

- $If\ 1 < d \leq 3, then\ s_{cat} = 1$

(The sentences are near to each other and retain their orders as in the original document)

- $If -3 \leq d < 0,\ then\ s_{cat} = 2$

(The sentences are adjacent or near to each other but the order is reversed)

- $If\ d > 3\ or\ d < -3, then\ s_{cat} = 3$

(The sentences are far from each other)

8) Select the sequence with the smallest value for $diff$ *, w_category and s_category to be the best sequence used to produce the phrase in the summary sentence.*
9) If a phrase is found, increase the number of phrases where,

$$p = p + 1$$

10) Repeat Step 4 until $diff = 0$ *or no more phrase is found in* s_i*.*

4.2.2 Summarizing strategies identification algorithm

After the best sequence of words positions is selected, the next step is to identify the summarizing strategies used to produce the sequence. This process involved the use of the heuristic rules where these rules are transformed into an algorithm as presented below:

Let m be the number of words in s_i *and* l *the number of words in* t_i

Rule 1:

$If\ (m < l)\ and\ (w_{cat} = 0\ or\ 1)\ then$

Deletion strategy

Rule 2:

$If\ (m = l)\ and\ (w_{cat} = 2)\ then$

Syntactic transformation strategy

Rule 3:

$If\ (m < l)\ and\ (w_{cat} = 2)\ then$

Deletion + Syntactic transformation strategies

Rule 4:

$If\ (m = l)\ and\ (w_{cat} = 0)\ then$

Copy-paste strategy

Rule 5:

$If\ (s_{cat} = 0)\ or\ (s_{cat} = 1)\ then$

Sentence combination strategy

Rule 6:

$If\ (s_{cat} = 2)\ then$

Sentence reordering + sentence combination strategies

```
Given a summary sentence, $s_i$ where it comprises p phrases,
If p = 1 then
    Check for Rule 1, Rule 2, Rule 3 and Rule 4
If p > 1 then
    Check for Rule 1, Rule 2, Rule 3, Rules 5 and Rule 6
```

5. Experiments

Consider a summary sentence consists of a phrase *"I started towards the shore"* where all the words are found in the original text which comprises 35 sentences. Each of the word is represented by their sentence position and word position, (q_{ij_k}, r_{ij_k}). However, most of the words occur more than once in the document as shown in Table 2 below. The common words like *'I'* was found in 30 locations and the word '*the'* occurs 29 times. The word *'shore'* was found in 6 locations while the words *'started'* and *'towards'* occur twice in the original text.

According to the possible sequence discussed in Section 4.1, the total number of the sequences for this summary sentence is 20880 (30 x 2 x 2 x 29 x 6). The best sequence is the sequence of the words positions in the original text which best used to represent the words in the summary sentence. To find the best sequence among all the possible sequences, sequencing method is used where the words locations are categorized according to their sentence positions, as shown in Table 3 below. Using the sequencing method, the number of possible sequences for this sentence, *k*, is 31 which has reduced the number of the search space to find the best sequence.

For each possible sequence, calculate the difference, *diff*, between the number of words in s_i, *m,* and the number of words in the sequence, *n,* where *m* is 5. Then, calculate the average distance, *d*, among words in the sequence and assign the *word category* according to their *d*, as shown in Table 4 below. However, some sequences like sequence 4, 6 and 7 have no distance since they only have one word position.

To select the best sequence among 31 sequences, choose the sequence which has the lowest *diff* and w_{cat}. In this example, the best sequence in *T* used to produce the phrase *"I started towards the shore"* is {(4 1) (4 2) (4 3) (4 4) (4 5)}. Using this sequence, the program searched the sentence in the original text used to produce the summary sentence. It indicates that the phrase is taken from the 4th sentence, *"I started towards the shore as I saw my father turned away"* in T where $\{s_{11}, s_{12}, s_{13}, s_{14}, s_{15}\} \equiv \{t_{41}, t_{42}, t_{43}, t_{44}, t_{45}\}$.

