

# Using string kernels for classification of Slovenian Web documents

Blaž Fortuna and Dunja Mladenič

J.Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia

**Abstract.** In this paper we present an approach for classifying web pages obtained from the Slovenian Internet directory where the web sites covering different topics are organized into a topic ontology. We tested two different methods for representing text documents, both in combination with the linear SVM classification algorithm. The first representation that we have used is a standard bag-of-words approach with TFIDF weights and cosine distance used as similarity measure. We compared this to String kernels where text documents are compared not by words but by substrings. This removes the need for stemming or lemmatisation which can be important issue when documents are in languages other than English and tools for stemming or lemmatisation are unavailable or are expensive to make or learn. In highly inflected natural languages, such as Slovene language, the same word can have many different forms, thus String kernels have an advantage here over the bag-of-words. In this paper we show that on classification of documents written in highly inflected natural language the situation is opposite and String Kernels significantly outperform the standard bag-of-words representation. Our experiments also show that the advantage of String kernels is more evident for domains with unbalanced class distribution.

## 1 Introduction

Classification of documents is usually performed by representing documents as word-vectors using the bag-of-words document representation and applying some classification algorithm on the vectors (Sebastiani, 2002). In the usual setting, bag-of-words document representation cuts a document text into words and represents the document with frequency of words that occur in the document. Even though it ignores the ordering of words, it was found to perform well in combination with different classification algorithms and usually outperforms alternative representations on the standard problems of document categorization. However, experiments are usually performed on standard document categorization datasets and most of them contain documents written in English.

There are mixed results on the performance change due to using word stemming as a pre-processing on English documents. However, when dealing with non-English documents, especially documents written in highly inflected languages, applying stemming or lemmatisation can be crucial. Namely, in highly inflected natural languages, a word having the same or very similar meaning can occur in several tens of, slightly different, forms (depending on

gender, number, case, etc.). Unfortunately, it is not always the case that we have stammer or lemmatiser available for a particular natural language (it may not be publicly available or even it may not exist).

This paper investigates performance of an alternative document representation, String kernels, on non-English documents. String kernels cut the document text into sequences of characters regardless of the word boundaries. This can be seen as an alternative approach to handling the problem of having slightly different words carrying almost the same meaning. Namely, in most cases, these words differ in the word suffix, so taking the first  $k$  letters of the word (where  $k$  is smaller than the average length of the words) can be seen as a way of obtaining a word stem.

Previous research has shown that on categorization of English documents with linear SVM, the bag-of-words document representation outperforms String kernels (Saunders et al., 2002). We show that String kernels outperform the bag-of-words representation on documents written in highly inflected natural language, namely Slovenian. The difference in performance is larger on problems with unbalanced class distribution. To the best of our knowledge this is the first experimental comparison of these two document representations on documents written in highly inflected natural language.

This paper is organized as follows. Section 2 describes the used methodology including the Support Vector Machine classifier and String kernels. Section 3 describes the used datasets. Experimental comparison of the two document representations is provided in Section 4, followed by discussion in Section 5.

## 2 Methodology

### 2.1 Support Vector Machine

The most common technique for representing text documents is *bag-of-words* (BOW) using word frequency with TFIDF weighting. In the bag-of-words representation there is a dimension for each word; a document is then encoded as a feature vector with word frequencies as elements. Document classification has been performed using different classification algorithms on the bag-of-words document representation. The linear Support vector machine (SVM) (Bose et al., 1992) algorithm is known to be one of the best performing for text categorization eg., in (Joachims, 1999). Thus, in this paper we report on experiments using linear SVM for classifying web documents.

Support vector machine is a family of algorithms that has gained a wide recognition in the recent years as one of the state-of-the-art machine learning algorithms for tasks such as classification, regression, etc. In the basic formulation they try to separate two sets of training examples by hyperplane that maximizes the margin (distance between the hyperplane and the closest points). In addition one usually permits few training examples to be misclassified. For unbalanced datasets, different cost can be assigned to

examples according to the class value (Morik et al., 1999). The cost is controlled by parameters  $j$  and  $C$ , where  $C$  corresponds to the misclassification cost ( $C_+ = jC$  and  $C_- = C$ ). An alternative approach to handling unbalanced datasets based on shifting the SVM induced hyperplane was proposed in (Brank et al., 2003). In this paper we consider only changing the value of SVM parameter  $j$  in order to improve performance on unbalanced datasets. We avoided hyperplane shifting by using a measure for experiments that does not depend on the threshold.

When constructing the SVM model, only the inner product between training examples is needed for learning the separation hyperplane. This allows the use of so called kernel function. The kernel function is a function that calculates inner product between two mapped examples in feature space. Since explicit extraction of features can have a very high computational cost, a kernel function can be used to tackle this problem by implicit use of mapped feature vectors.

