An Algorithm for One-page Summarization of a Long Text Based on Thematic Hierarchy Detection

Yoshio Nakao
Fujitsu Laboratories Ltd.
Kamikodanaka 4-1-1, Nakahara-ku, Kawasaki, Japan, 211-8588
nakao@flab.fujitsu.co.jp

Abstract

This paper presents an algorithm for text summarization using the thematic hierarchy of a text. The algorithm is intended to generate a one-page summary for the user, thereby enabling the user to skim large volumes of an electronic book on a computer display. The algorithm first detects the thematic hierarchy of a source text with lexical cohesion measured by term repetitions. Then, it identifies boundary sentences at which a topic of appropriate grading probably starts. Finally, it generates a structured summary indicating the outline of the thematic hierarchy. This paper mainly describes and evaluates the part for boundary sentence identification in the algorithm, and then briefly discusses the readability of one-page summaries.

1 Introduction

This paper presents an algorithm for text summarization using the thematic hierarchy of a long text, especially for use by readers who want to skim an electronic book of several dozens of pages on a computer display.

For those who want an outline to quickly understand important parts of a long text, a one-page summary is more useful than a quarter-size summary, such as that generated by a typical automatic text summarizer. Moreover, a one-page summary helps users reading a long text online because the whole summary can appear at one time on the screen of a computer display.

To make such a highly compressed summary, topics of appropriate grading must be extracted according to the size of the summary to be output, and selected topics must be condensed as much as possible. The proposed algorithm decomposes a text into an appropriate number of textual units by their subtopics, and then generates short extracts for each unit. For example, if a thirty-sentence summary is required to contain as many topics as possible, the proposed algorithm decomposes a source text into approximately ten textual units, and then generates a summary composed of two- or three-sentence extracts of these units.

The proposed algorithm consists of three stages. In the first stage, it detects the thematic hierarchy of a source text to decompose a source text into an appropriate number of textual units of approximately the same size. In the second stage, it adjusts each boundary between these textual units to identify a *boundary sentence*, indicating where a topic corresponding to a textual unit probably starts. It then selects a *lead sentence* that probably indicates the contents of subsequent parts in the same textual unit. In the last stage, it generates a structured summary of these sentences, thereby providing an outline of the thematic hierarchy of the source text.

The remainder of this paper includes the following: an explanation of problems in one-page summarization that the proposed algorithm is intended to solve; brief explanations of a previously published algorithm for thematic hierarchy detection (Nakao, 1999) and

a problem that must be solved to successfully realize one-page summarization; a description and evaluation of the algorithm for boundary sentence identification; a brief explanation of an algorithm for structured summary construction; and some points of discussion on one-page summarization for further research.

2 Problems in one-page summarization of a long text

This section examines problems in one-page summarization. The proposed algorithm is intended to solve three such problems.

The first problem is related to text decomposition. Newspaper editorials or technical papers can be decomposed based on their rhetorical structures. However, a long aggregated text, such as a long technical survey report, cannot be decomposed in the same way, because large textual units, such as those longer than one section, are usually constructed with only weak and vague relationships. Likewise, their arrangement may seem almost at random if analyzed according to their logical or rhetorical relationships. Thus, a method for detecting such large textual units is required.

Since a large textual unit often corresponds to a logical document element, such as a part or section, rendering features of logical elements can have an important role in detecting such a unit. For example, a section header is distinguishable because it often consists of a decimal number followed by capitalized words. However, a method for detecting a large textual unit by rendering features is not expected to have wide range of applicability. In other words, since the process for rendering features of logical elements varies according to document type, heuristic rules for detection must be prepared for every document type. That is a problem. Moreover, the logical structure of a text does not always correspond to its thematic hierarchy, especially if a section consists of an overview clause followed by other clauses that can be divided into several groups by their subtopics.

