Text Data Mining: Discovery of Important Keywords in the Cyberspace

Arimura, Hiroki
PRESTO, JST.

Abe, Junichiro
Department of Informatics, Faculty of Information Science and Electrical Engineering, Kyushu University

Fujino, Ryoichi
ENICOM

Sakamoto, Hiroshi
Department of Informatics, Faculty of Information Science and Electrical Engineering, Kyushu University

他

出版情報：Digital Libraries: Research and Practice, pp.220-226, 2000. IEEE
バージョン：

Text Data Mining: Discovery of Important Keywords in the Cyberspace*

Hiroki Arimura
Department of Informatics
Kyushu University
PRESTO, JST.
arim@i.kyushu-u.ac.jp

Junichiro Abe
Department of Informatics
Kyushu University
j-abe@i.kyushu-u.ac.jp

Ryoichi Fujino
ENICOM
fujino@po0.kix.or.jp

Hiroshi Sakamoto
Department of Informatics
Kyushu University
hiroshi@i.kyushu-u.ac.jp

Shinichi Shimozono
Department of Artif. Intell.
Kyushu Inst. Tech.
sin@ai.kyutech.ac.jp

Setsuo Arikawa
Department of Informatics
Kyushu University
arikawa@i.kyushu-u.ac.jp

Abstract

This paper describes applications of the optimized pattern discovery framework to text and Web mining. In particular, we introduce a class of simple combinatorial patterns over phrases, called proximity phrase association patterns, and consider the problem of finding the patterns that optimize a given statistical measure within the whole class of patterns in a large collection of unstructured texts. For this class of patterns, we develop fast and robust text mining algorithms based on techniques in computational geometry and string matching. Finally, we successfully apply the developed text mining algorithms to the experiments on interactive document browsing in a large text database and keyword discovery from Web bases.

1. Introduction

The rapid progress of computer and network technologies makes it easy to collect and store a large amount of unstructured or semi-structured texts such as webpages, HTML/XML archives, E-mails, and text files. These text data can be thought of large scale text databases, and thus it becomes important to develop an efficient tools to discover interesting knowledge from such text databases.

There are a large body of data mining researches to discover interesting rules or patterns from well-structured data such as transaction databases with boolean or numeric attributes [1, 8]. However, it is difficult to directly apply the traditional data mining technologies to such text or semi-structured data since these text databases consist of (i) heterogeneous and (ii) huge collections of (iii) un-structured or semi-structured data. Therefore, there still are a small number of studies on text data mining, e.g. [5, 6].

Our research goal is to devise an efficient semi-automatic tool that supports human discovery from large text databases. Therefore, we require a fast pattern discovery algorithm that can work in time $O(n)$ to $O(n \log n)$ to respond in real time on an unstructured data set of total size n. Furthermore, such algorithm has to be robust in the sense that it can work on a large amount of noisy and incomplete unstructured data without the assumption of an unknown hypothesis class. To achieve this goal, we adopt the framework of *optimized pattern discovery* [8] described below, and develop efficient and robust pattern discovery algorithms combining the advanced technologies in string algorithm, computational geometry, and computational learning theory.

2. Framework of text mining

2.1. Optimized pattern discovery

The framework of *optimized pattern discovery*, adopted in this paper, is originally proposed by Fukuda et al. [8] in the field of data mining and also known as *Agnostic PAC learning* [11] in computational learning theory. In optimized pattern discovery, a pattern discovery algorithm tries to find a pattern from a given hypothesis space that optimizes a statistical measure function, such as *classification*

*This research is supported in part by the Ministry of Education, Science, Sports, and Culture of Japan, Grant-in-Aid for Scientific Research on Priority Areas "Discovery Science."

error [11], *information entropy* [13], *Gini index* [3], and χ^2 *index* [13] to discriminate a given target (or *positive*) data set from another background (*negative*) data set [3, 13].

More precisely, we define the optimized pattern discovery as follows. Suppose that we are given a set $S = \{s_1, \ldots, s_m\}$ of texts and an objective function $\xi : S \to \{0, 1\}$, where each s_i is called a *document*. The value of the objective function $\xi(s_i)$ indicates that the document s_i is interesting (positive) if $\xi(s_i) = 1$ and not interesting (negative) otherwise.

