July 27, 2000
RT0372
Computer Science 8 pages

Research Report

Natural Language Processing for Text Mining

Tohru Nagano, Tetsuya Nasukawa, Kohichi Takeda and Matthew Hurst

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

Natural Language Processing for Text Mining

Tohru Nagano, Tetsuya Nasukawa, Kohichi Takeda and Matthew Hurst
IBM Research, Tokyo Research Laboratory, IBM Japan
1623-14 Shimotsuruma, Yamato, Kanagawa, Japan
nagano@trl.ibm.co.jp

Abstract

Text mining aims to find hidden regularities/associations in textual data. Even though it might be conceived as a knowledge discovery technology quite opposite to information retrieval – a user-initiated information seeking technology, we will show that text mining can also be used to facilitate information retrieval by providing aggregate information as a content-based overview of the underlying textual database. We propose an ontology of significant terms and dependencies to capture the textual contents.

1 Introduction

Text mining is a technology, analogous to data mining (Agrawal 1993), to extract new information and find hidden regularities/associations from a large collection of textual data, while information retrieval (IR) is a technology to identify a collection of documents that satisfies a user's query. As Hearst said (Hearst 1999) "*The fact that an information retrieval system can return a document that contains the information a user requested implies that no new discovery is being made.*" Although these two technologies sound quite opposite, text mining can be used to facilitate IR for a particular collection of documents, since the collection can be characterized with aggregate yet informative summary data by using text mining. For example, text mining would provide an overview of a diverse set of textual data such as one million customer claims reported to a Help center in a year, then a user can retrieve a collection of specific customer claims such as virus issues.

It has been well known that a wide range of knowledge can be extracted from textual data, such as linguistic knowledge for Natural Language Processing (NLP)(Knight 1999) and that domain-specific lexical and semantic information may be stored in a database (Hahn 1997). Such linguistic/conceptual entities are used to capture trends and frequently-asked-questions (FAQs) from text. In this paper, we propose a notion of significant terms and dependencies (noun-noun and noun-verb relationship) as informative entities suitable to represent the contents of textual data.

As shown in (Mladenic 1999), most of the research to date on handling textual content in large textual databases has employed bag-of-word techniques. It is hard for a user to capture textual content only from a discrete set of keywords. Labeling a cluster with just nouns and noun phrases is sometimes misleading because a problematic sentence ("modem is broken") and its negation ("modem is not broken") are easily mixed in such a cluster. Significant terms and dependencies, however, can successfully identify these contents (i.e., modem-broken and modem-not-broken). We will show that from a collection of 46,000 customer claims (one month's data), our significant terms and dependencies can improve the content aggregations better than keywords.

2 Information Extraction for Text Mining

2.1 Significant Term

In order to analyze texts in a database statistically, some terms representative of the contents have to be extracted from the text using NLP. Then, the mining process can analyze a set of terms which is extracted from the text.

Research into techniques to deal with extracting terms can be divided into term weighting and automatic term recognition. In other word, a statistically-oriented approach and a linguistically-oriented approach (Kageura 1996). Salton provides vector space model (Salton 1983) that has been widely applied to information retrieval. This technique features terms which are significant for each vector/cluster, and it is useful for document clustering. On the other hands, automatic term recognition is a technique to recognize and extract distinctive terms and terminology from a text as a sequence of words. Several methods to derive the terms have been proposed such as a method using mutual information (Church 1990).

Text mining is a statistical and linguistic technology. In other words, text mining requires considering both frequency and significance. When a statistical process such as mining is carried out, if many trivial terms are included in the extracted terms, it may make the result of the mining seem trivial.

2.2 Customer Call Center Data

We focused on Customer Call Center documents as a first application for text mining, and we examined the relationship between the frequency and significance of terms. The experiments reported in this paper uses this PC call center's data. Each document contains : Transaction ID (automatically filled), Date (automatically filled), Formatted Field, Title of dialog and Dialogue.

The content of these fields are filled by the call-taker. The data in the formatted fields are chosen from a selection list by a call-taker. The selection list contains 10 ~ 30 choices. For instance, the field `Component` contains 30 choices such as "`Windows95`", "`memory`", "`hard disk`", etc. Call-takers summarize the dialogue between the customer and her/himself, and types it into the dialogue field. This text part, which is denoted as `Call` in Table 1, is divided into the customer's question and the call-taker's answer. Our study looks only at the question part of the data.

