A preliminary approach to the multilabel classification problem of Portuguese juridical documents

Teresa Gonçalves and Paulo Quaresma

Departamento de Informática,
Universidade de Évora,
7000 Évora, Portugal
tcg|pq@di.uevora.pt

Abstract. Portuguese juridical documents from Supreme Courts and the Attorney General's Office are manually classified by juridical experts into a set of classes belonging to a taxonomy of concepts.
In this paper, a preliminary approach to develop techniques to automatically classify these juridical documents, is proposed. As basic strategy, the integration of natural language processing techniques with machine learning ones is used. Support Vector Machines (SVM) are used as learning algorithm and the obtained results are presented and compared with other approaches, such as C4.5 and Naïve Bayes.

1 Introduction

Automatic classification of documents is an important problem in many domains. For instance, it is needed by web search engines and information retrieval systems in order to organize the text bases into sets of semantic categories.

In order to develop better algorithms for document classification it is necessary to integrate research from several areas, such as machine learning, natural language processing and information retrieval.

A methodology for the automatic classification of documents is proposed and applied to a set of documents written in the European Portuguese language. This methodology integrates:

- Machine learning algorithms, namely, a kernel-based learning algorithm – Support Vector Machines;
- Natural language processing techniques, such as, lemmatization (transforming each word inito its lemma without Portuguese symbols) and part-of-speech tagging;
- Information retrieval techniques, such as the use of stop words, the representation of documents as bag-of-words and evaluation procedures.

Since the work of Joachims [**?**] it is known that Support Vector Machines (SVM) perform quite well compared with other approaches to the text classification problem. In his approach, documents are represented as bag-of-words

(without word order information) [**?**] and some words are not represented (words belonging to the set of the so called "stop words"). Then, a kernel based learning algorithm is applied (SVM [**?**]) and the results are evaluated using error measures and information retrieval ones.

In this paper, we follow Joachims' proposal, applying it to the set of Portuguese juridical documents from the Attorney General's Office. This set is composed by 7089 documents and it is being manually classified by juridical experts into a set of concepts from a law taxonomy. However, our proposal is quite distinct from Joachim's work because we aim to prove the importance of linguistic information in the classification problem. At present, we are only using part-of-speech information to eliminate words from the bag-of-words but we intend to use syntactical and semantical information and to propose and evaluate specific kernels (following the ideas of word sequence kernels [**?**]).

The SVM classification results are analyzed and compared with other machine learning algorithms, such as C4.5 and Naïve Bayes, through the accurate rate (Acc %) and information retrieval measures (K and *true* and *false* F-measures).

In section **??** our classification problem is described and characterized. In section **??** a brief description of the Support Vector Machines theory is presented. Section **??** describes our experiments and evaluates the results. Finally, in section **??**, some conclusions and future work are pointed out.

2 Text Classification

Our goal is to automatically classify documents written in the European Portuguese language into sets of concepts. This problem is usually called a multi-label classification because each document can be classified into multiple concepts/topics.

The typical approach to the multi-label classification problem is to divide it into a set of binary classification problems, where each concept is considered independently. In this way, the initial problem is reduced to solve several binary classification problems.

Binary classification problems can be characterized by the inference of a classification rule assigning one of two possible values $(-1, 1)$ to each document. A value of -1 means the document does not belong to the concept and a value of 1 means that it belongs to it.

In this work we are using the set of documents from the Portuguese Attorney General's Office (in portuguese, Procuradoria Geral da República – PGR)[1]. These documents represent the decisions of the Attorney General's Office since 1940 and they define a set with cardinality 7089 and around 96MB of characters. All documents were manually classified by juridical experts into a set of classes belonging to a taxonomy of law concepts with around 6000 terms. However, a preliminary evaluation showed that only around 3000 terms are used in the multi-label classification.

[1] These documents can be found at the PGR site (http://www.pgr.pt)

As final goal we intend to develop a binary classification model for each concept but, for the scope of this work, we have only dealt with the top 50 most used concepts. As an example, we present the top five concepts and its frequency:

- pgr_2572: deficiente das forcas armadas (647)
- pgr_1391: pensao por servicos excepcionais e relevantes (539)
- pgr_744: aposentacao (494)
- pgr_16: funcionario publico (404)
- pgr_1877: competencia (358)

Another important open problem is the representation of the documents. In this preliminary work, we will use the standard vector representation [**?**], where each document is represented as a bag-of-words and where order information is lost and no syntactical or semantical information is used. As future work, we intend to explore the use of word order and to use syntactic and semantic information in the classification.