Let $\mathcal{P}$ is a (possibly infinite) class of *patterns*, where for any pattern $\pi \in \mathcal{P}$ and any string s, we define $\pi(s) = 1$ if π *matches* s and $\pi(s) = 0$ otherwise. Let S be a set of documents and ξ be an objective function. Then, a pattern π defines a contingency table (M_1, M_0, N_1, N_0), where N_1 (resp., N_0) is the number of all positive (resp., negative) documents in S and M_1 (resp., M_0) is the number of all positive (resp., negative) documents $s \in S$ such that $\pi(s) = 1$.

An impurity function is any real-valued function $\psi : [0, 1] \to \mathbf{R}$ such that (i) it takes the maximum value $\psi(1/2)$, (ii) the minimum value $\psi(0) = \psi(1) = 0$, and (iii) ψ is convex, i.e., $\psi((x + y)/2) \geq (\psi(x) + \psi(y))/2$ for every $x, y \in [0, 1]$. The followings are examples of impurity functions:

- The prediction error: $\psi_1(x) = \min(x, 1 - x)$.
- The information entropy: $\psi_2(x) = -x \log x - (1 - x) \log(1 - x)$.
- The gini index: $\psi_3(x) = 2x(1 - x)$.

Then, the evaluation function based on ψ over S and ξ is

$$G^{\psi}_{S,\xi}(\pi) = \psi(M_1/N_1)N_1 + \psi(M_0/N_0)N_0,$$

where (M_1, M_0, N_1, N_0) is the contingency table defined by the pattern π over S and ξ.

Now, we state the our data mining problem called the *optimal pattern discovery problem* as follows. Let $\mathcal{P}$ be the class of candidate patterns and ψ be any impurity function.

Optimal Pattern Discovery Problem

Given: a set S of documents and an objective function $\xi : S \to \{0, 1\}$.

Problem: Find an optimal pattern $\pi \in \mathcal{P}$ that minimizes the cost $G^{\psi}_{S,\xi}(\pi)$ within $\mathcal{P}$.

In what follows, we consider the information entropy measure only, but not limited to it. From recent development in learning theory, it is known that any algorithm that efficiently solves, e.g., classification error minimization, can approximate arbitrary unknown probability distributions and thus can work with noisy environments [11].

2.2. Keyword discovery by optimized pattern discovery

An intuition behind the application of the optimized pattern discovery to text mining can be explained as follows. Suppose that we are given as the target set a collection of text files with unknown vocabulary, say, Reuter newswires for one year of 1987 [12]. We want to take a look at the contents and find a set of topic keywords characterizing the major topics arising for one year of 1987.

A possible way to find such keywords or phrases is to find the keywords that frequently appear in the target set as in traditional data mining. However, this does not works in most text collections because in a typical English text, the most frequent keywords are stopwords like "the" or "an" (see Table 1 (a) and Table 3 (a)). These keywords are basic constituents of English grammars and convey no information on the contents of the text collection. Such frequent but less informative stopwords may hide less frequent informative keywords. The traditional information retrieval technique called *stopword elimination* may not work, too.

A basic idea behind our method is to use an average set of texts as the control set used for canceling the occurrences of frequent and non-informative keywords. The control set will be a set of documents randomly drawn from the whole text collection or the internet. We can easily observe that most stopwords appear evenly in the target and the control set, while informative keywords appear more frequently in the target set than the control set. Therefore, the optimized pattern discovery algorithm will find those keywords or phrases that appear more frequently in the target set than the control set by minimizing a given statistical measure such as the information entropy or the prediction error (See Table 1 (b)-(c) and Table 3 (b)-(c)).

