Serbian Text Categorization Using Byte Level n-Grams

Jelena Graovac
Faculty of Mathematics, University of Belgrade
Studentski trg 16
11000 Belgrade, Serbia
jgraovac@matf.bg.ac.rs

ABSTRACT

This paper presents the results of classifying Serbian text documents using the byte-level n-gram based frequency statistics technique, employing four different dissimilarity measures. Results show that the byte-level n-grams text categorization, although very simple and language independent, achieves very good accuracy.

Keywords

N-gram, text categorization, Serbian

1. INTRODUCTION

Text categorization is the task of classifying unlabelled natural language documents into a predefined set of categories. The increasing volume of available documents in the World Wide Web has turned the document indexing and searching more and more complex. This issue has motivated the development of automated text and document categorization techniques that are capable of automatically organizing and classifying documents. There are many different techniques for solving text categorization problem with very good results, including *Naive Bayes* classifier, *Decision trees*, *Decision rule classifiers*, *Regression methods*, *Rocchio*'s method, *Neural networks*, *K nearest neighbors*, *Support vector machines* etc.[4] Most of them have concentrated on English text documents, and therefore are not applicable to documents written in some other language such as Serbian.

Serbian language has several properties that significantly influence text categorization: the use of two alphabets (Cyrillic or Latin alphabet), phonologically based orthography, the rich morphological system, free word order of the subject, predicate, object and other sentence constituents, special placement of enclitics and complex agreement system.[6] All these characteristics make the preprocessing steps, such as feature extraction and feature selection, to be more complex. There is a need for a dictionaries, finite-state transducers for the description of the interactions between text and dictionary and other complex natural language processing tools.[6]

This paper presents the results of Serbian text categorization using a technique which is language independent and very simple, that overcomes the above difficulties. This technique is based on byte-level n-gram frequency statistics method for documents representation and *K nearest neighbors* machine learning algorithm for categorization process. It is derived from Kešelj's[3] method to solving the authorship attribution problem by using n-grams and some Tomović's ideas from [5], where the problem of automated categorization of genome isolates was examined. Kešelj defines an author profile as an ordered set of pairs (x_1, f_1), (x_2, f_2),..., (x_L, f_L) of the L most frequent byte n-grams x_i and their normalized frequencies f_i. The authorship is determined based on the dissimilarity between two profiles, comparing the most frequent n-grams. Based on this work, a wide range of dissimilarity measures are introduced in [5]. This technique, including all these measures and one newly introduced measure, has been tested in the work [2], to solve the problem of text categorization in English, Chinese and Serbian. In this paper is presented results of categorization only Serbian text documents. The same technique and the same corpus is used as in the [2] except that the categorization is carried out at five rather than three classes and only dissimilarity measures from [3] and [5] were examined. Since it is based on byte level n-grams technique do not need any text preprocessing or higher level processing, such as tagging, parsing, feature selection, or other language-dependent and non-trivial natural language processing tasks.[3] The approach is also tolerant to typing, spelling and grammatical errors and word stemming is got essentially for free.[1]

Overview of the paper: Some background information about n-grams are presented in Section 2. Section 3 describes methodology for categorization of text documents used in this paper. This section also presents several dissimilarity measures, the data set used for text categorization and the set of evaluation metrics that are used to assess the performance of this technique. Section 4 reports on experimental results and shows comparison of dissimilarity measures. Finally, Section 5 concludes the paper.

2. N-GRAMS

Given a sequence of tokens $S = (s_1, s_2, ..., s_{N+(n-1)})$ over the token alphabet $\mathcal{A}$, where N and n are positive integers, an n-gram of the sequence S is any n-long subsequence of consecutive tokens. The i^{th} n-gram of S is the sequence $(s_i, s_{i+1}, ..., s_{i+n-1})$.[5]

For example, if $\mathcal{A}$ is the English alphabet, and l string on alphabet $\mathcal{A}$, l = ”life is a miracle” then 1-grams are: l,i,f,e,_,s,a,m,r,c; 2-grams are: li,if, fe, e_, _i, is, s_, _a, ...; 3-grams are: lif, ife, fe_, e_i, ...; 4-grams are: life, ife_, fe_i,

BCI'12, September 16–20, 2012, Novi Sad, Serbia.

Copyright © 2012 by the paper's authors. Copying permitted only for private and academic purposes. This volume is published and copyrighted by its editors.

Local Proceedings also appeared in ISBN 978-86-7031-200-5, Faculty of Sciences, University of Novi Sad.

