Rule Discovery from Textual Data based on Key Phrase Patterns

Shigeaki Sakurai
Corporate Research & Development Center,
Toshiba Corporation
1, Komukai Toshiba-cho Saiwai-ku,
Kawasaki, 212-8582, Japan
shigeaki.sakurai@toshiba.co.jp

Akihiro Suyama
Corporate Research & Development Center,
Toshiba Corporation
1, Komukai Toshiba-cho Saiwai-ku,
Kawasaki, 212-8582, Japan
akihiro.suyama@toshiba.co.jp

ABSTRACT

This paper proposes a new method for discovering rules from textual data. The method decomposes textual data into word sets by using lexical analysis, generates training examples from both key phrase relations extracted from the word sets by using key phrase patterns and text classes given by the user, and acquires key phrase relation rules from the examples by using a fuzzy inductive learning algorithm. The method is also able to deal with textual data that requires word segmentation, such as Japanese text. This paper reports on the application of the method to e-mail analysis tasks for a customer center. The e-mails are written in Japanese and have two analytical criteria: a product criterion and a contents criterion. We evaluate the acquired rules in each criterion.

Categories and Subject Descriptors

I.2.6 [**Artificial Intelligence**]: Learning—*Induction*

General Terms

Algorithms, Experimentation

Keywords

E-mail, fuzzy decision tree, key phrase pattern, text mining

1. INTRODUCTION

Although large amounts of textual data are stored on computers, the ability to process them is limited and it is not always possible to process them efficiently. This is the context in which text mining techniques have been studied with a view to using textual data efficiently [2] [4] [6].

One paper [2] proposed a method that discovers characteristic patterns by using hierarchical relations of words and then classifies textual data. This method processes textual data written in English, and cannot deal with textual data written in a language that requires word segmentation, such as Japanese. Another paper [6] proposed a method that visualizes relations among texts by generating a two-dimensional map. This method is unable to give a meaning of the map to the user. The user is responsible for noticing any meaning expressed in the map.

On the other hand, other paper [4] have proposed a method that classifies textual data by using lexical analysis and two kinds of background knowledge: a key concept dictionary and a concept relation dictionary. This method is able to process textual data written in a language that requires word segmentation. In this method, the key concept dictionary expresses important words and phrases concerning a target task. Also, the concept relation dictionary expresses relations between some key concepts and a text class. These dictionaries are created by a human expert through trial and error. This method is able to classify textual data more appropriately than classification methods based on words, because the method deals with the relations among words. Another paper [11] proposed a method that discovers concept relations from training examples. However, the method still has used a key concept dictionary created by a human expert. It is difficult to apply these methods to many target tasks, because the number of human experts is limited and an expert is unable to create a key concept dictionary for each target task.

In this paper, we propose a new method that discovers rules from textual data without using a key concept dictionary. This method generates attributes and their values by using key phrase patterns and acquires key phrase relation rules from training examples composed of the values and text classes. In this method, each key phrase pattern represents a relation among parts of speech and words. The key phrase patterns are usable for many target tasks and have high generality, because the patterns are created using linguistic knowledge and do not excessively depend on the target tasks. This paper demonstrates the effectiveness of this method by applying it to e-mails stored in the customer center. The e-mails are written in Japanese and have two analytical criteria: a product criterion and a contents criterion.

