Using Conceptual Graphs for Text Mining in Technical Support Services

Michael Bogatyrev and Alexey Kolossoff

Tula State University,
Lenin ave. 92, 300600 Tula, Russia
okkambo@mail.ru, alexey.kolosoff@gmail.com

Abstract. Text mining problems of natural text classification and fact extraction are important in developing information systems for Technical Support Services. An approach which is based on joining acquisition of conceptual graphs and keywords search technique is presented to their solution. Conceptual graphs have been created from e-mail queries sent to Technical Support Service. Correct conceptual graphs acquired from e-mail texts represent facts and situations which become patterns to search in systems resources to resolve users problems. Experimental results of implementing proposed approach are presented.

Keywords: natural language texts classification, conceptual graphs, correctness of conceptual graphs, technical support services.

1 Introduction

Text mining strategies share many techniques such as machine learning, natural language processing, text categorization, clustering, filtering, etc. These techniques can be classified as ones which use texts words and others which use semantic models constructed from text. Two known strategies, Latent Semantic Analysis (LSA) [1] and Formal Concept Analysis (FCA) [2] illustrate that difference. LSA uses term-document matrices which describe the occurrences of terms in textual documents and has been created from documents words. FCA uses conceptual models - conceptual graphs [3] and conceptual structures (formal concept lattices) which are formal models. The mentioned strategies also have different mathematical nature: LSA is founded on geometry and statistics whereas FCA is founded on logic and algebra (the lattice theory).

Traditionally only one approach, based on keywords or formal models, is applied in industrial text mining systems. Nevertheless modern problems of textual analysis may have significant complexity and it becomes necessary to apply hybrid approaches to solve them.

In our work namely that complex problem is investigated. As the result we decided to apply keywords technique and conceptual graphs in our Text Mining system. Although each separate technique does not solve the problem, their combination produces good preliminary results.

2 Problem Statement

Technical Support Services (TSS) have been intended to help users to solve specific problems with a product - electronics, goods or software. Users send queries to TSS as natural language e-mail texts. It is needed to resolve queries and to find an appropriate decision represented as help topics, useful URLs or e-mail reply. As a rule the system's reply is prepared manually by support team using system's resources as it is shown on Fig.1.

Fig. 1. The structure of a Technical Support Service

When the number of queries significantly grows, automation of creating TSS replies becomes very important. That automation is implemented in the TSS Search Engine shown on Fig.1.

There are two basic text mining problems solved by the Search Engine. The first one is the problem of *natural text classification*. The second is the problem of *fact extraction*. These problems have some peculiarities. The query text must be classified according to various resources of the TSS. TSS database contains documentation, help topics, e-mails of queries and replies. To find an appropriate decision it is needed to refer to all these resources. The decision may exist as an example answer ready to be sent to a user or it can be constructed from separate pieces. Fact extraction problem is to find two kinds of objects in the query texts: *things* which being analyzed text is about and *situations* which took users attention.

3 General Approach to Solution

Considering the problems of natural text classification and fact extraction described in the previous section, we propose the following general approach to their solution:

1. Having a flow of user e-mail queries we nevertheless do not apply classical machine learning technique because the style and contents of queries are very individual. But it is worth to collect queries and corresponding replies in the system's database to apply them in further analysis. So, a kind of self-learning is possible in the system.
2. *Things* and *situations* described in a query are represented by *words* and *phrases*. So we need to find keywords in the query text which correspond to *terms* described in system's database texts. Since learning technique is impossible in the system, another way of keywords extraction is needed. Text filtering is standard and evidently necessary technique for e-mail texts in natural language. We apply *term filtering* to find *direct terms* corresponding to the terms described in systems database texts. For example, the *driver* word in a query in the software TSS has high probability to be the term. So a text containing this word can be classified as referring to the topic "Drivers" in the system's database texts. To implement term filtering a thesaurus as an additional resource must be created in the system.
3. Text filtering is not exhaustive technique for classification. Besides terms, a query text contains many words which can also be useful for analysis. The personal style of an author has certain representation in query text as a set of specific words and language grammar distortions (slang). Nevertheless our analysis of real queries shows that the following heuristic principle is valid: *despite the personal style, every author uses grammatically correct phrases when describes problematic situations*. Therefore semantics of these grammatically correct phrases in a query text may represent useful information about situations we need to extract. We apply conceptual graphs for modelling semantics of sentences or phrases of the text and use their concepts and relations for further analysis.

Conceptual graphs acquisition from natural language texts is the problem which has no closed solution for arbitrary texts. We assume that the following circumstances cause the success of creating conceptual graphs:

1. query texts are not long and all their sentences may be processed for acquisition in appropriate time;
2. grammatically correct phrases in the sentences produce correct conceptual graphs possibly being sub graphs in incorrect conceptual graph of the whole sentence.

The following rough criterion of correctness of conceptual graph is admissible here: *correct conceptual graph has no isolated concepts*. An *isolated concept* is a concept which has no connection to any relation.

4 System Implementation and Experimental Results

The TSS Text Mining system works according to the following stages.

1. **Text documents indexing.** All TSS documents have been indexed according to selected terms. These terms represent topics and main notes presented in system documentation. Terms are either single words or several words phrases (no more than 4 words). Term weights are calculated via the well-known tf-idf formula [4]. The TSS complex index is the only additional modification of TSS information resource realized in standard database technology (MS SQL Server).
2. **Conceptual graphs acquisition and processing.** Conceptual graphs are applied as an instrument of extracting keywords and key phrases according to the principle described below.
3. **Search relevant documents in TSS database.** Keywords and key phrases corresponding to each e-mail text and extracted by conceptual graphs processing have been used as queries for full-text search in TSS indexed database.

Consider the last two stages in some detail.