... and so on. The underscore character ("_") is used here to represent blanks. For $n \leq 5$ Latin names are commonly used for n-grams (e.g., trigrams) and for $n > 5$ numeric prefixes are common (e.g., 6-grams).[5]

The use of n-gram models and techniques based on n-gram probability distribution in natural language processing is a relatively simple idea, but it turned out to be effective in many applications.[3] Some of them are text compression, spelling error detection and correction, information retrieval, language identification, authorship attribution. It also proved useful in domains not related to language processing such as music representation, computational immunology, protein classification etc.

The term n-gram could be defined on word, character or byte level. In the case of Latin-alphabet languages, character-level and byte-level n-gram models are quite similar according to the fact that one character is usually represented by one byte. The only difference is that character-level n-grams use letters only and typically ignore digits, punctuation, and whitespace while byte-level n-grams use all printing and non-printing characters.[2]

3. METHODOLOGY AND DATA

The technique used in this paper is based on calculating and comparing profiles of N-gram frequencies. First, profiles on training set data that represent the various categories are computed. Then the profile for a particular testing document that is to be classified is computed. Finally, a dissimilarity measure between the document's profile and each of the category profiles is computed. The category whose profile has the smallest value of dissimilarity measure with the document's profile is selected. Detailed text categorization procedure is presented in [2].

Dissimilarity measures: Dissimilarity measure d is a function that maps the Cartesian product of two sets of sequences $\mathcal{P}_1$ and $\mathcal{P}_2$ (defining specific profiles) into the set of positive real numbers. It should reflect the dissimilarity between these two profiles and it should meet the following conditions:[5]

- $d(\mathcal{P}, \mathcal{P}) = 0$;
- $d(\mathcal{P}_1, \mathcal{P}_2) = d(\mathcal{P}_2, \mathcal{P}_1)$;
- the value $d(\mathcal{P}_1, \mathcal{P}_2)$ should be *small* if $\mathcal{P}_1$ and $\mathcal{P}_2$ are *similar*;
- the value $d(\mathcal{P}_1, \mathcal{P}_2)$ should be *large* if $\mathcal{P}_1$ and $\mathcal{P}_2$ are *not similar*.

The last two conditions are informal as the notion of similarity (and thus the dissimilarity) is not strictly defined.

In this paper is used four dissimilarity measures. First of them is measure used by Kešelj[3] and it has a form of relative distance:

$$d(\mathcal{P}_1, \mathcal{P}_2) = \sum_{n \in profile} \Big(\frac{2 \cdot (f_1(n) - f_2(n))}{f_1(n) + f_2(n)} \Big)^2 \qquad (1)$$

where $f_1(n)$ and $f_2(n)$ are frequencies of an n-gram n in the author profile $\mathcal{P}_1$ and the document profile $\mathcal{P}_2$.

Other three measures are measures from [5] that performed best on considered data set:

$$d_1(\mathcal{P}_1, \mathcal{P}_2) = \sum_{n \in profile} \frac{2 \cdot (f_1(n) - f_2(n))}{f_1(n) + f_2(n)} \qquad (2)$$

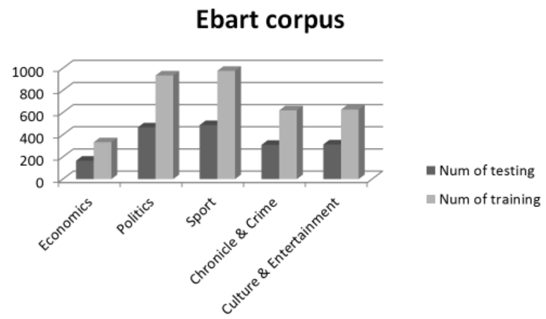

Figure 1: Category distribution of Ebart corpus.

$$d_2(\mathcal{P}_1, \mathcal{P}_2) = \sum_{n \in profile} \Big(\frac{\sqrt{2} \cdot |f_1(n) - f_2(n)|}{\sqrt{f_1(n)^2 + f_2(n)^2}} \Big)^2 \qquad (3)$$

$$d_3(\mathcal{P}_1, \mathcal{P}_2) = \sum_{n \in profile} \frac{\sqrt{2} \cdot |f_1(n) - f_2(n)|}{\sqrt{f_1(n)^2 + f_2(n)^2}} \qquad (4)$$

In this paper is presented categorization of text documents in Serbian. For this purpose, Ebart corpus is used.

Ebart: Ebart[1] is the largest digital media corpus in Serbia with almost one million news articles from daily and weekly newspapers archived by early 2003 onwards. The current archive is classified into thematic sections following the model of regular newspaper columns. In this paper a subset of the Ebart corpus is taken into consideration - articles from the Serbian daily newspaper "Politika" that belong to columns Sport, Economics, Politics, Chronicle & Crime and Culture & Entertainment published from 2003 to 2006. There are 5235 such articles. This data set was split into the training and testing set in the ratio 2 : 1. Fig. 1 shows the distribution of this corpus.[2]

Performance evaluation: The standard metrics for the categorization performance evaluation is considered, namely micro- and macro-averaged F_1 measures. As usual, micro-averaged F_1 measure is computed for all documents over all document categories. Macro-averaged F_1 measure represents the averaged value determined from F_1 values computed for each classification category separately. The standard definition of these measures and measures of *precision* and *recall* can be found, e.g., in [2].

