May 24, 1999
RT0319
Computer Science 16 pages

Research Report

Customer Claim Mining: Discovering knowledge in vast amounts of textual data

Tetsuya Nasukawa, Masayuki Morohashi, Tohru Nagano

IBM Research, Tokyo Research Laboratory
IBM Japan, Ltd.
1623-14 Shimotsuruma, Yamato
Kanagawa 242-8502, Japan

Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

Customer Claim Mining:
Discovering knowledge in vast amounts of textual data

Tetsuya Nasukawa Masayuki Morohashi Tohru Nagano
IBM Research, Tokyo Research Laboratory
1623-14 Shimotsuruma, Yamato, Kanagawa 242-8502, Japan
(nasukawa, moro, tohru3)@jp.ibm.com

Abstract

This paper illustrates our results and methodology to discover useful knowledge in a collection of over 400,000 reports by call takers on customers' calls received in a PC Help Center. Since each report consists of free text describing the actual dialog with the customer as well as formatted data such as the date and product name, it is very hard to analyze such reports manually.

By integrating natural language processing technology, to extract concepts with their categories and customers' intentions, mining technology, to apply statistical analysis to the concepts, and visualization technology, to provide an overview of the contents of the data collection and to allow interactive analysis of the results, a user can obtain various information such as trends in customers' claims, and the causes of those trends.

A feasibility study in which the reports were examined by human analysts revealed that the information obtained through this automated analysis is effective in enabling call centers to reduce the number of calls, detect product failures in their early stages, and evaluate the skills of call takers for the purpose of allocation and education.

1. Introduction

A huge collection of important documents is naturally a valuable knowledge resource. However, because of the limitations of human document-handling capability, it is difficult to make good use of such documents.

For example, recent advances in call centers' systems make it easier for call takers to write reports on calls from various customers. The reports are considered valuable resources for obtaining various information on products and services as well as customers' behaviors and opinions.

In many cases, a call taker's report consists of a free-format text describing the

actual dialog with the customer and a number of coded data such as the date of the call, the names of products involved, and the type of call. Although the text part is usually considered the most important part of the report, it is often analyzed in only a small proportion of the total number of reports, of which thousands may be stored in the call center every day.

To support such analysis of documents, text clustering/classification technologies such as (Cohen, 1998; Zamir, 1997) may be used to organize the reports, as well as information retrieval technology with relevant ranking such as (Salton & McGill, 1983) in order to focus on documents with specific topics. However, a great deal of effort is still required for human analysts to read through the documents even though the numbers of documents can be reduced. Since the unit of output of these technologies is basically a document rather than the contents or the topic of a document, even when each document contains several topics, the output documents have to be read to be analyzed.

Thus, different types of technologies are needed to extract valuable information from the contents of documents, and they should distinguish various concepts within a document instead of considering one text as a single object.

In this paper, we explore technology for discovering knowledge in vast amounts of documents on customer claims that may directly lead to some various desirable results such as fewer calls and faster detection of problems. After overviewing the related work and summarizing the requirements and our approach to the extraction of knowledge from customers' claims in the next section, we illustrate actual procedures for customer claim mining in section 3, and demonstrate the results in section 4. Finally, section 5 presents our conclusions.

2. Mining the contents of customer claims

As technologies for handling the contents of individual documents, Feldman and Dagan (1995) presented a framework for finding interesting patterns in the distributions of concepts in documents, and Feldman et al. (1997) introduced a method for visualizing

associations of concepts in a certain context. Lent et al. (1997) presented a technique for finding trends in the use of words and phrases in documents. However, the outputs of these previous methods are limited by the shallow conceptual level (unit of analysis). A set of character strings (words and phrases) appearing in the textual data has been used as the basis for a concept in previous methods, whereas the same expression (set of character strings) may indicate various concepts, and various expressions may indicate the same concept. For example, the character string "Washington" may indicate a person or a place. Such ambiguity and synomity should be considered in order to improve the accuracy of the analysis. Moreover, manipulation of these character-based concepts does not allow handling of sentence-level messages that convey the intentions of documents. Since such messages with intentions are an essential factor in the analysis of customer claims, we developed a framework for extracting concepts that consist of predicate argument pairs in association with information on modality, which often indicates intention. Polarity, which may be either negative or positive, is also an important factor in analyzing intentions (Hatzivassiloglou and McKeown, 1997). For example, from the sentence, "*I couldn't send files.*" a predicate argument pair that consists of the canonical forms of "*send*" and "*files*" (namely [*send*, *file*] in our framework) along with the modality of "*can*" and "*not*" is extracted. Since this information on modality indicates a problem, "a problem with sending a file" is identified from this sentence. As a result of introducing this framework, we verified an improvement in the power to express contents, and incorporation of this concept actually improved the accuracy of document categorization.

Another issue as regards handling concepts in documents is the treatment of diversity in concepts. To extract meaningful knowledge from various statistical analyses using a huge variety of concepts, we need to set some criteria for analysis. We found it highly beneficial to categorize each concept according to its features, so that we can compare the features of each concept in the same category. This allows us to apply techniques such as that of Suzuki (1997) in order to discover singular facts. Furthermore, these categories provide useful viewpoints for analyzing the contents of documents.