Since then, based on Hearst's work (1994), an algorithm for detecting the thematic hierarchy of a text using only lexical cohesion (Haliday and Hasan, 1976) measured by term repetitions was developed (Nakao, 1999). In comparison with some alternatives (Salton et al., 1996; Yaari, 1998), one of the features of the algorithm is that it can decompose a text into thematic textual units of approximately the same size, ranging from units just smaller than the entire text to units of about one paragraph. In this paper, a summarization algorithm based on this feature is proposed.

The second problem is related to the textual coherence of a one-page summary itself. A three-sentence extract of a large text, which the proposed algorithm is designed to generate for an appropriate grading topic, tend to form a collection of unrelated sentences if it is generated by simple extraction of important sentences. Furthermore, the summary should provides new information to a reader, so an introduction is necessary to help a reader understand it. Figure 4 shows a summary example of a technical survey report consisting of one hundred thousand characters. It was generated by extracting sentences with multiple significant terms as determined by the likelihood ratio test of goodness-of-fit for term frequency distribution. It seems to have sentences with some important concepts (keywords), but they do not relate much to one another. Moreover, inferring the contexts in which they appear is difficult.

To prevent this problem, the proposed algorithm is designed to extract sentences from only the lead part of every topic.

The third problem is related to the readability of a summary. A one-page summary is much shorter than a very long text, such as a one-hundred-page book, but is too long to read easily without some breaks indicating segues of topics. Even for an entire expository text, for which a method for displaying the thematic hierarchy with generated headers was proposed to assist a reader to explore the content (Yaari, 1998), a good summary is required to help a user understand quickly.

To improve readability, the proposed algorithm divides every one-page summary into

several parts, each of which consists of a heading-like sentence followed by some paragraphs.

3 Text Summarization Algorithm

3.1 Thematic Hierarchy Detection

In the first stage, the proposed algorithm uses the previously published algorithm (Nakao, 1999) to detect the thematic hierarchy of a text based on lexical cohesion measured by term repetitions. The output of this stage is a set of lists consisting of thematic boundary candidate sections (TBCS). The lists correspond individually to every layer of the hierarchy and are composed of TBCSs that separate the source text into thematic textual units of approximately the same size.

3.1.1 Thematic Hierarchy Detection Algorithm

First, the algorithm calculates a cohesion score at fixed-width intervals in a source text. According to Hearst's work (1994), a cohesion score is calculated based on the lexical similarity of two adjacent fixed-width windows (which are eight times larger than the interval width) set at a specific point by the following formula:

$$c(b_l, b_r) = \frac{\Sigma_t w_{t,b_l} w_{t,b_r}}{\sqrt{\Sigma_t w_{t,b_l}^2 \Sigma_t w_{t,b_r}^2}}$$

where b_l and b_r are the textual block in the left and right windows, respectively, and w_{t,b_l} is the frequency of term[1] t for b_l, and w_{t,b_r} is the frequency t for b_r. Hereafter, the point between the left and right windows is referred to as the reference point of a cohesion score.

The algorithm then detects thematic boundaries according to the minimal points of four-item moving average (arithmetic mean of four consecutive scores) of the cohesion score series. After that, it selects the textual area contributing the most to every minimal value and identifies it as a TBCS.

Figure 1 shows the results of a TBCS detection example, where FC is, Forward Cohesion, a series of average values plotted at the reference point of the first averaged score, and BC is, Backward Cohesion, a series of averaged values plotted at the reference point of the last averaged score. Since the textual area just before the point at which FC plotted is always in the left window when one of the averaged cohesion scores is calculated, FC indicates the strength of forward (left-to-right) cohesion at a point. Conversely, BC indicates the strength of backward cohesion at a point. In the figure, EP is, Equilibrium Point, the point at which FC and BC have an identical value. The algorithm checks for FC and BC starting from the beginning till the end of the source text; and it records a $TBCS$, as depicted by the rectangle, whenever an equilibrium point is detected (see (Nakao, 1999) for more information).

