Mining in the Phrasal Frontier[⋆,⋆⋆]

Helena Ahonen, Oskari Heinonen, Mika Klemettinen, and A. Inkeri Verkamo

University of Helsinki, Department of Computer Science
P.O. Box 26, FIN–00014 University of Helsinki, Finland

Abstract. Data mining methods have been applied to a wide variety of domains. Surprisingly enough, only a few examples of data mining in text are available. However, considering the amount of existing document collections, text mining would be most useful. Traditionally, texts have been analysed using various information retrieval related methods and natural language processing. In this paper, we present our first experiments in applying general methods of data mining to discovering phrases and co-occurring terms. We also describe the text mining process developed. Our results show that data mining methods — with appropriate preprocessing — can be used in text processing, and that by shifting the focus the process can be used to obtain results for various purposes.

1 Introduction

During recent years, data mining, or knowledge discovery, has become a popular research area. Data mining methods have been applied on a wide variety of domains, from supermarket basket data and telecommunication alarm data to the analysis of satellite pictures and human genomes. Aside the growing interest towards knowledge discovery, there has been an explosion in the amount of information available. Therefore, data mining from existing document collections, including the World Wide Web, which contains large amounts of semi-structured text in HTML format, would be most useful. Surprisingly enough, only a few examples of data mining in text analysis are available. The most notable are the KDT and FACT systems [4] used in mining Reuters news articles. Their approach, however, requires a substantially large amount of background knowledge, and is not applicable as such to text analysis in general.

The aim of our study is to apply data mining techniques — more specifically, discovery of episode rules — together with fast morphological analysis, to examine unrestricted text. As the first application we consider discovering *co-occurring terms*, words that frequently appear together in the text. Several cases can be distinguished, for instance, the words may have a fixed order or they can appear in any order. We can also permit other words between the co-occurring words, or require the words to be tightly coupled.

Discovery of co-occurring words has applications within several fields. For instance, in information retrieval, keywords and keyphrases are commonly used

⋆ This work was partially supported by the Finnish Technology Development Centre (TEKES). We thank Hannu Toivonen, Ph.D., for the episode rule algorithm implementation. Authors' e-mail: {*hahonen,oheinone,mklemett,verkamo*}*@cs.helsinki.fi*.

⋆⋆ To appear in *Proceedings of PKDD'97 – 1st European Symposium on Principles of Data Mining and Knowledge Discovery*, Trondheim, Norway, June 1997.

to boost query processing [8, 5]. Consider a common information retrieval task: The user expresses his/her information needs, e.g., by giving a query, and the system executes the search by matching the query with the documents. With large collections simply scanning the text is not feasible. Hence, a set of representative keywords must be selected and attached to the documents.

Often, however, single-term keywords are too broad to be used alone. *Phrases* consisting of sequences of related words carry a more specific meaning than the single terms included in the phrases, cf. for instance *processing* and *industrial processing*. In a sense, a set of phrases can be regarded as a content descriptor that should distinguish the document from other documents in the collection. In addition to simple queries, content descriptors can be used for various text classification tasks. For instance, documents can be clustered according to their similarity, to visualize a large document collection [2].

More elaborate ways of utilizing co-occurrent terms can be found in natural language processing techniques. A specific class of co-occurring terms are so-called *collocations*, i.e., recurrent combinations of words that correspond to arbitrary word usages [9]. Opposite to typical phrases used in information retrieval, collocations may often contain prepositions and inflected words. In addition to the linguistic interest, collocations may be useful in retrieval tasks. It has been shown that some types of collocations are domain-dependent and, hence, good indicators of the topics covered by the document. Although indexing and selecting keywords are well-studied within information retrieval, new challenges have been recently set by the sudden appearance of very large heterogeneous full text document collections. Lewis and Spärck Jones [5] consider compound keyterms as one essential possibility to improve the quality of text retrieval in the new situation; they also emphasise the need of exhaustive experimenting.

In this paper, we present our first experiments in applying general methods of data mining to discovering phrases and co-occurring terms. First, in Section 2 we describe the knowledge discovery process and methods used in this article. Then, in Section 3, we describe our experiments with text documents, and show that data mining methods — with appropriate preprocessing — can be used in text processing, and that by shifting the focus the process can be used to obtain results for various purposes. Finally, Section 4 is a short conclusion.