Thus, in our approach to customer claim mining, we focus on deep analysis of original documents in order to extract well-defined concepts with proper categories in the first step, then apply mining functions so as to take advantage of the categorized concepts, and finally exhibit the results by means of an adaptive visualizer that enables users to analyze the contents of vast amounts of documents by changing their viewpoints interactively.

3. Claim-mining technologies

The claim-mining process has three major components: the feature extractor, the miner, and the visualizer (Information Outliner). Fig. 1 shows the process flow through these components, using a sample document.

3.1 Feature extractor

The first component, the feature extractor, is a process for cleansing data from text. It is common in information retrieval to make a table of document ids and their keyword set (more precisely, a set of nouns used in every document). However, such a table has a large number of features (= the number of nouns), which causes it to be very sparse. We therefore have to consider several points before applying it to the mining process. We took the following linguistic measures to reduce its sparseness:

- Canonization of word representations
  - Domain-independent canonization (e.g., thru -> through)
    This type of process is more important in Japanese than in English, because we have more variations in word spelling.
  - Canonization of numeric word representations (e.g., twelve -> 12)
  - Domain-dependent canonization

  Examples from the PC call center reports are:

  cu, cus -> customer;
  TP, T/P -> ThinkPad
- Word categorization

  This is the process that finds the category of each word, such as Windows -> S/W.
  Almost all categories are domain-dependent; S/W, H/W and technical terms are sample

categories of a PC call center application.

- Compound and noun phrase identification

  Identification of compounds and noun phrases is useful for finding their canonical representations and their categories (e.g., "Microsoft Word" -> MSWord -> S/W).

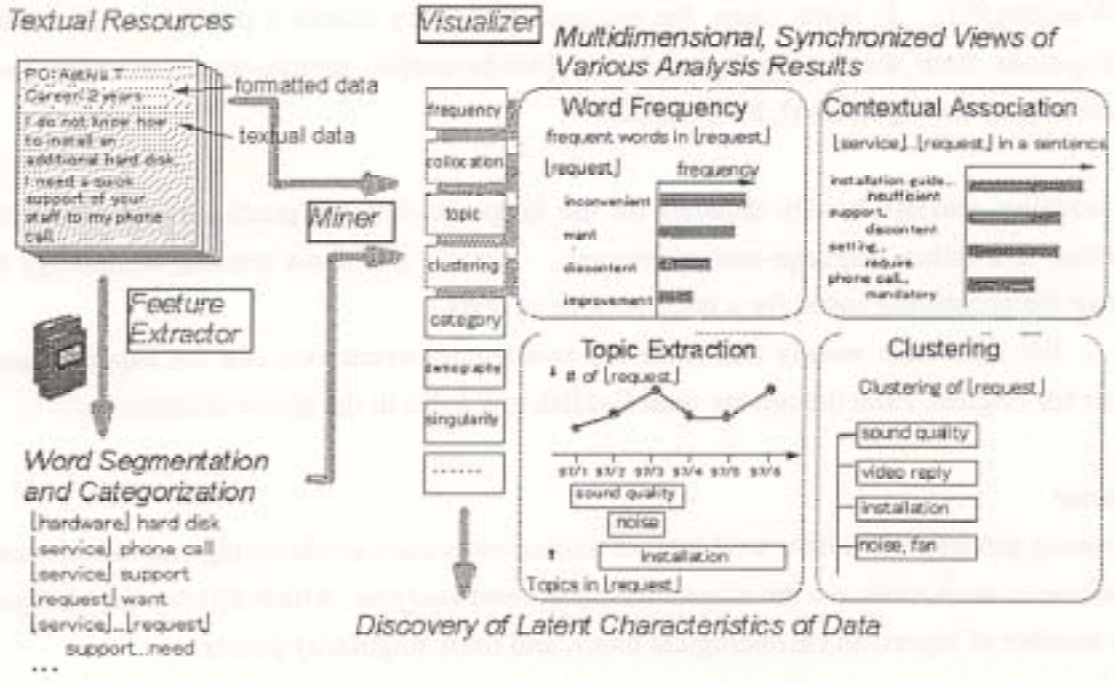

Fig. 1 Text mining process flow

Another consideration is how we can extract more condensed features than those of nouns. Especially in call center applications, it is important to identify customers' intentions. This cannot be done by simply extracting nouns and noun-based compounds. We perform **intent extraction** based on the following linguistic analysis:

- Modality identification

Morphological analysis makes it possible to determine the modality of a string, such as requirement, complaint, question, and so on (e.g., "would like to" -> requirement).

- Predicate-argument pair extraction

  Word combinations based on a sentence structure provide much information. A typical combination is a predicate (main verb) and its governing arguments (subject, object, etc.). For instance, [install, Windows98] represents the customer's

intention/description much more accurately than the single noun [Windows98] does. These types of combinations are collected not just by finding co-existing word pairs in a report, but by using a language parser. We developed a shallow parser, which finds every possible predicate-argument pair in a sentence at a reasonable speed. Negations and some modality information are included in those pairs (e.g., [cannot-install, Windows98]). In some cases, the antonym dictionary makes it possible to eliminate negations from word pairs by replacing [not-be-useful, mouse-operation] with [be-useless, mouse-operation], for example.

