Fragments and Text Categorization

Jan Blaťák and **Eva Mráková** and **Luboš Popelínský**
Knowledge Discovery Lab
Faculty of Informatics, Masaryk University
602 00 Brno,
Czech Republic
{xblatak, glum, popel}@fi.muni.cz

Abstract

We introduce two novel methods of text categorization in which documents are split into fragments. We conducted experiments on English, French and Czech. In all cases, the problems referred to a binary document classification. We find that both methods increase the accuracy of text categorization. For the Naïve Bayes classifier this increase is significant.

1 Motivation

In the process of automatic classifying documents into several predefined classes – text categorization (Sebastiani, 2002) – text documents are usually seen as sets or bags of all the words that have appeared in a document, maybe after removing words in a stop-list. In this paper we describe a novel approach to text categorization in which each documents is first split into subparts, called *fragments*. Each fragment is consequently seen as a new document which shares the same label with its source document. We introduce two variants of this approach – `skip-tail` and `fragments`. Both of these methods are briefly described below. We demonstrate the increased accuracy that we observed.

1.1 Skipping the tail of a document

The first method uses only the first X sentences of a document and is henceforth referred to as `skip-tail`. The idea behind this approach is that the beginning of each document contains enough information for the classification. In the process of learning, each document is first replaced by its initial part. The learning algorithm then uses only these initial fragments as learning (test) examples. We also sought the minimum length of initial fragments that preserve the accuracy of the classification.

1.2 Splitting a document into fragments

The second method splits the documents into fragments which are classified independently of each others. This method is henceforth referred to as `fragments`. Initially, the classifier is used to generate a model from these fragments. Subsequently, the model is utilized to classify unseen documents (test set) which have also been split into fragments.

2 Data

We conducted experiments using English, French and Czech documents. In all cases, the problems referred to a binary document classification. The main characteristics of the data are in Table 1. Three kinds of English documents were used:

20 Newsgroups[1] (202 randomly chosen documents from each class were used. The mail header was removed so that the text contained only the body of the message and in some cases, replies)

Reuters-21578, Distribution 1.0[2] (only documents from `money-fx`, `money-supply`, `trade` classified into a single class were chosen). All documents marked as BRIEF and UNPROC were removed. The classification tasks involved `money-fx+money-supply` vs. `trade`, `money-fx` vs. `money-supply`, `money-fx` vs. `trade` and `money-supply` vs. `trade`.

MEDLINE data[3] (235 abstracts of medical papers that concerned gynecology and assisted reproduction)

	n	docs	ave$_s$	sdev$_s$
20 Newsgroups	138	4040	15.79	5.99
Reuters-21578	4	1022	11.03	2.02
Medline	1	235	12.54	0.22
French cooking	36	1370	9.41	1.24
Czech newspaper	15	2545	22.04	4.22

Table 1: Data (n=number of classification tasks, docs=number of documents, ave$_s$=average number of sentences per document, sdev$_s$=standard deviation)

[1] `http://www.ai.mit.edu/~jrennie/20Newsgroups/`
[2] `http://www.research.att.com/~lewis`
[3] `http://www.fi.muni.cz/~zizka/medocs`

The French documents contained French recipes. Examples of the classification tasks are Accompagnements vs. Cremes, Cremes vs. Pates-Pains-Crepes, Desserts vs. Douceurs, Entrees vs. Plats-Chauds and Pates-Pains-Crepes vs. Sauces, among others.

We also used both methods for classifying *Czech documents*. The data involved fifteen classification tasks. The articles used had been taken from Czech newspapers. Six tasks concerned authorship recognition, the other seven to find a document source – either a newspaper or a particular page (or column). Topic recognition was the goal of two tasks.

The structure of the rest of this paper is as follows. The method for computing the classification of the whole document from classifying fragments (`fragments` method) is described in Section 3. Experimental settings are introduced in Section 4. Section 5 presents the main results. We conclude with an overview of related works and with directions for potential future research in Sections 6 and 7.

3 Classification by means of fragments of documents

The class of the whole document is determined as follows. Let us take a document T which consists of fragments $l_1, \ldots, l_n$ such that $T = \bigcup_{i=1}^{n} l_i$ and $\forall i, j : i \neq j \wedge l_i \cap l_j = \emptyset$. The value of n depends on the length of the document T and on the number of sentences in the fragments. Let $L = \{l_1, \ldots, l_n\}$, and C denotes the set of possible classes. We than use the learned model to assign a class $c(l) \in C$ to each of the fragments $l \in L$. Let $p(l, c(l))$ be the confidence of the classification fragment l into the class $c(l)$. This confidence measure is computed as an estimated probability of the predicted class. Then for each fragment $l \in L$ classified to the class $c \in C$ we define $c(L, c) = \{l \in L | c(l) = c\}$. The confidence of the classification of the whole document T into c is computed as follows

$$P(L,c) = \begin{cases} 0, & c(L,c) = \emptyset \\ \frac{1}{|c(L,c)|} \cdot \sum_{l \in c(L,c)} p(l,c), & otherwise \end{cases}$$