2.3. The class of patterns

The class of patterns we consider is the class of proximity phrase association patterns [3]. We mean by a *phrase* any string of tokens, which may be either letters or words, of arbitrary length. A *phrase association pattern* (phrase pattern) is an expression of the form

(⟨attacks⟩,⟨iranian oil platform⟩;8)

which expresses that phrase ⟨attacks⟩ first appears in a document and then phrase ⟨iranian oil platform⟩ follows within eight words. A phrase pattern can contain arbitrary many but bounded number of phrases as its components. If the order of the phrases in a pattern matters as in the above example then we call it *ordered* and otherwise *unordered*. Proximity phrase association patterns can be regarded as a generalization of association rules in transaction databases [1] such that (i) each item is a phrase of arbitrary

Table 1. Phrase mining with entropy optimization to capture `ship` category. To see the effectiveness of entropy optimization, we mine only patterns with $d = 1$, i.e., single phrases. (a) The best ten frequent phrases found by traditional frequent pattern mining. (b) Short phrases of smallest rank, $1 \sim 10$ and (c) Long phrases of middle rank, $261 \sim 270$ found by entropy minimization mining. The data set consists of 19,043 articles of 15.2MB from Reuters Newswires in 1987.

(a) Frequency maximization	(b) Entropy minimization	(c) Entropy minimization
1 <reuter >	1 <gulf >	261 <mhi >
2 <the >	2 <ships >	262 <mclean >
3 <to >	3 <shipping >	263 <lloyds shipping intelligence >
4 <said >	4 <iranian >	264 <iranian oil platform >
5 <of >	5 <iran >	265 <herald of free >
6 <and >	6 <port >	266 <began on >
7 <in >	7 <the gulf >	267 <bagged >
8 <a >	8 <strike >	268 <24 - >
9 <s >	9 <vessels >	269 <18 - >
10 <on >	10 <attack >	270 <120 pct >

length, (ii) items are ordered, and (iii) a proximity constraint is introduced.

3. A fast and robust text mining algorithm for ordered patterns

If the maximum number of phrases in a pattern is bounded by a constant d then the frequent pattern problems for both unordered and ordered proximity phrase association patterns are solvable by *Enumerate-Scan* algorithm [15], a modification of a naive generate-and-test algorithm, in $O(n^{d+1})$ time and $O(n^d)$ scans although it is still too slow to apply real world problems.

Adopting the framework of optimized pattern discovery, we have developed an efficient algorithm, called *Split-Merge*, that finds all the optimal patterns for the class of *ordered* k-proximity d-phrase association patterns for various measures including the classification error and information entropy [3, 4]. The algorithm quickly searches the hypothesis space using dynamic reconstruction of the content index, called a *suffix array* with combining several techniques from computational geometry and string algorithms.

We showed that the Split-Merge algorithm runs in *almost linear time in average*, more precisely in $O(k^{d-1}N(\log N)^{d+1})$ time using $O(k^{d-1}N)$ space for nearly random texts of size N [4]. We also show that the problem to find one of the best phrase patterns with arbitrarily many strings is MAX SNP-hard [4]. Thus, we see that there is no efficient approximation algorithm with arbitrary small error for the problem when the number d of phrases is unbounded.

4. Developing a scan-based algorithm for unordered patterns

In Web mining, the unordered version of the phrase patterns are more suitable than the ordered version. Besides this, we also have to deal with huge text databases that cannot fit into main memory. For the purpose, we developed another pattern discovery algorithm, called *Levelwise-Scan*, for mining unordered phrase patterns from large disk-resident text data [7].

Based on the design principle of the Apriori algorithm of Agrawal [1], the Levelwise-Scan algorithm quickly discovers most frequent unordered patterns with d phrases and proximity k in time $O(n^2 + N(\log n)^d)$ and space $O(n \log n + R)$ on nearly random texts using a random sample of size n, where N is the total size of input text and R is the output size [7].

To cope with the problem of the huge feature space of phrase patterns, the algorithm combines the techniques of random sampling, the generalized suffix tree, and the pattern matching automaton. By computer experiments on large text data, the Levelwise-Scan algorithm quickly finds patterns for various ranges of parameters and linearly scales up on a large disk-resident text database. For example, the estimated running time of Levelwise-Scan algorithm on a text database of 500MB is around three hours on a Sun workstation (UltraSPARC-II 300MHz, g++ on Solaris2.6) with a sample size 500KB [7].

5. Application to Interactive document browsing

We applied our text mining method to *interactive document browsing*. Based on the Split-Merge algorithm [4], we developed a prototype system on a unix workstation, and run experiments on a medium sized English text collection.

