Information Gain Ratio as Term Weight
— The case of Summarization of IR Results —

Tatsunori Mori
Graduate School of Environment and Information Sciences, Yokohama National University
79-7 Tokiwadai, Hodogaya, Yokohama 240-8501, Japan
mori@forest.eis.ynu.ac.jp

Abstract

This paper proposes a new term weighting method for summarizing documents retrieved by IR system. Unlike query-biased summarization, our method utilizes not the information of query, but the similarity information among original documents by hierarchical clustering. To map the similarity structure of the clusters into the weight of each word, we adopt the information gain ratio of probabilistic distribution of each word as term weight.

1 Introduction

Information retrieval (IR) becomes widely used in daily life to search for a variety of information. Those systems usually show not only the titles of documents but also the small pieces of documents, namely "summaries." Such summary information is expected to be helpful for users to judge the relevance of each (original) document to users' information need.

In general, most of search engines adopt simple strategies like presenting several portions of document which include the keywords in queries. Quality of summaries generated by such simple strategies is not usually enough for users to judge the relevance correctly. We need more sophisticated summarization methods.

The most basic and main way of automatic document summarization is the sentence extraction(Luhn, 1958). The importance of each sentence is calculated as the combination of several factors, like importance of each word (e.g. frequency, clue words etc.), position of the sentence in the document, the role of sentence(e.g. title etc.), and so forth(Okumura and Nanba, 2000). Especially, the sentence extraction based on importance of words is one of primary ways to summarize documents.

As the importance of words, term frequency (TF) is widely used, because it can be easily calculated within each document and works well. However, in order to improve the quality of summaries, we have to also consider other types of available information. The query-biased summarization(Tombros and Sanderson, 1998) is the mainstream of such methodologies. Although the strategy is very intuitive and works well, there are the following drawbacks.

- The strategy gives higher weight to the words in the queries. However, typical search engines usually make some efforts to improve the accuracy by modifying original queries, such as relevance feedbacks, query expansions, and so on. Those efforts are not reflected on summaries.
- Search engines may also retrieve irrelevant documents to queries. Since irrelevant documents scarcely contain the words in queries, the summarization of such documents falls into the single-document summarization with no bias.

In this paper, we propose a novel way to utilize the information lying in the set of retrieved documents in order to summarize the documents. Unlike query-biased summarization, our method utilizes not the information of query, but the similarity information among retrieved documents by hierarchical clustering. In order to map the similarity structure of documents into the weight of each word, we adopt the information gain ratio (IGR) of the probabilistic distribution of each word as a term weight. We will show the effectiveness of our method based on IGR by comparison with other systems.

2 Term Weighting Method based on Information Gain Ratio

2.1 Overview of Proposed Method

Our proposed method consists of the following steps as shown in Figure 1:

1. Make a hierarchical clustering structure of documents to obtain the information of similarity among them.
2. Calculate the weight of words according to document clusters and the probabilistic distribution of words.

3. Calculate the weight of each sentence according to the weight of words, and extract sentences of higher weights.

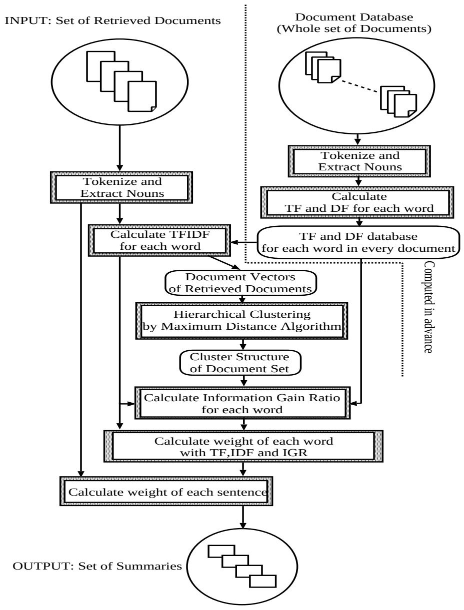


Figure 1: Overview of our Scheme

Through Step 1, it is expected that documents relevant to query and irrelevant documents are separately organized into different clusters. Note that we have to take account of the documents which are *not* retrieved but exist in the document database in order to obtain the information what words really contribute to selecting the retrieved documents. Therefore, as shown in Figure 2 we introduce another layer of cluster, which corresponds to the set of the whole document database. The cluster consists of two sub-clusters. One sub-cluster is the cluster of retrieved documents, which is the target of further clustering. The other one corresponds to the rest of database. The contrast between those sub-clusters should implicitly carry the information of query.