Nevertheless, PGR documents were pre-processed in order to obtain the part-of-speech tags for each word (its morpho-syntactical information) and to transform each word in its lemma (for instance, each verb is transformed into its infinitive form and each noun to the singular form). This work is done using the results of a previous project, PGR project, which aimed to develop an intelligent information retrieval system for PGR decisions [**?**]. In this project, a lexical database – POLARIS – is used to perform the lemmatization and the part-of-speech (POS) tagging is done with this lexical database and a neural network.

Using the POS tags we were able to eliminate words with non relevant information, such as, articles and prepositions, and with the lemmatization procedure it was also possible to reduce the number of distinct words. As final result, we obtained a total of 38703 distinct words. In section **??**, some experiments done trying to reduce the number of words (features) are described.

3 Support Vector Machines

In this section a brief introduction to kernel classifiers and support vector machines is presented[2]. More detailed information can be obtained in several specialized books, such as [**?**,**?**].

Kernel learning algorithms are based on theoretical work on statistical learning theory, namely the structural risk minimization [**?**,**?**].

A binary classifier is a function from an input space X into the set of binary labels $\{-1, +1\}$. A supervised learning algorithm is a function assigning, to each labeled training set, a binary classifier

$$h : X \to \{-1, +1\} \tag{1}$$

[2] This introduction is based on a similar section in [**?**]

Whenever X is a vector space, a simple binary classifier is given by:

$$h(x) = sign(< w, x > +b) \tag{2}$$

where $< ., . >$ stands for the vector dot-product.

Learning the linear classifier is equivalent to finding values for w and b, which maximize an evaluation measure.

Linear classifiers fail when the boundary between the two classes is not linear. In this situation the approach followed is to project X into a new feature space F and to try to define a linear separation between the two classes in F. If the projection function is defined by $\phi : X \rightarrow F$ then the linear classifier is:

$$h(x) = sign(< w, \phi(x) > +b) \tag{3}$$

Support Vector Machines (SVM) are specific learning algorithms for linear classifiers, trying to obtain values for w and b. In SVM w is assumed to be defined as a linear combination of the projections of the training data:

$$w = \sum_{i=1}^{l} y_i \alpha_i \phi(x_i) \tag{4}$$

where α_i is the weight of the training example i with input x_i and label y_i.

The optimal weights are the solution of a high dimensional quadratic problem, which can be expressed in terms of the dot product of the projection of the training data $< \phi(x_i), \phi(x_j) >$.

It was proved that it is not necessary to map the input data into the feature space F, as long as it is defined a kernel function $K : X * X \rightarrow R$, such that $K(x, y) =< \phi(x), \phi(y) >$. This is known as the *kernel trick*. On the other hand Mercer's theorem [**?**] states that any positive semi-definite symmetric function corresponds to some mapping in some space and it is a valid kernel.

In the scope of this work only linear kernels are used and each document is represented by a vector where each dimension value stands for the frequency of a specific word in that document. As future work we intend to propose and evaluate specific kernels trying to take into account linguistic knowledge.

4 Experiments

As it was referred in the previous sections, the SVM learning algorithm was applied to the problem of multi-label classification of the Portuguese Attorney General's Office decisions.

The text base is composed by 7089 documents and the number of existent distinct words was reduced through the application of part-of-speech tagging techniques and through the lemmatization of every word. In this way it was possible to exclude non-relevant words, such as, articles and prepositions and to reduce distinct forms of every word to its lemma (verbal forms to the infinitive form; noun forms to the singular, masculine form). As final result, we obtained

a set of 38703 distinct words. After the selection of the relevant words, each document was represented by a vector having 38703 dimensions where each value stands for the occurrence's frequency of the correspondent word in the document.

The 6000 classification labels/concepts were sorted in a decrescent number of occurrences in the documents and the top concepts were selected for the application of learning algorithms (section **??** presents the top five concepts).