The modality analysis strictly depends on the language, but the predicate-argument pair extraction is a rather language-neutral process. We use a shallow parsing technology to improve the processing speed for a huge volume of texts.

We must note merely that there are two feature extractors, one for Japanese and another for English, even though we used English examples in the above discussion.

3.2 Miner

The mining process uses rather well-known mining tools such as clustering and association. In addition to such tools, we developed a simple trend analyzer, which follows the changes in the number of reports in chronological order, and finds singularity points.

3.3 Visualizer

Even though the first component (the feature extractor) pays much attention to the avoidance of sparseness, the results are still sparse in comparison with the well-structured numerical features. The interactive mining-visualization mechanism helps us to find useful facts from the mining database by drilling down the data set to be analyzed. For instance, after extraction of a special report set strongly related to software materials, the software-related features of this set are not sparse any more. Thus, combined operations of information retrieval and dynamic mining are the key to mining a text database. In other words, we need an **adaptive visualizer** to support what-if-type questions.

Information Outlining (Morohashi et al., 1995) is a system developed for this purpose. In the Information Outliner, categories (identified by the feature extractor) and report creation dates are used to extract a special report set. For instance, if you select reports that include

software related terms, then you have a chance of finding software-related associations such as strong support between [Product A] and [not-work some-function].

4. A call center application

The configuration of our prototype system named TAKMI (Text Analysis and Knowledge MIning) is shown in Fig. 2. As can be seen in the figure, all the extraction and mining are done on the server machine. The client system (the latest version of our system supports Web clients) generates graphical views from the text-based results.

4.1 Experiments

Details of the PC Help Center reports used in the experiments are as follows:

- Each report contains a dialog between a customer and a call-taker (in Japanese).
- Data size: 40,000 reports per month (400,000 reports in total)
- Contents:
  Formatted fields: report id, contact date, call-taker's id, call-type, machine-name, etc.
  Text fields: dialogs divided into parts headed Q and A (the feature extraction was done only for the Q (question) parts)
- Text-part statistics:
  Text length: 150 characters on average
  35 words per report excluding role words (e.g., at, in, on, ...)
  Low variety of word usage: 37,000 words appeared more than once.

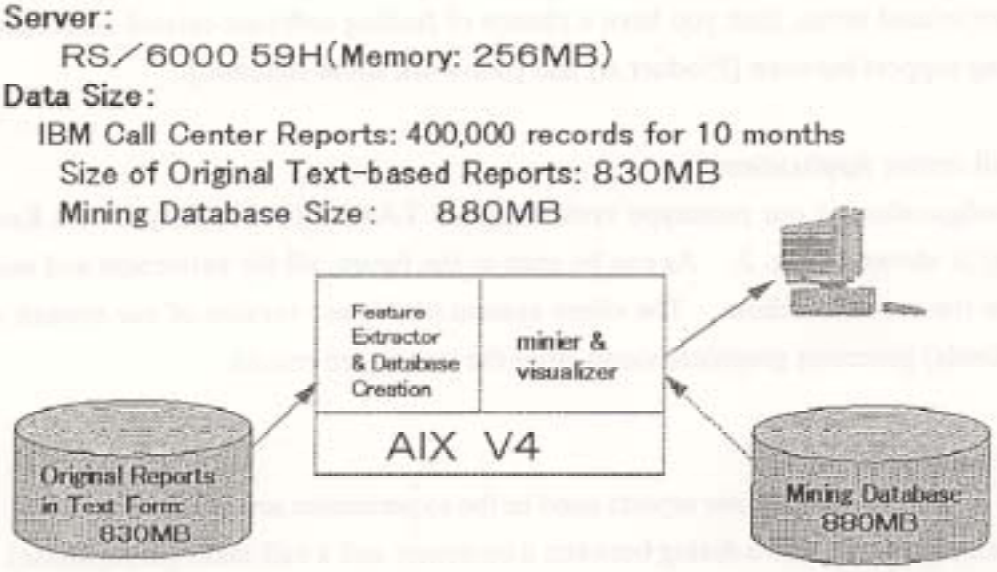

Fig. 2 Prototype system configuration

Before the analysis, we defined the following application-dependent categories:

- Noun-type categories: H/W, S/W, commands, company names, technical terms, etc.
- Predicate-type categories:

action (load, work, install, . . .), requirements (require, need, want, . . .), evaluations (be-good, be-bad, be-useful, . . .), etc.

- Predicate-argument pair categories:

[act ... H/W], [act ... S/W], [act ... technical-terms], [evaluate ... H/W], etc.

Then, we created a domain-specific dictionary according to the pre-defined categories. A candidate word list was prepared automatically by the system that analyzed the original texts. About 16,000 words were registered in the dictionary with their appropriate category codes; this work took two people 5 days.