Our test collection contains 43,378 documents, and each document contains $\simeq$ 14 compound nouns. The number of different terms (simple nouns) is 31,609, and total occurrence of the terms is 1,119,039. The number of different compound nouns and simple nouns is 600,494. Therefore a compound noun has $\simeq$ 2 component words.

2.3 Significance for Text Mining

As a measure of significance, we use the entropy value of terms. The entropy value is used to measure the dispersion of the data set into given a range of the data set or given categories, (Resnik 1995) used the value to measure a semantic similarity between terms.

All documents have been categorized by call-takers in terms of a `Component`, a `Call-Type` and an `Answer-Type`. The `Component` field has over 30 category values, and is divided into software/hardware components such as "`Windows95`", "`Memory`", "`hard disk`". In the experiments reported below, the category `Component` is used. We assume that a meaningful term is a term that can be used for determing the category value assigned by the call-taker.

The entropy of terms w_i is defined as:

$$H(w_i) = \Sigma_{t=0}^{N} P(c_t \in w_i) \log_2 \frac{1}{P(c_t \in w_i)}$$

$P(w_i)$ is the probability of w_i which categorized into category c_t (i.e. "`Windows95`", "`memory`", "`Hard-disk`", and so on). N is a number of the categories.

And redundancy is given as:

$$r(w_i) = 1 - \frac{H(w_i)}{\log_2 r}$$

$(log_2 r$ gives the upper limit of $H(w_i))$

This value $H(w_i)$ indicates the diversity of term distribution. A lower value means more diverse term distribution. The value $r(w_i)$ is the normalized and reversed value of $H(w_i)$. If this value is high, then the term is significant term.

For example, if a term w_i is distributed into these three categories C = {"`Windows95`", "`Memory`", "`HardDisk`"}, and P("`scandisk`") = {0.26, 0.02, 0.72}, P("`failure`") = {0.41, 0.31, 0.38}, then r("`scandisk`") and r("`failure`") are calculated as 0.53 and 0.01. In this case, "`scandisk`" is more significant term.

3 Characteristic of Term

In order to investigate what kind of terms are significant, we examined the significance of terms between various set of terms. The text part of all documents is tokenized and part-of-speech tagged. Then a parser produces a dependency analysis. On the other hand, the precision of the parser is not as good as those for other NL applications which require precision such as machine translation, but it is sufficient for our statistical analysis.

Noun vs. Compound noun

A noun group consists of a sequence of nouns. If a noun phrase has N elements, the number of possible candidates of compound nouns is $N(N+1)/2$. For example, the noun phrase "`Microsoft Internet Explorer`" has three elements, possible candidates for compound noun are "`Microsoft`", "`Internet`", "`Explorer`", "`Microsoft Internet`", "`Internet Explorer`" and "`Microsoft Internet Explorer`". We call "`Microsoft Internet Explorer`" a long term, and "`Microsoft`", "`Internet`", "`Explorer`" short terms. If the noun phrase has only one element, long term and short term is same.

Figure 1-left shows the comparison of the frequency of terms (log scale) vs. $r(S)$ between long term and short term. A set of term S contains terms which have the same frequency. This value averaged by the value of terms which are included in the log-scaled frequency range.

Figure 1-right shows the coverage of all terms. In this case, the top 100 simple terms share approximately 40%. The terms with the same frequency have same significance for both simple terms and compound nouns.

Table 1: Sample Data

ID	03240210233
Date	10/21/1999
Component	Windows95
Call-Type	Software Problem
Answer-Type	Information needed
Title	Windows95 doesn't start up
Call	Q: Ethernet card changed. After replace driver, Windows95 doesn't start up. A: Could you use new Ethernet card driver? Once, shutdown windows ....