4.1 Feature reduction

The first experiment was to evaluate the overall results of the SVM for the top concepts and to evaluate the impact of the reduction of features/words in the algorithm. The idea behind this reduction was to try to reduce the algorithm complexity without loosing performance. In fact, 38703 attributes is a large number and it creates some computational problems to the learning algorithms.

The reduction was done by eliminating words that appear in less than a specific number of documents. For instance, $R55$ means that all words appearing in less than 55 documents were eliminated.

The results for the top concept were the following (we used a 10-fold cross-validation evaluation procedure and all experiments were done using the WEKA software package [**?**] from Waikato University[3] with default parameters for all experiments):

pgr_2572

	Acc %	K	t Fmeasure	f Fmeasure
1	99,535	0,972	0,975	0,997
55	99,577	0,975	0,977	0,998
100	99,549	0,973	0,975	0,998
200	99,605	0,976	0,978	0,998
400	99,619	0,977	0,979	0,998
600	99,591	0,975	0,978	0,998
800	99,408	0,964	0,967	0,997
1200	99,069	0,945	0,950	0,995
1600	98,575	0,910	0,918	0,992
2000	98,279	0,890	0,899	0,991
2400	98,180	0,883	0,893	0,990
2800	97,926	0,865	0,876	0,989
3200	94,400	0,543	0,568	0,970
3600	94,118	0,511	0,536	0,969
4000	93,610	0,449	0,474	0,966

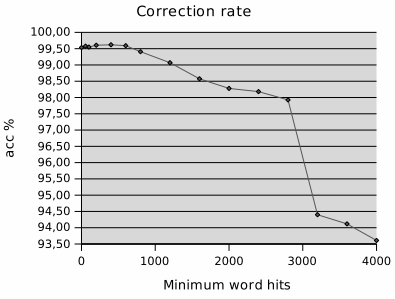

Fig. 1. Results for concept *pgr_2572*

Note the high results obtained for the classification – 99.5% accurate classifications. Quite good are also the results for the F-measure of the class *true* and the class *false*. F-measure is a standard information retrieval measure, which combines the precision and the recall measures [**?**]. Precision and recall are calculated from the contingency table of the classification (prediction vs manual classification). Precision is given by the number of correct classified documents

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

divided by the number of documents classified into the class. Recall is given by the number of correct classified documents divided by the number of documents belonging to the class. K-measure is also an important measure, which tries to obtain the degree of concordance between the two classifiers (manual and SVM). It is commonly accepted that a value of K higher than 0.7 stands for a relevant degree of concordance.

From the analysis of the results it appears that the first 6 experiments had no significant loss of performance. This is a quite interesting and important result because, for instance, $R55$ has only 5388 attributes and $R600$ has only 1518 attributes!

In order to test this hypothesis we performed similar experiments for the next top concepts (we will only show here the results for the next two top concepts).

pgr_1391

	Acc %	K	t Fmeasure	f Fmeasure
1	98,900	0,920	0,926	0,994
55	98,928	0,922	0,928	0,994
100	98,886	0,919	0,925	0,994
200	98,928	0,922	0,928	0,994
400	98,928	0,922	0,927	0,994
600	98,928	0,922	0,927	0,994
800	98,773	0,909	0,916	0,993
1200	98,378	0,876	0,885	0,991
1600	97,785	0,822	0,834	0,988
2000	96,741	0,718	0,734	0,983
2400	96,586	0,701	0,718	0,982
2800	96,558	0,699	0,716	0,982
3200	96,586	0,702	0,719	0,982
3600	96,600	0,703	0,719	0,982
4000	96,600	0,703	0,719	0,982

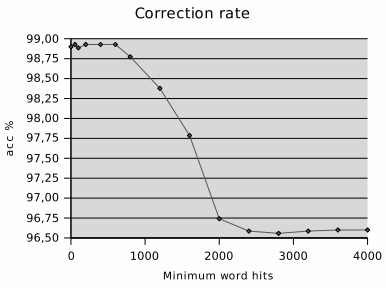

Fig. 2. Results for concept pgr_1391

As figure **??** shows, the classifier for concept pgr_1391 has a similar behavior (only after $R600$ decreases performance) but its results are not so good.