## 2.2 String kernels

The main idea of string kernels (Lodhi et al., 2002; Saunders et al., 2002) is to compare documents not by words, but by the substrings they contain – there is a dimension for each possible substring and each document is encoded as a feature vector with substring weights as elements. These substrings do not need to appear contiguous in the document, but they receive different weighting according to the degree of contiguity. For example: substring 'c-a-r' is present both in the words 'card' and 'custard' but with different weighting. Weight depends on the length of substring and the decay factor  $\lambda$ . In previous example, substring 'car' would receive weight  $\lambda^3$  as part of 'card' and  $\lambda^6$  as part of 'custard'. Feature vectors for documents are not computed explicitly because it is computationally very expensive. However, an efficient dynamic algorithm exists (Lodhi et al., 2002) that computes the inner product between two feature vectors. We use this algorithm as a kernel in the SVM. The advantage of this approach is that it can detect words with different suffixes or prefixes: the words 'micro**computer**', '**computers**' and '**computer**based' all share common substrings. The disadvantage of this approach is that computational cost is higher than that of BOW.

We have used our own implementation of SVM, bag-of-words and string kernels which are all part of our Text Garden<sup>1</sup> suite of tools for text mining. The SVM implementation is very efficient and gives similar performance to SVMlight. Its advantage is tight integration with the rest of Text Garden.

---

<sup>1</sup> <http://www.textmining.net>

### 3 Dataset description

We compared performance of bag-of-words and String kernels on several domains containing document from Matkurja directory of Slovenian web documents (such as Open directory or Yahoo!). Each web page is described with a few sentences and is assigned to a topic from the directory's taxonomy. The whole directory contains 52,217 documents and 63,591 words.

Similar as proposed in some previous experiments on Yahoo! documents (Mladenic and Grobelnik 2002), we have selected some top-level categories and treated each as a separate problem. Top level category 'Arts' having 3557 documents and 'Science and Education' having 4046 documents were used ignoring hierarchical structure of the documents they contain. From each of them we select three subcategories of different sizes thus having different percentage of positive examples. In this way we obtained domains with different proportion of positive examples ranging from unbalanced (where only 4 % of examples are positive and 96 % are negative) to balanced with 45% of examples being positive. The selected domains are as follows. From Arts we have selected three subcategories: Music having 45 % of documents, Painting having 7 % of documents and Theatre having 4 % of documents. From 'Science and Education' the following three subcategories were selected: Schools having 25 % of documents, Medicine having 14 % of documents and Students having 12 % of documents. For each subcategory we define a separate domain having all the documents from the subcategory as positive documents and all the documents from other subcategories of the same top-level category as negative documents.

### 4 Experiments

All the experimental results are averaged over five random splits using hold-out method, randomly splitting each category into a training part (30%) and a testing part (70%). A classifier is generated from training documents and evaluated on the testing documents. The evaluation is performed using Break Even Point (BEP) – a hypothetical point at which precision (ratio of positive documents among retrieved ones) and recall (ratio of retrieved positive documents among all positive documents) are the same. There was no special pre-processing performed on the documents used in experiments except removing html-tags and changing all the characters to lowercase.

For classification we use linear SVM algorithm with cost parameter  $C$  set to 1.0. We ran experiments for different values of the length of substrings used in string kernel, of the value of decay parameter  $\lambda$  and of parameter  $j$ .

We have tested the following hypotheses all assuming usage of linear SVM classifier for document classification. String kernels outperform bag-of-words on documents written in highly inflected natural languages (Section 4.1) with the difference being more evident on data with unbalanced class distribution

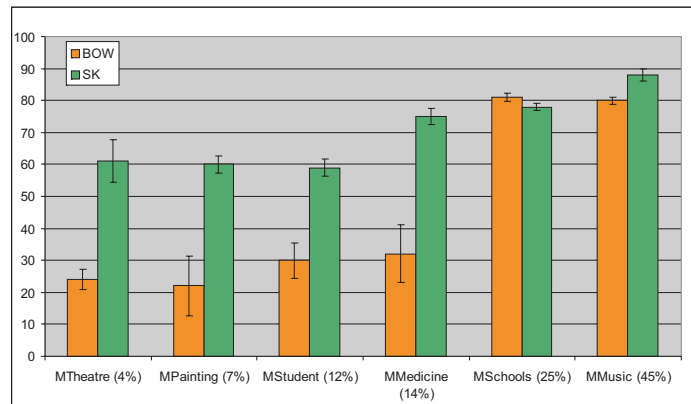
(Section 4.2). Using SVM mechanism for handling unbalanced data improves performance of the two representations (Section 4.3). We have also to limited extent investigated the influence of two String kernel parameters with the main hypothesis being that using too short String kernels hurt the performance (Section 4.4).