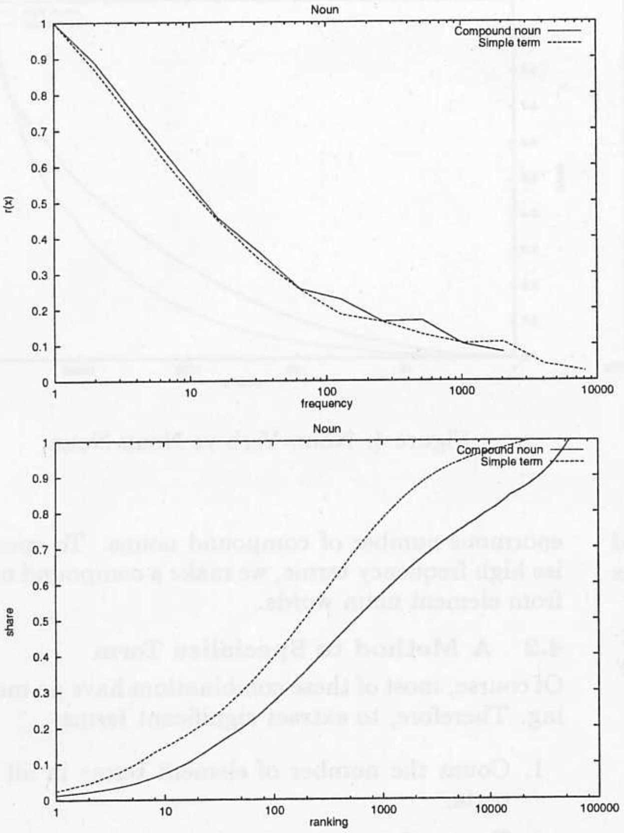

Figure 1: Noun vs. Compound noun

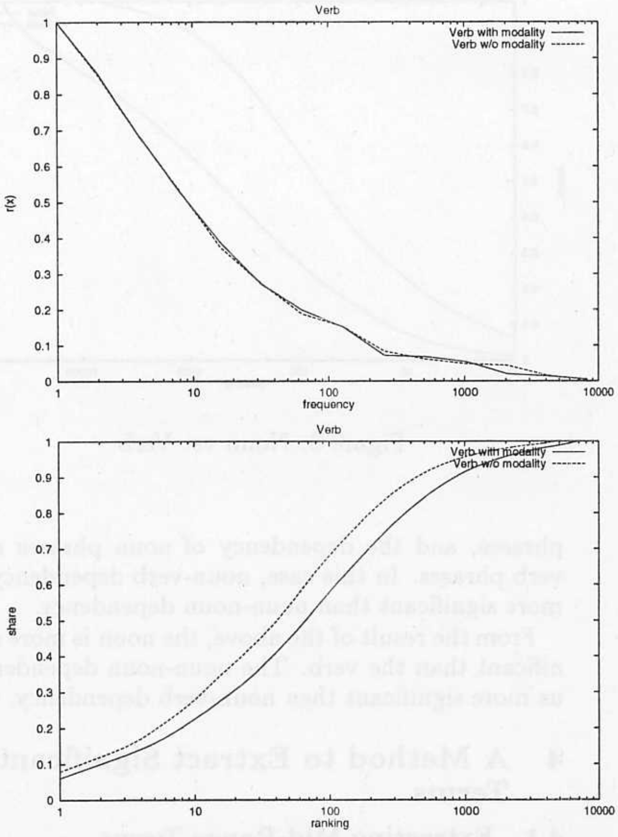

Figure 2: Verb (stem) vs. Verb (with modality)

Verb vs. Verb with modality

In Japanese, modality is included in the suffix of the verb, such as "-hosii (want)", "-tai (hope)", "-nai (negative)". This figure shows the effect by modality on the significance of terms. Figure 2-left shows the comparison between verb and verb with the modal component. In this case, the significance of terms which occur with the same frequency is not different.

Noun vs. Verb

The comparison between noun group and verb group is shown in Figure 1-left. In this case, the noun is more significant than the verb in all frequency ranges.