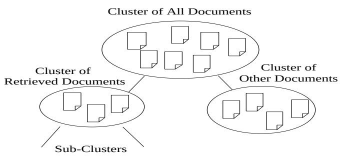


Figure 2: Clustering Retrieved Documents

For Step 1, we adopt a recursive version of Maximum distance clustering algorithm(Tou and Gonzalez, 1974), in which one cluster may have more than two sub-clusters according to distance among documents. Similarity among documents are defined in terms of the vector space model.

The hierarchical structure in Figure 2 represents not only the similarity structure among the retrieved documents but also the structure of contrast between the retrieved documents and the non-retrieved documents. Therefore, in Step 2 we would like to map such structural information into weight of words. As for the step, we introduce a way to measure the contribution of each word to forming a given cluster structure. It is based on a measure, called *information gain ratio* (IGR).

By combining the weight of IGR with other weights such as TF and the inverse document frequency (IDF), we obtain a composite weight for each word in documents. Note that those three types of weight have different features. IGR, TF, IDF represent the importance of a word in a cluster, in a document, and in the whole document database, respectively. Therefore, we expect that the composite weight would be an overall weight in retrieved documents.

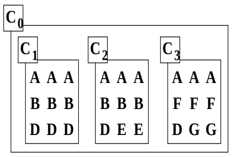
2.2 Term Weighting based on Information Gain Ratio

Each inner node of the cluster tree represents the partition of a cluster based on the similarity among documents. Therefore, the information about similarity among documents can be mapped into the weight of words by the following steps.

1. For each cluster, calculate the weight of each word according to the partition.
2. Since each document is specified by the series of partitions from the root node to a leaf of the cluster tree, calculate the total weight of each word in a document by integrating the weights of each word in the series of partitions.

Step 1 is the very core of our method. The basic idea is that we assign a higher weight to the word that makes more contribution to determine the structure of the sub-clusters. The degree of contribution can be measured by the degree of consistency between the distribution of a word and the partition of a cluster.

For example, let us consider partitioning the cluster into three sub-clusters as shown in Figure 3. Since the word A is the most frequent word in the cluster C_0, A can be regarded as a characteristic word of C_0. Several methods proposed so far, indeed, adopt the most frequent words as the keywords of clusters. However, we can see that the word A is not useful for selecting one of sub-clusters, because the word uniformly appears in all of sub-clusters. On the other hand, the word F is not high frequency in C_0 but concentrates on the cluster C_3. Therefore, the appearance of the word F is a good clue to select the cluster C_3. In this case, we may consider that the word F contributes to deciding the sub-structure of the cluster better than the word A. In the next section, we propose the use of *Information Gain Ratio*(IGR) as a measure of such contribution.



A∼G: Words

Figure 3: Word Distribution and Partition of Cluster

2.2.1 Information Gain Ratio

The IGR originally is the measure of goodness for attributes used in the decision tree leaning algorithm C4.5(Quinlan, 1993). It represents how precisely the attributes predict the classes of example cases. By regarding a cluster structure of documents as a decision tree, we may use the IGR as a measure of the consistency between the distribution of a word and the partition of a cluster.

The IGR value of the word w in the cluster C, $gain\_r(w,C)$, is calculated as follow:

$$
\begin{aligned}
gain\_r(w,C) &= \frac{gain(w,C)}{split\_info(C)}, \qquad (1)\\
gain(w,C) &= entropy(w,C) - entropy_p(w,C),\\
entropy(w,C) &= -p(w|C)\log_2 p(w|C)\\
&\quad -(1-p(w|C))\log_2(1-p(w|C)),\\
p(w|C) &= freq(w,C)/|C|,\\
entropy_p(w,C) &= \sum_i \frac{|C_i|}{|C|} entropy(w,C_i),\\
split\_info(C) &= -\sum_i \frac{|C_i|}{|C|}\log\frac{|C_i|}{|C|},
\end{aligned}
$$

where $freq(w,C)$, C_i and $|C_i|$ are the frequency of the word w in C, the i-th sub-cluster of C, and the number of words in C_i, respectively.