2 Methods

In this section, we describe the methods we have used for discovering useful information in text. The general discovery process, adapted to the task of text processing, is represented in Figure 1. The starting point of the discovery process is textual data in SGML representation, but we could likewise use the representation produced by any text processing program. The end product of the process are the episodes and episode rules describing phenomena that are frequent in the data, in the case of textual data, e.g., phrases or co-occurring terms. Let us first give a brief description of episodes and episode rules.

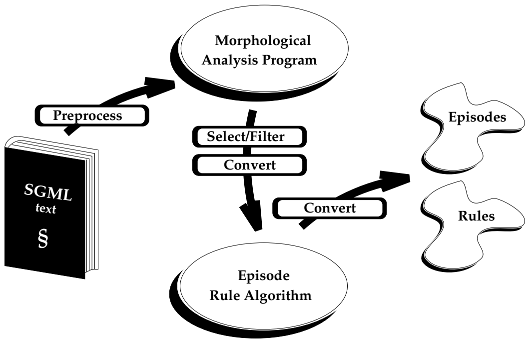

Fig. 1. Knowledge discovery from SGML representation into episodes and episode rules.

2.1 Episodes

Episode rules and *episodes* are a modification of the concept of *association rules* and *frequent sets* (see, e.g., [1]), applied to sequential data. Sequential data can be seen as a sequence of events, where each event is a pair *(event type, time)*. As an example, let us think of WWW log data, where an event is a pair consisting of a page name and a time of reference to the page, e.g., *(page A, Jan 22 19:46:49)*. An episode constitutes of the event types of events that occur closely enough, i.e., within a given time window. Thus, if the time window is 60 s, the sequence

(page A, Jan 22 19:46:49) (page B, Jan 22 19:47:19) (page C, Jan 22 19:48:49)

contains the episode *(page A, page B)*, but not the episode *(page A, page C)*.

An episode is *parallel*, if there is no requirement for the order of the events within the time window, and *serial*, if a total partial order of the events is required. For example, episode *(page A, page B)* is parallel, if pages A and B can be accessed in any order, whereas the episode is serial, if the pages must be accessed in the given order. An episode is said to be *frequent*, if the episode occurs in the event sequence often enough, i.e. at least so many times as indicated by the *support threshold*. Hence, the essential parameters for defining all frequent episodes in an event sequence are the time window, and the support threshold.

An episode rule is a rule of the form

$$\textit{page } A,\ \textit{page } B\ [20\ \mathrm{s}] \Rightarrow \textit{page } C\ [40\ \mathrm{s}]\ (63\ \%),$$

which tells us that in 63 % of the cases, where A and B were accessed within 20 s, also C was accessed within 40 s. In general, we are not interested in rules with a negligible *confidence*, e.g., less than 20 %; it is common to select only those rules with a confidence exceeding a given *confidence threshold*. For finding all interesting episode rules in a given event sequence, the necessary parameters are the support threshold, two window sizes (one for the left-hand side and one covering the entire episode), and the confidence threshold.

The method that we have used to discover frequent episodes and episode rules in our data is described in [7]. This method allows us to discover serial and parallel episodes of a given support threshold and episode rules of a given confidence threshold for a collection of window sizes with a fixed upper limit.

2.2 Phases of the Discovery Process

As we saw in Figure 1, the episode and rule discovery phase is only a part of the task of obtaining useful data. To get appropriate data for the analysis, substantial effort must usually be directed to the preprocessing phase. Referring to the results in different domain areas and applications, preprocessing can take as much as 80 per cent of the total effort [6].

In the context of our knowledge presentation formats, a KDD process, adapted, e.g., from [3], consists of (1) data preprocessing (selection, cleaning, etc.), (2) data transformation and input selection for discovery phase, (3) discovery of episodes and episode rules, (4) presentation of the results, and (5) interpretation and utilization of the results. The preprocessing and transformation operations necessary for the text data that we used are described in Section 3.1.