#### 4.1 String kernels vs. bag-of-words on inflected languages

The first hypothesis we have tested is that String kernels achieve better results than bag-of-words on document written in highly inflected language. Our experiments confirm that hypothesis, as on eight out of nine domains String kernels achieve significantly higher BEP than bag-of-words (with significance level 0.001 on seven domains and 0.05 on one domain). Table 2 gives the results of categorization for six domains of Slovenian documents.

Category	Subcategory	BOW [%]	SK [%]
M-Arts	Music	$80 \pm 1.9$	<b><math>88 \pm 0.4</math></b>
	Painting	$22 \pm 5.5$	<b><math>60 \pm 2.6</math></b>
	Theatre	$24 \pm 3.1$	<b><math>61 \pm 6.6</math></b>
M-Science	Schools	$81 \pm 3.8$	$78 \pm 2.6$
	Medicine	$32 \pm 1.9$	<b><math>75 \pm 2.0</math></b>
	Students	$30 \pm 4.0$	<b><math>59 \pm 1.1</math></b>

**Table 1.** Results for classification task, BEP is used as evaluation measure. String kernel of length 5 and  $\lambda = 0.2$ . Bold numbers are significantly higher.



**Fig. 1.** Comparison of SVM performance on Slovenian documents using bag-of-words (BOW) and String kernels (SK). The domains are sorted according to the percentage of positive examples, from M-Theatre (4%) to M-Music (45%).

#### 4.2 String kernels vs. bag-of-words on unbalanced datasets

From initial experiments on a few domains, we have noticed that the difference in performance between the two representations varies between the

domains. Thus we have performed experiments on more domains, selecting them to have different percentage of positive examples.

Our hypothesis was that the difference is larger on domains with more unbalanced class distributions. As can be seen from Figure 1 this is the case, on domains having less than 15% of positive examples (the four leftmost domains in Figure 1), String kernels achieve much higher BEP compared to bag-of-words. On one domain having 25% of positive examples the difference in performance is not significant, while on the balanced class domains (the last column in Figure 1) String kernels are again significantly better than bag-of-words (but the absolute difference in performance is much lower).

### 4.3 Setting SVM parameters to handle unbalanced datasets

The categorization algorithm we are using, SVM, already has a mechanism for handling domains with unbalanced class distribution (commonly referred to as parameter  $j$ ). The  $j$  parameter enables assigning different misclassification cost to examples of positive and of negative class. The default value of  $j$  is 1.0. Setting it to some value greater than 1.0 is equivalent to over-sampling by having  $j$  copies of each positive example.

J	Bag-of-words			String kernels		
	1.0	5.0	10.0	1.0	5.0	10.0
Music	80 ± 1.8	84 ± 1.0	84 ± 0.8	88 ± 0.4	87 ± 0.8	87 ± 0.8
Painting	22 ± 5.5	48 ± 2.6	48 ± 2.6	60 ± 2.6	58 ± 2.7	58 ± 2.7
Theatre	24 ± 3.1	38 ± 7.8	38 ± 7.7	61 ± 6.6	62 ± 5.8	62 ± 5.8
Schools	81 ± 3.9	80 ± 0.8	80 ± 1.1	78 ± 2.6	77 ± 2.1	77 ± 1.9
Medicine	32 ± 1.8	55 ± 3.1	55 ± 3.1	75 ± 2.0	73 ± 0.8	73 ± 0.6
Students	30 ± 4.0	50 ± 3.3	50 ± 3.0	59 ± 1.1	58 ± 0.8	58 ± 1.0

**Table 2.** Influence of SVM parameter  $j$  on six domains of Slovenian documents using bag-of-words using String kernel of length 5 and  $\lambda = 0.2$ .

We have investigated influence of changing the value of  $j$  and found that changing it from 1.0 to 5.0 significantly improves performance (significance level 0.01) of bag-of-words on all but one domain (see Table 3). Setting  $j$  to higher values ( $j = 10.0$ ) does not significantly change the performance. Changing the value of parameter  $j$  when using String kernels, does not significantly influence the performance of SVM, as can be seen in Table 3.

### 4.4 Changing parameters of String kernels

String kernels work with sequences of characters. It was shown in previous work on English documents (Lodhi et al., 2002) that the length of the sequence significantly influences the performance to some degree. As expected our experiments have confirmed that finding also for Slovenian documents.

Namely, using too short string kernels (in our case 3 characters) results in significantly (significance level 0.05) lower performance than using longer string kernels, achieving in average over the six domains BEP of 65.5 compared to BEP 70 achieved when using String kernels of length 4. Having length 4, 5 or 6 results in similar performance on Slovenian documents. However, one would expect that this might depend on the natural language, as in some cases having length 4 or 5 may still be too short.