4.2 Results

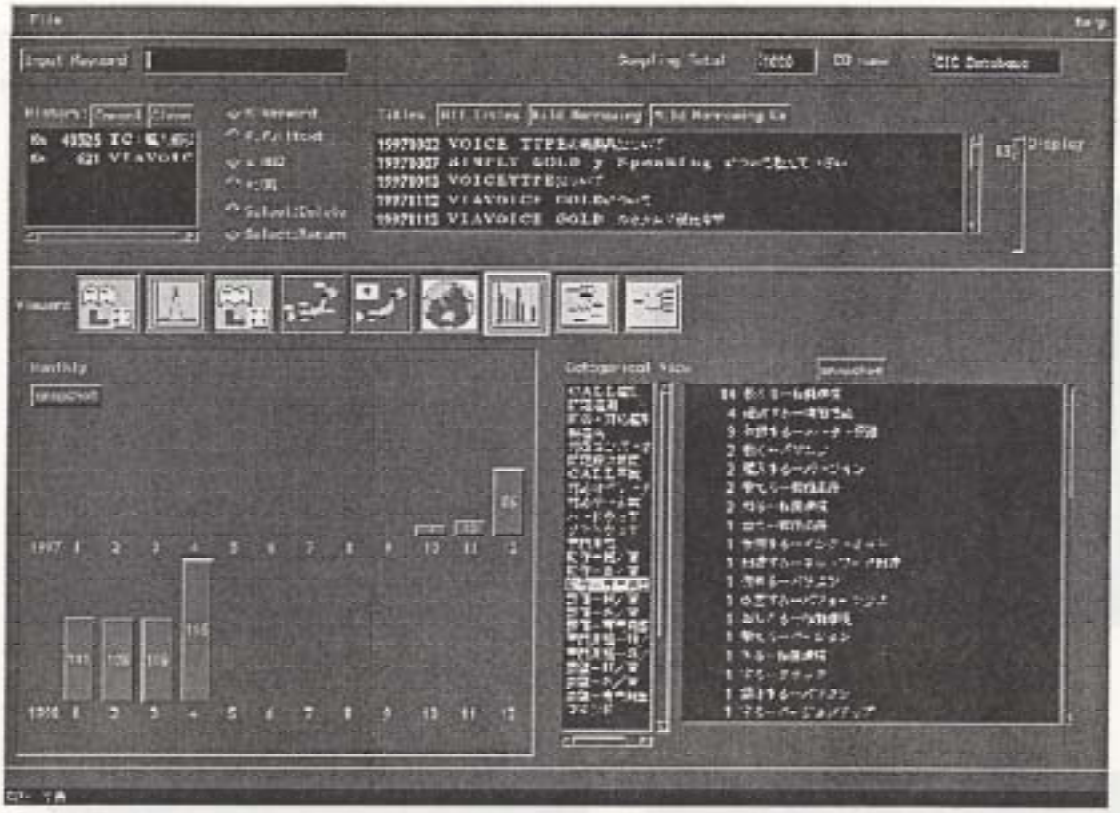

Fig. 3 Results of analysis (1)

Figures 3-6 show some results of analysis by our system, TAKMI. The first set of results, shown in Fig. 3, reveals trends in reports on "ViaVoice" (an IBM software product). The upper left window shows that the system found 621 reports that include the keyword "VIAVOICE" before the mining process. This is the log of the information retrieval function, which shows that the user entered "VIAVOICE" as a search keyword. The lower left window shows the chronological distributions of the 621 reports. The lower right window lists [act...technical-term] type word pairs that appeared frequently in the above 621 reports; [tell...working-environment], [confirm...working-environment], and [register...user-group] are the top three of the frequently appearing pairs.

A second example of the results found by our system is shown in Fig. 4. Two windows (both displayed on the screen simultaneously) describe a typical life cycle of "VoiceType," the predecessor of "ViaVoice." The major contents of the related reports shift from

"guidance on purchasing" (July-August) to "general guidance" (October-November), then to "requirements" (January) in the right window (Nomiyama, 1996). The system cannot find any noteworthy items in February-April.

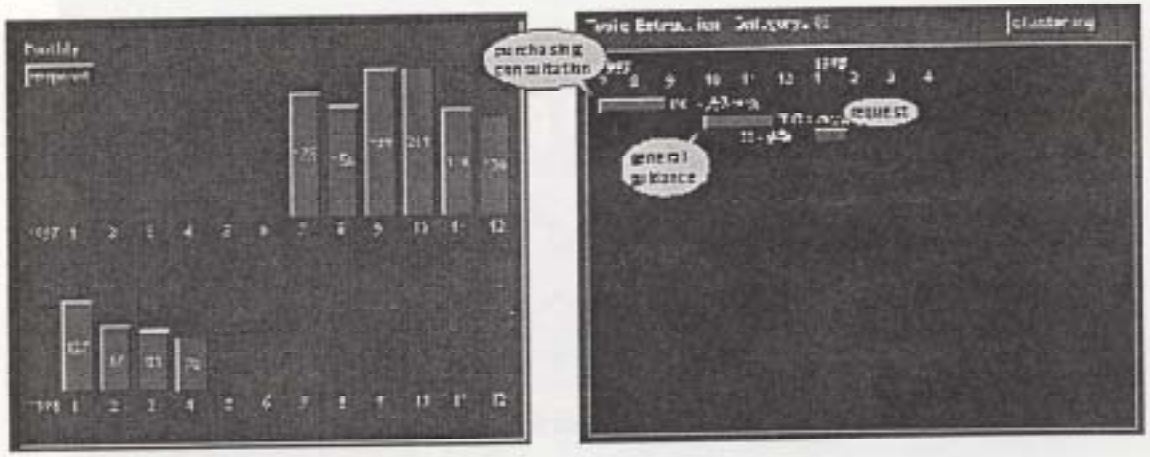

Fig. 4 Results of analysis (2)