2. ACQUISITION OF KEY PHRASE RELATION RULES

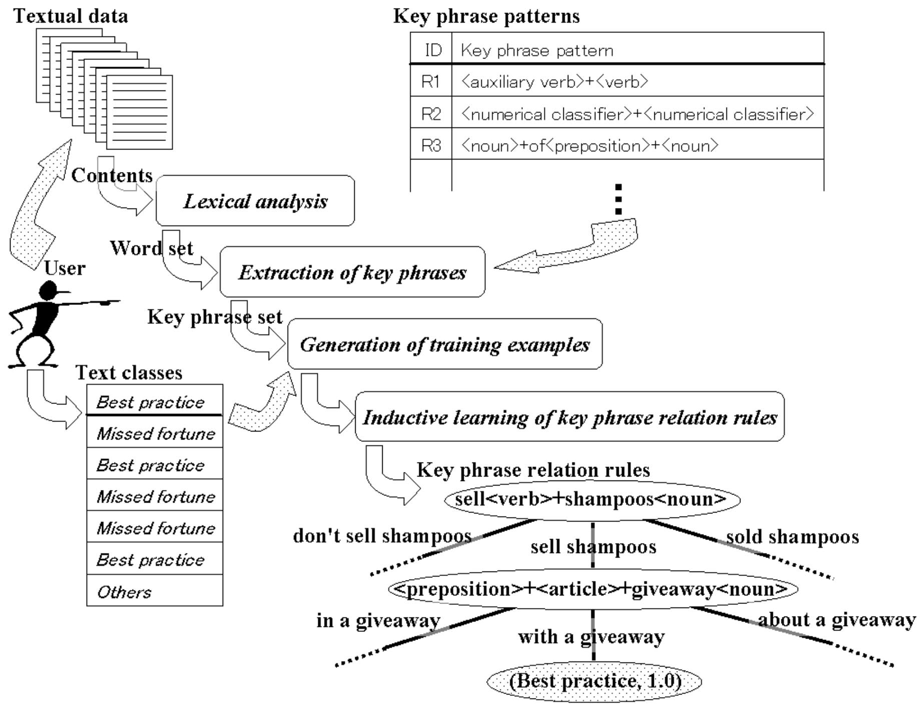

Figure 1: A learning flow for key phrase relation rules

2.1 Learning Flow

We propose a new learning method that makes discoveries from textual data by using lexical analysis and key phrase patterns. This method decomposes textual data into words by using lexical analysis [4], and extracts key phrases from the words by using key phrase patterns. Here, each key phrase is a combination of words. It also evaluates the key phrase and assigns the key phrase to a corresponding key phrase group. The key phrase group is a set which gathers key phrases including the same word. In subsection 2.2, we explain a method that extracts key phrases and a method that generates key phrase groups in detail. The user reads textual data and assigns a text class to each set of textual data. The text class shows the criterion which analyzes textual data from a viewpoint. This method generates a training example set from key phrase groups, key phrases, and text classes. In subsection 2.3, we explain a method that generates a training example set in detail. Lastly, this method applies the training example set to an inductive learning algorithm and acquires key phrase relation rules with a decision tree format. We explain the inductive learning algorithm and the format of the key phase relation rules in subsection 2.4. Figure 1 shows the learning flow.

We expect that this method acquires more understandable rules than rules based on words, because it is able to reflect relations among words. We can also create key phrase patterns more universally because a key phrase pattern describes a pattern composed of parts of speech and words, and the pattern does not excessively depend on target tasks. Therefore, we expect this method to be easily applied to many target tasks.

2.2 Extraction of Key Phrases

Each key phrase pattern is composed of parts of speech and words. Samples of key phrase patterns in English are shown at the upper-right side in Figure 1. We are able to apply the idea of the patterns to languages that require word segmentation, such as Japanese. But, for textual data written in such a language it is necessary to perform lexical analysis beforehand and the analyzed textual data has to be applied to key phrase patterns. In these samples, the R1 rule extracts a word set in which an "auxiliary verb" and "verb" occur together in textual data. The R3 rule also extracts a word set in which a "noun", "of<preposition>", and "noun" occur together in textual data. The extracted word set based on the patterns is regarded as a key phrase. For example, if the text, "He will sell shampoos at the supermarket.", is given, the R1 rule extracts a word set, "will sell", as a key phrase, and if the text, "He displays a mascot of wood at the entrance.", is given, the R3 rule extracts a word set, "mascot of wood", as a key phrase.