For example, the IGR values of words in Figure 3 hold the following relation:

$$
\begin{aligned}
&gain\_r(\mathrm{B},C_0) \simeq gain\_r(\mathrm{F},C_0)\\
&\quad > gain\_r(\mathrm{E},C_0) = gain\_r(\mathrm{G},C_0)\\
&\qquad > gain\_r(\mathrm{D},C_0) > gain\_r(\mathrm{A},C_0).
\end{aligned}
$$

$$
\begin{aligned}
&gain\_r(\mathrm{A},C_0) = 0.000, gain\_r(\mathrm{B},C_0) = 0.161,\\
&gain\_r(\mathrm{D},C_0) = 0.031, gain\_r(\mathrm{E},C_0) = 0.080,\\
&gain\_r(\mathrm{F},C_0) = 0.157, gain\_r(\mathrm{G},C_0) = 0.080.
\end{aligned}
$$

2.2.2 Weighting Terms based on Information Gain Ratio

For each word in every document, we can collect a set of IGR values by tracing the path in the cluster tree from the root to the leaf corresponding to the document. There would be several ways to use the set of IGR values according to the design of user interfaces.

For instance, we need to integrate the set of IGR values into one value if we adopt a list-style user interface, which displays the ranked list of documents along with each summary, like the interfaces of Web search engines. There would be several ways of integration, e.g., summation of all values, the maximum value, and so on.

In this paper, we suppose the list-style user interface, where all summaries of retrieved documents are shown to the user at once as a list, and adopt the integration given by the summation shown in (2) and Figure 4:

$$igr(w,D) = \sum_{C \in Cset(D)} gain\_r(w,C), \qquad (2)$$

where $Cset(D)$ is the set of all clusters to which the document D belongs. The integration method equally takes account of each IGR value in a path of cluster. It is based on our assumption that all of cluster partitions evenly contribute to selecting corresponding documents. With the weight $igr(w,D)$, we define the weight

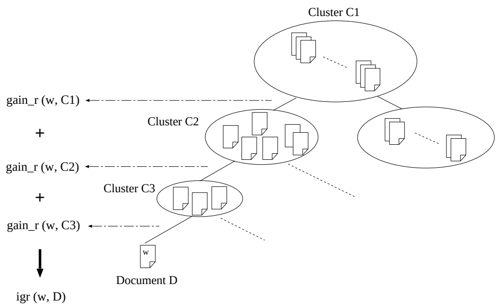


Figure 4: Integration of IGR values

$weight(w,D)$ of the word w in the document D as the combination of the three types of fundamental weights, TF, IDF and IGR.

$$weight(w,D) = igr(w,D)\cdot tf(w,D)\cdot idf(w) \qquad (3)$$

3 Experimental Evaluation

3.1 Summarization by Sentence Extraction

Our aim in this paper is to show that our weighting method is effective in summarizing retrieved documents. Therefore, we use the most fundamental scheme of summarization, which is the sentence extraction based on the term weighting as follows.

1. Let the importance $s\_imp(s, D)$ of the sentence s in the document D be the average weight of keywords in the sentence.

$$s\_imp(s, D) = \frac{1}{|s|} \sum_{w \in keyw(s)} weight(w, D),$$

where $|s|$ and $keyw(s)$ are the number of words in s and the list of keywords in s.

2. Extract sentences with higher importance from the original document, until the total length of selected sentences exceeds a certain predetermined length of summary.

Our experiment is performed under the conditions that keywords are nouns, and the cutoff length is 150 words.

3.2 Evaluation of Summarization in IR Tasks

We evaluated our term weighting method in IR tasks of NTCIR2 Text Summarization Challenge 1 (TSC1)(Fukusima and Okumura, 2001). The data set distributed by TSC1 committee consists of 12 topics. Each topic has one query and 50 retrieved documents. Those documents are retrieved by a search engine from all of Mainichi Shimbun Newspaper articles in 1998.

Figure 5 shows the scheme of the evaluation. Every participant made a summary for each document with his/her system and submitted 600 summaries to TSC1 committee. TSC1 committee evaluated the summaries by presenting the queries and the summaries to 36 subjects(36 students). Three subjects were assigned to one of topics and they judged the relevance between the query and each summary. The quality of summaries are evaluated by comparing subjects' relevance judgments for summaries with the relevance judgment for the original documents, which is carefully assigned by TSC1 committee. If those two relevance judgments are highly consistent with each other, we may conclude that the system is very effective in summary generation for retrieved documents.