Since $\boldsymbol{diff} = \mathbf{0}$, the program stop searching for other phrase, thus, $\boldsymbol{p} = \mathbf{1}$. According to the algorithm, if $\boldsymbol{p} = \mathbf{1}$, then, the program checks for Rule 1, Rule 2, Rule 3 and Rule 4. However, before this can be done, the program needs to determine the number of words, $\boldsymbol{l}$, of the original sentence where in this case, $\boldsymbol{l} = \mathbf{12}.$ Since it complies with Rule 1 where $\boldsymbol{m} < \boldsymbol{l}$ and *w_cat = 0*, then it is found that the summary sentence *"I started towards the shore"* is produced from the original sentence *"I started towards the shore as I saw my father turned away*" using the deletion strategy.

6. Conclusion

This paper presents a development of an algorithm to identify students' summarizing strategies. The rules to identify the strategies are identified from the analysis of the experts' summaries and are translated into a set of heuristic rules. The result from the experiments shows that, the sentence decomposition algorithm can reduce the number of possible sequences using the sequencing method and consequently reduce the number of search space to find the best sequence. It also shows that the algorithm can be used to identify the summarizing strategy used to produce the summary sentence. This algorithm can be embedded in an educational application to help teachers identify the ability of their students in applying the strategies in summary writing.

References

Brown, A. L., & Day, J. D. (1983). Macrorules for summarizing texts: The development of expertise. *Journal of Verbal Learning and Verbal Behavior, 22*, 1 - 14.

Franzke, M., Kintsch, E., Caccamise, D., Johnson, N., & Dooley, S. (2005). Summary Street®: Computer support for comprehension and writing. *Journal Educational Computing Research, 33*(1), 53 - 80.

Hidi, S., & Anderson, V. (1986). Producing written Summaries: Task demands, cognitive operations and Implications for instruction. *Review of Educational Research, 56*, 473 - 493.

Jing, H., & McKeown, K. (1999). *The decomposition of human-written summary sentences.* Paper presented at the 22th Annual International ACM SIGIR Conference on Research & Development in Information Retrieval.

Kaur, G. (1997). *The Role of Summarization Instruction in the Comprehension of Expository Texts.* University of Malaya, Kuala Lumpur.

Lemaire, B., Mandin, S., Dessus, P., & Denhière, G. (2005, July 21-23). *Computational cognitive models of summarization assessment skills.* Paper presented at the 27th Annual Meeting of the Cognitive Science Society (CogSci' 2005), Stresa, Italy.

Norisma, I., Mohd. Sapiyan Baba, & Abdullah, R. (2007). An Analysis on student-written summaries: A Step towards Developing an Automated Summarization Assessment, *International Conference on Electrical Engineering and Informatics (ICEEI2007)* (pp. 550 - 553). Institut Teknologi Bandung, Indonesia.

Wade-Stein, D., & Kintsch, E. (2004). Summary Street: Interactive computer support for writing. *Cognition and Instruction, 22*, 333 - 362.

Winograd, P. (1984). Strategic difficulties in summarizing texts. *Reading Research Quarterly, 19*(4), 404 - 425.

Zipitria, I., Arruarte, A., Elorriaga, J. A., & Díaz de Ilarraza, A. (2004). *Towards a cognitive model of summary evaluation.* Paper presented at the ITS'2004 on Modeling Human Teaching Tactics and Strategies, Maceió, Brazil.

Zurina, M. H. (2003). *Summary writing skills by selected form 4 ESL learners utilizing narrative and expository texts.* University of Malaya, Kuala Lumpur.

Table 1. A set of heuristic rules for identifying summarizing strategies

Strategy	Heuristic Rules
Deletion	s_i *is produced by deletion if:* • s_i *contains a phrase of words where these words,* o *are found in the same sentence,* t_k *in T* o *are located near to each other in T and retain their orders, where the average distance, d is* $1 \geq d \geq 3$ • *the number of words in* s_i *is less than* t_k
Sentence combination	s_i *is produced by sentence combination if:* • s_i *contains phrases of words which are found in different sentences in T where these sentences,* o *are adjacent or are located near to each other and retain their order where the distance between the sentences, d is* $1 \geq d \geq 3$ o *are combined using conjunction words or commas.*
Syntactic transformation	s_i *is produced by syntactic transformation if:* • s_i *contains a phrase where the words in the phrase,* o *are found in the same sentence,* t_k *in T* o *are located near but the order of the word position is reversed where the average distance, d is* $-3 \leq d < 1$
Sentence reordering	s_i *is produced by sentence reordering if:* • s_i *contains phrases of words which are found in different sentences in T where these sentences,* o *are adjacent or are located near to each other but the order of the sentence position is reversed where the distance between the sentences, d is* $-2 \leq d \leq -1$
Copy-paste	s_i *is produced by copy-paste if:* • *the words in* s_i *are found in the same sentence,* t_k *in T* • *the positions of the words in* s_i *are the same as in* t_k • *the number of word in* s_i *are equal to* t_k

