A Method to Extract Sentences Referenced by Students' Technical Reports Using Parse Trees and Word concepts

Yasutoshi Haga Masayuki Arai
Graduate School of Sciences and Engineering, Teikyo University
06m108@uccl.teikyo-u.ac.jp, arai@ics.teikyo-u.ac.jp

Abstract

We are developing a system that automatically assesses Japanese technical reports submitted by university students. An important function of the system is to evaluate the accuracy of references in the reports. In this paper we propose a method to extract the sentences referenced in a report from the actual reference. The method extracts sentences using parse trees and word concepts from the reference. The experimental result shows that the fourth cumulative extraction rate of the method is about 90%.

1. Introduction

The teaching staffs of universities take a lot of time to assess students' reports, although the assessment standards are apt to differ among staff members. In order to resolve this problem, we are developing a system to automatically assess students' technical reports written in Japanese. The system has several functions. One of the important functions is to evaluate the accuracy of references in the reports.

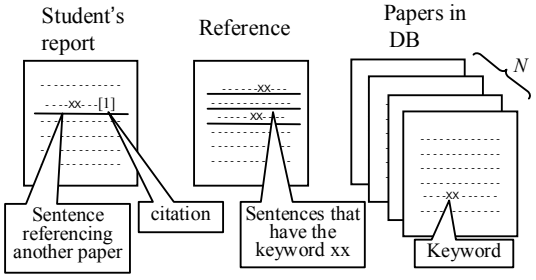
To implement this function, we propose a method to extract referenced sentences in a student's report from an actual reference. Using these two processes, we determine the similarity of sentences: (1) extracting a keyword from both the report and the reference, then choosing sentences using the keyword, and (2) selecting several sentences using parse trees and word concepts from the reference.

2. Proposed method

This section describes our proposed method to extract keywords and select several similar sentences from the reference.

2.1. Extracting a keyword and choosing sentences

First, the system chooses one sentence that has a citation in the student's report and extracts the same words existing in the sentence from both the report and the reference. Second, the system finds a word in the reference that has the max *tf・idf* (term frequency・inverse document frequency) value, which is often used in natural language processing, and establishes this as the keyword. Figure 1 depicts the calculation of *tf・idf*. Third, the system chooses sentences in the reference paper that have this keyword.



$$tf \cdot idf = tf \cdot \log\left(\frac{N}{df}\right) + 1 \cdots (1)$$

Figure 1. Calculation of *tf・idf*, where *xx* means the keyword, *tf* is the frequency of the keyword *xx* in the reference, *df* is the document frequency that has the keyword *xx* in the database DB and *N* is the number of all papers in the database.

2.2. Extracting several sentences from a reference using parse trees and words concepts

The sentences referencing other papers have the same paraphrasing characteristic: paraphrasing of the words and sentences written in the references. In this

978-0-7695-3167-0/08 $25.00 © 2008 IEEE
DOI 10.1109/ICALT.2008.48

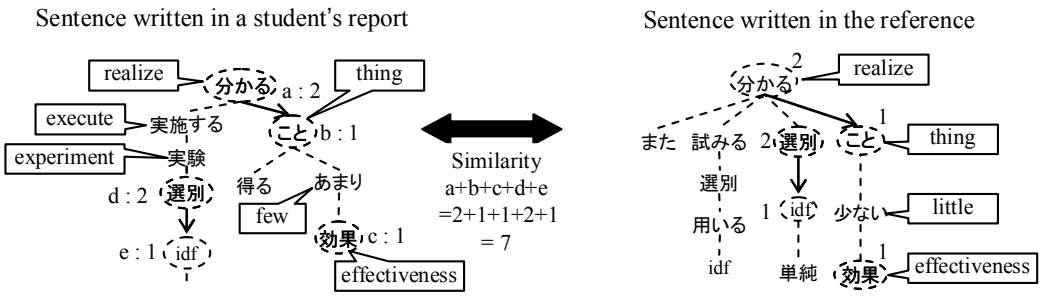


Figure 2. Calculating the similarity between two sentences using the *Tree Kernel* method. The similarity between two nodes—that is, nodes having the same word and no child-nodes, such as the node "effectiveness"—is 1. However, the similarity between nodes that have the same child-nodes, such as the node "realize," is 2. Consequently, the similarity between the above two parse trees is 7. The dashed circular nodes indicate same words.

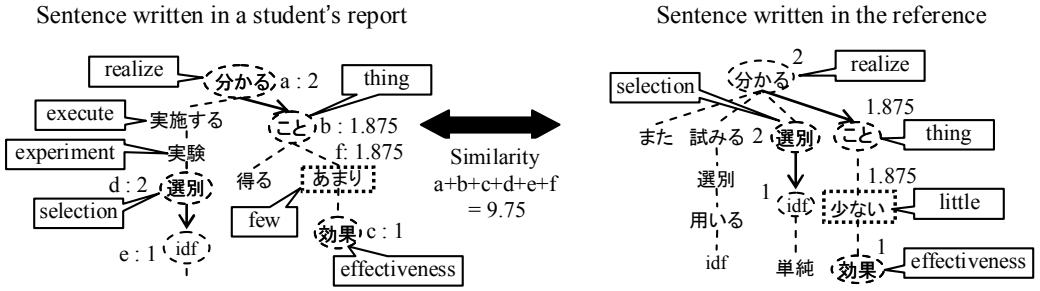


Figure 4. The system adds the distances of the word concepts and calculates the similarity between two sentences using the *Tree Kernel* method. The dashed rectangular nodes indicate similar words.

subsection, we propose a method to extract the referenced sentences with the characteristic.