4.1 Conceptual Graphs Acquisition and Processing

We use our software [5] for conceptual graphs acquisition from natural language texts. The software is based on existing approaches of lexical, morphological and semantic analysis. *Semantic roles labeling* [6] is applied as the main instrument for constructing relations in acquisition algorithm. The acquisition algorithm works with our recently developed *controllable grammatical templates*. Using these templates, it is possible to adapt acquisition algorithm as to certain language grammar (Russian or English in the current version of the system) as to some peculiarities of concrete language. User interface has also tools for recognizing incorrect conceptual graphs.

Conceptual graphs being acquired from all sentences of a query text are applied to detect keywords and key phrases. As a rule, incorrect conceptual graphs indicate that there is no useful information in processed text. For example, conceptual graph acquired from the "Thanks in advance" phrase, $G1 = \{[advance:] [thank:]\}^1$ is incorrect since it has no relation. TSS user can handle any acquired graph by using interface tools including visualization. That helps finding possibly valid keywords in incorrect conceptual graphs.

All acquired correct conceptual graphs considered as potential source of keywords and key phrases for the next search. Concepts connected with the *agent* relation may represent *terms* and have been picked as keywords. Some term may consist of several words, for example Remote Agent Service. The relation *genitive* in its graph $G2 = \{[remote*a:] [service*b:] [agent*c:] (genitive?b?c) (attribute?c?a)\}$ indicates that Agent Service is the single whole. All graphs having simple structure with *genitive* and *attribute* relations are considered as sources of keywords and key phrases.

¹ Here we use the CGIF format [3] for representing conceptual graphs.

It is known that relations in conceptual graphs have linguistic meaning at first. But some of them can directly indicate a *situation*. That is the *location* relation and it is also considered as key phrases indicator. For example, it is illustrated by the phrase "stop on error" and its conceptual graph $G3 = \{[error*a:"][stop*b:"](location?b?a)\}$.

4.2 Search Relevant Documents

All keywords and key phrases extracted by conceptual graphs processing are then treated as queries for full-text search in TSS indexed database. For each e-mail text they constitute a query vector. We devoted special attention to applying LSA search strategy for such queries. We also compared it with other methods which use ranking functions of Okapi BM25 [7], SQL Server iFTS [8] and ranking function of Google. An experiment was conducted on the textual database with more than 7000 help topics belonging to online help systems of three different software products. Employees of the products vendor company were asked to rate (from 1 to 4) the quality of search results (including their ranking) for top 10 most popular queries retrieved from the users queries statistics. A short summary is presented in the table below.

Table 1. Search results ratings for 10 most popular queries

Search Query	LSA	Okapi BM25	SQL Server iFTS	Google
working with grids	3	1	4	4
load testing	4	2	3	3
web testing	4	3	2	3
Remote Agent Service	4	2	2	4
name mapping template	4	3	3	2
stop on error	3	2	4	4
object not found	3	4	4	3
UI Automation Silverlight	4	4	4	3
testing flash applications	4	4	3	4
web service testing	4	4	3	4
Total (max. 40):	37	29	32	34

Here Google refers to Google web search in the online help systems of three different software products (with URLs filtering). As one can see from the table, the LSA search gives the best result. We can explain it by the following informal conclusion: *Latent Semantic Analysis pretends to detect texts which are semantically similar*. Conceptual graphs processing produces a set of keywords which are semantically connected. So, the query produced with conceptual graphs has certain *portion of semantics* which can *resonate* with semantics of TSS documents. It seems that LSA, according to its mathematical nature, is namely that method which can find such peculiar *semantic resonance* of texts.

5 Conclusion and Future Work

Hybrid approach to textual analysis in Technical Support Services is presented. It is based on using conceptual graphs for extracting keywords and key phrases from query text and applying standard full-text search technique. Experimental results show that conceptual graphs represent a valid tool for extracting keywords and key phrases since this tool provides semantic connection between words in key phrases. Conceptual graphs technique usually produces less number of keywords and key phrases than there are in a query text that shortens the time for further search.

Future development of presented technology is planned on the way of creating additional information resource in the TSS system. This resource will be in the form of *conceptual lattice*. Having conceptual lattices as system's information resource, we will apply conceptual graphs as immediate queries in search strategy according to the principles of FCA.

References

1. Landauer, T., Foltz, P.W., Laham, D.: Introduction to Latent Semantic Analysis. *Discourse Processes* 25, 259–284 (1998)
2. Ganter, B., Wille, R.: *Formal Concept Analysis. Mathematical Foundations.* Springer, Heidelberg (1999)
3. Sowa, J.F.: *Conceptual Structures: Information Processing in Mind and Machine.* Addison-Wesley, London (1984)
4. Salton, G., McGill, M.J.: *Introduction to modern information retrieval.* McGraw-Hill, New York (1983)
5. Bogatyrev, M.Y., Mitrofanova, O.A., Tuhtin, V.V.: Building Conceptual Graphs for Articles Abstracts in Digital Libraries. In: *Fourth Conceptual Structures Tool Interoperability Workshop (CS-TIW 2009) at 17th International Conference on Conceptual Structures (ICCS 2009)*, Moscow, pp. 50–57 (2009)
6. Gildea, D., Jurafsky, D.: Automatic labeling of semantic roles. *Computational Linguistics* 28, 245–288 (2002)
7. Robertson, S., Walker, S., Jones, S., et al.: Okapi at TREC-3. In: *Proceedings of the Third Text Retrieval Conference (TREC 1994)*, Gaithersburg, USA (1994)
8. Langit, L., Goff, K., Mauri, D., Malik, S.: *Smart Business Intelligence Solutions with Microsoft SQL Server 2008.* Microsoft Press (2009)