pgr_744

	Acc %	K	t Fmeasure	f Fmeasure
1	93,695	0,375	0,405	0,967
55	93,709	0,329	0,355	0,967
100	93,836	0,311	0,335	0,968
200	93,553	0,232	0,252	0,966
400	93,032	0,056	0,064	0,964
600	92,961	0,029	0,035	0,963
800	92,961	0,026	0,031	0,963
1200	93,046	0,031	0,035	0,964
1600	93,046	0,027	0,031	0,964
2000	93,060	0,028	0,031	0,964
2400	93,088	0,032	0,035	0,964
2800	93,088	0,032	0,035	0,964
3200	93,088	0,032	0,035	0,964
3600	93,116	0,036	0,039	0,964
4000	93,102	0,032	0,036	0,964

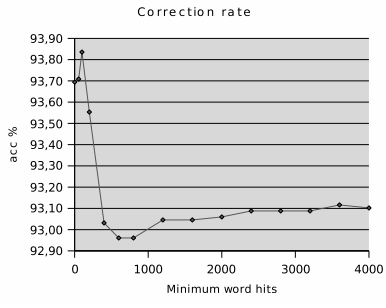

Fig. 3. Results for concept pgr_744

Concept *pgr*_744 (figure **??**) shows a quite different behavior. The percentage of correct classified documents is high (although not so high as the previous concepts) but the K and F_{true} measures are quite low. These results show the importance of these information retrieval measures in the evaluation of document classifiers. One possible origin of these problems is the fact that we have much more negative examples (around 90%) than positive (around 10%). The consequences of this situation is that even the simpler classifier (assigning always the negative class) obtains 90% correct results but it will get a low value for the IR measures.

Figure **??** shows the contingency table for the four top concepts. Table lines show the values for the manual classification and table columns show the values obtained by the classifier. For instance, line 2, column 1, represents the number of documents classified as *true* by the classifier, which belong to the $false$ class.

pgr_2572	ct	cf
t	633	14
f	16	6426

pgr_744	ct	cf
t	123	371
f	75	6520

pgr_1391	ct	cf
t	487	52
f	24	6526

pgr_16	ct	cf
t	5	399
f	32	6653

Fig. 4. Contingency table for the top 4 concepts

As it can be seen, concepts *pgr*_744 and *pgr*_16 show a high level of false negatives and, as a consequence, the IR measures are quite low. Further work needs to be done in order to explain why some concepts are modeled so well by the linear SVM and others perform so poorly.

As a consequence of these experiments, we decided to focus our work in the $R55$ documents (documents represented by the words that appear in at least 55 documents) because they showed no loss of performance and they have a smaller complexity (5388 attributes versus 38703 attributes for $R1$).

4.2 SVM evaluation

As explained in the previous section we focused our experiments in the $R55$ set of documents and, in this section, we will evaluate the obtained results against two other standard learning methods: Naïve Bayes and the decision-tree C4.5 classifier.

Naïve Bayes classifiers uses a probabilistic model of text to estimate the probability of a document d to be in class y – $P(y|d)$. However, in order to make the estimation of parameters possible, some assumptions are made. For instance, words are assumed to occur independently of the other words in documents, given its class. Moreover, all documents associated with a particular class are assumed to be modeled accordingly with a unique model for that category. Naïve Bayes

classifiers try to maximize $P(y|d)$ using these assumptions and the well-known Bayes rule for conditional probabilities (see, for instance, [**?**] for a description of experiments using Naïve Bayes classifiers).

C4.5 [**?**] is one of the most well-known decision tree classifiers and it has shown good results in a quite diversity of classification problems. We have used the WEKA Java latest version – J48 – with its default parameters.

Figure **??** shows the results obtained for the top concepts (accurate rate and computation time in a P4 at 2.8GHz with 1GB RAM).

pgr_2572

test set

SMO	NaiveBayes	J48
99,577	25,081	99,619

time consumed

SMO	NaiveBayes	J48
11m	41m	4h10m

pgr_1391

test set

SMO	NaiveBayes	J48
98,928	26,661	99,267

time consumed

SMO	NaiveBayes	J48
7m	38m	8h11m

Fig. 5. Classification comparison

From these results it is quite clear that Naïve Bayes classifier performs quite worse than the other two classifiers: 25-26% vs 98-99%! The computation time showed also quite different values: from a minimum of 7 minutes (SVM) to a maximum of 8 hours (J48).