5.1. Data sets and a prototype system

We used an English text collection, called Reuters-21578 data [12], which consists of news articles of 27MB on international affairs and trades from February to August in 1987. The sample set is a collection of unstructured texts of the total size 15.2MB obtained from Reuters-21578 by removing all but category tags. The target (positive) set consists of 285 articles with category `ship` and the background (negative) set consists of 18,758 articles with other categories such as `trade`, `grain`, `metal`, and so on. The average length of articles is 799 letters.

By experiments on Reuters-21578 data above of 15.2MB, the prototype system finds the best 600 patterns at the entropy measure in seconds for $d = 2$ and a few minutes for $d = 4$ and with $k = 2$ words using a few hundreds mega-bytes of main memory on Sun Ultra 60 (Ultra SPARC II 300MHz, g++ on Solaris 2.6, 512MB main memory) [9].

5.2. The first experiment

In Table 1, we show the list of the phrase patterns discovered by our mining system, which capture the category `ship` relative to other categories of Reuters newswires. In Fig. 1 (a), we show the list of most frequent keywords discovered by traditional frequency maximization method. On the other hands, we list in Fig. 1 (b) and (c), we show the list of optimal patterns discovered by Entropy minimization method. The patterns of smallest rank ($1 \sim 10$) contain the topic keywords in the major news stories for the period in 1987 (Fig. 1(b)). Such keywords are hard to find by traditional frequent pattern discovery because of the existence of the high frequency words such as ⟨`the`⟩ and ⟨`are`⟩. The patterns of medium rank ($261 \sim 270$) are long phrases, such that ⟨`lloyds shipping intelligence`⟩ and ⟨`iranian oil platform`⟩, as a summary (Table 1 (c)), which cannot be represented by any combination of non-contiguous keywords.

5.3. The second experiment

Table 2 shows an experiment on interactive text browsing, where we try to find an article containing a specific topic from a collection of documents by using optimized pattern discovery combined with keyword search.

First, we suppose that a user is looking for articles related to the labor problem, but he does not know any specific keywords enough to identify such articles. (a) Starting with the original target and the background sets related to `ship` category in the last section, the user first finds topic keywords in the original target set using optimized pattern mining with $d = 1$, i.e., phrase mining. Let `union` be a keyword found. (b) Then, he builds a new target set by drawing all articles with keyword `union` from the original target set. The last target set is used as the new background set. As a result, we obtained a list of topic phrases concerned to `union`. (c) Using a term `seamen` found in the last stage, we try to find long patterns consisting of four phrases such that the first phrase is `seamen` using proximity $k = 2$ words. In the table, a pattern ⟨`seamen `⟩⟨`were `⟩⟨`still on strike`⟩ found by the algorithm indicates that there is a strike by a union of seamen.

6. Discovery of important keywords in Web

In this section, we apply our text mining system to discovery of important keywords that characterize a community of web pages. In the Web search, a user may use a search engine to find a set of Web pages by giving a keyword relevant to an interesting topic. However, for too ambiguous keywords the search engine often returns a large amount of documents as the answers, and thus a postprocessing is required to select the pages that the user is really interested.

6.1. Method

In the experiments, the mining system receives two keywords, called the *target* and the *control* as inputs, and computes the collections of Web pages called the base sets of the target and the control keywords as follows. For a keyword w, let the *base set* of w be the set of best, say 200, pages obtained by making a query w to a Web search engine together with those pages pointed by the hyperlinks from the first set of pages. It is well known that the base set of a keyword forms a well connected community [10]. Then, the base sets corresponding to the target and the control keywords are given the text mining system as positive and negative examples, respectively, and the system discovers the phrases that characterize the base set of the target keyword relative to the base set of the control keyword by using entropy minimization.

6.2. Data

We used *AltaVista* [2] as an Web search engine and Perl and Java scripts to collect Web pages from internet.

Table 2. Document browsing by optimized pattern discovery. (a) First, a user tries to mine the original target set using optimized pattern discovery over phrases using the background set ($d = 1, k = 0$). The user selected a term `union` of rank 51. (b) Next, the user mines a subset of articles relating to `union` and mine this set again by optimized phrases ($d = 1, k = 0$). He obtained topic terms on `seamen`. (c) Finally, the user tries to discover optimized patterns starting with `seamen` on the same target set ($d = 4, k = 2$ words). The underlined pattern indicates a strike by a union of seamen.