4. RESULTS

This section presents the experimental results of text categorization obtained for Ebart corpus on five categories: Sport, Economics, Politics, Chronicle & Crime and Culture & Entertainment. No preprocessing is done on texts, and a simple byte n-grams representation is used, treating text documents simply as byte sequences. Only the measures selected from [3] and [5] that give the best results on the Ebart corpus are considered: d, d_1, d_2 and d_3 (see Sec. 3). For producing n-grams and their normalized frequencies the software package *Ngrams* written by Kešelj[3] is used. In

[1] Ebart current archive is available at *http://www.arhiv.rs*

[2] All experimental data can be obtained on request from the author.

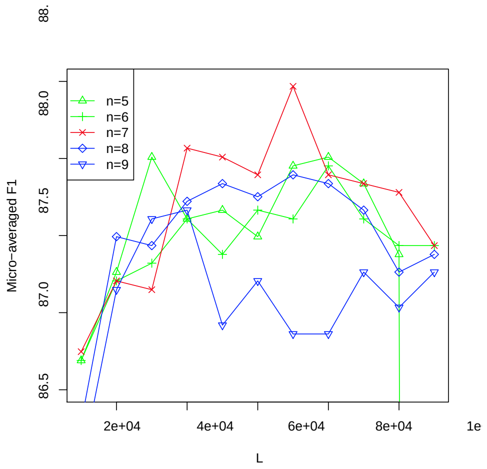

Figure 2: Micro-averaged F_1 in percentages for Ebart corpus, for different values of n-gram size n and dissimilarity measure d.

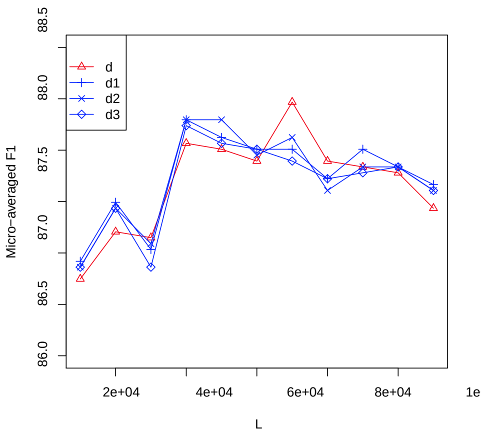

Figure 4: Micro-averaged F_1 in percentages for Ebart corpus, for $n = 7$ and different dissimilarity measures.

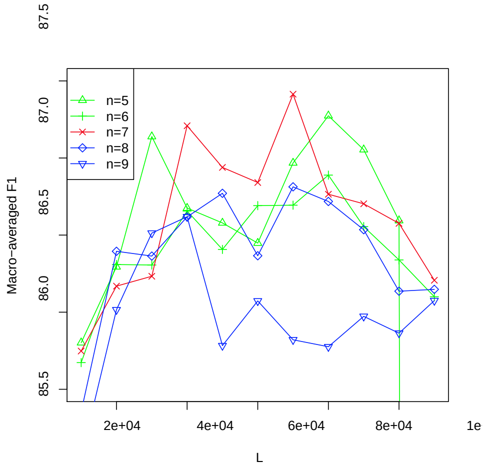

Figure 3: Macro-averaged F_1 in percentages for Ebart corpus, for different values of n-gram size n and dissimilarity measure d.

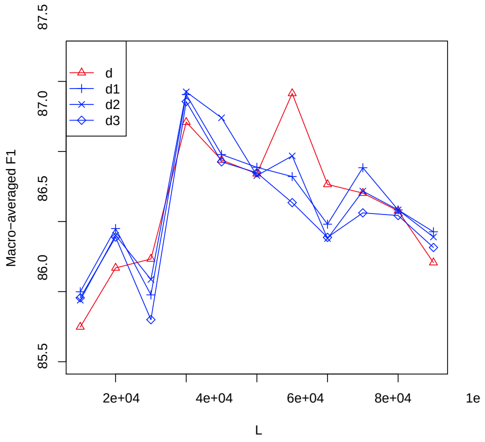

Figure 5: Macro-averaged F_1 in percentages for Ebart corpus, for $n = 7$ and different dissimilarity measures.

the process of separating testing from training documents and in the process of categorization, the software package *NgramsClassification*[3] is used.

