An Empirical Study on Two-Class Categorization of Czech Documents

XXX

XXX

Abstract. We present results that are based on experiments with the pre-processing phase, namely lemmatisation, chunking and various feature selection methods, and with different learning algorithms. In the second part of this paper we analyze the results that concern the exploitation of sentence chunks as new features. We showed that a use of NLP tools and feature selection methods, namely ig and chi, with Support Vector Machines and Naive Bayes achieved very good results in terms of accuracy. Adding noun, verb and prepositional phrases (NPVGs) leads to a slight increase in accuracy. However, this increase is not significant and does not depend on the feature selection method used. Lemmatization does not play a significant role.

1 Motivation

Text categorization [9] – automatically classifying documents into several classes – by means of machine learning is one of the most successful applications of text mining. Very good results have been achieved mainly for categorization of English documents [9]. Various aspects of text categorization by means of machine learning can be found in [3, 6, 10].

For highly inflective languages like Czech the task involves additional effort. Pre-processing steps – first of all lemmatization and feature (important word) selection – are emerging as important factors in categorizing Czech texts. As far as we know there have not been current attempts to solve the problem of document classification for Czech nor for other highly inflective languages such as Russian, Slovak, Polish, Bulgarian, Rumanian. A part of the IST project Clarity [1] is devoted to hierarchical document categorisation for Latvian and Lithuanian.

Our work is a part of an extensive research project into the automatic classification of Czech texts. We first present results that are based on experiments with the pre-processing phase, namely lemmatisation, chunking and various feature selection methods, and with different learning algorithms.

In the second part of this paper we analyze the results that concern the exploitation of sentence chunks as new features. In text categorization a text document is typically seen as a set or a bag of words and each document is represented as a list of features, one feature corresponding to one word. In this way information about document structure, e.g. the word order or syntactic patterns, is ignored. Various attempts to enriching this representation is described in [2, 4, 5]. To the best of our knowledge, there are no similar works concerning Slavonic languages. In this paper we focused on the use of syntactic chunks, namely noun, prepositional and verb groups, henceforth NPVGs, as additional features.

We aim at answering the following questions.

1. Which are the best learning algorithms?
2. Which feature selection metrics results in the best performance of classification?
3. Is lemmatization needed?
4. Does the use of NPVGs result in an increase of classification accuracy?
5. If so, how does this increase depend on a feature selection method and on a classifier and is this increase significant?

The structure of this paper is as follows. In Section 2 we describe the preprocessing tools used in our work, namely NLP tools – a lemmatizer and a shallow parser, and feature selection metrics. Section 3 describes the data and the learning algorthms used. Technical details of the experiments are in Section 4. The main results are presented in Section 5. Comparison of three different categorization tasks is discussed in Section 6.

2 Preprocessing tools

2.1 Lemmatization and shallow parsering

To find sentence chunks (NPVGs) we used the VaDis shallow parser [7]. VaDis consists of a lemmatizer, a robust grammar for the main sentence chunks in Czech and the parsing mechanism. It also processes verb valencies. VaDis is able to recognize chunks with a recall higher than 99 %. On the other hand, the precision of this parser is only 66 %. This low precision results from the number morphological interpretations of many chunks in Czech. For example, the chunk *mírové síly* can be interpreted as *peace keeping forces, for peace keeping forces* or *of peace keeping forces* depending on the context. The lemmatized form of these chunks remains the same. As we employ lemmatized forms of found chunks only, and not their morphological tags and interpretations, the lower precision of VaDis is not a disadvantage..

We have used the output from the VaDis shallow parser and the built-in lemmatizer in the following way:

dis – all noun, prepositional and verb groups (NPVGs) identified by VaDis. Groups are lemmatized – every constituent of a group is represented by its lemma. Lemmata of all input words which are not parts of any identified group are also included.

lemmata – separated lemmata of all constituents of groups identified by VaDis. Again, lemmata of all input words which are not parts of any identified group are also included.

Table 1. Example of exploited data combinations.

dis	lemmata	dis+words	dis+lemmata
český_republika	český	český_republika	český_republika
být_zastoupit	republika	být_zastoupit	být_zastoupit
ministr_stanislav_gross	být	ministr_stanislav_gross	ministr_stanislav_gross
	zastoupit	česká	český
	ministr	republika	republika
	stanislav	byla	být
	gross	zastoupena	zastoupit
		ministrem	ministr
		stanislavem	stanislav
		grossem	gross

dis+words – concatenation of corresponding **dis** file and the original data
dis+lemmata – concatenation of corresponding **dis** and the **lemmata** file

An example of these different data combinations is presented in Table 1. The original source sentence is *Česká republika byla zastoupena ministrem Stanislavem Grossem. (The Czech Republic was represented by minister Stanislav Gross.).*

2.2 Feature selection metrics

A feature selection is fundamental in text classification. Dimensionality reduction of a data (removing unimportant terms) can increase the speed of learning and can reduce overfitting. It was shown [3, 10] that we can achieve the same or better performance in terms of accuracy on the less than 10 % of terms from document (with a level of aggresivity [1] greater than .90). In our experiments we used feature selection methods based on feature filtering. We computed the value of a metric (function of "importance") for each term occuring in the training set and selected k terms with the highest value. We used four metrics – Chi-Squared, Information Gain, F_1-measure and Probability Ratio which are frequently used in text classification. We also tried MI score, Odds and BNS [3] measures but the results were inferior to those of other methods regardless of the data that had been explored.