Textual data is characterized by sparse key phrases, because textual data contains many kinds of expressions. If a key phrase is regarded as an attribute, the attribute is apt to have little robustness, because textual data is characterized by having only a few parts of many key phrases. The attribute may not characterize textual data appropriately. It is necessary to decrease the ratio of an attribute that does not have a key phrase. Thus, we gather key phrases with the same word set and generate a key phrase group. We call the word set a center word set in the following. We regard a key phrase group as an attribute and regard the occurrence of key phrases included in the key phrase group as an attribute value. That is, an attribute has attribute values that are subsets of key phrases included in a key phrase group and a special value, "nothing". The special value is given in the case that key phrases corresponding to a key phrase group are not extracted. For example, key phrases, "will sell", "must sell", and "do not sell", are given, and a center word set, "sell", is given. A key phrase group, "sell", is regarded as an attribute. The attribute has attribute values, "will sell", "must sell", "do not sell", "{will sell, do not sell}", $\cdots$, "{will sell, must sell, do not sell}", and "nothing". The text, "He will sell shampoos at the supermarket.", has "will sell" as an attribute value of the attribute.

If we have to select specific word sets as center word sets, the sets are apt to depend on a specific target task. Thus, we point out important parts of speech, such as "verbs", "adjectives", and so on, as center word sets. Center word sets are extracted from textual data using the parts of speech. It is possible for the framework to select center word sets without depending on a specific target task.

2.3 Generation of Training Examples

Each text is characterized by attribute values. Also, each text has a text class given by a user. We generate a training example for each text. The training example is composed of attribute values and a class, which corresponds to a text class. For example, three key phrases, "will sell", "big supermarket", and "cheerful supermarket", are extracted from textual data and a text class "Best practice" is assigned to the textual data. Three key phrase groups that have "supermarket", "sell", and "shampoo" as center word sets are given. Then, the attribute "supermarket" has" { big supermarket, cheerful supermarket }" as an attribute value, the attribute "sell" has "will sell", the attribute "shampoo" has "nothing", and the class of the training example is "Best practice". Figure 2 shows a generated training example.

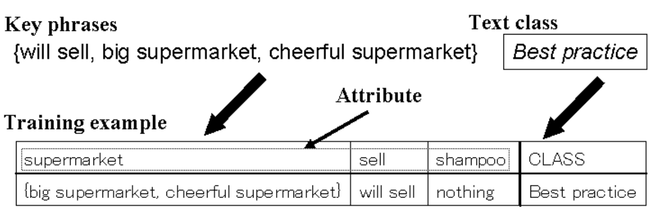

Figure 2: Generation of a training example

2.4 Inductive Learning of Key Phrase Relation Rules

In a generated training example, an attribute value is composed of a key phrase set. An inductive learning algorithm, which acquires key phrase relation rules from training examples, has to process an attribute value composed of a set of values. A fuzzy inductive algorithm, IDTF (Inductive Decision Tree with Fuzziness) [10], is able to process the attribute value by defining membership functions for basic attribute values. Here, each basic attribute value is an attribute value composed of an element. Each membership function is defined in the latter half of this subsection. The learning method, IDTF, is able to acquire key phrase relation rules with a fuzzy decision tree format from the training examples.

The fuzzy decision tree is composed of two kinds of nodes and branches connecting an upper node to a lower node. One kind of node has an attribute and is called a branch node. The other kind of node has classes with degrees of certainty and is called a leaf node. Each branch has a basic attribute value corresponding to an attribute of an upper node. A fuzzy decision tree is able to describe key phrase relation rules by assigning a key phrase group to a branch node, assigning text classes with degrees of certainty to a leaf node, and assigning a key phrase to a branch. The path connecting from the root node to a leaf node corresponds to a key phrase relation rule. Here, the degrees of certainty in the leaf node are calculated by summing degrees of certainty of training examples included in the terminal node for each class and normalizing the total values. But, each training example has 1.0 degree of certainty in the root node. The degree of certainty for the training example is updated by proportional distribution based on values of membership functions when the training example is transferred from a node to its lower node.