Noun-Verb vs. Noun-Noun

The dependencies between phrases are analyzed by the parser. Figure 1-left shows the comparison between the dependency of noun phrases and noun

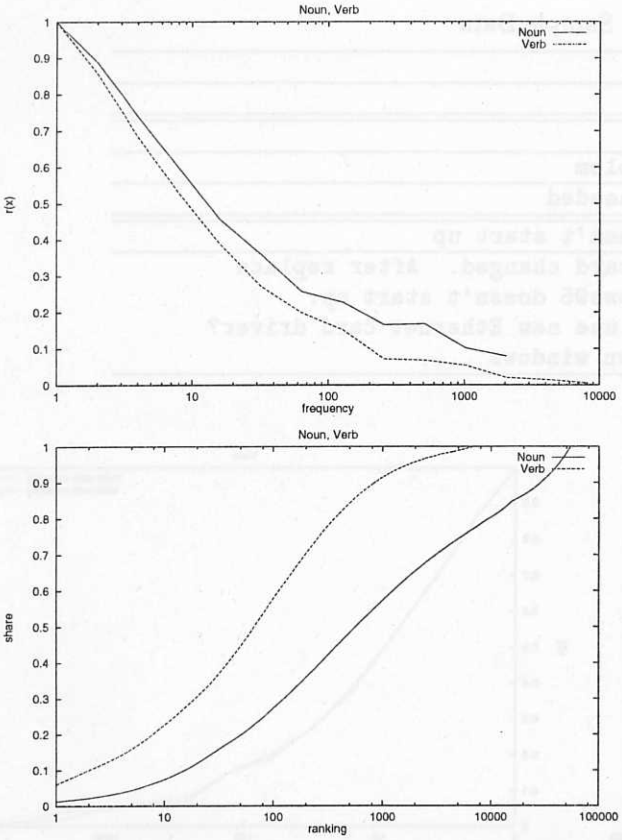

Figure 3: Noun vs. Verb

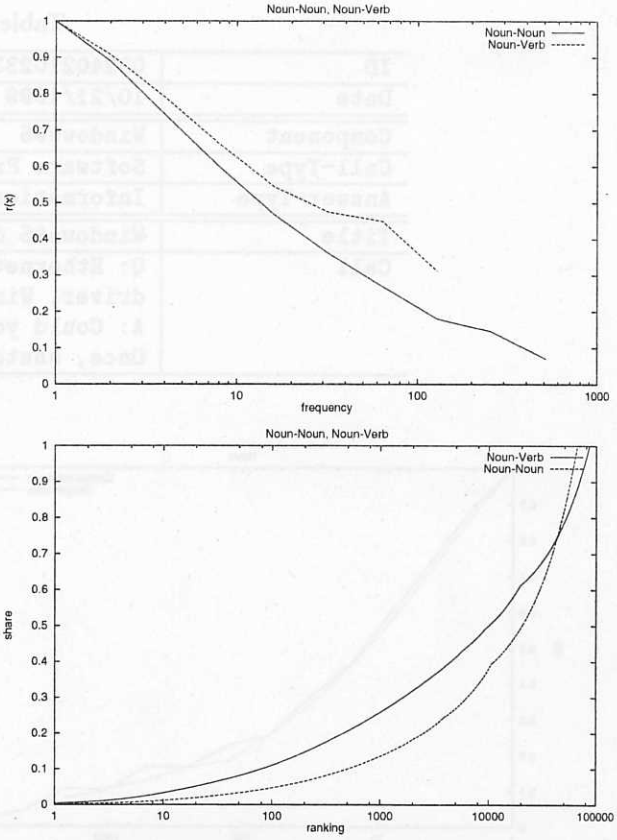

Figure 4: Noun-Verb vs Noun-Noun

phrases, and the dependency of noun phrases and verb phrases. In this case, noun-verb dependency is more significant than noun-noun dependency.

From the result of the above, the noun is more significant than the verb. The noun-noun dependency us more significant then noun-verb dependency.

4 A Method to Extract Significant Terms

4.1 Extracting Mid-Range Terms

Via the above results, we conclude that the significance of a term depends on its frequency. The terms in the mid-range frequency include more effective terms for text mining, therefore the mid-range frequency terms have both significance and reasonable frequency. Figure 5 shows the basic concept used to gather meaningful terms.