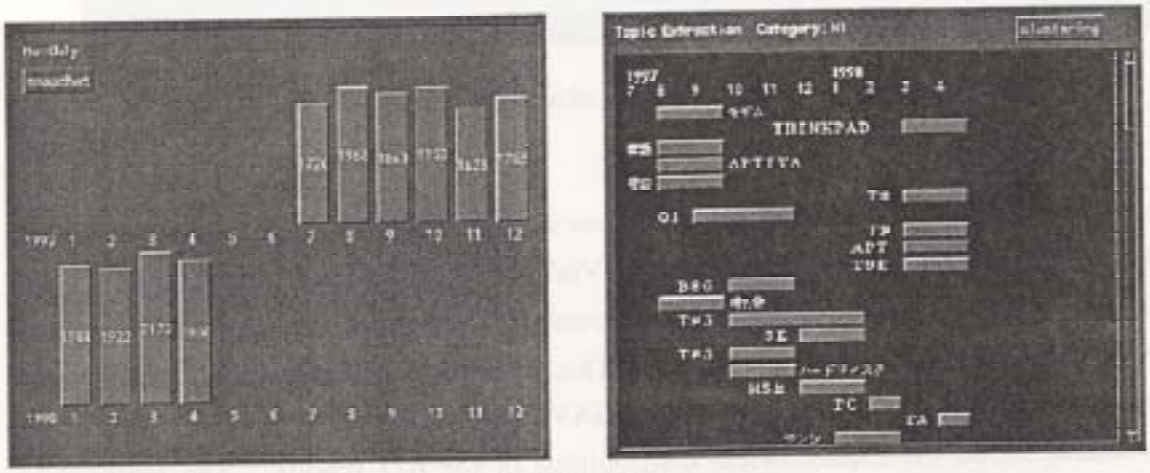

Fig. 5 Results of analysis (3)

Fig. 5 investigates another topic, "Internet." The total number of reports in each month did not change significantly (see the left window). However, the hardware items changed rather drastically. You can see the PC names Aptiva, B86, T95, or T85 (all of which are models in the Aptiva family) in the early stages, but these were replaced by the names of

new machines such as ThinkPad, T8, T9, or T9E (models in the ThinkPad family) in the later stages.

The last example shown in Fig. 6, reveals the characteristics of reports related to "Internet on ThinkPad." The distribution goes up, and right-hand window lists noteworthy software words such as "Internet connection wizard," "modem use," "dialer," and so on.

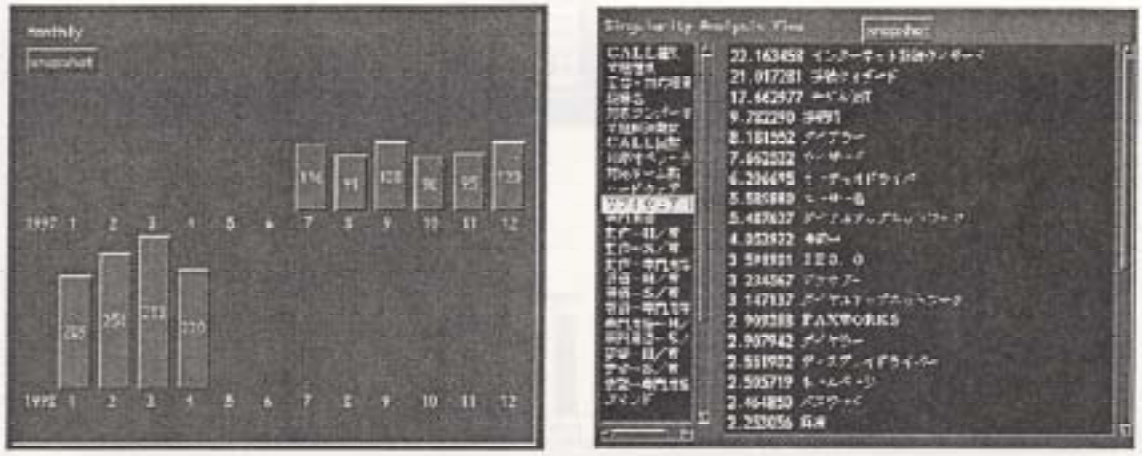

Fig. 6 Results of analysis (4)

4.3 Practical application

As a result of the experiments, we began to optimize the system for practical use in help centers. The latest TAKMI, with a new GUI, is shown in Fig. 7. In this figure, the monthly distribution of the number of cases (reports) handled at the IBM PC Help Center in Japan from January 1998 to February 1999 is shown in the bar graph in the lower part.

Fig. 8 shows an example of the output of a function in the latest system for showing trends in the number of concepts in the reports. It shows the trends of concepts categorized as software from the middle of June '98 to the beginning of July '98, and indicates that the most rapid growth during this period was for "Windows 98." From this interface, by clicking the "Windows 98" cell, a user can focus on reports that contain the concept "Windows 98," sometimes written as "Win 98."