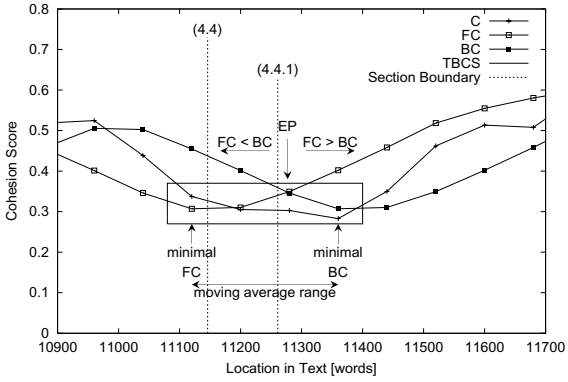

Figure 1: Example of TBCS Detection

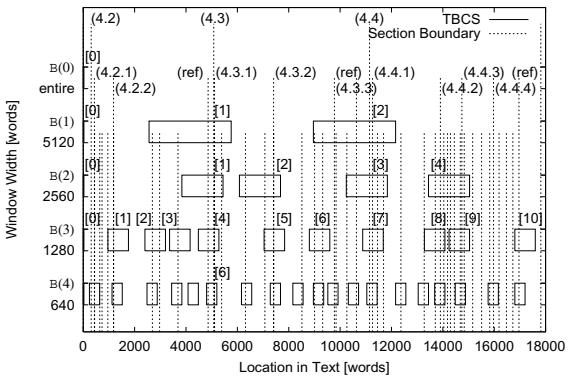

Figure 2: Example of Thematic Hierarchy

For a sample text, Figure 2 shows the resulting thematic hierarchy that was detected

[1] All content words (i.e., verbs, nouns, and adjectives) extracted by a tokenizer for Japanese sentences.

Table 1: Accuracy of Thematic Hierarchy Detection

Window width	Boundary #		Original TBCS				Unified TBCS			
	cor.	res.	Recall		Precision		Recall		Precision	
5120	1	2	100	(22)	50	(11)	100	(0.3)	50	(0.1)
2560	2	4	100	(22)	50	(11)	50	(0.5)	25	(0.3)
1280	3	10	100	(27)	30	(8.1)	67	(1.4)	20	(0.4)
640	30	42	90	(23)	64	(16)	57	(2.3)	40	(1.7)
320	114	163	67	(22)	47	(16)	46	(4.5)	33	(3.2)
160	184	365	70	(22)	35	(11)	51	(9.1)	25	(4.6)
80	322	813	57	(25)	23	(10)	57	(21)	23	(8.2)
40	403	1681	52	(25)	13	(6.2)	71	(42)	17	(10)

The figures in parentheses are the baseline rates.

by the aforementioned procedure using varying window widths (the ordinates). Each horizontal sequence of rectangles depicts a list of TBCSs detected using a specific window width.

To narrow the width of candidate sections, the algorithm then unifies a TBCS with another TBCS in the layer immediate below. It continued the process until TBCSs in all layers, from the top to the bottom, are unified. After that, it outputs the thematic hierarchy as a set of lists of TBCS data:

- i: layer index of the thematic hierarchy
- $B(i)[j]$: TBCS data containing the following data members:
  - *ep*: equilibrium point
  - *range*: thematic boundary candidate section.

In Figure 2, for example, $B(1)[1]$ is unified with $B(2)[1], B(3)[4], B(4)[6], \ldots$, and the values of its data members (ep and range) are replaced by those of the unified TBCS in the bottom layer, which has been detected using the minimum window width (40 words).

3.1.2 Results of Thematic Hierarchy Detection

Table 1 summarizes the accuracy of thematic hierarchy detection in an experiment using the following three kinds of Japanese text as test data: a technical survey report[2] that consists of three main sections and contains 17,816 content words; eight series of newspaper columns[3], each of which consists of 4 to 24 articles containing about 400 words; and twelve economic research reports[4], each of which consists of about ten articles containing 33 to 2,375 words.