IDTF is an ID3-like inductive learning algorithm. IDTF makes a fuzzy decision tree grow by recursively decomposing a training example set into subsets with a selected attribute. The growth is repeated until either of two conditions is satisfied in nodes. One condition represents whether the maximum occupation ratio, defined by Formula (1), is more than a threshold. The other condition represents whether the decomposition ratio, defined by Formula (2), is less than a threshold. The default values of the thresholds are 100 and 0, respectively. The values lead to a fuzzy decision tree which classifies training examples most precisely. Also, IDTF prunes branches in the fuzzy decision tree until MDL (Minimum Description Length) [7] of the tree is not revised. IDTF creates the pruned fuzzy decision tree.

$$\frac{\max_i \{p_{i,j}\}}{\sum_i p_{i,j}} \cdot 100 \tag{1}$$

$$\frac{\sum_i p_{i,j}}{\sum_i p_{i,0}} \cdot 100 \tag{2}$$

Here, $p_{i,j}$ is sum of degree of certainty for training examples included in the j-th node with the i-th class. The 0-th node is the root node in the fuzzy decision tree.

IDTF is able to deal with ambiguity included in textual data because IDTF decomposes the training example set by using fuzzy sets defined by the attribute.

For example, parts of key phrase relation rules are shown at the lower-right side in Figure 1. A relation described in the center has a meaning such that if a key phrase "sell<verb> + shampoos<noun>" has a key phrase value "sell shampoos" and another key phrase "<preposition> + <article> + giveaway<noun>" has a key phrase value "with a giveaway"; then the text class is "Best practice" with a 1.0 degree of certainty.

Lastly, we explain the membership function defined in a basic attribute value. The membership function is defined by Formula (3).

$$\text{If} \quad l_{ikr} \in v_i, \quad \text{then} \quad grade_{ikr} = \frac{1}{|v_i|} + \frac{1-\alpha}{|L_{ik}|}$$

$$\text{If} \quad l_{ikr} \notin v_i, \quad \text{then} \quad grade_{ikr} = \frac{1-\alpha}{|L_{ik}|} \qquad (3)$$

$$\alpha = \frac{|v_i \cap L_{ik}|}{|v_i|}$$

Here, v_i is a subset of the basic attribute values included in the i-th attribute of an example, L_{ik} is a subset of the basic attribute values included in the i-th attribute corresponding to the k-th branch node, l_{ikr} is the r-th element of L_{ik}, and $|\cdot|$ is an operation that calculates the number of elements included in a set.

The formula has the following meaning. When a basic attribute value included in the i-th attribute of an example is equal to one of the basic attribute values corresponding to the k-th branch node, the formula gives a weight $\frac{1}{|v_i|}$ to a lower node connecting to the branch with the basic attribute value. When the basic attribute in the attribute is not equal to any basic attribute values corresponding to the branch node, the formula gives an equal weight $\frac{1-\alpha}{|L_{ik}|}$ to all lower nodes connecting to the branch node. Then, we note that L_{ik} is composed of basic attribute values included in the attributes of examples, which are given to the branch node in the learning phase. That is, α is equal to 1 in the learning phase, because $v_i \cap L_{ik}$ is equal to v_i. On the other hand, $v_i \cap L_{ik}$ is not always equal to v_i in the case that a class is inferred based on the acquired rules, because there are basic attribute values that occur only in the inference phase and IDTF does not generate lower nodes corresponding to the basic attribute values. In this case, it is impossible to evaluate an example in the attribute. Therefore, equal weight is given to all lower nodes in order to inspect all possibilities.

3. APPLICATION TO E-MAIL ANALYSIS TASKS

Recently, the decisive importance of delivering the highest possible level of customer satisfaction has been widely recognized. Consequently, customer centers dealing with requests and complaints from customers are assuming a more important role. On the other hand, the number of inquiries made to customer centers using e-mail is rapidly increasing. There are two reasons for this increase: e-mail is increasingly selected as the inquiry medium for companies, and it makes it easier for customers to make inquiries. Analysis of these e-mails leads to improvements in customer satisfaction, because they reveal the real voices of customers. However, the quantity of e-mail is too great for sufficient analysis. Customer centers need a method which facilitates the analysis of e-mails.