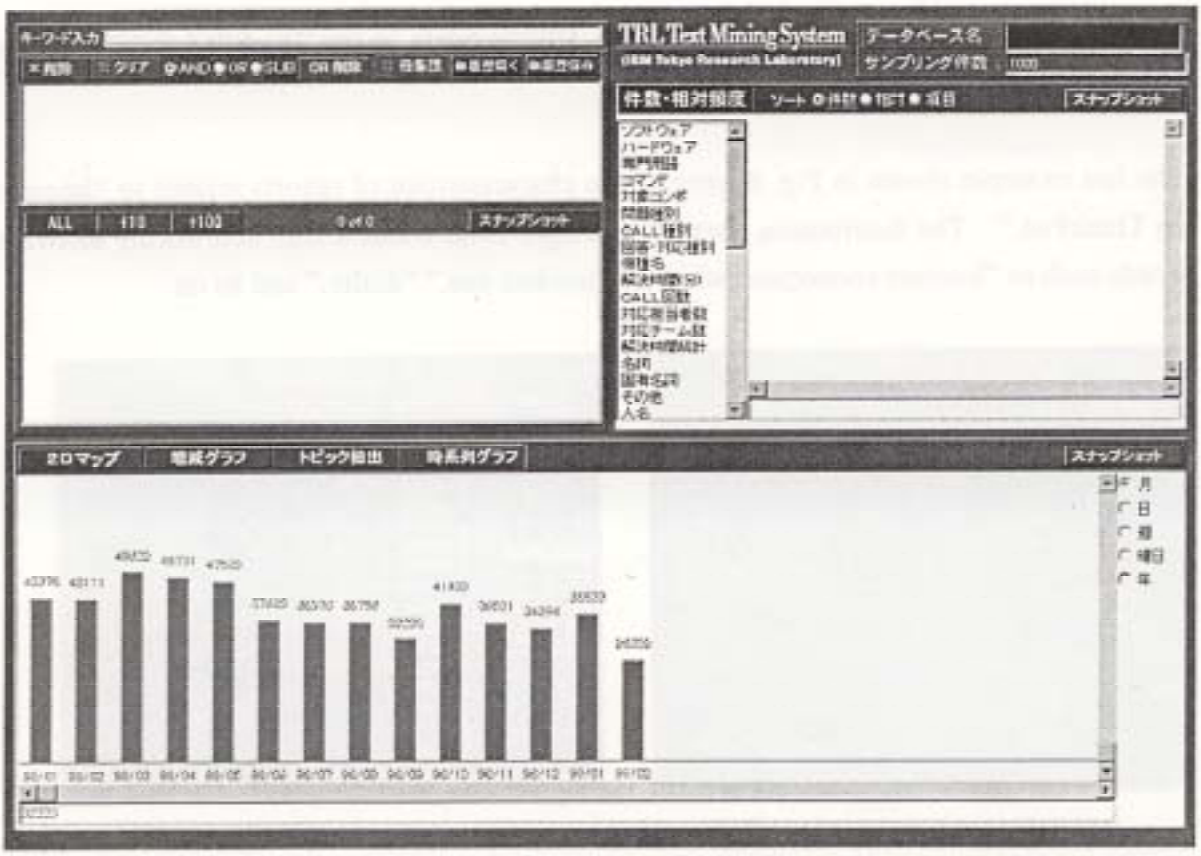

Fig. 7 New GUI for customer claim mining

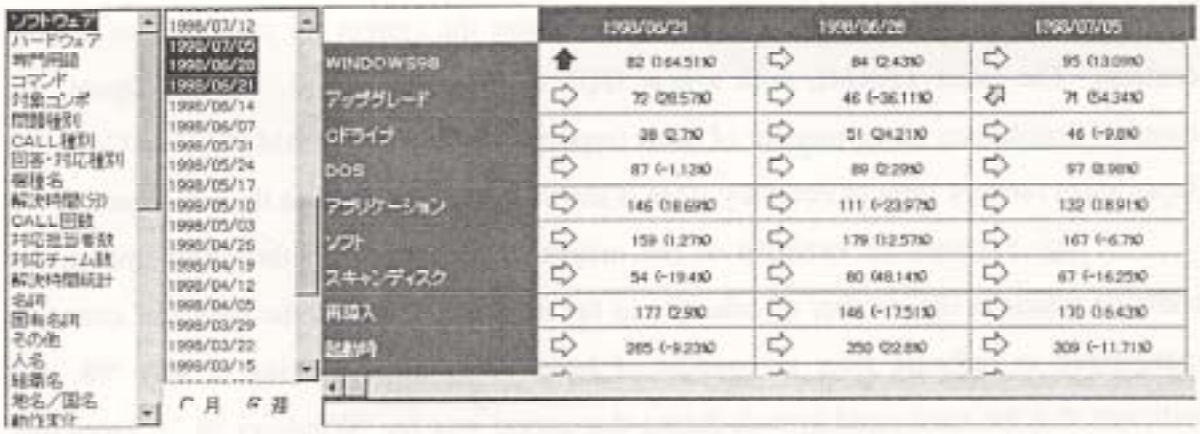

Fig. 8 Results of trend analysis for software

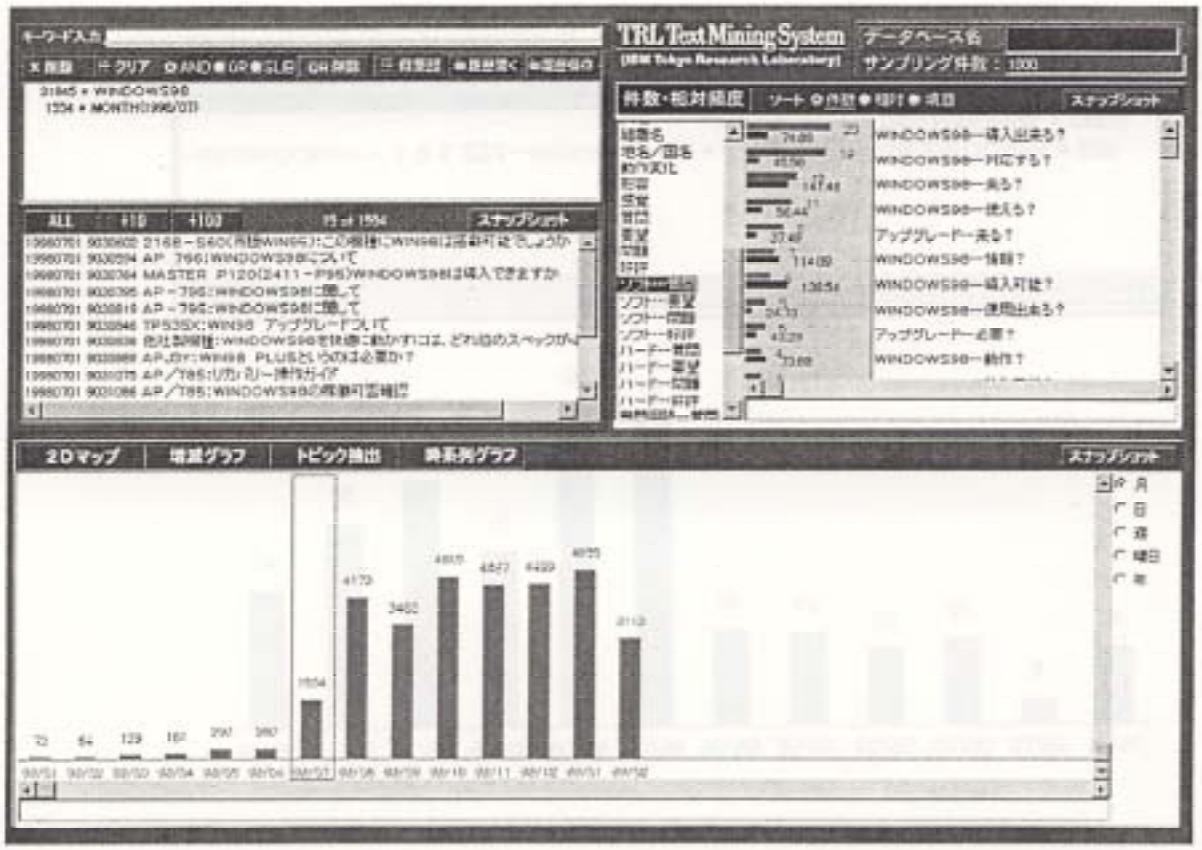

Fig. 9 Analysis of calls on Windows 98

Fig. 9 shows information about reports on "Windows 98." As shown in the monthly distribution below, the number of calls about "Windows 98" was increasing rapidly around July '98. The upper right corner of this figure shows a list of concepts that consists of questions on software. This list reveals that customers were asking if they could install or use Windows 98 (in their PC). As a result of this analysis, the PC Help Center can prepare answers and put the information on the WWW home page of IBM Japan in order to reduce the number of calls from customers, as well as the workload of call takers, by preparing quick answers.

In the mean time, the effects of such actions can be examined by using our system. Fig. 10 shows the monthly distribution of the number of reports on customers' calls in which customers asked if they could install or use "Windows 98" on their PC.