In the table, *cor.* denotes the number of the correct data values composed of the starting points of sections that contain the same number of words or more than the window width listed in the same row[5]. In addition, *res.* denotes the number of TBCSs. The *original TBCS* columns list the recall and precision rates of detected TBCSs before TBCS unification, and the *unified TBCS* columns list those rates after TBCS unification. On each layer, the width of candidate sections for *original TBCS* is about half of the window width; and that of *unified TBCS* is 25 words (about half of the minimum window width). The figures shown in parentheses are the baseline rates corresponding to random selection. That is, parts are randomly selected from the source text whose total size is equal to the total area size of TBCSs.

As the boundary figures indicate, the proposed algorithm decomposes a text into textual units of about equivalent window widths. In addition, the rates of detected TBCSs are clearly larger than their baselines. Further-

[2] "Progress Report of Technical Committee on Network Access" in *Survey on Natural Language Processing Systems* by Japan Electronic Industry Development Association, chapter 4, pp. 117–197, Mar. 1997.

[3] Obtained from the Daily Yomiuri On-line (http://www.yomiuri.co.jp/).

[4] Monthly reports written for a Japanese company by a Japanese professor living in the U.S.A.

[5] Only headings and intentional breaks, such as symbol lines inserted to separate a prologue or epilogue from a main body, are used as correct boundaries. As a result, the precision rates of using smaller window widths tend to degrade because of insufficient amounts of correct data.

more, for two relatively large series of newspaper columns, the major boundaries were detected properly. That is, using larger window widths, those boundaries were selectively detected that separate groups of columns by their subtopics. For example, the starting point of a set of three consecutive columns identically entitled "The Great Cultural Revolution" in the "Chinese Revolution" series was detected using 1,280 word width window, as well as those of other three sets of consecutive columns entitled identically. Thus, the proposed algorithm is expected to be effective for arbitrarily selecting the size of textual units corresponding to different grading topics.

However, there are problems about how to determine a boundary point in the range defined by a TBCS. Although the previously published algorithm (Nakao, 1999) determines a boundary point with minimal points of cohesion scores for the smallest window width, the accuracy degrades substantially (see Table 3). The boundary sentence identification algorithm given below is a solution to this problem.

3.2 Boundary Sentence Identification

In the second stage, from sentences in a TBCS, the algorithm identifies a *boundary sentence*, indicating where a topic corresponding to a textual unit probably starts, and selects a *lead sentence* that probably indicates the contents of subsequent parts in the same textual unit. Figure 3 shows the algorithm in detail.

3.2.1 Forward/Backward Relevance Calculation

In steps 2 and 3, boundaries are identified and lead sentences are selected based on two kinds of relevance scores for a sentence: *forward relevance* indicating the sentence relevance to the textual unit immediately after the sentence, and *backward relevance* indicating the sentence relevance to the textual unit immediately before the sentence. The difference between the forward and the backward relevance is referred to as *relative forward relevance*.

1. Assign the target layer as the bottom layer of the thematic hierarchy: $i \leftarrow i_{max}$.
2. For each TBCS in the target layer, $B(i)[j]$, do he following:
   (a) If $i \leftarrow i_{max}$, then select and identify all sentences in $B(i)[j].range$ as *Boundary Sentence Candicates (B.S.C.)*; otherwise, select and identify the sentences in $B(i)[j].range$ located before or identical to the boundary sentence of $B(i+1)$ as *B.S.C.*
   (b) From the *B.S.C.*, identify a sentence as a *Boundary Sentence (B.S.)*, whose relative forward relevance is greater than 0 and has the most increment from that of the previous sentence.
   (c) Among the sentences in the *B.S.C.* located after or identical to the B.S., select the sentence that has the greatest forward relevance as a *Lead Sentence (L.S.)*.
3. If $i > 1$, then $i \leftarrow i-1$, and repeat from step 2.