For this reason we have excluded Naïve Bayes classifier from the other experiments.

Our next experiment was to evaluate and to compare the results for the $R55$ document classification using SVM and C4.5/J48. Figure **??** shows the results obtained for the top-5 concepts.

	SMO				J48			
	acc %	K	t Fmeasure	f Fmeasure	acc %	K	t Fmeasure	f Fmeasure
pgr_2572	99,577	0,975	0,977	0,998	99,619	0,977	0,979	0,998
pgr_1391	98,928	0,922	0,928	0,994	99,267	0,948	0,951	0,996
pgr_744	93,709	0,329	0,461	0,973	94,738	0,592	0,620	0,972
pgr_16	93,920	0,013	0,023	0,969	93,144	0,267	0,302	0,964
pgr_1877	94,555	0,138	0,157	0,972	93,441	0,218	0,251	0,966

Fig. 6. SVM vs. J48

After analyzing the results it is possible to conclude that the overall correction rate is similar (although a little bit better for J48) but J48 statistics for K and F_{true} are better for the worst classified concepts. This values can be explained by the capability of J48 to build quite complex models with decision trees with many levels. However, it is important to point out that the temporal complexity of C4.5/J48 is much higher than SVM algorithms (10min vs. 8 hours) and the worst SVM classification models remain bad classification models in C4.5/J48.

As a conclusion of this evaluation section, we may point out that SVM linear learning algorithms for documents written in the Portuguese language showed to be, a least, as good as the two other learning algorithms (Naïve Bayes and C4.5) and they produced quite good results.

Similar results were already obtained for other sets of documents, such as the Reuters [**?**]. Nevertheless, our results showed to be better than the results obtained by Joachims in his experiments. Further work needs to be done in order to explain these differences.

5 Conclusions and Future Work

A methodology for the automatic classification of the Portuguese documents from the Attorney General's Office was proposed. The methodology tries to integrate machine learning algorithms (SVM) with natural language processing tools (part-of-speech tagging and lemmatization) and information retrieval techniques (stop words, documents as bag-of-words, evaluation measures).

The obtained results showed to be, at least, equivalent with similar approaches and they proved to be adequated for the Portuguese language and for the law domain.

As future work, we intend to evaluate our approach against standard document sets, such as the Reuters set. In this way, we will be able to fully compare our results with others researchers' results.

Nevertheless, for some concepts, the obtained results were not quite good and further work needs to be done in order to explain them and to improve the classifiers. Our hypothesis is that these classifiers need more powerful document representations. As a consequence, we intend to use more linguistic knowledge in the document representation, namely, moving from a vector-based representation into a structured syntactical and/or semantical representation. This document representation change will have, as a consequence, the need for new and more adapted kernels.

References

1. N. Cancedda, E. Gaussier, C. Goutte, and J. Renders. Word sequence kernels. *Journal of Machine Learning Research*, 3:1059–1082, 2003.
2. C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.
3. N. Cristianini and J. Shawe-Taylor. *Support Vector Machines*. Cambridge University Press, 2000.
4. Thorsten Joachims. *Learning to Classify Text Using Support Vector Machines*. Kluwer academic Publishers, 2002.
5. A. McCallum and K. Nigam. A comparison of event models for naive bayes text classification. In *Learning for text categorization Workshop of the ICML/AAAI-98 conference*. AAAI Press, 1998.

6. Paulo Quaresma and Irene Pimenta Rodrigues. PGR: Portuguese attorney general's office decisions on the web. In Bartenstein, Geske, Hannebauer, and Yoshie, editors, *Web-Knowledge Management and Decision Support*, Lecture Notes in Artificial Intelligence LNCS/LNAI 2543, pages 51–61. Springer-Verlag, 2003.
7. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993.
8. G. Salton and M. McGill. *Introduction to Modern Informatin Retrieval*. McGraw-Hill, 1983.
9. B. Schölkopf and A. Smola. *Learning with Kernels*. MIT Press, 2002.
10. V. Vapnik. *Estimation of Dependencies based on Empirical Data*. Springer, 1982.
11. V. Vapnik. *The nature of statistical learning theory*. Springer, 1995.
12. I. Witten and E. Frank. *Data Mining: Practical machine learning tools with Java implementations*. Morgan Kaufmann, 1999.