The output formats for frequent episodes and episode rules are as follows. For each frequent episode, its frequency count is given (see Figure 3 a), and the episodes are sorted in the decreasing order of frequency. For each episode rule, the IF part denotes the rule left-hand side, and the THEN part denotes the rule right-hand side; the WITH part contains the window size for the left-hand side and the entire episode, the confidence of the rule, and the exact frequencies of the entire episode and the left-hand side (see Figure 3 b). Different window sizes may produce several instances of the same rule with different confidence values.

3 Experiments

To survey the usefulness of knowledge discovery methods and the discovered knowledge in the context of text documents, we have made experiments with real data sets. In the experiments, we have used the methods described in the previous section, namely the discovery of episodes and episode rules.

3.1 Used Data Sets and Preprocessing

For the experiments we used Finnish legal texts, originally in SGML format. The document collection consisted of 759 separate documents, which in turn contained over 30 000 different “words” or symbols.[3] From the statutes, i.e. acts and decrees, we selected a random sample of 14 documents to be used in the experiments. The test material is described in Table 1.

To begin with, we replaced special characters (some marked as entities in SGML, e.g., “§” for “§”), parentheses, commas, etc., by symbols such as “LPAREN”, “RPAREN” and “COMMA”, thus enabling later recognition. In addition, full stops were recognized with simple heuristics and labelled “FSTOP”. Finally, all SGML tagging was removed, except for the structural information to be used in future experiments.

[3] In this article, by “word” we mean not only normal words, but also commas, full stops, exclamation marks, etc.

Statute (act/decree)	Size of Orig. SGML file (in kilobytes)	Number of Real Words	Number of Words	Number of Words + Interprets	Number of Sentences
Painovapauslaki	44	3 349	4 273	5 233	144
Oikeudenkäymiskaari	381	28 535	38 970	47 250	1 381
Ilmansuojelulaki	29	1 920	2 622	3 143	87
Osuuskuntalaki	218	15 720	19 682	24 646	767
Kolttalaki	74	5 513	6 427	7 869	259
Vesilaki	644	49 330	61 559	75 889	2 053
Tiekuljetussopimuslaki	63	4 938	5 815	7 307	206
Kansaneläkelaki	211	15 497	20 756	25 065	724
Kemikaaliasetus	43	3 161	3 997	4 781	89
Sosiaalihuoltolaki	61	4 190	5 491	6 636	187
Kemikaalilaki	79	5 260	7 169	8 353	241
Laki rikosvahinkojen korvaamisesta...	34	2 502	3 445	4 234	124
Tilintarkastuslaki	56	3 857	4 576	5 516	159
Jäteasetus	54	4 187	5 239	6 153	269

Table 1. The test material. Number of real words is the actual word count without special characters, commas, etc., included in the number of words.

After cleaning the data, the statutes were fed to a morphological analyser program (FINTWOL[4]), which gives us the basic dictionary form and the morphological analysis of each word (capitalized symbols, e.g. "FSTOP", are passed and marked "unknown"). Note that FINTWOL only looks at one word at the time and does not try to disambiguate using the word context. An example of the output is "`rikoslain rikoslaki N GEN SG`", which tells us that the word which occurred in the text is *rikoslain*, its basic dictionary form is *rikoslaki* (in English, Criminal Act). Furthermore, the morphological analysis reveals that the word is noun (N), genitive (GEN), and singular (SG).

Getting the basic forms is important since in the Finnish language words most often appear inflected. Mostly the words are correctly and unambiguously recognized by FINTWOL. It is possible, however, that a word has several interpretations, even being inflections of separate basic forms. Nevertheless, telling a noun from a verb or an adjective is usually not a problem in Finnish. In these preliminary experiments, we only used information about the part of speech and discarded the more detailed morphological information.

3.2 Discovery of Episodes and Episode Rules

Data Transformation After the preprocessing phase, the data format consists of one word and its part of speech per line, possible multiple interpretations of the words on consecutive lines, with one blank line separating different words.