Although the relevance of each original document is graded either 'A(relevant)', 'B(related)' or 'C(not relevant)', each subject answers the relevance of each summary with 'Yes' or 'No'. Therefore, there are, at least, two criteria for evaluating consistency: *Answer Level A*, in which documents of the grade A are regarded as 'relevant' (Strict Evaluation), and *Answer Level B*, in which documents of either the grade A or the grade B are regarded as 'relevant' (Loose Evaluation).

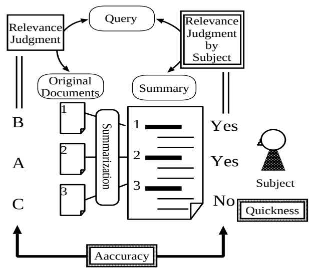


Figure 5: Evaluation of Summaries in IR task

3.3 Experimental Result

The experimental result of our system in TSC1 is shown in Table 1 along with the results of other participating systems and three baseline systems, 'Fulltext', 'TF with QB' and 'Lead'.

In the summarization for retrieved documents, it is important to improve both of the accuracy of the judgments of relevance and the time required to make the judgments, simultaneously. However, there is a trade-off relation between them. Therefore, we plot the relation between the time for judgment and the other measures in Figure 6.

4 Discussion

4.1 Accuracy of Performance of Task

4.1.1 Answer Level A

In this section, we consider the evaluation of 'Answer Level A'. Our system outperforms other participating systems in terms of all of measures, the average precision, the average recall and the average F measure. Although the precision of the 'Lead' method is 1.5 point higher than our system, our system outperforms all baseline systems in other measures.

The F measure of 'Lead' method is 7.7 point lower than our system, because the precision of the method is the lowest. The 'Lead' can be regard as a precision-oriented method.

In comparison with 'TF with QB' method, our system is 10.9 point higher in the recall, 2.7 point higher in the precision and 7.0 point higher in the F-measure. It would show that we can make effective summaries even if we do not use the information of query.

Note that the performance of 'Fulltext' is not the best one. It shows that summarized presentation can reduce unnecessary information that would degrade the performance through information overload, as well as Okumura et al.(Okumura and Mochizuki, 2000) pointed out.

Next, let us consider the relation between the accuracy of the relevance judgments and the

Table 1: Experimental Result in TSC1

Ans. Lev.		Proposed	Sys 1	Sys 2	Sys 3	Sys 4	Sys 6	Sys 7	Sys 8	Sys 9	Fulltext	TF with QB	Lead
A	Rec.	**0.907**	0.833	0.899	0.793	0.818	0.858	0.831	0.824	0.849	0.843	0.798	0.740
	Pre.	0.751	0.728	0.717	0.685	0.674	0.718	0.739	0.738	0.741	0.711	0.724	**0.766**
	F	**0.808**	0.761	0.785	0.715	0.718	0.763	0.766	0.749	0.768	0.751	0.738	0.731
B	Rec.	0.754	0.741	**0.793**	0.715	0.737	0.745	0.719	0.719	0.752	0.736	0.700	0.625
	Pre.	0.897	0.921	0.904	0.898	0.875	0.892	0.908	0.913	**0.923**	0.888	0.913	0.921
	F	0.797	0.808	**0.828**	0.776	0.773	0.785	0.779	0.775	0.805	0.773	0.776	0.712
	Time	8:33	9:41	12:48	**6:25**	6:44	9:01	10:16	9:16	9:31	13:46	8:44	7:32
	Len.	234.4	297.8	585.7	**89.5**	136.4	288.4	292.9	266.1	262.5	819.4	253.6	174.5

Sys 1 ~ 9:	Other participating systems.
Fulltext:	The system which just returns the orignal documents.
TF with QB:	The system which generates summaries with TF-based sentence extraction. The weight of words in the query is doubled. Compression ratio is 20%.
Lead:	The system which returns the lead of document. Compression ratio is 20%.
Ans.Lev.:	Answer Level.
Rec.,Pre.,F:	Recall, Precision and F-measure.
Time:	Average time to accomplish the relevance judgment of one task (50 summaries).
Len.:	Average length of Summaries in character.