Table 2. Words location in T

Words	*I*	*started*	*towards*	*the*	*shore*
$(q_{ij}\ ,\ r_{ij})$	(1 3)	(4 2)	(4 3)	(1 9)	(1 15)
	(4 1)	(22 7)	(6 7)	(1 14)	(3 9)
	(4 7)			(3 8)	(4 5)
	(5 3)			(3 17)	(17 22)
	(6 1)			(4 4)	(22 5)
	(7 1)			(6 10)	(33 10)
	(8 1)			(6 14)	
	(9 5)			(9 19)	
	(10 2)			(10 7)	
	(10 19)			(10 13)	
	(11 1)			(12 12)	
	(12 2)			(13 2)	
	(14 5)			(13 8)	
	(15 1)			(14 10)	
	(16 2)			(17 6)	
	(18 15)			(17 16)	
	(23 2)			(17 21)	
	(23 4)			(18 12)	
	(24 1)			(19 5)	
	(25 6)			(21 10)	
	(25 11)			(25 2)	
	(28 3)			(25 16)	
	(28 7)			(26 6)	
	(30 2)			(27 7)	
	(31 1)			(27 10)	
	(31 9)			(29 2)	
	(33 1)			(30 18)	
	(34 1)			(31 4)	
	(35 7)			(33 4)	
	(35 9)				

Table 3. Sequencing the words locations according to sentences

k	q_{ij}	$(q_{ij}\,, r_{ij})$					n
1	1	(1 3)	(1 9)	(1 15)			3
2	4	(4 1)	(4 2)	(4 3)	(4 4)	(4 5)	5
3	4	(4 7)	(4 2)	(4 3)	(4 4)	(4 5)	5
4	5	(5 3)					1
5	6	(6 1)	(6 7)	(6 10)			3
6	7	(7 1)					1
7	8	(8 1)					1
8	9	(9 5)	(9 19)				2
9	10	(10 2)	(10 7)				2
10	10	(10 19)	(10 7)				2
11	11	(11 1)					1
12	12	(12 2)	(12 12)				2
13	14	(14 5)	(14 10)				2
14	15	(15 1)					1
15	16	(16 2)					1
16	18	(18 15)	(18 12)				2
17	23	(23 2)					1
18	23	(23 4)					1
19	24	(24 1)					1
20	25	(25 6)	(25 2)				2
21	25	(25 11)	(25 2)				2
22	28	(28 3)					1
23	28	(28 7)					1
24	30	(30 2)	(30 18)				2
25	31	(31 1)	(31 4)				2
26	31	(31 9)	(31 4)				2
27	32	(32 1)					1
28	33	(33 1)	(33 4)	(33 10)			3
29	34	(34 1)					1
30	35	(35 7)					1
31	35	(35 9)					1

Table 4. Classifying the sequences according to the *distance* and selecting the sequence according to the *diff* and w_cat

k	q_{ij}	(q_{ij}, r_{ij})					*diff* $(m-n)$	d	w_{cat}
1	1	(1 3)	(1 9)	(1 15)			2	6	3
2	4	(4 1)	(4 2)	(4 3)	(4 4)	(4 5)	0	1	0
3	4	(4 7)	(4 2)	(4 3)	(4 4)	(4 5)	0	-0.5	2
4	5	(5 3)					4	-	-
5	6	(6 1)	(6 7)	(6 10)			2	4.5	3
6	7	(7 1)					4	-	-
7	8	(8 1)					4	-	-
8	9	(9 5)	(9 19)				3	14	3
9	10	(10 2)	(10 7)				3	5	3
10	10	(10 19)	(10 7)				3	-12	3
11	11	(11 1)					4	-	-
12	12	(12 2)	(12 12)				3	10	3
13	14	(14 5)	(14 10)				3	5	3
14	15	(15 1)					4	-	-
15	16	(16 2)					4	-	-
16	18	(18 15)	(18 12)				3	-3	2
17	23	(23 2)					4	-	-
18	23	(23 4)					4	-	-
19	24	(24 1)					4	-	-
20	25	(25 6)	(25 2)				3	-4	3
21	25	(25 11)	(25 2)				3	-9	3
22	28	(28 3)					4	-	-
23	28	(28 7)					4	-	-
24	30	(30 2)	(30 18)				3	16	3
25	31	(31 1)	(31 4)				3	3	1
26	31	(31 9)	(31 4)				3	-5	3
27	32	(32 1)					4	-	-
28	33	(33 1)	(33 4)	(33 10)			2	4.5	3
29	34	(34 1)					4	-	-
30	35	(35 7)					4	-	-
31	35	(35 9)					4	-	-