To be used with the episode and episode rule generation algorithms, we supplied the words with an index indicating the "time stamp" of the word (i.e., first word is numbered "1", the second is numbered "2", and so on). Because the episode algorithms can only take *(event type, time)* pairs, we preprocessed the data to contain two fields: (a) the selected attributes concatenated as one aggregate attribute and (b) the index.

[4] FINTWOL is a product of Lingsoft, Inc.

```
(a) painovapauslaki_N 1      (b) monistaa_V 72
    eduskunta_N 9                monistettu_A 72
    päätös_N 10                  painokirjoitus_N 85
    seuraava_A 13                sanoa_V 86
    luku_N 16                    koskea_V 88
    painokirjoitus_N 17          kuvallinen_A 90
    julkaisuoikeus_N 18          esitys_N 91
```

Fig. 2. The input formats for the discovery of phrases (a) and co-occurring terms (b). The former contains only nouns (N), proper nouns (PROP), and adjectives (A). Additionally, the latter contains also verbs (V).

Discovery We first made some experiments using all the words in the documents but while analysing rather large documents, we soon discovered that the attribute space was too large (tens of thousands of attributes), thus requiring an unfeasible amount of main memory and computation time.

Based on that, we selected only certain interesting parts of speech and considered two simple test cases, the discovery of *phrases* (using serial episodes) and *co-occurring terms* (using parallel episodes). Both cases are potentially useful for various purposes and rather well studied using different methods; the latter case is particularly important in Finnish, where the word order is, in general, very flexible. We also wanted to compare the results between the analysis of the words with and without separating successive sentences. The former was implemented by adding enough space or null events between sentences; i.e., we increased the index between the sentences corresponding the size of the maximum time window. To be more concrete, if the last word in the sentence has a time stamp "10" and the maximum time window is 5, then the time stamp of the first word of the next sentence is "16".

For phrase discovery, we selected only nouns, proper nouns, and adjectives. The search for co-occurring terms, on the other hand, was first carried through with nouns and proper nouns, and then with nouns, proper nouns, verbs, and adjectives. In the experiments, the words not included in the selected set were bypassed by increasing the index by 1. The input formats for the experiments are sketched in Figure 2. The parameters used for discovering phrases/co-occurring terms were episode support threshold 10/10, episode rule confidence threshold 0.2/–(no rule generation), and maximum window size 3/10.

3.3 Results Analysis

Phrases Generally speaking, the given results verify that linguistically reasonable phrases can be found with serial episode discovery. Examples of the phrases we found are the terms *teollinen käsittely* (industrial processing) and *vesioikeus päätös* (Water Rights Court judgement) in Figure 3.

The episodes and episode rules can be studied separately, but also in parallel: we can first search for frequent episodes and then study them more carefully by looking at the rules. For instance, consider the examples in Figure 3. If we take the episode (a) that occurs in the Chemical Act (statute #9) rather frequently then by looking at the rule (b) we can conclude that the phrase *teollinen käsittely*

```
(a) 37:  teollinen (A)          (c) 44:  vesioikeus (N)
         käsittely (N)                   päätös (N)

(b) IF   teollinen (A)          (d) IF   vesioikeus (N)
    THEN käsittely (N)              THEN päätös (N)
    WITH [0] [1] 0.0000 (0/38)      WITH [0] [1] 0.0000  (0/558)
         [0] [2] 0.9737 (37/38)          [0] [2] 0.0681  (38/558)
         [0] [3] 0.9737 (37/38)          [0] [3] 0.0735  (41/558)
```

Fig. 3. *Examplary results from phrase discovery: episodes and episode rules from the Chemical Act (a and b; statute #9) and Water Rights Act (c and d; statute #6).*

is not only a common phrase, but in practice the term *teollinen* always implies an immediate occurrence of the term *käsittely*. On the contrary, with an equally frequent episode (c) in the Water Rights Act (statute #6), the rule (d) tells us that *vesioikeus* is actually quite rarely immediately followed by *päätös*. This kind of analysis can be completed by looking at all rules that have either *vesioikeus* on the left-hand side or *päätös* on the right-hand side.

Despite the applicability of our methods, it is not necessarily always practical to use them for all text analysis needs. For instance, if we want to use fixed "phrase templates", e.g., "N+N, A+N, or A+A+N", then it is more efficient to just make a couple of scripts to scan and filter the text, and finally count the term frequencies.

