Categorization of Czech written documents using WEBSOM methods

Roman Mouček, and Pavel Mautner,
Department of Computer Science and Engineering,
Faculty of Applied Sciences, University of West Bohemia,
Pilsen, Czech Republic

Abstract— **The method called WEBSOM was designed for automatic processing and categorization of English and Finnish written documents and the following information retrieval in these documents. We applied this method (based on two layer architecture) to categorization of Czech written documents. Our research was focused on the syntactic and semantic relationship within word categories of word category map (WCM) and on the results provided by document category map (DCM) with respect to the content of WCM. The document classification system was tested on a subset of 100 documents (manual work was necessary) from the corpus of Czech News Agency documents. The result confirmed that not only WEBSOM method but also humans have problems with natural language semantics and determination of semantic domains from word categories.**

I. Introduction

Nowadays, finding relevant information from the vast material in the electronic form (mostly available in the web) is a difficult and time consuming task. Therefore, an enormous scientific and commercial effort is paid to development of new methods and approaches, which help people to find and refer to (or extract) required information in accessible electronic sources. Some approaches try to involve as many aspects of natural language as possible whereas some of them are strictly limited by elaborated domain or processed language aspects.

However, the following question is rarely asked: which approaches are useful and which of them people will really use. We got used to enter key words using search engines and go through a set of returned documents to find the right one. Since entering key words does not limit or annoy people in general, scanning a large set of documents is a tiring and unpleasant work.

II. Semantic web

Inability to find required information in documents properly led to idea of semantic web. Semantic web provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries [1]. This idea is based on common formats for interchange of data (not interchange of documents) from various sources. It supposes that documents are designed for humans to read, not for computer programs to manipulate them meaningfully. It is believed that computers have no reliable way to process the semantics of documents.

Searching for documents means to work with semantics of natural language. The processing of natural language is still a serious problem for computer systems and applications. Natural language gives freedom to express a real word in various ways; to choose between synonyms, to use different styles, emphasis, different levels of abstractions, anaphoric and metaphoric expressions, etc. Then the idea of semantic web corresponds to the idea that there is no reliable way to process natural language semantics. The second idea of semantic web is a language, which records relationship between data and objects in real world; this issue is out of scope of this article.

There are two necessary conditions to succeed in the next development of semantic web. However, acceptance of these conditions is very indeterminate, because they relate more to common human behavior then to technical solutions. The first condition is a general agreement of people working in the elaborated domain because only widely accepted domain ontology can be respected and used. The second condition deals with the human ability and willingness to organize data respecting domain ontology; people naturally write documents. It is clear that both conditions can be hardly solved technically.

III. Document organization

The actual progress in the development of semantic web leads to the suggestion that a lot of people will prefer writing documents in the future. Then there is a question if we can help people with document organization, eventually with parsing techniques, which extract relevant data from previously organized documents. We focus on the first step of this process: organization of a set of large documents.

We suppose a common scenario of searching for relevant documents. This scenario is based on asking a question (query including keywords from a domain area), and the following matching of the keywords with document content. One possibility to accelerate information retrieval in large document collections is a categorization of documents into classes with similar content. Based on the keywords

Manuscript received September 1, 2008. This work was supported by Grant no. 2C06009 Cot-Sewing.

Roman Mouček is with Department of Computer Science and Engineering, Faculty of Applied Sciences, University of West Bohemia, Pilsen, Czech Republic (e-mail: moucek@kiv.zcu.cz).

Pavel Mautner is with Department of Computer Science and Engineering, Faculty of Applied Sciences, University of West Bohemia, Pilsen, Czech Republic (e-mail: mautner@kiv.zcu.cz).

included in the query, we suggest that it is possible to estimate the document domain and to search only in the documents from this domain. In this case, search time and a list of returned documents are strongly reduced.

In the past, some methods of document classification into domains were developed. These methods usually require a suitable representation of the stored documents. Documents are most often represented by the vector model [2]. The main problem of this representation is the large vocabulary of document collection and the high dimensionality of document vectors. Then methods for reducing this dimensionality have to be used. The common technique called Latent Semantic Indexing uses singular value decomposition (SVD). The resulting latent representation is reduced by discarding the least significant elements.

Grouping similar items together is a technique used by methods based on word clustering. Documents are than represented as histograms of word clusters. One from various approaches to word clustering is the self-organizing map, which is based on distribution of words in their immediate context.