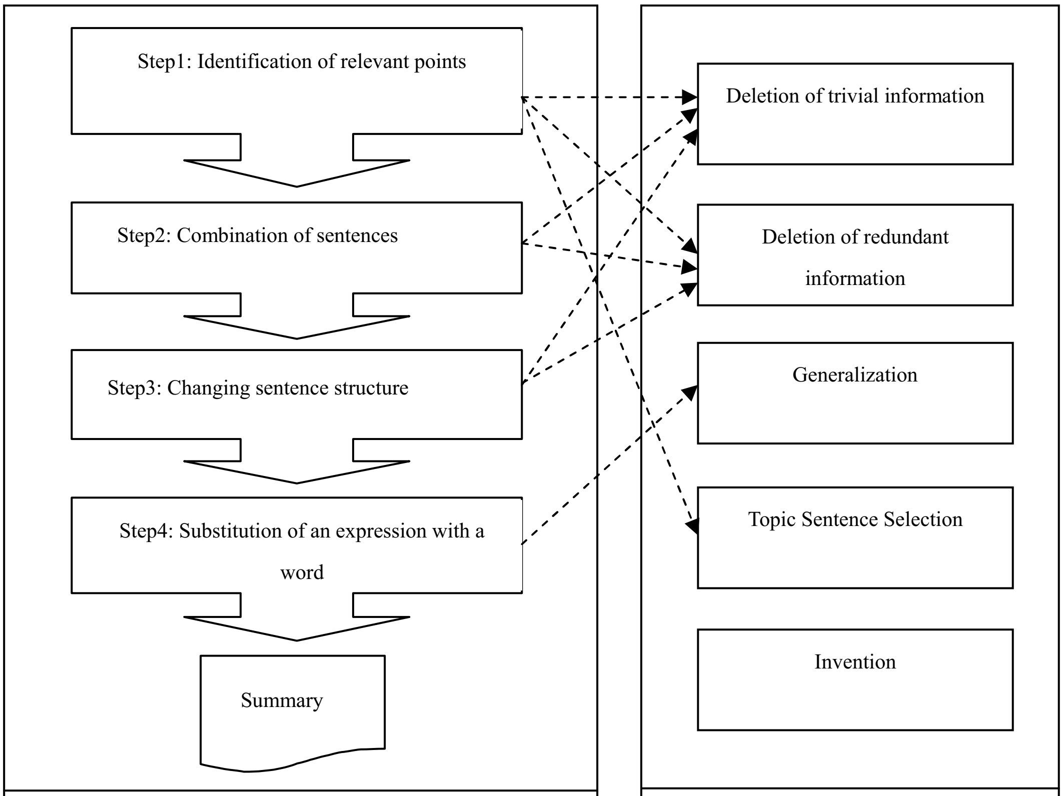
Step1: Identification of relevant points
Step2: Combination of sentences
Step3: Changing sentence structure
Step4: Substitution of an expression with a word
Summary
Deletion of trivial information
Deletion of redundant information
Generalization
Topic Sentence Selection
Invention

Figure 1(a). Instructional rules step-by-step used in summary writing

Figure 1(b): Basic rules for summarization (Brown & Day,

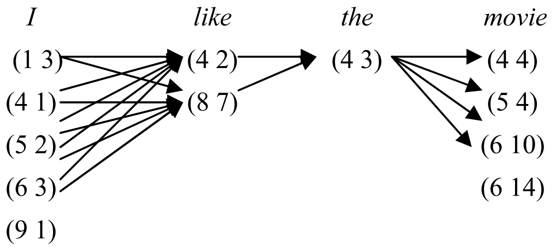

Figure 2. All possible sequences of words positions