(a) Stage1

Rank	Pattern
46	⟨loading ⟩
47	⟨u s flag ⟩
48	⟨platforms ⟩
49	⟨s flag ⟩
50	⟨the strike ⟩
51	⟨**union** ⟩
52	⟨kuwait ⟩
53	⟨waters ⟩
54	⟨missiles ⟩
55	⟨navy ⟩

(b) Stage 2

Rank	Pattern
1	⟨union ⟩
2	⟨u.s. ⟩
3	⟨**seamen** ⟩
4	⟨strike ⟩
5	⟨pay ⟩
6	⟨gulf ⟩
7	⟨employers ⟩
8	⟨labor ⟩
9	⟨de ⟩
10	⟨redundancies ⟩

(c) Stage 3

Rank	Pattern
1	⟨seamen ⟩⟨and ⟩⟨the ⟩
2	⟨seamen ⟩⟨a ⟩⟨pay ⟩
3	⟨seamen ⟩⟨were ⟩⟨strike. ⟩
4	⟨**seamen** ⟩⟨**were** ⟩⟨**still on strike.** ⟩
5	⟨seamen ⟩⟨were ⟩⟨on strike. ⟩
6	⟨seamen ⟩⟨to ⟩⟨after ⟩
7	⟨seamen ⟩⟨said ⟩⟨were ⟩
8	⟨seamen ⟩⟨pct ⟩⟨pay ⟩
9	⟨seamen ⟩⟨on ⟩⟨strike. ⟩
10	⟨seamen ⟩⟨leaders ⟩⟨of ⟩

Table 3. Mining the Web. Characterizing communities of Web pages. In each table the first, the second, the third, the fourth columns show, repectively, the rank, the number of occurrences in the positive and the negative documents, and the phrase found. The number of phrases is $d = 1$.

(a) Frequency maximization
POS = honda

1	17893	<the >
2	12377	<and >
3	11904	<to >
4	11291	<a >
5	8728	<of >
6	8239	<for >
7	7278	<in >
8	5752	<is >
9	5626	<i >
10	5477	<honda >
11	4838	<on >
12	4584	<s >
13	4571	<with >
14	4545	<you >
15	3880	<it >
16	3650	<or >
17	3447	<this >
18	3279	<that >
19	3115	<are >
20	3085	<99 >

(b) Entropy minimization
POS = honda, NEG = softbank

1	5477	4	<honda >
2	2125	0	<prelude >
3	5626	1099	<i >
4	1863	68	<car >
5	1337	24	<parts >
6	1472	92	<engine >
7	862	1	<rear >
8	744	0	<vtec >
9	3085	769	<99 >
10	718	0	<exhaust >
11	754	16	<miles >
12	629	2	<motorcycle >
13	662	6	<bike >
14	734	20	<racing >
15	732	29	<si >
16	802	45	<black >
17	526	1	<tires >
18	586	13	<fuel >
19	555	9	<civic >
20	1307	219	<me >

(c) Entropy minimization
POS = honda, NEG = toyota

1	5477	302	<honda >
2	2125	9	<prelude >
3	744	5	<vtec >
4	732	16	<si >
5	662	23	<bike >
6	629	23	<motorcycle >
7	3085	942	<99 >
8	376	0	<prelude si >
9	448	9	<the honda >
10	555	32	<civic >
11	396	5	<honda prelude >
12	310	0	<valkyrie >
13	309	0	<99 time >
14	390	12	<honda s >
15	1208	266	<98 >
16	274	0	<scooters >
17	396	17	<rims >
18	271	0	<date 10 >
19	299	3	<92 96 >
20	435	29	<accord >

The target keyword is ⟨honda⟩, the name of an automobile company, and the control keywords are ⟨softbank⟩, the company on internet business, and ⟨toyota⟩, another automobile company. From these keywords ⟨honda⟩, ⟨softbank⟩, and ⟨toyota⟩, we obtained collections of plain text files of size 4.90MB (52,535 sentences), 3.65MB (34,170 sentences), and 5.05MB (40,757 sentences), respectively, after removing all HTML tags. Although the keywords ⟨honda⟩ and ⟨toyota⟩ can be used as the name of persons, they were used as the names of automobile companies in most pages of high ranks.