Finally, the class $c(T)$ which is assigned to a document T is computed according to the following definition:

$$\begin{aligned} c(T) = c \quad &\Leftrightarrow \quad |c(L,c)| = \max\{|c(L,c_1)|\} \\ &\wedge \quad P(L,c) = \max\{P(L,c_2)\} \end{aligned}$$

for $\forall c_1 \in C$ and $\forall c_2 \in \{c_3 \in C | \; |c(L,c_3)| = \max\{|c(L,c_1)|\}\}$.

In other words, a document T is classified to a $c \in C$, which was assigned to the most fragments from L (the most frequent class). If there are two classes with the same cardinality, the confidence measure $P(L, c)$ is employed. We also tested another method that exploited the confidence of classification but the results were not satisfactory.

4 Experiments

For feature (i.e. significant word) selection, we tested four methods (Forman, 2002; Yang and Liu, 1999) – Chi-Squared (**chi**), Information Gain (**ig**), F_1-measure (**f1**) and Probability Ratio (**pr**). Eventually, we chose **ig** because it yielded the best results. We utilized three learning algorithms from the Weka[4] system – the decision tree learner J48, the Naïve Bayes, the SVM *Sequential Minimal Optimization* (SMO). All the algorithms were used with default settings. The entire documents have been split to fragments containing 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 20, 25, 30, and 40 sentences. For the `skip-tail` classification which uses only the beginnings of documents we also employed these values.

As an evaluation criterion we used the accuracy defined as the percentage of correctly classified documents from the test set. All the results have been obtained by a 10-fold cross validation.

5 Results

5.1 General

We observed that for both `skip-tail` and `fragments` there is always a consistent size of fragments for which the accuracy increased. It is the most important result. More details can be found in the next two paragraphs.

Among the learning algorithms, the highest accuracy was achieved for all the three languages with the Naïve Bayes. It is surprising because for full versions of documents it was the SMO algorithm that was even slightly better than the Naïve Bayes in terms of accuracy. On the other hand, the highest impact was observed for J48. Thus, for instance for Czech, it was observed for `fragments` that the accuracy was higher for 14 out of 15 tasks when J48 had been used, and for 12 out of 15 in the case of the Naïve Bayes and the Support Vector Machines. However, the performance of J48 was far inferior to that of the other algorithms. In only three tasks J48

[4] `http://www.cs.waikato.ac.nz/ml/weka`

resulted in a higher accuracy than the Naïve Bayes and the Support Vector Machines. The similar situation appeared for English and French.

5.2 `skip-tail`

`skip-tail` method was successful for all the three languages (see Table 2). It results in increased accuracy even for a very small initial fragment. In Figure 1 there are results for `skip-tail` and initial fragments of the length from 40% up to 100% of the average length of documents in the learning set.

	n	NB	stail	lngth	incr
English	143	90.96	92.04	1.3	++105
French	36	92.04	92.56	0.9	+ 25
Czech	15	79.51	81.13	0.9	+ 12

Table 2: Results for `skip-tail` and the Naïve Bayes (n=number of classification tasks, NB=average of error rates for full documents, stail=average of error rates for `skip-tail`, lngth=optimal length of the fragment, incr=number of tasks with the increase of accuracy: +, ++ means significant on level 95% resp 99%, the sign test.)

For example, for English, taking only the first 40% of sentences in a document results in a slightly increased accuracy. Figure 2 displays the relative increase of accuracy for fragments of the length up to 40 sentences for different learning algorithms for English. It is important to stress that even for the initial fragment of the length of 5 sentences, the accuracy is the same as for full documents. When the initial fragment is longer the classification accuracy further increase until the length of 12 sentences. We observed similar behaviour for `skip-tail` when employed on other languages, and also for the `fragments` method.