One of the most important question in the byte n-gram categorization is what are the values of n and L that produce the best results. To give an answer to this question, the accuracy (micro- and macro-averaged F_1) of the technique was tested for all values of n-gram size n and the profile length L that makes sense to do.

Fig. 2 and 3 shows graphical representation of this extensive set of experiments taking into account only a few values of n for which the highest accuracy is achieved and only dissimilarity measure d (all other measures achieve the maximum accuracy for the same value of n as d). It can be seen that the maximum values for micro- and macro-averaged F_1 measures are reached for $n = 7$. For that particulary value of n, comparison between measures d, d_1, d_2 and d_3 is performed. The results of these experiments are shown in Fig. 4 and 5.

Additionally, Fig. 6 presents micro- and macro-averaged F_1 values for the best results for each measure. All these results show that the all introduced measures achieves comparable results.

Quality of classification is assessed also using a confusion matrix, i.e., records of correctly and incorrectly recognized documents for each category. Table 1 give the confusion matrix for the experiments using the 70000 most frequent 7-grams and dissimilarity measure d. It shows the information about actual and predicted categorizations done by our system. Each column of the matrix represents the number of documents in a predicted class, while each row represents the number of documents in an actual class. The diagonal elements show the number of correct classified documents, and the off-diagonal elements show the number of wrongly classi-

[3] Source code can be obtained on request from the author.

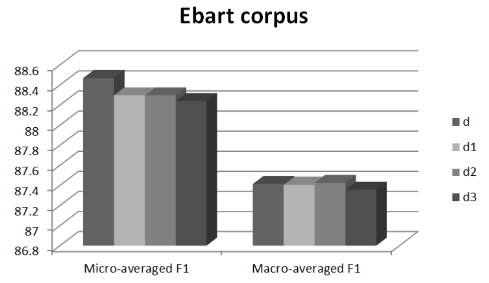

Figure 6: Best micro- and macro-averaged F_1 in percentages for Ebart corpus, and different measures.

Table 1: Confusion Matrix for Classification of Ebart Corpus

Categories	Economy	Politics	Sport	H&C	C&E
Economy	140	22*	0	3	1
Politics	12*	413	3	35*	4
Sport	3	9	466	8	2
H&C	14	61*	0	229	5
C&E	3	9	5	2	294

fied documents. The main reason for some wrongly classified documents comes from the similarities between categories in real world. For example, the category "Chronicle & Crime" and the category "Politics" are close to each other. The same stands for "Politics" and "Economy". In the Table 1, the numbers with a label of "*" are the numbers of documents wrongly classified in a category that is close to the correct category.

5. CONCLUSION AND FUTURE WORK

In this paper is presented the results of classifying Serbian data set using a new variant of a document categorization approach based on byte-level n-grams, including all printing and non-printing characters. The approach relies on a profile document representation of restricted size and a very simple algorithm for comparing profiles. It provides an inexpensive and effective way of classifying documents.

Dissimilarity measures are subject to further investigation and improvement, as well as categorization methods themselves. The plan is to compare the results obtained by presented method with results of other supervised methods on the same data sets. This method, being based on a sequence of bytes, is applicable to different domains and problems and will be further tested on specific corpora such as bioinformatics and multi-lingual corpora and on tuning Internet search engines.

6. ACKNOWLEDGEMENTS

The work presented has been financially supported by the Ministry of Science and Technological Development, Republic of Serbia, through Projects No. III47003 and No. 174021. The author is grateful to prof. Gordana Pavlović-Lažetić for her unfailing support and supervision of my research.

7. REFERENCES

[1] W. B. Cavnar and J. M. Trenkle. N-gram-based text categorization. In *In Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175, 1994.

[2] J. Graovac. A variant of n-gram based language-independent text categorization. *Submitted*, 2012.

[3] V. Keselj, F. Peng, N. Cercone, and C. Thomas. N-gram-based author profiles for authorship attribution. In *In Proceedings of the Pacific Association for Computational Linguistics*, pages 255–264, 2003.

[4] F. Sebastiani and C. N. D. Ricerche. Machine learning in automated text categorization. *ACM Computing Surveys*, 34:1–47, 2002.

[5] A. Tomovic and et al. N-gram-based classification and unsupervised hierarchical clustering of genome sequences. In *Computer Methods and Programs in Biomedicine*, pages 137–153, 2006.

[6] D. Vitas, C. Krstev, I. Obradovic, L. Popovic, and G. Pavlovic-lazetic. An overview of resources and basic tools for processing of serbian written texts. In *In Proc. of the Workshop on Balkan Language Resources, 1st Balkan Conference in Informatics*, pages 97–104, 2003.