Significance and Frequency which text mining requires, are alternatives. In order to gather significant terms for text mining, high frequency terms should be specialized. Low frequency terms should be aggregated with synonyms, which may be found in thesauri. However, the thesauri may not contain low-frequency terms since there is usually an enormous number of compound nouns. To specialize high frequency terms, we make a compound noun from element noun words.

4.2 A Method to Specialize Term

Of course, most of these combinations have no meaning. Therefore, to extract significant terms:

1. Count the number of element terms in all the texts.
2. Extract frequent terms that may be defined by a threshold $T_{high-freq}$.
3. The consecutive terms exceeding threshold $T_{combination}$ are unified to one compound noun.
4. If the re-calculation of significance $r(S)$ of the unified term is larger than $r(S)$ of each element term, it is designated a compound noun.
5. The above processes are repeated for each frequent term.

In this method, the thresholds should be formulated and validated.

This method can use all terms in the text. The state-of-the-art methods to extract terminology

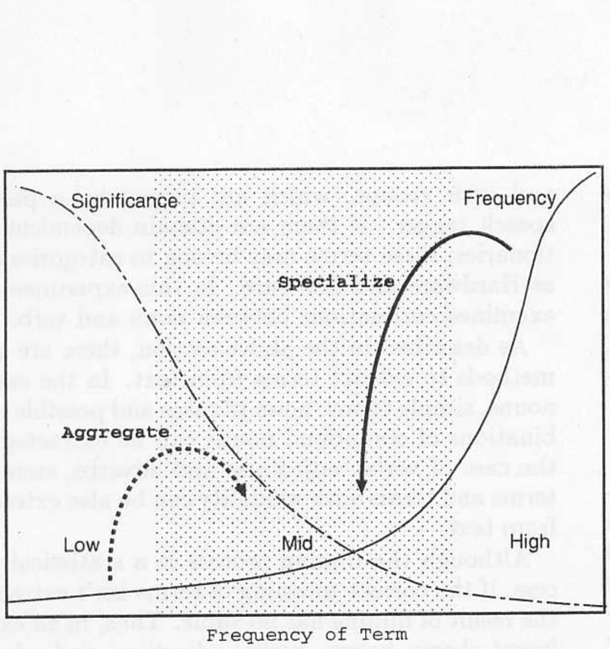

Figure 5: Concept of the Algorithm

which use term weighting are not for statistical analysis, so they are not mentioned.

5 Experiment and Evaluation

5.1 Natural Language Processing

NLP Overview

Text mining shows an overview of the entire data set. Thus the result of text mining has to be informative linguistically and significant statistically.

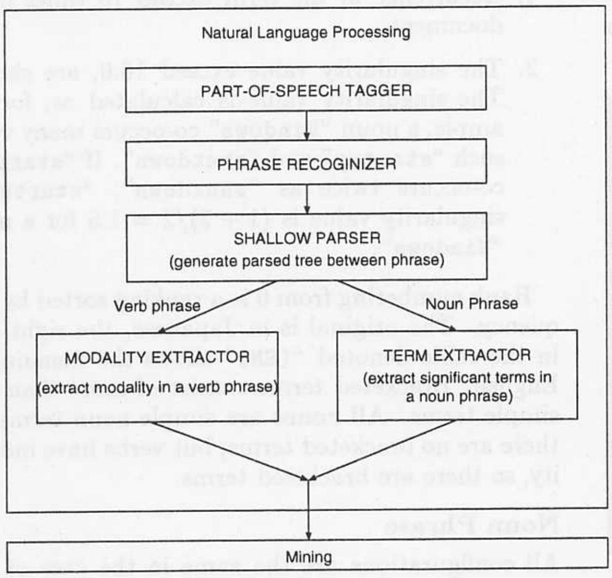
Natural Language Processing
PART-OF-SPEECH TAGGER
PHRASE RECOGINIZER
SHALLOW PARSER
(generate parsed tree between phrase)
Verb phrase
Noun Phrase
MODALITY EXTRACTOR
(extract modality in a verb phrase)
TERM EXTRACTOR
(extract significant terms in a noun phrase)
Mining

Figure 6: NLP Overview

Our text mining system consists of an NLP module and a mining module. The NLP part of the system has some functions to extract informative and significant terms from the document set. The system contains, dependency analysis, modality extraction, term extraction.