To compute the values of these metrics we used Forman's [3] definitions which are simplified for classification into two classes. In the definitions below terms tp, fp, tn and fn denote: *true positives* (a number of documents from *positive* classes – referred to as a *positive document* – containing the term t), *false positives* (the number of occurences of t in *negative documents*), *true negatives* and *false negatives*. Other terms are computed as follows: $pos = tp+fn$ (number of positive documents), $neg = fp + tn$ (number of negative documents), $all = pos + neg$ (the size of training set), $P(c_i)$ is a probability of class c_i, and $P(t)$ resp. $P(\bar{t})$ is a probability that term t occures in document or it does not respectively.

[1] The value of the *aggresivity* function is computed as $1 - \frac{\bar{r}}{r}$ where r is the number of original features and $\bar{r}$ is a number of features in reduced set.

Chi-Squared (chi) – is based on the χ^2 statistic which measures independence between term t and class c (it can be seen as the χ^2 distribution with one degree of freedom). For a two class problem it is defined as:

$$\begin{aligned} chi(t) = {} & \frac{(tp - P(c_p)(tp + fp))^2}{P(c_p)(tp + fp)} + \frac{(fn - P(c_p)(fn + tn))^2}{P(c_p)(fn + tn)} + \\ & + \frac{(fp - P(c_n)(tp + fp))^2}{P(c_n)(tp + fp)} + \frac{(tn - P(c_n)(fn + tn))^2}{P(c_n)(fn + tn)} \end{aligned}$$

Information Gain (ig) – measures the decrease of entropy when a term occurs in document and when not. This measure is frequently used in machine learning algorithms (e.g. in the induction of decision trees [6]).

The value of this metric is computed as follows:

$$\begin{aligned} ig(t) &= H(pos, neg) - [P(t) \cdot H(tp, fp) + P(\bar{t}) \cdot H(tn, fn)] \\ H(c_p, c_n) &= -\frac{c_p}{c_p + c_n} \log_2 \frac{c_p}{c_p + c_n} - \frac{c_n}{c_p + c_n} \log_2 \frac{c_n}{c_p + c_n} \end{aligned}$$

where $H(c_p, c_n)$ computes the entropy for two classes.

F$_1$-measure (f1) – is a direct application of F$_\beta$-measure (with $\beta = 1$) $f_1(t) = (2 \cdot tp)/(pos + tp + fp)$

Probability Ratio (pr) – is defined as an estimate probability of the occurence of term t in the positive class divided by the estimate probability of the term in negative class. For $fp \neq 0$ we define $pr(t) = tp/pos \cdot (fp/neg)^{-1}$, otherwise we use the value 0.0005 instead of the second fraction.

3 Data and learning algorithms

3.1 Data

In our experiments we used 15 data sets of articles from the Czech newspapers Mladá fronta, Lidové noviny, Hospodářské noviny and Právo. Six tasks concerned authorship recognition: last year's articles from Mladá fronta, by Jana Bendová, Martin Komárek and Karel Steigerwald; with the number of documents for each task ranging from 400 to 600. The other six aimed at identifuing a document source, namely the front page of Mladá fronta, commentary, 'Zajimavosti'; from 170 to 400 documents. The goal of three tasks was topic recognition from different newspaper; informative article vs. commentary, nuclear power stations, and the European Union; from 34 to 157 documents. In all the cases, problems referred to a binary document classification. There was no significant difference between the numbers of positive and negative examples for each task.

3.2 Learning algorithms

We compared the performances of three learning algorithms – Naïve Bayes, the Support Vector Machine SMO, and the decision tree learner J4.8. We have chosen the Naïve Bayes classifier because it is well established in text classification domain and it is widely respected for its good results. SVM is now very popular in text categorization because it very often provides comparable or even better results than the Naïve Bayes. Lastly the decision trees are easily interpreted and can be used to analyze properties of data. In a preliminary phase we also tested the instance-based learner IB1, however, the accuracy was lower than that obtained with the other algorithms.