Most of the information included in e-mails is textual data. We are able to apply text mining techniques to the analysis of e-mails. The methods are discussed in various papers [1] [12] [14]. One paper [1] proposed a method which extracts examples by using a random sampling method. A human evaluates the examples. The method extracts data patterns based on the evaluation result. The paper reported on the application of the method to a task in which addresses are extracted from e-mails. Another paper [14] proposed a method that uses the number of words, the number of lines, and the frequency of important keywords as characteristic values, and identifies the person who wrote the e-mail. On the other hand, the third paper [12] proposed a method that classifies e-mails using lexical analysis and two kinds of background knowledge. The paper reported on the application of the method to e-mail analysis tasks for a customer center.

The method proposed in subsection 2.1 is applicable to the analysis of e-mails. It is possible for the method to discover rules that characterize a set of e-mails by using viewpoints given by operators of the customer center. The method uses the subjects and contents included in the e-mails. The textual data is analyzed by using the proposed method.

4. NUMERICAL EXPERIMENTS

4.1 A key phrase pattern dictionary

In generation of a key phrase pattern dictionary for this experiment, we enumerated rows of parts of speech that characterize textual data and created key phrase patterns based on the rows. On the other hand, we noticed that the lexical analysis engine used in this experiment is apt to decompose textual data into wrong lexicons and is apt to regard a proper noun as an unknown word in the case that a proper noun is included in the textual data. Also, we noticed that wrong characters are described with high probability and unknown words are created in the case that words in katakana are included in the textual data. Thus, we created key phrase patterns corresponding to unknown words. So, a key phrase pattern dictionary has 37 key phrase patterns such as <noun> + of <preposition> + <noun>, <adverb> + <verb>, and <unknown word> + <unknown word>.

4.2 Experimental Data

The customer center of Toshiba Corp. (our employer) collects e-mails sent by customers as well as Toshiba's own responses in a database. This experiment used 466 e-mails in the database. The e-mails were written in Japanese. The e-mails were analyzed with two criteria by operators of the customer center. One criterion is a product criterion that analyzes e-mails with five kinds of text classes, “Washing machine”, “Vacuum cleaner”, “Refrigerator”, “Microwave oven”, and “Others”. Here, each e-mail assigned to “Washing machine”, “Vacuum cleaner”, “Refrigerator”, and “Microwave oven” has a topic relating to the product. Each e-mail assigned to “Others” has a topic relating to a product other than the four products. In the case that an e-mail included some topics, the operator assigned the e-mail to a text class including the main topic. The other criterion is a contents criterion that analyzes e-mails with five kinds of text class, “Question”, “Request”, “Suggestion”, “Complaint”, and “Others”. Here, each e-mail assigned to “Question”, “Request”, “Suggestion”, and “Complaint” has a topic relating to four kinds of customer voice. Each e-mail assigned to “Others” has a topic relating to another kind of voice, such as “Thanks”, or “Comments”, and so on.

These e-mails included personal information such as names, addresses, and telephone numbers. Care must be exercised in dealing with personal information. Thus, we excluded personal information from the e-mails by replacing the information with special character strings and then used the e-mails without the information in the experiments. The information is not particularly important for the e-mail analysis tasks because we are interested in a generalized rule. Therefore, the lack of personal information does not present a problem.

4.3 Experimental Methods

We decomposed the given e-mails into 10 subsets and generated key phrase relation rules from the 9 subsets. We generated 10 key phrase relation rule sets by changing the selection of the subsets. Then, we used key phrases whose frequency in the e-mails is bigger than 2, because key phrases with a lower frequency are not important. There are 3,821 key phrases in the e-mails. We regarded verbs, adjectives, and so on, as center word sets and made key phrase groups from the key phrases including these words. We were able to make 1,948 key phrase groups. We acquired key phrase relation rules from the data using the proposed method. Here, IDTF uses default values and pruned fuzzy decision trees are evaluated.