IV. SELF-ORGANIZING FEATURE MAP AND WEBSOM METHOD

A. *Self-organizing feature map*

Self-organizing feature map (SOFM) has been developed by T. Kohonen and it has been described in several research papers and books [3], [4]. The purpose of SOFM is to map a continuous high-dimensional space into discrete space of lower dimension (usually 1 or 2). The map contains one layer of neurons, arranged to a two-dimensional grid, and two layers of connections. In the first layer of connections, each neuron is fully connected (through weights) to all feature vector components. The computations are feed-forward in the first layer of connection: the network computes the Euclidean distance between the input feature vector and each of the neuron weight vectors. The second layer of connections acts as a recurrent excitatory/inhibitory network. The aim of this network is to implement the winner-take-all strategy, i.e. only one neuron is selected and labeled as the best matching unit (BMU). Detailed description of Kohonen self organizing feature map and training algorithm can be found in [3], [4].

B. *WEBSOM architecture*

WEBSOM method [5] is based on SOFM. This method was designed for automatic processing and categorization of arbitrary English and Finish written documents accessible on internet and the following information retrieval in these documents. Like WEBSOM, our classifier is based on two layer architecture (Fig. 1).

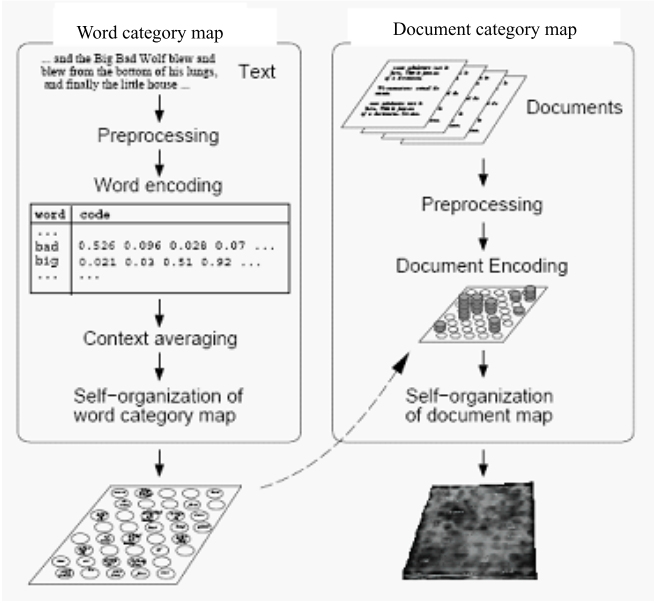

Fig. 1. Architecture of WEBSOM (from [5])

The first layer processes the input feature vector representing the document words and creates the word category map (WCM). The second layer (document category map - DCM) processes the output from WCM and creates the clusters corresponding to document categories. Both layers are based on SOFM.

C. *Document preprocessing*

Each document in a collection can be initially preprocessed using various techniques to reduce the computational load: lemmatization is done, non-textual information is removed, numerical expressions are replaced by textual forms, words occurring only a few times or common words not distinguishing document topics are removed.

D. *Word category map*

Word category map is supposed as "self-organizing semantic map" [6] because describes relation of words based on their averaged contexts. The word category map is trained by context vectors (input feature vector, which includes word context), which are created by the following procedure:

1. The unique random n-dimensional v_i real vector (called representing vector) is assigned for $i-th$ word in a domain dictionary ($i \epsilon (1,n)$, n is a number of words in a domain dictionary)
2. The given text documents are searched for all occurrences of the word represented by vector v_i
3. The context, in which the word v_i occurs in documents, is found, i.e. the immediately preceding and succeeding words of the word v_i in all documents are found and average value p_i (or n_i) of all preceding (or succeeding) representing vectors of the word v_i are evaluated .
4. The context vector cv_i of the word represented by v_i is created from p_i, v_i, and n_i values:

$$cv_i = \begin{bmatrix} p_i \\ \varepsilon v_i \\ n_i \end{bmatrix},$$

where ε is a weight of representing vector v_i of the word i.

It is suggested that the words occurring in the similar context in the given document will have a similar representing vector v_i and they will also belong to the same word category.

In Fig. **2** we can see an example of the word category map trained by the words from the set of 100 documents. We can see that some map units respond to the words from certain syntactic categories (e.g. verbs, proper nouns etc.), whereas other units respond to the words from various syntactic categories (in detail in V.C).