We used JMA (Japanese Morphological Analyzer) (Maruyama 1994) as a part-of-speech tagger which include a tokenizer. The precision of this parser is more than 0.99 according to their experiments (newspaper article). This parser also produces phrase boundaries. Then the sequence of terms delimited by sentence boundary information, is parsed and analyzed by the Modality Extractor and the Term Extractor, semantically and statistically.

Shallow Parser

Extracting dependencies for keyword expansion (Strzalkowski 1992) had been researched. They extracted head-modifier pairs using full parser and suffix trimmer, in order to measure the relative strength of connection between the words in syntactic pairs. We extract the dependencies, in order to get more informative relations.

In order to generate the dependencies between phrases, we developed a shallow parser. One of the reasons that we don't use a full parser but a shallow parser, is to generate dependency information quickly, since text mining deals with a number of sentences. Our test collections contains about 0.21 million sentences. Another reason is that the length of the sentences (the length of phrases in a sentence) in a dialogue such as those Call-Center logs is shorter than it is in formatted text such as a newspaper articles. One sentence contains about nine noun and verb phrases in average. Third, the dialogue of the Call Center is not written in correct syntax, since the main purpose of the call-taker is not to input the dialog, but to solve the customer's claims. Thus, it is better to generate all the dependencies that may be important.

JMA produce a sequence of terms with sentence boundaries. One sentence T has some phrases $p(i)$ $(0 \leq i \leq N_T)$, each phrase consist of a sequence of terms $t_i(j)(0 \leq j \leq N_{Ti})$. N_T is the length of sentence T, and N_{Ti} is the length of terms in a phrase pi. This parser is rule based parser. It has some rules to determine the dependencies. Each rule consist of a sequence of the combinations of part-of-speech. For example, a rule r is $r = \{(\texttt{prop}, \texttt{conj}_1), (\texttt{prop}, \texttt{punct})\}$ where prop is proper noun, conj is conjunction and punct is punctuation. A bracketed () combination of part-of-speech means $(contentword, functionword)$ in a sentence. Then a sentence $\texttt{T} = \{(\texttt{prop}, \texttt{conj}_1), (\texttt{noun}, \texttt{conj}_5), (\texttt{prop}, \texttt{punct})\}$ matches this rule, and a dependency between $p(0)$ and $p(2)$ is extracted.

Modality Extractor

When a phrase is a verb phrase, this module is applied. This module extract the modality which is included in a verb phrase, and assign them to some kind of intention: Question, Request, Negation, Hope, If and so on. For simple example, the question mark "?" means modality Question. The rules to extract modality are described in a term which indicate modality or a correlation of terms. These modality are extracted by a rule based pattern matching engine.

Term Extractor

Term Extractor generates domain-specific compound nouns from given noun phrases. When a phrase is a noun phrase, this module is applied it. This module make some simple nouns to be one compound noun. The list and rules of compound nouns are given as a sequence of terms or a sequence of part-of-speech. If there is a rule $r = (\texttt{noun}_{\texttt{firstname}}, \texttt{noun}_{\texttt{lastname}})$ in given rules, a sequence of term which has the same sequence of part-of-speech : "John" "Smith" is to be "John Smith".

These rule can be generated manually, however it cost much since noun and proper noun are open classes. Thus, we tried to extract significant terms for text mining. The list of compound nouns are given from the result of extracting significant terms.

5.2 Text Mining Application

The purpose of text mining is to discover and extract knowledge from text databases. We have developed a text mining system, which incorporates NLP. It can analyze time-sequence trends and discover associations in textual data, and has functions for both keyword-based and full-text search.