Figure 3: Boundary Sentence Identification Algorithm

Forward or backward relevance is calculated using the formula below, where every textual unit is partitioned at the equilibrium points of two adjacent TBCSs in the target layer, the equilibrium point of each TBCS is initially set by the thematic hierarchy detection algorithm, and the point is replaced by the location of the boundary sentence after the boundary sentence is identified (i.e., step 2b is completed).

$$r_{S,u} = \frac{1}{|S|} \sum_{t \in S} \frac{tf_{t,u}}{|u|} \times \log\left(\frac{|D|}{df_t}\right))$$

$\|S\|$	total number of terms in sentence S
$\|u\|$	total number of terms in textual unit u
$tf_{t,u}$	frequency of term t in textual unit u
$\|D\|$	total number of fixed-width (80 words) blocks in the source text
df_t	total number of fixed-width blocks where term t appears

The use of this formula was proposed as an effective and simple measure for term importance estimation (Nakao, 1998)[6]. It is a

[6] An experiment reported in (Nakao, 1998) indi-

Table 2: Example of Boundary Sentence Identification

Location	Relevance			Sentence [partially presented] (translation)
	Backward	Forward	Relative	
O.R. 11122	0	0.017	0.017	[吉村他, 86] ([Yoshimura et. al])
11124	0.021	0.004	-0.017	吉村賢治…: "…の自動抽出システム", …, pp.33-40, 1986 (Yoshimura, Kenji ... : Automatic Extraction System of ...)
B.S. 11146	0	0.016	0.016	4.4. 検索エンジン (Search Engine)
L.S. 11148	0.005	0.022	0.017	ここでは…知的情報アクセスにおける…ついて報告する。 (This section reports on ... of intelligent information access.)
11170	0.010	0.016	0.006	以下の各節の報告に共通するテーマは、…である。 (The key issue of the reports in the following clauses is ...)

modified version of entropy, where information bit (log part of the formula) is calculated by reducing the effect of term repetitions in a short period. The modification was done to increase the scores for an important term higher, based on the reported observation that content bearing words tend to occur in clumps (Bookstein et al., 1998).

3.2.2 Example of Boundary Sentence Identification

Table 2 summarizes an example of boundary sentence identification of a TBCS located just before the 12,000th word in Figure 2. Every row in the table except the first row, which is marked with *O.R.*, shows a candidate sentence. The row marked *B.S.* shows a boundary sentence, which has positive relative forward relevance (0.016 in the fourth column of the row) and the greatest increment from the previous value (-0.017). The row marked *L.S.* shows a lead sentence, which has the greatest forward relevance (0.022 in the third column of the row) among all sentences after the boundary sentence.

3.2.3 Evaluation of Boundary Identification

Table 3 shows recall and precision rates of the boundary identification algorithm in the same format as Table 1. Compared with the results obtained using the previous version of the algorithm (Nakao, 1999), as shown in the *minimal cohesion* columns, the proposed algorithm identifies more accurate boundaries (the *boundary sentence* columns). In addition, boundary sentence identification was successful for 75% of the correct TBCSs, that is, TBCSs including correct boundaries[7] (see *unified TBCS* in Table 1). Thus, the proposed boundary sentence identification algorithm is judged to be effective.

Table 3 also summarizes a feature of the proposed algorithm that it tends to detect and identify headings as boundary sentences (the *heading rate* columns). For the part corresponding to larger textual units, which the proposed algorithm mainly used, the figures in the *overall* columns indicate that half of boundary sentences or more are identical to headings in the original text; and the figures in the *identification* columns indicate that the proposed algorithm identifies headings as boundary sentences for more than 80% of the case where TBCSs including headings.

3.3 Summary Construction

In the third and last stage, the algorithm outputs the boundary and lead sentences of TBCSs on a layer that probably corresponds to topics of appropriate grading. Based on the ratio of source text size to a given summary size, the algorithm chooses a layer that contains an appropriate number of TBCSs, and generates a summary with some breaks to indicate thematic changes.

For example, to generate a 1,000-character summary consisting of several parts of approximately 200 characters for each topic, a text decomposition consisting of five textual

cates that heading terms (i.e., terms appeared in headings) are effectively detected by scoring terms with the part of the formula in the summation operator.