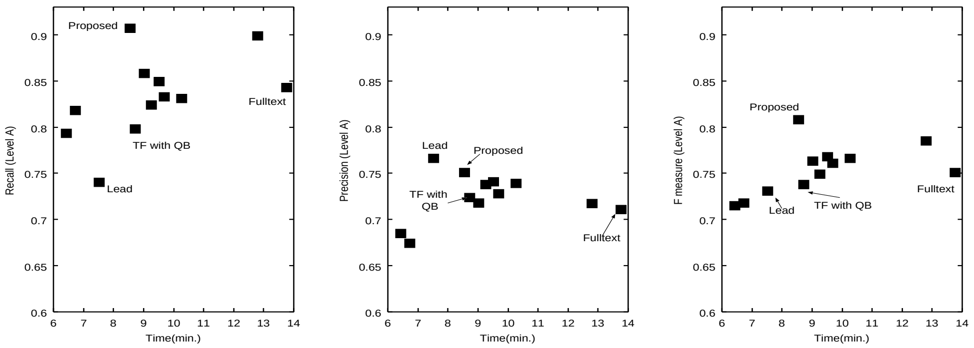


Figure 6: Time for judgment v.s. Recall, Precision and F-measure in Answer Level A

time for judgment. Since, in Figure 6, the effective systems will be located in the upper-left corner, we may conclude that our system out performed other systems. Especially, the difference of recall rate among systems is wider than other measures and we can see the our system is the highest.

4.1.2 Answer Level B

Since the number of relevant documents increases in 'Answer Level B', the precision grows and the recall decreases. If a system has high precision in 'Answer Level A,' the recall in 'Answer Level B' will remarkably fall. On the other hand, a system will gain in precision if the main cause of error in 'Answer Level A' is that the documents of the relevance level B are judged as relevant.

Although the recall of our system decreases from 0.907 to 0.754, it is still in second place among participating systems. Thus, our system generate more summaries which are judged correctly as relevant than other systems.

On the other hand, the precision does not grow as other systems do and is degraded to the seventh place. It shows that our system generates more inappropriate summaries, which are originally the grade-C documents but are judged as relevant, than other systems. It means that our system can extract the sentences related to the topic successfully, but drops some contexts for them crucial to the relevance judgment. The main reason for it is that our system is based on simplest sentence extraction and does not use the the information about cohesion among sentences.

4.2 Evaluation on Term Weighting

Since we do not have 'correct answer' of weighting terms, it is difficult to evaluate the quality of term weighting directly. Thus, in this section, we show an example of term weighting by our method, along with the weighting by the TF method and TFIDF method. In order to examine the typical result of term weighting, let us consider Topic 1027 of TSC1 Task B (Table 2), on which our system get top marks in F-measure. From the set of given retrieved documents, we pick up three documents in relevance grade A,B and C respectively, and calculate the term weight for each documents as shown in Table 3,4 and 5. In those tables, 'IGRsum' is the summation of IGR values defined by (2), and 'TFIDFIGRsum' is the our final weight defined by (3).

As for the Grade A document, the topic-related words, like *BS*, *program*, *digital*, *appli-*

cation, *satellite*, *MPT*[1], are ranked in higher places by TFIDFIGRsum as well as the words explicitly appeared in the topic description. On the other hand, the irrelevant word *SDTV*[2], which ranks higher by the TFIDF method, disappears from the top ten list.

As we can see in those tables, the factor of IGRsum mainly contributes to the effect. In the case of the word *SDTV*, the IGR value of the upper-most cluster (9.03×10^{-5}) is much lower than the word *HDTV* (7.06×10^{-4}), because *SDTV* appears in not only the retrieved documents but also the non-retrieved documents.

Table 2: TSC Task B Topic 1027

Description	Hi-vision Television
Narrative	Articles on the information about Hi-vision Television (HDTV) from the government, companies, users and so on. ...

The topic is originally written in Japanese.

5 Related Works

Summarization of retrieved documents has the features that the following kinds of extra information are given:

1. A query,
2. A set of documents to be summarized.

As described before, the method which utilize the information 1 is called query-biased summarization. Tombros et al.(Tombros and Sanderson, 1998) and Shiomi et al.(Shiomi et al., 1998) independently propose the method to give the higher weight to the terms in queries and confirm the effectiveness of it. Although those methods, which use queries directly, are very intuitive and works well, there are the drawbacks as described in Section 1.