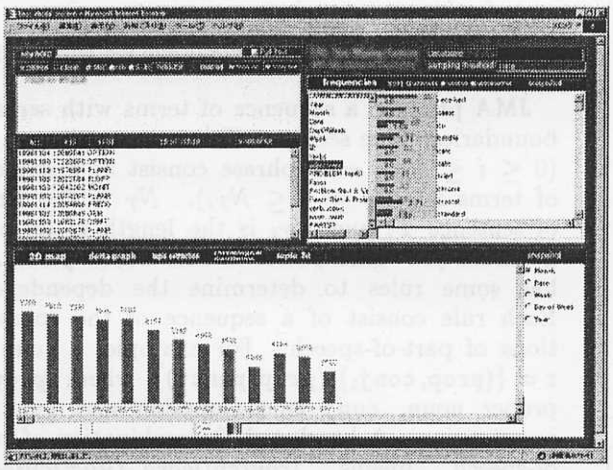

Figure 7: Our Text Mining Application

5.3 Result of Mining

The text mining system includes a function for discovering associations between terms in the textual data. This function can show correlations of noun and verb groups, which are tagged by a part-of-speech tagger. If there are domain dependent dictionaries, these terms may belong to categories such as Hardware and Software. In this experiment, we examined correlations between noun and verb.

As described in the above section, there are some methods to extract terms from text. In the case of nouns, simple terms, noun phrases and possible combinations of compound nouns can be extracted. In the case of verbs, adjectives and adverbs, stemmed terms and terms with modality can be also extracted from text.

Although the mining process is a statistical process, if the correct meaning of terms isn't extracted, the result of mining has no value. Thus, in an experiment shown below, verbs, adjectives and adverbs with modality are used (e.g. "hot" and "not hot" must be distinguished.). This experiment compares the results of correlations among simple noun terms, noun phrases and terms which are extracted by our method.

Simple noun

Table 2 shows correlations between nouns and verbs (including adjectives and adverbs) using simple noun terms. In this table, not all correlations are shown. These correlations shown in the table are only the peculiar combinations. The combinations which satisfy both:

1. Occurrence of the term exceed 10 times in all document.
2. The singularity value exceed 10.0, are shown. The singularity value is calculated as, for example, a noun "Windows" co-occurs many verbs such "startup" and "shutdown". If "startup" co-occurs twice as "shutdown", "startup"'s singularity value is $(1+2)/2 = 1.5$ for a noun "Windows".

Rank numbering from 0 is a ranking sorted by frequency. The original is in Japanese, the right side in the table denoted "(EN)" shows the meaning in English. Bracketed terms consist of more than two simple terms. All nouns are simple noun terms, so there are no bracketed terms, but verbs have modality, so there are bracketed terms.

Noun Phrase

All configurations are the same in the case of the simple noun. Most of these term combinations are frequent, but they are not useful as the representation of phenomena.

Our method

Table 4 shows the result using our method. Many compound nouns are in the result, and they seems to represent what customer want to say.

Table 2: Correlations: Simple noun

Rank	Freq.	Noun	Verb	Noun (EN)	Verb (EN)
0	327	onegai	itasu	ask	please
1	229	key	osu	key	push
2	79	tensou	negau	routing	wish
3	74	me-ru	jusinsuru	mail	receive
4	73	scandisk	kakaru	scandisk	do
5	73	konsento	nuku	concent	pull
6	33	lanpu	tentousuru	lamp	lit
8	65	HD	zousetusuru	HD	add
10	62	hassin	(kikoe)-(nai)	call	(don't)-(hear)
11	58	network	(deki)-(nai)	network	(don't)-(work)

Table 3: Correlations: Noun Phrase

Rank	Freq.	Noun	Verb	Noun (EN)	Verb (EN)
0	260	onegai	itasu	ask	please
3	58	(hassin)-(on)	(kikoe)-(nai)	(call)-(sound)	(don't)-(hear)
5	41	gazou	torikomu	picture	scan
8	30	(taiou)-(onegai)	itasu	(response)-(ask)	please
10	25	hikiage	negau	sendback	want
12	24	jikkou	shiteisuru	command	indicate
13	23	(dengen)-(koncento)	nuku	lamp	lit
15	22	(hidari)-(ue)	tenmetusuru	(left)-(up)	lit
17	21	(orikaesi)-(go)-(renraku)	itasu	(call)-(.)-(back)	please
19	17	(kounyuu)-(jiki)	sousinsuru	(purchase)-(date)	send

Figure 8 shows a comparison of frequency among these three methods. X axis is the ranking shown in Table 2 ~ 4 of terms and Y axis is the frequency of terms. Simple terms gain high frequency combinations, and noun phrase have low frequency. By using our method, the frequency of term combinations have mid-range frequency.