[7] For the correct TBCSs, the average number of boundary sentence candidates is 4.4.

units is appropriate for summarization. Since the sample text used here was decomposed into five textual units on the $B(2)$ layer (see Figure 2), it outputs the boundary sentences and lead sentences of all TBCSs in $B(2)$.

4 Discussion

Figure 5 shows a one-page summary of a technical survey report, where (a) is a part of the summary automatically generated, and (b) is its translation. It corresponds to the part of the source text between $B(1)[1]$ and $B(1)[2]$ (in Figure 2). It is composed of three parts corresponding to $B(2)[1]$, $B(2)[2]$, and $B(3)[6]$. Each part consists of a boundary sentence, presented as a heading, followed by a lead sentence.

In comparison with the keyword-based summary shown in Figure 4, generated in the process described in Section 2, the one-page summary gives a good impression as being easy to understand. In fact, when we informally asked more than five colleagues to state their impression of these summaries, they agreed with this point. As described in Section 2, one of the reasons for the good impression should be the difference in coherence. The relationship among sentences in the keyword-based summary is not clear; conversely, the second sentence of the one-page summary introduces the outline of the clause, and it is closely related to the sentences that follow it. The fact that the one-page summary provides at least two sentences, including a heading, for each topic is also considered to make coherence strong.

As shown in Table 3, the proposed algorithm is expected to extract headings effectively. However, there is a problem that detected headings do not always correspond to topics of appropriate grading. For example, the second boundary sentence in the example is not appropriate because it is a heading of a subclause much smaller than the window width corresponding to $B(2)[2]$, and its previous sentence "4.3.2 Technical Trend of IR Techniques" is more appropriate one.

This example is also related to another limitation of the proposed algorithm. Since there is no outline description in the subsequent part of the heading of clause 4.3.2, the proposed algorithm could not generate a coherent extract if it had identified the heading as a boundary sentence.

It is a future issue to develop more elaborated algorithm for summarizing detected topics especially for the user who wants richer information than that can be provided in a extract consisting of two or three sentences.

5 Conclusion

This paper has proposed an algorithm for one-page summarization to help a user skim a long text. It has mainly described and reported the effectiveness of the boundary sentence identification part of the algorithm. It has also discussed the readability of one-page summaries. The effectiveness of structured summaries using the thematic hierarchy is an issue for future evaluation.

References

A. Bookstein, S. T. Klein, and T. Raita. 1998. Clumping properties of content-bearing words. *Journal of the American Society for Information Science*, 49(2):102–114.

Michael A.K. Haliday and Ruqaiya Hasan. 1976. *Cohesion in English*. Longman, London.

Marti A. Hearst. 1994. Multi-paragraph segmentation of expository text. In *Proc. of the 32nd Annual Meeting of Association for Computational Linguistics*, pages 9–16.

Yoshio Nakao. 1998. Automatic keyword extraction based on the topic structure of a text. IPSJ SIG Notes FI-50-1. (in Japanese).

Yoshio Nakao. 1999. Thematic hierarchy detection of a text using lexical cohesion. *Journal of the Association for Natural Language Processing*, 6(6):83–112. (in Japanese).

Gerard Salton, Amit Singhal, Chris Buckley, and Mandar Mitra. 1996. Automatic text decomposition using text segments and text themes. In *Proc. of Hypertext '96*, pages 53–65. the Association for Computing Machinery.

Yaakov Yaari. 1998. Texplore – exploring expository texts via hierarchical representation. In *Proc. of CVIF '98*, pages 25–31. Association for Computational Linguistics.