5.3 `fragments`

This method was successful for classifying English and Czech documents (significant on level 99% for English and 95% for Czech). In the case of French cooking recipes, a small, but not significant impact has been observed, too. This may have been caused by the special format of recipes.

	n	NB	frag	lngth	incr
English	143	91.12	93.21	1.1	++ 96
French	36	92.04	92.27	1.0	19
Czech	15	82.36	84.07	1.0	+ 12

Table 3: Results for `fragments` (for the description see Table 2)

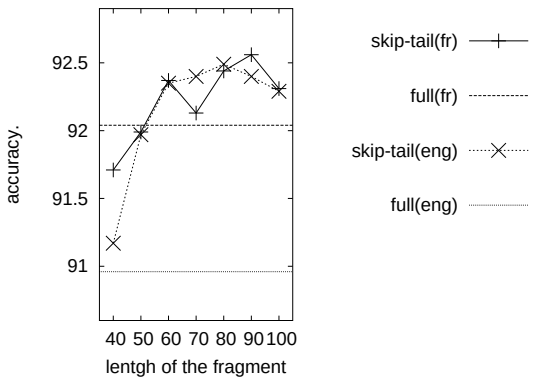

Figure 1: `skip-tail`, Naïve Bayes. (lentgh of the fragment = percentage of the average document length)

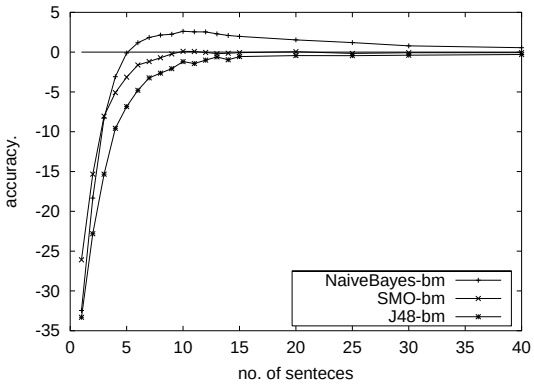

Figure 2: Relative increase of accuracy: English, `skip-tail`

5.4 Optimal length of fragments

We also looked for the optimal length of fragments. We found that for the lengths of fragments for the range about the average document length (in the learning set), the accuracy increased for the significant number of the data sets (the sign test 95%). It holds for `skip-tail` and for all languages. and for English and Czech in the case of `fragments`. However, an increase of accuracy is observed even for 60% of the average length (see Fig. 1). Moreover, for the average length this increase is significant for Czech at a level 95% (t-test).

6 Discussion and related work

Two possible reasons may result in an accuracy increase for `skip-tail`. As a rule, the beginning of a document contains the most relevant information. The concluding part, on the other hand, often includes the author's interpretation and cross-reference to other documents which can cause confusion. However, these statements are yet to be verified.

Additional information, namely lexical or syntactic, may result in even higher accuracy of classification. We performed several experiments for Czech. We observed that adding noun, verb and prepositional phrases led to a small increase in the accuracy but that increase was not significant.

Other kinds of fragments should be checked, for instance intersecting fragments or sliding fragments. So far we have ignored the structure of the documents (titles, splitting into paragraphs) and focused only on plain text. In the next stage, we will apply these methods to classifying HTML and XML documents.

Larkey (Larkey, 1999) employed a method similar to `skip-tail` for classifying patent documents. He exploited the structure of documents – the title, the abstract, and the first twenty lines of the summary – assigning different weights to each part. We showed that this approach can be used even for non-structured texts like newspaper articles. Tombros et al. (Tombros et al., 2003) combined text summarization when clustering so called top-ranking sentences (TRS). It will be interesting to check how fragments are related to the TRS.

7 Conclusion

We have introduced two methods – `skip-tail` and `fragments` – utilized for document categorization which are based on splitting documents into its subparts. We observed that both methods resulted in significant increase of accuracy. We also tested a method which exploited only the most confident fragments. However, this did not result in any accuracy increase. However, use of the most confident fragments for text summarization should also be checked.

8 Acknowledgements

We thank James Mayfield, James Thomas and Martin Dvořák for their assistance. This work has been partially supported by the Czech Ministry of Education under the Grant No. 143300003.

References

G. Forman. 2002. Choose your words carefully. In T. Elomaa, H. Mannila, and H. Toivonen, editors, *Proceedings of the 6th Eur. Conf. on Principles Data Mining and Knowledge Discovery (PKDD), Helsinki, 2002, LNCS* vol. 2431, pages 150–162. Springer Verlag.

L. S. Larkey. 1999. A patent search and classification system. In *Proceedings of the fourth ACM conference on Digital libraries*, pages 179–187. ACM Press.

F. Sebastiani. 2002. Machine learning in automated text categorization. *ACM Comput. Surv.*, 34(1):1–47.

A. Tombros, J. M. Jose, and I. Ruthven. 2003. Clustering top-ranking sentences for information access. In T. Koch and I. Sølvberg, editors, *Proceedings of the 7^{th} European Conference on Research and Advanced Technology for Digital Libraries (ECDL), Trondheim 2003, LNCS* vl. 2769, pages 523–528. Springer Verlag.

Y. Yang and X. Liu. 1999. A re-examination of text categorization methods. In *Proceedings of the 22^{nd} annual international ACM SIGIR conference on Research and development in information retrieval*, pages 42–49. ACM Press.