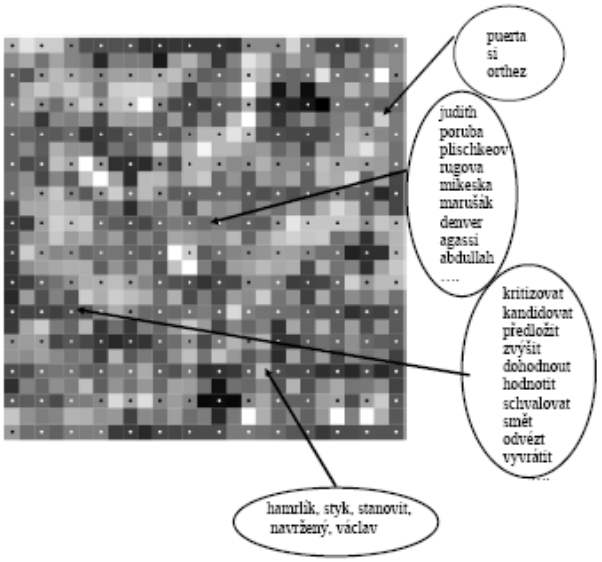

Fig. 2. Example of word category map

E. WEBSOM architecture

Document category map (DCM) classifies the input document to given class. The size of input vector of DCM, i.e. the word category vector, is the same as the number of neurons in WCM. Each component of this vector represents a frequency of occurrence of the given word category in the input document. It is assumed that documents with the similar or the same content will have the similar word category vector. Based on this assumption, it is possible to use these vectors for training of DCM. Since a Kohonen map is unsupervised learning paradigm, only the clusters of similar documents are created during the training. The given categories are assigned to these clusters afterwards.

V. EXPERIMENTS AND RESULTS

A. Document collection

The document classification system described in the previous sections was tested on the corpus of Czech News Agency documents. Generally, there were 7600 documents from 6 domains, containing 145766 words (stop words were removed). SOM-PAK [7], SOM toolbox [8] and own implementation of SOFM have been used for Kohonen map simulation. Both layers, WCM and DCM, were trained by the sequential training algorithm. The documents were classified by hand into 6 classes according to the document topics.

Our main task was to examine syntactic and semantic relationships within the word categories of WCM. Basic syntactic categories (nouns, adjectives, etc.) can be easily detected automatically, whereas semantic relationships have to be marked manually. Thus we worked only with a limited number of documents to manage this tiring and time consuming process of word categories evaluation. Finally we randomly selected 100 documents from the document collection. These documents contained 7421 different words after lemmatization and stop words removing.

B. Word category map

All words from the selected collection of 100 documents appeared in WCM; no threshold was applied to frequency of word occurrence, because a lot of words, which occurred only once, had impact on document semantics. The size of WCM was 437 neurons (19 x 23 grid), i.e. on average 17 words were placed into each category. The dimension of context vector cv_i was set to 60; ε was set to 0.2.

C. Syntactic evaluation of word categories

The syntactic evaluation of word categories was done by the following process. Distribution of words into three basic word classes (nouns, adjectives, verbs) within word categories was observed. The fourth class named “others” was settled for all other word classes. Word categories contained in total 55.0% of nouns, 19.2% of adjectives and 13.5% of verbs (document collection contained a large number of geographical names and proper nouns). Fig. 3 represents distribution of word classes within word categories. It is obvious that adjectives and verbs usually create up to 20% of words in the word category, while 280 word categories contain between 40% and 60% of nouns. Because document collection contains a higher number of nouns, this word distribution corresponds to standard distribution of investigated word classes within text documents.

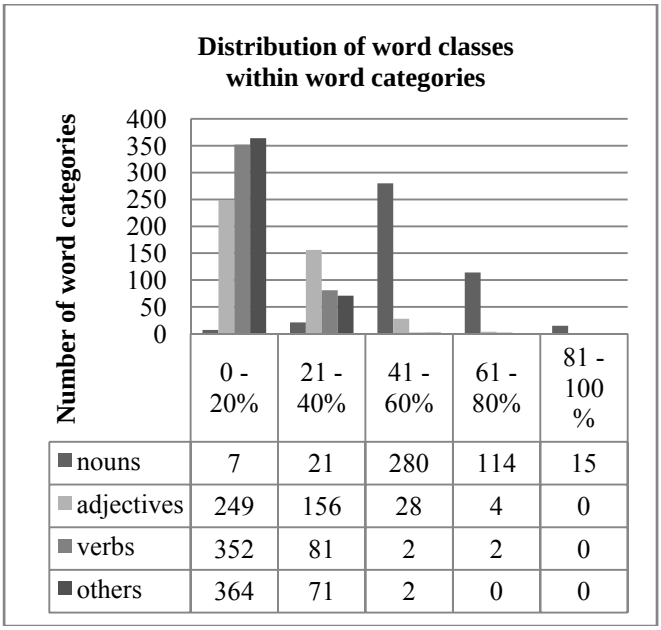

Fig. 3 Distribution of word classes (nouns, adjectives, verbs) within word categories, the percentage shares (five groups) of the word class within word category is presented on X-axis, each column indicates the number of word categories, in which the given word class appears with given percentage share, e.g. there are seven word categories in which the percentage share of nouns is up to 20%, otherwise there are 15 word categories, where the percentage share of nouns is greater than or equal to 81%.