Also, we performed experiments based on words included in the e-mails. We decomposed each e-mail into words by lexical analysis and calculated the $tfidf$ value [13] for each word. We extracted words whose $tfidf$ value was bigger than a threshold. This method regards a word as an attribute, whether a word appears or not as an attribute value, and acquires relations between words and text classes by using IDTF. We performed preparatory experiments to decide the threshold before the experiments based on words. We generated key phrase relation rule sets from a combination of the subsets by changing thresholds and selected a threshold corresponding to the rule set with the highest precision ratio. In the experiment, the threshold was 0.005 and 2,098 words were extracted from the e-mails.

On the other hand, we inferred text classes of e-mails included in the subset which were not used in the learning phase and verified the validity of the acquired rules. If we pay attention to the classification, it has been reported that SVM (Support Vector Machine) [15] provides a high performance for classification of textual data [5] [8]. Therefore, we investigated the classification performance using SVM. In the experiment, we used SVM software given by [3] and selected linear kernel and default parameters as its options. But, SVM regards a key phrase as an attribute and regards whether a key phrase appears or not as an attribute value, because SVM is unable to deal with a set value as an attribute value.

4.4 Experimental Results

Figure 3 and Figure 4 show a representative structure of key phrase rules acquired by 10 sub-experiments. Here, Figure 3 shows the case of a product criterion and Figure 4 shows the case of a contents criterion. In each figure, a plain square represents a branch node and a notched square represents a leaf node. An attribute is assigned to a branch node and a class is assigned to a leaf node. A certainty factor is not assigned to each class, because the certainty factor is subtly different in each sub-experiment. Lines between squares indicate branches and character strings on the lines are attribute values. An attribute is a key phrase group and an attribute value is a key phrase. However, we described English words with a similar meaning instead of a key phrase group and a key phrase because we could not correctly describe them in English. Both “TA” and “TW” correspond to identification numbers of products. A wavy line on the branch shows that some conditions may be included in a key phrase relation rule. That is, in Figure 4, a path connecting the “Energy conservation” root node to the “Request” leaf node shows that if “Energy conservation” is “Nothing”, “Catalog” is “Catalog”, and if other conditions are satisfied, then the class is “Request”.

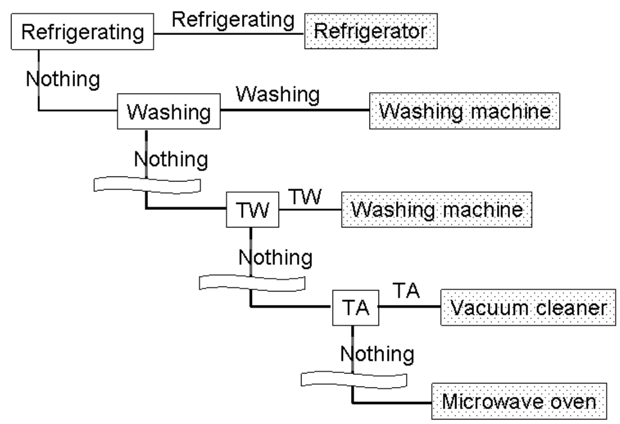

Figure 3: A part of the key phrase relation rules in a product criterion

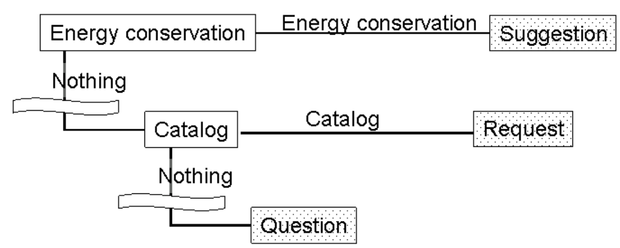

Figure 4: A part of the key phrase relation rules in a contents criterion

Table 1 shows the average size of fuzzy decision trees in 10 sub-experiments. Here, “Key phrase” shows the results in the case where key phrase relations were used and “Word” shows the results for the case based on words. Also, “Branch” shows the number of branch nodes, “Leaf” shows the number of leaf nodes, and “Total” shows the sum of branch nodes and leaf nodes. The number of leaf nodes is equal to the number of key phrase relation rules.