Table 3: Evaluation of Boundary Sentence Identification

Window width	Boundary #		Minimal cohesion				Boundary sentence				Heading rate			
	cor.	res.	Recall		Precision		Recall		Precision		Overall		Identification	
5120	1	2	0	(0.1)	0	(.05)	100	(0.1)	50	(.05)	100	(6.6)	100	(29)
2560	2	4	0	(0.2)	0	(0.1)	100	(0.2)	50	(.05)	100	(6.6)	100	(29)
1280	3	10	33	(0.5)	10	(0.2)	67	(0.5)	20	(0.2)	80	(6.6)	80	(30)
640	30	42	27	(1.0)	19	(0.7)	47	(1.0)	33	(0.7)	67	(6.3)	88	(34)
320	114	163	26	(1.8)	18	(1.3)	40	(1.8)	28	(1.3)	54	(5.0)	82	(31)
160	184	365	28	(3.5)	14	(1.8)	43	(3.5)	22	(1.8)	37	(4.8)	77	(28)
80	322	813	29	(7.8)	12	(3.1)	45	(7.8)	18	(3.1)	23	(4.8)	70	(26)
40	403	1681	37	(17)	9	(3.9)	46	(16)	11	(3.9)	12	(4.8)	58	(26)

The figures in parentheses are the baseline rates.

4.3 **ネットワーク上の検索サービス**

…また検索精度を高めるために、高頻度語は検索の対象としない、タイトルや見出しに含まれる語に重みをつける、などの工夫がなされている。

…また、検索サービスが収集したページ数が膨大になるにつれて、ヒット数も膨大になってきたため、すばやく必要な情報を探すために、よりわかりやすい自動抄録作成技術が必要となる。…

…tf・idf 方式とは、単語に分割された文章の各単語の重要度を、その単語が文書中に出現する頻度 tf と、その単語を含む文書が文書集合中に出現する頻度の逆数 idf の積によってその単語の重要さを数値化する手法である。

…[河合, 92] の研究 キーワードのカイ二乗値から各キーワードの分類に対する得点を計算する場合に、シソーラス辞書から得られる抽象的な意味を得点に加える手法である。…

†a part of a summary condensed to 1.3% of the source text

(a) Original

4.3 Internet Services

... They are also enhanced with some techniques, such as eliminating high frequency words, weighing a term in document titles and headings, etc., to achieve high precision. ...

... In addition, since the greatly increasing amount of pages provided by an Internet service causes a great increase of average hit number for a query, more effective automatic text summarization technique is required for helping a user to find out required information quickly. ...

... Tf·idf method weighs a term in a document with a product of the term frequency (tf) in a document and inverse document frequency (idf), i.e., inverse of the number of document that the term appears. ...

... [Kawai, 92] A document classification method calculates a score based on χ^2 values of not only keyword frequencies but also semantic frequencies corresponding to occurrences of abstracted semantic category in target divisions. ...

(b) Translation

Figure 4: Example of Keyword-based Summary (partially presented)

ネットワーク上の検索サービス [4.3 参照]

本節では、 WWW 上の検索サービスと電子出版及び電子図書館について、現在行われている各サービスの特徴、技術的なポイント、問題点等を調査すると同時に、関連する研究分野も調査し、将来どのようなサービスが望まれるか、また、そこに必要となる技術は何であるか、についてまとめる。…

キーワード抽出 [(1) 参照]

ネットワーク上の文書をアクセスする方法の 1 つとしてキーワード検索がある。…

分散検索 [(4) 参照]

情報を一ヶ所に集中登録するタイプの検索サービスでは、今後ますます肥大化・多様化していく WWW には対応しきれなくなることが予想される。…

†a part of a summary condensed to 1% of the source text

(a) Original

Internet Services [see 4.3]

This clause surveys internet services, electronic publishing, and digital libraries, reports on their features, technical points, and problems observed in their typical cases, and suggests the desired services in the future and the required technology for their realization based on the investigation of related research areas. ...

Keyword Extraction [see (1)]

Keyword-based IR is a popular access method for retrieving document on the networks. ...

Distributed IR Systems [see (4)]

In near future, it will be impossible for a single IR system storing all resources in a single database to handle the increasing number of large WWW text collections. ...

(b) Translation

Figure 5: Example of One-page Summary (partially presented)