D. Semantic evaluation of word categories

Semantic content of word categories can be hardly evaluated automatically. It is not possible to compare word categories e.g. to WordNet sets and expect some level of similarity. Thus semantic processing of word categories was done by hand. We used the following method: 4 students were asked to go through 437 word categories three times in three weeks (the week break was necessary to ensure that students forgot the content of word categories from the previous task). Each round they got a different task concerning semantics of word categories.. All tasks were time limited (1 second of reading time for each five words in a word category). The response time was different according to task complexity.

The first task was to resolve if the given word category represents a semantic domain; the answer was simply yes or no. The response time was 3 seconds for each category. The results are shown in Table 1.

Student/ Answer	Yes	No
1	81,50%	18,5%
2	68,00%	32,0%
3	57,90%	42,1%
4	71,20%	28,8%
Average	**69,65%**	**30,35%**

Table 1 Responses of students to the question: Does a given word category represent a semantic domain?

The percentage share of word categories considered as semantic domains was 69.65%, but there was a significant difference between students.

The second task was to go through the set of word categories and name each category, which is supposed to be a semantic domain. The response time was 6 seconds for each category. The results are shown in Table 2.

Student/Category name	Yes	No
1	55,1%	44,9%
2	35,2%	64,8%
3	29,3%	70,7%
4	25,9%	74,1%
Average	**36,38%**	**63,63%**

Table 2 Responses of students to the task: If a given word category represents a semantic domain, write its name (Answer 'Yes' means that word category was named, answer 'No' means none or senseless answer).

There is obvious that students had problems to name a semantic category even in the case they marked it as a semantic domain in the previous task.

The third task was to classify a given word category to four predefined domains (sport, politics, legislation, and society). Students had a possibility to answer that a given word category did not match any from the predefined set of domains. The response time was 3 seconds. The results are available in Table 3.

Student/ Predefined Category name	Yes	No
1	67,5%	32,5%
2	71,4%	28,6%
3	52,9%	47,1%
4	65,4%	34,6%
Average	**64,30%**	**35,70%**

Table 3 Responses of students to the task: Classify a given word category to the predefined domains (sport, politics, legislation, and society). If a given word category does not match any from predefined domains, give no answer (Answer 'Yes' means classification in a domain from the predefined set of domains).

Students classified 64.30% of word categories into a domain selected from the predefined set of domains.

E. Document Category Map

Document Category Map (the second layer of WEBSOM architecture) consisted of 9 neurons arranged to 3 x 3 grid. The map receives and processes the vectors from the output of WCM convolved by Gaussian mask. Then it produces the output which corresponds to the category of the given input document.

The results of document classification using DCM are presented in Table 4.

SOM categories output neuron number	Number of documents for category				Total number of documents
	Sport	Politics	Legislation	Society	
1	1	7	1	0	9
2	18	4	2	1	25
3	8	1	0	0	9
4	9	11	3	0	23
5	0	0	0	0	0
6	4	11	5	3	23
7	0	0	0	0	0
8	0	0	0	0	0
9	0	9	1	1	11
Total number	**40**	**43**	**12**	**5**	**100**

Table 4 Results from DCM (100 documents).

We can see that the results of categorization are not too convincing. They are strongly affected by the output of WCM, but we can hardly find a meaningful criterion to compare the results of DCM with the results obtained from students. We can only express an idea that not only WEBSOM method but also humans have problems with document semantics and document classification.

VI. Conclusion

The results obtained by application of WEBSOM method to a collection of Czech written documents confirmed a general problem connected with document semantics (i.e. with semantics of natural language) and document classification. Not only WEBSOM method but also humans had problems with classification of word categories into semantic domains. Moreover, there were significant differences between students undergoing the semantic experiment. However, an effort to interpret these differences would lead only to a speculative result. It is possible that the obtained results correspond to an idea that the semantics of natural language cannot be processed with computer in any reliable way.

References

[1] Semantic Web (2008, August). Available: http://www.w3.org/2001/sw
[2] C. D. Manning, P. Raghavan, H. Schütze, "Introduction to Information Retrieval", preliminary draft, Cambridge University Press, 2007.
[3] T. Kohonen, "Self-organizing map", Berlin Heidelberg: Springer-Verlag, 2001.
[4] L. V. Fausset:"Fundamentals of neural networks", Prentice Hall, Engelwood Cliffs, NY, 1994.
[5] S. Kaski, T. Honkela, K. Lagus, T. Kohonen, "WEBSOM – Self-Organizing Maps of Document Collections", Neurocomputer, 1998, pp. 101 – 117.
[6] H. Ritter, T. Kohonen, "Self-organizing semantic maps", Biological Cybernetics, 1989, pp. 61:241-254.
[7] T. Kohonen, J. Hynninen, J. Kangas, J. Laaksonen, SOM-PAK, The self-organizing map program package, 1996.
[8] J. Vesanto, J. Himberg, E. Alhoniemi, J. Parhankangas, SOM Toolbox for Matlab, 2000.