Table 1: Size of fuzzy decision trees

	Key phrase			Word		
	Branch	Leaf	Total	Branch	Leaf	Total
Product	33.9	35.1	69.0	35.7	36.7	72.4
Contents	52.6	61.2	113.8	61.7	62.7	124.4

Lastly, Table 2 shows the average precision ratios in 10 sub-experiments. Here, the precision ratios are calculated for the example subset that is not used in the learning phase of each sub-experiment. "Products" shows the average ratios for the product analysis task and "Contents" shows the average ratios for the contents analysis task. Also, "Key phrase" shows the results in the case where key phrase relations are used, "Word" shows the results in the case based on words, and "SVM" shows the results in the case based on SVM.

Table 2: Average precision ratios

	Key phrase	Word	SVM
Product	80.0	79.6	81.1
Contents	64.4	59.7	76.6

4.5 Discussions

Discovered rules:

In a product criterion, the proposed method discovers key phrase relation rules that are related with the basic function and identification number of products. That is, a refrigerator, a washing machine, and a vacuum cleaner correspond to key phrases relevant to refrigerating, washing, and sweeping, respectively. They also are corresponding to "GR", "TW", and "TA", which are part of the identification number of their products. In cases where the topics of some products are described, key phrases including parts and goods relevant to products are added to a conditional part in order to identify products that relate to the main topic. The rules correspond to the intuition of the operators. We believe that this method acquires valid rules.

In a contents criterion, this method discovers a key phrase relation rule between "Energy conservation" and "Suggestion" and a key phrase relation rule between "Catalog" and "Request". The operators felt that there were many e-mails requesting the sending of a product catalog. Discovery of the rule between "Catalog" and "Request" was anticipated to some degree. On the other hand, the operators did not become aware that there were e-mails which suggested energy conservation. The rule gave new knowledge to the operators.

In the case where key phrase relations were used, it was possible to deal with relevant words at the same time. In the case based on words, relevant words were apt to appear on remote nodes in a fuzzy decision tree because it deals with relevant words separately. Unimportant words were also apt to be included in the tree. Moreover, trees were larger in the latter case than in the former case, as shown in Table 1. Therefore, it was possible to easily understand rules in the case where key phrase relations were used.

Inference:

We evaluated an example subset which was not used in the learning phase with an acquired fuzzy decision tree. The case where the key phrase relations were used gave a higher precision ratio than the case based on the words, as shown in Table 2. We think that a key phrase is a more appropriate characteristic value than a word because it is possible for a key phrase to deal with relevant words at the same time. Also, it is possible for a key phrase group to decrease the sensitivity of the acquired fuzzy decision tree because the group deals with relevant key phrases at the same time and leads to a compact fuzzy decision tree. Therefore, in the case where key phrase relations were used, the precision ratios for the analysis tasks were improved. Especially in the case of the contents analysis task, combined lexicons were apt to connect with a text class, because in most Japanese sentences the last lexicon determines the kind of sentence, i.e., an interrogative sentence or a negative sentence, and the combination of the last lexicon and the front lexicons is more important. The contents analysis task is a more appropriate task for key phrase relations than the product analysis task. Therefore, in the contents analysis task, the case where the key phrase relations were used gave higher precision ratios than the case based on the word.

On the other hand, the case based on SVM gave a higher precision ratio than the case where the key phrase relations were used. SVM is able to generate complicated boundaries among text classes by mapping training examples into high dimensional space. SVM may be more appropriate than IDTF from the viewpoint of identification of text classes. However, SVM generates hyperplanes as a learning result. It is difficult to understand what the hyperplanes mean. It is not possible to discover new knowledge from the hyperplanes. For the viewpoint of discovery of new knowledge, IDTF is more appropriate.

If we pay attention to the revision of classification performance, it is necessary for SVM to be revised. In the experiments, SVM used only the information of key phrases and did not use the information of key phrase groups because it was not possible to deal with a set value as an attribute value. If it is possible for SVM to deal with the set value, SVM may give better results.