6.3. Results

In Table 3, we show the best 20 phrases found in the target set ⟨honda⟩ varying the control set. In the table (a), we show the phrases found by traditional frequent pattern mining with the empty control [1]. In the next two tables, we show the phrases found by mining based on entropy minimization, where the target is ⟨honda⟩ and the control varies between ⟨softbank⟩ and ⟨toyota⟩. We see that (b) the mining system found more general phrases, e.g. `cars, parts, engine` in the target ⟨honda⟩ with the control ⟨softbank⟩, while the system found more specific phrases, e.g. `prelude si,valkyrie, accord`, with the control ⟨toyota⟩. This difference on patterns come from the fact that the target keyword ⟨honda⟩ is more similar to the first control keyword ⟨toyota⟩ than the second control keyword ⟨softbank⟩.

7. Conclusion

In this paper, we investigate a new access method for large text databases on internet based on text mining. First, we formalized text mining problem as the optimized pattern discovery problem using a statistical measure. Then, we gave fast and robust pattern discovery algorithms, which was applicable for a large collection of unstructured text data in real time. We ran computer experiments on interactive document browsing and keyword discovery from Web, which showed the efficiency of our method on real text databases.

In this paper, we regard Web pages as unstructured texts. However, it is often more natural to consider Web pages as markup texts or semi-structured data [6, 14]. Using such a structure information, we may obtain more interesting pattern that characterize a given dataset well. Thus, it will be an interesting future problem to develop an efficient algorithm for finding optimal pattern in such semi-structured data.

References

[1] R. Agrawal, R. Srikant, Fast algorithms for mining association rules, In Proc. VLDB'94, 487–499, 1994.

[2] Compaq Computer K.K., AltaVista. `http:///www.altavista.com/`, 2000.

[3] Arimura, H., Wataki, A., Fujino, R., Arikawa, S., A fast algorithm for discovering optimal string patterns in large text databases, In Proc. ALT'98, LNAI 1501, 247–261, 1998.

[4] H. Arimura, S. Arikawa, S. Shimozono, Efficient discovery of optimal word-association patterns in large text databases *New Generation Computing*, 18, 49–60, 2000.

[5] W. W. Cohen, Y. Singer, Context-sensitive learning methods for text categorization, J. ACM, 17(2), 141–173, 1999.

[6] M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery, Learning to construct knowledge bases from the World Wide Web, Artificial Intelligence, 118, 69–114, 2000.

[7] R. Fujino, H. Arimura, S. Arikawa, Discovering unordered and ordered phrase association patterns for text mining. Proc. PAKDD2000, LNAI 1805, 281–293, 2000.

[8] T. Fukuda, Y. Morimoto, S. Morishita, and T. Tokuyama, Data mining using two-dimensional optimized association rules, In Proc. SIGMOD'96, 13–23, 1996.

[9] T. Kasai, T. Itai, H. Arimura, S. Arikawa, Exploratory document browsing using optimized text data mining, In *Proc. Data Mining Workshop*, 24–30, 1999 (In Japanese).

[10] J. M. Kleinberg, Authoritative sources in a hyperlinked environment. In Proc. SODA'98, 668–677, 1998.

[11] M. J. Kearns, R. E. Shapire, L. M. Sellie, Toward efficient agnostic learning. *Machine Learning*, 17(2–3), 115–141, 1994.

[12] D. Lewis, Reuters-21578 text categorization test collection, Distribution 1.0, AT&T Labs-Research, `http://www.research/.att.com`, 1997.

[13] S. Morishita, On classification and regression, Proc. DS'98, LNAI 1532, 49–59, 1998.

[14] H. Sakamoto, H. Arimura, S. Arikawa, Identification of Tree Translation Rules from Examples, Proc. the 5th International Colloquium on Grammatical Inference (ICGI 2000), LNAI 1891, 241-255, 2000.

[15] J. T. L. Wang, G. W. Chirn, T. G. Marr, B. Shapiro, D. Shasha and K. Zhang, Combinatorial pattern discovery for scientific data: Some preliminary results, In *Proc. SIGMOD'94*, 115–125, 1994.