We have also varied the value of decay factor of string kernel (parameter *lambda*) from 0.1 to 0.4 and found that it does not influence the performance on our domains.

## 5 Discussion

We have tested two methods for representing text documents, bag-of-words and String kernels, both in combination with linear SVM classification algorithm. We have shown that when dealing with documents written in highly inflected natural language, such as Slovene, String kernels significantly outperform a commonly used bag-of-words representation. Moreover, the advantage of String kernels is more evident for the domains with unbalanced class distribution having less than 15% of positive examples. As string kernels use substrings instead of whole words for representing document content, this seems to compensate for stemming or lemmatisation which can be important for documents in highly inflected languages. This is especially important when tools for stemming or lemmatisation are unavailable or expensive.

Because we are dealing with highly inflected natural languages bag-of-words fails to match different forms of the same word. On the other hand, String kernels are able to match them because they use substrings (in our case of length 5, not words) as features and allow gaps between parts of the substrings. For illustration, in the following examples of Slovenian sentences, talking about traffic problems, bag-of-words does not find any connection between them. However, String kernels identify that the words 'cesti', 'obcestnega', 'cestisce' and 'cestninsko', all different forms of word 'road', share common substrings. Note that in the case of String kernels of length 5, the substring 'cesti' does not necessary contain letters from the same words (see bold letters in the example).

- 'Prevrnjeni tovornjak povzroca zastoje na **cesti** ...'
- 'Zaradi zamasenega ob**cestnega** jarka in odtoka je popljavneno **cestisce** na ...'
- 'Pred **cestninsko** postajo nastajajo daljsi zastoji.'

We have also found that using the SVM mechanism (parameter *j*) for handling unbalanced domains, significantly improves the bag-of-words performance but it still stays significantly lower than the performance of String kernels. The same parameter does not significantly influence the performance of String kernels. The performance of String kernels is significantly influenced by the length of the kernel but only if the kernel is very short (using length

3 yields significantly worse performance than using length 4, but there is no difference between length 4 and length 5).

In the future work, it would be interesting to repeat the experiments on some other natural languages and possibly try to relate the advantage of String kernels over bag-of-words to the degree of the language inflection.

In our experiments we use Break even point as the evaluation measure, as commonly used in document categorization. However, we have noticed that if using the threshold proposed by SVM for predicting the class value, the value of precision or recall is very low, in most cases close to 0. A closer look has revealed that even though both bag-of-words and String kernels have problem with setting the right threshold, this is more evident for String kernels. In future work we want to investigate possibilities of improving the threshold, eg., as post-processing by shifting the SVM induced hyper-plane as proposed in (Brank et al 2003) for handling unbalanced domains using bag-of-words.

## Acknowledgements

This work was supported by the Slovenian Research Agency and the IST Programme of the European Community under SEKT Semantically Enabled Knowledge Technologies (IST-1-506826-IP) and PASCAL Network of Excellence (IST-2002-506778).

## References

- B. E. BOSER, I. M. GUYON, and V. N. VAPNIK (1992): *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory (pp. 144-152). Pittsburgh, PA, July 1992. ACM Press*
- J. BRANK, M. GROBELNIK, N. MILIC-FRAYLING, D. MLADENIC (2003): Training text classifiers with SVM on very few positive examples, *Technical report, MSR-TR-2003-34*.
- T. JOACHIMS (1999): Making large-scale svm learning practical. *In B. Scholkopf, C. Burges, and A. Smola, editors, Advances in Kernel Methods - Support Vector Learning. MIT-Press.*
- H. LODHI, C. SAUNDERS, J. SHAWE-TAYLOR, N. CRISTIANINI, AND WATKINS C. (2002): Text classification using string kernels. *Journal of Machine Learning Research, 2, 419-444.*
- MLADENIC D., GROBELNIK M. (2003): Feature selection on hierarchy of web documents. *Journal of Decision Support Systems, 35(1): 45-87.*
- K. MORIK AND P. BROCKHAUSEN AND T. JOACHIMS (1999): Combining statistical learning with a knowledge-based approach – A case study in intensive care monitoring. *International Conference on Machine Learning (ICML)*
- J. PLISSON, N. LAVRAC, D. MLADENIC, (2004): A rule based approach to word lemmatization. *Proceedings of the 7th International multi-conference Information Society IS-2004, Ljubljana: Institut Jozef Stefan, pp. 83-86.*
- C. SAUNDERS, H. TSCHACH, AND J. SHAWE-TAYLOR, (2002): Syllables and Other String Kernel extensions. *In Proceedings of Nineteenth International Conference on Machine Learning (ICML 02)*
- SEBASTIANI, F., (2002): Machine Learning for Automated Text Categorization, *ACM Computing Surveys, 34:1, pp.1–47.*