Co-occurring terms The results from discovering co-occurring terms were rather similar to the ones related to the phrase recognition. For the parallel episodes (for an example, see Figure 4), at least with our material, there were no dramatical differences between the results obtained with or without a gap between the sentences; the main effect was some increase or decrease in the term frequencies.

```
(a) 88: tuomioistuin (N)  (b) 112: tuomioistuin (N)
        asia (N)                   asia (N)
    13: tuomioistuin (N)       13: tuomioistuin (N)
        syyte (N)                  syyte (N)
        rikos (N)                  rikos (N)

(c) 49: laki (N)          (d) 59: laki (N)
        tulla (V)                 tulla (V)
        voima (N)                 voima (N)
    31: syy (N)               32: syy (N)
        erityinen (A)             erityinen (A)
```

Fig. 4. Examplary results from the search for co-occurring terms. Examples (a) and (c) have a gap of the maximum window size between sentences, while (b) and (d) do not.

4 Conclusions and Future Work

The preliminary tests we carried out showed that episode and episode rule techniques can be applied to document data, and appear to have potential in analysing or mining text. Especially, if we do not know the accurate syntax of the language or the syntax is very complex, or if we want to allow phrases to contain occasional gaps between words, then our approach clearly can alleviate and broaden the search with respect to more straightforward procedures.

Additionally, our results show, not surprisingly though, that there is one factor which has a very significant impact on the results: preprocessing. As far as it is not feasible to search for relationships just by scanning the text, due to both the methods used and the computational capacity, preprocessing has a leading role as the most time-consuming part of the process — at least when tailoring the system to be applied in a new domain. The time and effort consumed in preprocessing, however, pays back in better results. Consequently, when preprocessing the data, we really had to focus on (1) extracting the appropriate parts of speech, (2) filtering out superfluous words (i.e. stopwords) and erroneous inflections, and (3) converting the data into a suitable format.

In our experiments so far, we have used only a fraction of the available information about the words and the general structure of the analysed document. In further analysis, we are going to take the full morphological information into use and expand our approach to cover more detailed text and language analysis. Additionally, we will try to extract some information out of the SGML structure of the test material. Furthermore, we plan to survey which tools are needed to make the analysis of a presumably large collection of episodes and episode rules more efficient.

References

1. R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A. I. Verkamo. Fast discovery of association rules. In *Advances in Knowledge Discovery and Data Mining*, pages 307–328. AAAI Press, 1996.
2. D. R. Cutting, D. Karger, J. Pedersen, and J. W. Tukey. Scatter/Gather: A cluster-based approach to browsing large document collections. In *Proc. of the 15th Annual Int'l ACM/SIGIR Conference*, pages 318–329, Copenhagen, Denmark, June 1992.
3. U. M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From data mining to knowledge discovery: An overview. In *Advances in Knowledge Discovery and Data Mining*, pages 1–34. AAAI Press, 1996.
4. R. Feldman, I. Dagan, and W. Klösgen. Efficient algorithms for mining and manipulating associations in texts. In *Cybernetics and Systems, Vol. II, The 13th European Meeting on Cybernetics and Systems Research*, Vienna, Austria, April 1996.
5. D. D. Lewis and K. Spärck Jones. Natural language processing for information retrieval. *CACM*, 39(1):92–101, 1996.
6. H. Mannila. Data mining: machine learning, statistics, and databases. In *Proc. of the 8th Int'l Conference on Scientific and Statistical Database Management*, pages 1–6, Stockholm, Sweden, 1996.
7. H. Mannila and H. Toivonen. Discovering generalized episodes using minimal occurrences. In *Proc. of the Second International Conference on Knowledge Discovery and Data Mining (KDD'96)*, pages 146–151, Portland, Oregon, USA, August 1996. AAAI Press.
8. G. Salton. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley, 1988.
9. F. Smadja. Retrieving collocations from text: Xtract. *Computational Linguistics*, 19(1):143–177, 1993.

This article was processed using the LaTeX macro package with LLNCS style