4 Method

We used the Weka (`http://www.waikato.nj/~weka`) learning package. All algorithms were used with default settings. All combinations of lemmatization, chunking and feature selection methods have been explored in combination with these three learning algorithms. As an evaluation criterion we used accuracy defined as the percentage of correctly classified documents in a test set. All results have been obtained by 10-fold cross validation. It means that after possible lemmatization and/or chunking the data was split into 10 folds of equal size, 9 folds used for learning and one for testing. Then a feature selection metrics was employed to each fold. A term (a word, a lemma or a chunk) weight was set to the value of a particular feature selection metrics.

5 Results

For all data sets an accuracy was higher than 80%, for 9 out of 15 data was even higher than 90%. We observed that error rate did not decrease or decreased only slightly when the number of feature exceeded 500. For this reason we further tested only a number of features in the range from 1 to 500. Typical trends can be seen in Fig. 5. The results for all tasks can be found in Table 2. Concerning classifiers, the Support Vector Machines proved to be the best in terms of accuracy. The Naïve Bayes classifier achieved a slightly lower accuracy. J4.8 was in general inferior, despite having better results in two of the 15 tasks.

For each of these 15 classification tasks we have chosen a combination of preprocessing methods and learning algorithms which resulted in the highest accuracy. The results are shown in Table 2.

We further analyzed which combination of a feature selection method (pre in Table 2), chunking (NPVG), lemmatization (lemma), and a learning classifier (alg) is the best. Two feature selection methods – **chi** and **ig** – displayed the best results for all learning algorithms for the majority of the 15 tasks. The most impressive is **ig**, having been used for 8 out of the 13 tasks in which NPVGs helped. However, there is no significant difference between **ig** and **chi** for any of the learning algorithms. The main results are summarized below.

Fig. 1. bendovaXkomar+steiger and nazoryXzajim

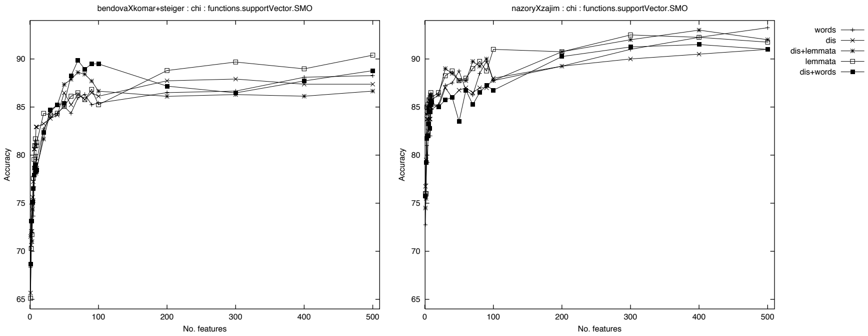

Table 2. Results

Task	NPVGs			words			lemma		
	pre	alg	acc	pre	alg	acc	pre	alg	acc
aut1	**chi**	**bay**	**95.78**	chi	bay	95.50	chi	bay	95.77
aut2	**ig**	**bay**	**89.57**	ig	bay	85.92	chi	smo	89.33
aut3	**ig**	**bay**	**97.24**	ig	bay	96.44	ig	bay	96.96
aut4	ig	smo	90.20	ig	bay	89.33	**chi**	**smo**	**90.40**
aut5	ig	bay	95.37	**ig**	**bay**	**95.91**	chi	bay	95.56
aut6	**f1**	**smo**	**90.40**	ig	smo	87.20	ig	smo	89.34
source1	**ig**	**smo**	**93.25**	**pr**	**bay**	**93.25**	ig	smo	93.00
source2	**ig**	**smo**	**91.00**	ig	smo	90.25	ig	smo	88.75
source3	**ig**	**smo**	**90.00**	pr	smo	84.25	ig	smo	89.25
source4	**chi**	**smo**	**88.67**	ig	smo	84.83	chi	smo	87.00
source5	ig	smo	87.83	chi	smo	81.67	**ig**	**smo**	**88.50**
source6	ig	smo	90.00	**ig**	**smo**	**90.33**	ig	smo	89.17
jadern	**f1**	**smo**	**82.50**	ig	bay	64.17	f1	smo	75.83
klaus	**chi**	**smo**	**83.38**	f1	smo	80.33	ig	smo	82.88
eu	**f1**	**j48**	**88.57**	f1	j48	87.85	ig	j48	88.04

1. For 12 out of 15 tasks, adding NPVGs as new features resulted in greater accuracy if compared with **words**. This increase varies from 0.3 to 18%. Conversely, a possible decrease in accuracy – as was observed for two tasks – was not greater than 0.6%.
2. However, the accuracy increase is not significant (t-test) for any of 15 tasks but `jadern`.
3. Verb phrases (about 20% of all NPVGs) are important for classification. After removing them, accuracy has.
4. There is no difference in preferences SMO and the Naïve Bayes neither for **ig** nor **chi** for data with NPVGs and without.
5. When lemmatization was employed, the addition of NPVGs found with VaDis resulted in an accuracy increase for 10 out of 15 tasks; for 2 tasks the highest accuracy was achieved on the lemmatized text.