In the same way as ours, Eguchi et al.(K.Eguchi et al., 1999), Fukuhara et al.(T.Fukuhara et al., 1999) and Radev et al.(Radev et al., 2000) use the information 2. Eguchi et al.(K.Eguchi et al., 1999) propose an IR system based on some kind of relevance feedback. The system partitions the set of retrieved documents into clusters. Then, it represent a "summary" of each cluster in order for the user to select a relevant cluster and feedback it to the system. The system proposed by Fukuhara et al.(T.Fukuhara et al., 1999) also makes clusters of retrieved documents, then, extracts topic words from the viewpoint of the notions of 'skewness' and 'kurtosis'. The summaries are generated by linking up sentences which have relevant topics. The method by Radev et al.(Radev et al., 2000; Radev et al., 2000) calculates the centroid vector for each cluster and the component of centroid vector is used as the main factor of term weight.

All of those systems uses the clustering of documents only to find groups of similar documents and calculate the term weight according to *with-in cluster* information. On the other hand, our method examines the structure of clusters in further detail and utilize it in order to weight terms.

In the evaluation of TSC1, the result of summarization is the set of summaries, although we take account of the relation among documents. Recently, the researches on making one summary document from multiple related documents have received much attention(Mani and Bloedorn, 1999; McKeown and Radev, 1999). Although our method for term weighting will have an effect in summarization of multiple documents, we will need some other methods to control the redundancy among sentences like *Maximum Marginal Relevance*(Carbonell and Goldstein, 1998).

[1]Ministry of Posts and Telecommunications (of Japan).

[2]Standard Definition Television.

6 Concluding Remarks

In this paper, we proposed a novel way to utilize the information lying in the set of retrieved documents in order to weight terms and summarize the documents. Our method utilizes the similarity information among original documents by hierarchical clustering. In order to map the similarity structure of documents into the weight of each word, we adopt the information gain ratio of the probabilistic distribution of a word as a term weight. In the experiments of TSC1, we showed that our term weighting method is very effective in summarization of retrieved documents.

In future work, we plan to investigate the utilization of our IGR-based term weighting method in an interactive user interface of IR. In this paper, every part of cluster structure is uniformly reflected in the weight of each term. On the other hand, as an interactive user interface of IR, we can imagine a system where the user selects one sub-cluster recursively to reach a desired document. In this case, words may be weighted according to the cluster structure presented to the user.

References

Jaime Carbonell and Jade Goldstein. 1998. The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Sum-

Table 3: Example of term weight (Topic 1027, Doc-ID 980822075, Relevance A)

TF		TFIDF		IGRsum		TFIDFIGRsum	
broadcasting	8	*broadcasting*	1.14×10^{-4}	*broadcasting*	9.52×10^{-3}	*broadcasting*	1.09×10^{-6}
application	6	*BS*	9.68×10^{-5}	*digital*	3.10×10^{-3}	*BS*	2.54×10^{-7}
program	5	*application*	9.41×10^{-5}	*BS*	2.63×10^{-3}	*program*	1.87×10^{-7}
BS	4	*SDTV*	8.23×10^{-5}	*program*	2.38×10^{-3}	*digital*	1.22×10^{-7}
satellite	3	*program*	7.86×10^{-5}	*TV*	2.20×10^{-3}	*HDTV*	1.10×10^{-7}
a program	2	*a program*	7.66×10^{-5}	*Hi-vision*	2.13×10^{-3}	*application*	8.37×10^{-8}
HDTV	2	*HDTV*	6.44×10^{-5}	*HDTV*	1.70×10^{-3}	*satellite*	7.95×10^{-8}
SDTV	2	*satellite*	5.42×10^{-5}	*MPT*	1.66×10^{-3}	*MPT*	6.20×10^{-8}
TV	2	*three programs*	4.34×10^{-5}	*satellite*	1.47×10^{-3}	*TV*	5.55×10^{-8}
digital	2	*twenty-two programs*	4.34×10^{-5}	*digital broadcasting*	1.37×10^{-3}	*Hi-vision*	5.44×10^{-8}

Each word is originally written in Japanese.