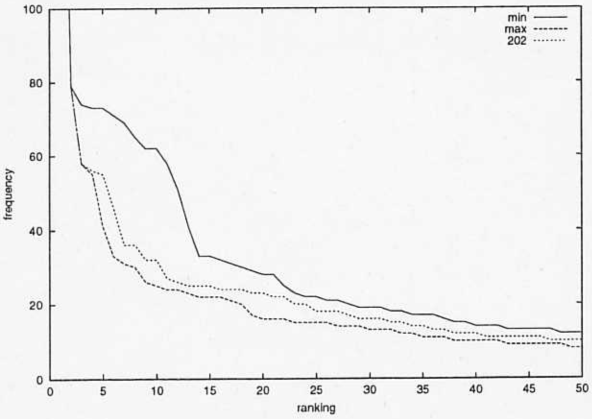

Figure 8: Ranking

6 Conclusion

Text mining aims not only to discover knowledge, but also to show an overview of a large collection of textual data. Since the mining results should be as informative as possible, we have proposed significant terms and dependencies to exclude trivial and rare content aggregations.

Significance terms are defined in terms of the entropy value, which successfully specifies a collection meaningful words, phrases, and dependencies. Our experiments showed that the 10 most significant terms and dependencies improved the content aggregation than simple keywords. In the future, we will further explore trend analysis and FAQ generations based on significant terms and dependencies.

References

Gerard Salton. 1983. SMART and SIRE Experimental Retrieval Systems. McGraw-Hill.

Hiroshi Maruyama, et al. 1994. Japanese Morphological Analysis Based on Regular Grammar Transactions of Information Processing Society of Japan (IPSJ), Vol.35, No.7, pages 1293-1299.

Tomek Strzalkowski, et al. 1992.
Information Retrieval using Robust Natural Lan-

Table 4: Correlations: Our method

Rank	Freq.	Noun	Verb	Noun (EN)	Verb (EN)
0	260	onegai	itasu	ask	please
2	79	tensou	negau	routing	please
3	58	(hassin)-(on)	(kikoe)-(nai)	(call)-(sound)	(don't)-(hear)
4	56	(kaihi)-(houhou)	osieru	(method)-(avoidance)	tell
8	36	(dengen)-(lampu)	tentousuru	(power)-(lamp)	lit
12	27	jikkou	shiteisuru	command	indicate
14	25	(syuusei)-(FD)	todoku	(fixpack)-(FD)	receive
15	25	hikiage	negau	sendback	want
19	23	(hidari)-(ue)	tenmetusuru	(left)-(up)	lit
20	23	(hidari)-(ue)	(kidousi)-(nai)	(left)-(up)	(don't)-(startup)

guage Processing. Proceedings of the Association for Computational Linguistics (ACL1992).

Rakesh Agrawal. 1993. Mining Association Rules between Sets of Items in Large Databases. Proceedings of the 1993 ACM SIGMOD, pages 207-216.

Hahn U. et al. 1997. Deep Knowledge Discovery from Natural Language Texts. Proceedings of KDD-97 pages 175-178.

Knight M. 1999. Mining Online Text. Communications of the ACM, Vol42, Number11, pages 58-61.

Mladenic D. et al. 1999. Text-Learning and Related Intelligent Agent: A Survey. IEEE Intelligent Systems. Volume14, Number4, pages 44-54.

Marti A Hearst. 1999. Untangling Text Data Mining. Proceedings of ACL-99, pages 3-10.

Kyo Kageura. 1996. Methods of Automatic Term Recognition - A Review - Tech rep., Terminology 3.

Philip Resnik. 1995. Using Information Content to Evaluate Semantic Similarity in a Taxonomy. Proceedings of 14th International Joint Conference on Artificial Intelligence (IJCAI95).

Tohru Hisamitsu, et al. 1999. Measuring Representativeness of Terms. Proceedings of the SIGNL133-16 Information Processing Society of Japan. pages 115-122.

Slava M Katz. 1995. Distribution of Content Words and Phrases in Text and Language Modeling. Natural Language Engineering 2(1), pages 15-59. Cambridge University Press 1996.