6 Usefulness of chunks as new features

As introduced above, we have experimented with three types of tasks, namely **authorship** recognition, determination of a document **source**, and **topic** recognition. For these three types of tasks we have compared the results of the **ig** data preprocessing method, especially its exploitation of NPVGs. Table 3 shows ten best NPVGs selected with **ig** for every task type. We have analysed the structure of NPVGs relevant for classification. Selected verb groups are usually very general ones, e.g. *muset_být (to have to be)*, the same property can be observed for preposition groups, e.g. *o_ten (about it)*. Selected noun groups can be split into name entities (organizations, location or personal names) like *václav_havel*, multiword expressions (nominal compounds) like *daňový_poplatník (tax-payer)* or *veřejný_finance (public finance)*, and general noun phrases. Percentages of particular chunks are displayed in Table 4 for every task type. NE+MWE means relative occurrence of name entities and multiword expressions among all the selected noun groups. Table 4 is based on one tens of the selected NPVGs for every task type. Below we list all name entities(NE) and multiword expressions (MWE) that appeared in the top twenty together with their position (the lower the better).

Table 3. Most relevant NPVGs-features for different task types.

authorship	**source**	**topic**
václav_klaus	u_my	sebe_líbit
václav_havel	k_ten	podle_můj_názor
na_ten	na_ten	v_oko
o_ten	muset_být	moct_být
v_nemocnice	o_ten	sebe_jednat
za_ten	v_ten	z_hnutí
na_rozdíl	v_česko	za_ten
moct_být	v_skutečnost	v_země
stanislav_gross	z_přidaný_hodnota	v_příprava
sebe_zdát	zdát_sebe	v_leden

Table 4. Frequency of chunks

chunk type	**authorship**	**source**	**topic**
noun groups (NE+MWE)	35.3% (23.5%)	15.7% (15.7%)	31.4% (17.6%)
verb groups	15.7%	17.6%	15.7%
prepositional groups	49.0%	66.7%	52.9%

- **authorship**: NE: *václav_klaus* (1), *václav_havel* (2), *stanislav_gross* (9); MWE: *veřejný_finance* (16), *daňový_poplatník* (18), *lidový_dům* (20);
- **source**: NE: *václav_klaus* (12), *stanislav_gross* (20);
- **topic**: NE: *vladimír_mlynář* (11); MWE: *právní_řád* (20);

7 Conclusion and future work

We showed that a use of NLP tools and feature selection methods, namely ig and chi, with Support Vector Machines and Naive Bayes achieved very good results in terms of accuracy. For all the data set explored an accuracy overcame 82%. The most important result lies in the fact that adding noun, verb and prepositional phrases (NPVGs) leads to a slight increase in accuracy. However, this increase is not significant and does not depend on the feature selection method used. Despite the complex morphology of Czech, this result is similar to those for English [5]. Lemmatization does not play a significant role.

8 Acknowledgement

We thank to members of Knowledge Discovery Group for fruitful discussions and to James Thomas for his proofreading. This work was partially supported by the grant of Czech Ministry of Education MSMT 143300003.

References

1. Clarity, IST-2000-25310, IST Project, 2000 (`http://www.dcs.shef.ac.uk/research/groups/nlp/clarity/`)
2. Cohen W., Singer Y. Context-Sensitive Learning Methods for Text Categorization (1996) Proceedings of SIGIR-96, 1996.
3. Forman G.: Choose Your Words Carefully. In Proceedings of PKDD 2002. LNAI 2431.
4. J. Fuernkranz, T. Mitchell and E. Riloff. A Case Study in Using Linguistic Phrases for Text Categorization on the WWW.. Proceedings of AAAI/ICML WS on Learning for Text Categorization. March 1998.
5. Lewis D. An evaluation of phrasal and clustered representations on a text categorization task In Proceedings of SIGIR pp. 37–50, Copenhagen, Denmark 1992.
6. Mitchell, Tom: Machine Learning. McGraw Hill 1997.
7. Mráková E., Sedláček R. From Czech Morphology through Partial Parsing to Disambiguation. In Proceedings of CICLING 2003.
8. Papka R., Allan J. Document classification using multiword features Proceedings of CIKM-98, 1998.
9. Sebastiani F. Machine learning in Automated Text Categorization. ACM Computing Surveys, Vol. 34, No. 1, March 2002, pp.1-47.
10. Yang Y., Liu X. A re-examination of text categorization methods. Proceedings of SIGIR, pp. 42–49, 1999.