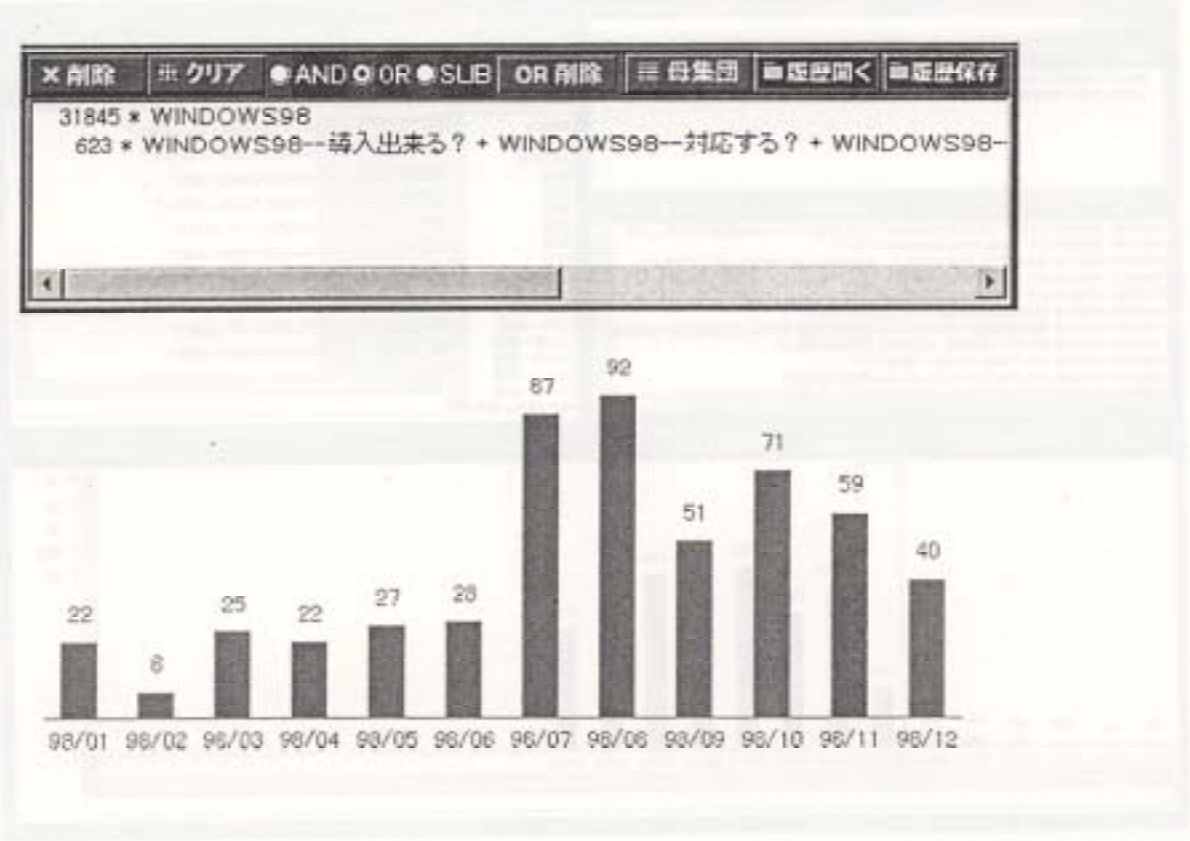

Fig. 10 Number of calls on the possibility of installing Windows 98

5. Conclusions

By applying our framework to real data in help centers, we verified that our technology was effective for detecting product defects in their early stages and studying customers' opinions on specific products, as well as finding frequently asked questions and evaluating the skills of call takers.

Our work is unique in applying deep but robust natural language processing technology to extract concepts with their categories and intentions in order to analyze documents in everyday language, and in visualizing the contents of the documents from various points of view with flexible interaction so that a user can fully understand the characteristics of the data set.

Since call takers are requested to type in a summary of each call in a short period

while they are communicating with customers, their reports often contain many unknown words based on personalized abbreviations and misspellings, and may contain too many words (sometimes over 100 words in English without any period); in addition, they are often ungrammatical. Our method was quite effective for extracting concepts from such documents, and the concepts were appropriate for discovering valuable knowledge.

While human analysts usually examine only a small portion of their vast amounts of textual data manually, this technology enables them to make use of all the data, and thus improves the quality of their analysis as well as reducing their workload.

This framework is not limited to call takers' reports. By applying this framework to the analysis of patent documents, we have verified that it could be used without difficulty to obtain patent portfolios such as a map of patented technologies, listed in accordance with the period in which a particular company frequently submitted patent applications on the technology and the frequency of their submissions, and also a map of organizations (mainly companies) that submitted patent applications related to the technology, listed in accordance with the period and frequency of their submissions.

References:

W. Cohen and H. Hirsh. 1998
Joins that Generalize: Text Classification Using WHIRL.
In *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining*, pp. 169-173.

O. Zamir, O. Etzioni, O. Madani, and R. Karp. 1997
Fast and Intuitive Clustering of Web Documents.
In *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, pp. 287-290.

R. Feldman and I. Dagan. 1995
Knowledge Discovery in Textual Databases (KDT).
In *Proceedings of the First International Conference on Knowledge Discovery and Data Mining*, pp. 112-117.

R. Feldman, W. Kloesgen, and A. Zilberstein. 1997
Visualization Techniques to Explore Data Mining Results for Document Collections.
In *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, pp. 16-23.

V. Hatzivassiloglou and K. McKeown, 1997,
Predicting the Semantic Orientation of Adjectives.
In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pp. 174-181.

B. Lent, R. Agrawal, and R. Srikant, 1997
Discovering Trends in Text Databases.
In *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, pp. 227-230.

M. Morohashi, K. Takeda, H. Nomiyama, and H. Maruyama, 1995
Information Outlining---Filling the Gap between Visualization and Navigation in Digital Libraries.
In *Proceedings of International Symposium on Digital Libraries*, pp. 151-158.

H. Nomiyama. 1996
Topic Analysis in Newspaper Articles,
Technical Report RT-0129, Tokyo Research Laboratory, IBM Research,

G. Salton and M. J. McGill, 1983
SMART and SIRE Experimental Retrieval Systems.
New York: McGraw-Hill

E. Suzuki, 1997
Autonomous Discovery of Reliable Exception Rules.
In *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, pp. 259-266.

O. Zamir, O. Etzioni, O. Madani, and R. Karp. 1997.
Fast and Intuitive Clustering of Web Documents.
In *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, pp. 287-290.