2.2.1. Calculating the similarity value using parse trees. First, the system parses the sentences written in the student's report and the reference [1]. Second, the system generates parse trees for these sentences. Third, the system calculates the similarity between the two parse trees using the *Tree Kernel* method [2]. Figure 2 depicts an example of calculating the similarity.

2.2.2. Calculating the distance between two words using the word concept. The system uses the Electronic Dictionary Research (EDR) Concept Dictionary [3] to calculate the distance between two words [4]. Figure 3 depicts the structure of the dictionary and the calculation of the distance between two words.

2.2.3. Extracting several sentences from the reference using parse trees and word concepts. The system adds the distances between the contents of the two words to the *Tree Kernel*, as shown in Figure 4.

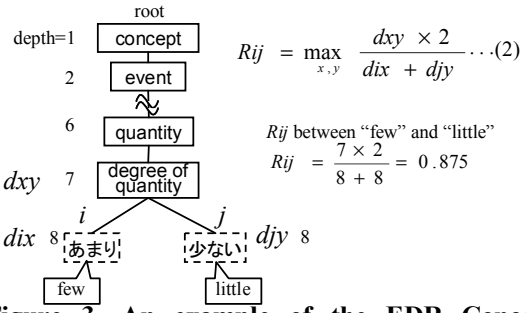


Figure 3. An example of the EDR Concept Dictionary and calculation of the distance between two words using the dictionary. The distance *Rij* between word *i* ("few") and word *j* ("little") is calculated with depths *dix* and *djy* from the root node and the depth of the same parent-node *dxy* from the root node. In this case, the distance between "few" and "little" is 0.875.

een the node "few" and the node "little" is zero because they are different words. However, in Figure 4, the distance between the two nodes becomes 1.875 because the distance between the two words is calculated at 0.875 according to the concept dictionary shown in Figure 3, and 1 is added because these two words have the same child-node, "effectiveness." As the result, the

similarity between two parse trees increases from 7.0, as shown in Figure 2, to 9.75, as shown in Figure 4. Finally, the system extracts several sentences that have a high similarity with the reference.

3. Experiment

We evaluated the method under the following conditions:

1. Eleven Japanese technical reports on computer science were used, and eighteen sentences referencing other papers were chosen from these reports.
2. Approximately twenty papers on computer science were prepared as the references, and they were stored in the database.
3. Several paragraphs from the references were extracted instead of sentences.
4. The N-th cumulative extraction rate, existing rate of the correct referenced sentence with in the best N candidates, was employed for evaluation.

In order to compare this method with other methods, we used *tf･idf* and *3-gram*, which are often utilized for natural language processing. Figure 5 shows the experimental result. Our method achieved the highest extraction rate among the three methods.

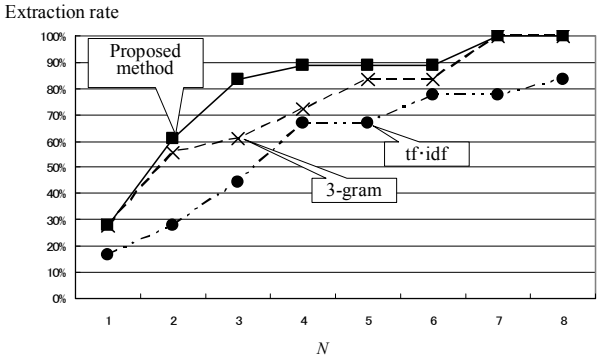


Figure 5. Experimental result (N-th cumulative extraction rate).

3-gram separates the sentences written in both the students' reports and references into three characters, and measures the frequency of the three characters to compare the sentences. Hence, *3-gram* has the following characteristic: it can obtain better results when it is applied to a few rephrased sentences or to sentences of almost the same length. In the technical papers, however, the sentences written in the references tend to be longer than those in the students' reports. Consequently, the extraction rate for this method is the lowest of the three methods.

tf･idf extracts distinctive words from the sentences and calculates the similarity between two sentences. In this experiment, however, we could not prepare many papers for the database; therefore, the *tf･idf* value of every word is very low and we could not achieve a high extraction rate.

4. Conclusion

We propose a method using parse trees and word concepts in reference papers to extract sentences referenced in the technical reports of university students. The experimental result shows the fourth cumulative extraction rate of the method is about 90%.

We have planned two future studies. First, in the current research, we used the EDR Concept Dictionary to compare words. However, the distance between two words that have different meanings but almost the same concept, such as "few" and "many," is very close. Hence, we plan to use the meaning of words instead of concepts.

Another study is to improve the *Tree Kernel* method. As shown in Figure 4, there is a parent-and-child relationship between the nodes "realize" and "selection" in the right parse tree. However, in the left parse tree, the two nodes do not have the parent-and-child relationship because the nodes "execute" and "experiment" are between these two nodes. Hence, we plan to propose a new *Tree Kernel* method that can skip non-relational nodes in order to obtain more accurate similarities between two parse trees.

5. References

[1] T. Kudo, and Y. Matsumoto, "Japanese Dependency Analysis Using Cascaded Chunking", Transactions of Information Processing Society of Japan, Vol.43, No.6, 2002, pp. 1834-1842 (in Japanese).

[2] M. Collins, and N. Duffy, "Convolution Kernels for Natural Language", In Proceedings of NIPS 2001.

[3] http://www2.nict.go.jp/r/r312/EDR/

[4] R. Fukaya, T. Yamamura, H. Kudo, T. Matsumoto, Y. Takeuchi, and N. Ohnishi, "Measuring Similarity between Documents Using Term Frequency", The transactions of the Institute of Electronics, Information and Communication Engineers, Vol.J87-D-II, No.2, 2004, pp. 661-672 (in Japanese).