Easy creation of key phrase patterns:

Each key phrase pattern is composed of parts of speech and words. For example, the words are prepositions, conjunctions, and so on. The key phrase is defined based on linguistic knowledge and do not depend on a specific target task. If we create appropriate key phrase patterns, it is possible to apply the key phrase to many different texts. Therefore, it is possible to easily discover new knowledge from textual data in many tasks.

5. SUMMARY AND FUTURE WORK

This paper proposes a new method that discovers rules from textual data. This method uses key phrase patterns that do not excessively depend on target tasks. Therefore, it is easy to apply the method to many target tasks. This paper also shows numerical experimental results for the application of the method to two kinds of e-mail analysis task: a product analysis task and a contents analysis task. The results show that the acquired key phrase relation rules give more valid rules than the method based on the words.

In the future, we are planning to improve key phrase patterns and the method that acquires key phrase groups. In addition, we are planning to analyze questionnaire data by combining the proposed text mining techniques with data mining techniques.

6. REFERENCES

[1] R. Caruana and P. G. Hodor. High Precision Information Extraction. *Proc. of KDD 2000 Workshop on Text Mining*, September 2000.

[2] R. Feldman, I. Dagan, and H. Hirsh. Mining Text using Keyword Distributions. *J. of Intelligent Information Systems*, 10:281-300, 1998.

[3] C. -W. Hsu, C. -C. Chang, and C. -J. Lin. A Practical Guide to Support Vector Classification. http://www.csie.ntu.edu.tw/˜cjlin/libsvm/.

[4] Y. Ichimura, Y. Nakayama, M. Miyoshi, T. Akahane, T. Sekiguchi, and Y. Fujiwara. Text Mining System for Analysis of a Salesperson's Daily Reports. *Proc. of Pacific Association for Computational Linguistics 2001*, 127-135, September 2001.

[5] J. Joachims. Text Categorization with Support Vector Machines: Learning with Many Relevant Features. *Technical Report LS-8 Report 23, Computer Science Department, University of Dortmund, Dortmund, Germany*, 1997.

[6] K. Lagus, T. Honkela, S. Kaski, and T. Kohonen. Websom for Textual Data Mining. *J. of Artificial Intelligence Review*, 13(5/6):345-364, 1999.

[7] J. R. Quinlan and R. L. Rivest. Inferring Decision Trees Using the Minimum Description Length Principle. *Information and Computation*, 80:227-248, 1989.

[8] B. Raskutti, H. Ferrá, and A. Kowalczyk. Combining Clustering and Co-training to Enhance Text Classification using Unlabelled Data. *Proc. of 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 620-625, July 2002.

[9] J. Rennie. ifile: An Application of Machine Learning to E-mail Filtering. *Proc. of KDD 2000 Workshop on Text Mining*, September 2000.

[10] S. Sakurai, Y. Ichimura, A. Suyama, and R. Orihara. Acquisition of a Knowledge Dictionary for a Text Mining System using an Inductive Learning Method. *Proc. of IJCAI 2001 Workshop on Text Learning: Beyond Supervision*, 45-52, August 2001.

[11] S. Sakurai, Y. Ichimura, and A. Suyama. Acquisition of a Knowledge Dictionary from Training Examples including Multiple Values. *Proc. of 13th International Symposium, ISMIS 2002*, 103-113, June 2002.

[12] S. Sakurai, A. Suyama, and K. Fume. Acquisition of a Concepts Relation Dictionary for Classifying E-mails. *Proc. of the IASTED International Conference on Artificial Intelligence and Applications, AIA2003*, 13-19, February 2003.

[13] G. Salton and M. J. McGill. Introduction to Modern Information Retrieval. McGraw-Hill, Inc., 1983.

[14] O. De Vel. Mining E-mail Authorship. *Proc. of KDD 2000 Workshop on Text Mining*, September 2000.

[15] V. N. Vapnik. The Nature of Statistical Learning Theory. Springer, 1995.