Table 4: Example of term weight (Topic 1027, Doc-ID 981215216, Relevance B)

TF		TFIDF		IGRsum		TFIDFIGRsum	
Kansai area	8	*Kansai area*	1.08×10^{-4}	*broadcasting*	9.52×10^{-3}	*broadcasting*	8.14×10^{-7}
broadcasting	6	*broadcasting*	8.55×10^{-5}	*enterprise*	3.15×10^{-3}	*BS*	1.27×10^{-7}
information	4	*CS*	7.17×10^{-5}	*information*	2.95×10^{-3}	*CS*	1.20×10^{-7}
CS	3	*program production*	6.59×10^{-5}	*BS*	2.63×10^{-3}	*Kansai area*	1.20×10^{-7}
satellite	3	*sending*	5.77×10^{-5}	*program*	2.38×10^{-3}	*information*	1.19×10^{-7}

Table 5: Example of term weight (Topic 1027, Doc-ID 981108028, Relevance C)

TF		TFIDF		IGRsum		TFIDFIGRsum	
broadcasting	7	*Radio Free Europe*	1.30×10^{-4}	*broadcasting*	9.52×10^{-3}	*broadcasting*	9.50×10^{-7}
Czech	6	*Czech*	1.26×10^{-4}	*enterprise*	3.15×10^{-3}	*program*	1.12×10^{-7}
free	5	*broadcasting*	9.98×10^{-5}	*information*	2.95×10^{-3}	*Czech*	1.09×10^{-7}
Iraq	4	*Prague*	7.95×10^{-5}	*program*	2.38×10^{-3}	*Radio Free Europe*	7.03×10^{-8}
Iran	4	*Iran*	7.64×10^{-5}	*station*	1.21×10^{-3}	*free*	5.01×10^{-8}

maries. In *Proceedings of the 21st Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 335–336.

Takahiro Fukusima and Manabu Okumura. 2001. Text Summarization Challenge Text summarization evaluation in Japan (TSC). In *Proceedings of NAACL 2001 Workshop on Automatic Summarization*, June.

K.Eguchi, H.Ito, A.Kumamoto, and Y.Kanata. 1999. Adaptive Query Expansion Based on Clustering Search Results. *Transaction of Information Processing Society of Japanese*, 40(5):2439–2449.

H.P. Luhn. 1958. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2):159–165.

Inderjeet Mani and Eric Bloedorn. 1999. Summarizing Similarities and Differences among Related Documents. *Information Retrieval*, 1(1-2):35–67.

Kathleen R. McKeown and Dragomir R. Radev. 1999. Generating Summaries of Multiple News Articles. In Inderjeet Mani and Mark T. Maybury, editors, *Advances in Automatic Text Summarization*, chapter 24, pages 381–389. The MIT Press.

Manabu Okumura and Hajime Mochizuki. 2000. Query-biased summarization based on lexical chaining. *Computational Intelligence*, 16(4):578–585.

Manabu Okumura and Hidetsugu Nanba. 2000. Recent advanced in automated text summarization. Technical Memorandum IS-TM-2000-001, School of Information Science, Japan Advanced Institute Of Science and Technology, 7. (in Japanese).

J. Ross Quinlan. 1993. *C4.5: progarms for machine learning*. Morgan Kaufmann Publishers, May.

Dragomir R. Radev, Hongyan Jing, and Malgorzata Budzikowska. 2000. Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. In *Proceedings of ANLP/NAACL Workshop on Summarization.*

Takakazu Shiomi, Katsumi Tokuda, Shoichi Aoyama, and Koji Kakigahara. 1998. An Abstraction Method Using Viewpoints. In *Proceedings of The 56-th Annual Meeting of Information Processing Society of Japan*, volume 3, pages 104–105. (in Japanese).

T.Fukuhara, H.Takeda, and T.Nishida. 1999. Multiple-text Summarization for Collective Knowledge Formation. In *Proceedings of Workshop on Social Aspects of Knoledge and Memmory.* IEEE Systems, Man and Cyberbetics Conference.

A. Tombros and M. Sanderson. 1998. Advantages of Query Biased Summaries in Information Retrieval. In *Proceedings of the 21st Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 2–10.

Julius T. Tou and Rafael C. Gonzalez. 1974. *Pattern recognition principles.* Addison-Wesley Pub. Co.