Is linguistic information relevant for the classification of legal texts?

Teresa Gonçalves
Departamento de Informática
Universidade de Évora
7000-671 Évora, Portugal
tcg@di.uevora.pt

Paulo Quaresma
Departamento de Informática
Universidade de Évora
7000-671 Évora, Portugal
pq@di.uevora.pt

ABSTRACT

Text classification is an important task in the legal domain. In fact, most of the legal information is stored as text in a quite unstructured format and it is important to be able to automatically classify these texts into a predefined set of concepts.

Support Vector Machines (SVM), a machine learning algorithm, has shown to be a good classifier for text bases [Joachims, 2002]. In this paper, SVMs are applied to the classification of European Portuguese legal texts – the Portuguese Attorney General's Office Decisions – and the relevance of linguistic information in this domain, namely lemmatisation and part-of-speech tags, is evaluated.

The obtained results show that some linguistic information (namely, lemmatisation and the part-of-speech tags) can be successfully used to improve the classification results and, simultaneously, to decrease the number of features needed by the learning algorithm.

1. INTRODUCTION

The learning problem can be described as finding a general rule that explains data, given a sample of limited size. In supervised learning, we have a sample of input-output pairs (the *training sample*) and the task is to find a deterministic function that maps any input to an output such that the disagreement with future input-output observations is minimised. If the output space has no structure except whether two elements are equal or not, we have a *classification* task. Each element of the output space is called a *class*. The supervised classification task of natural language texts is known as *text classification*.

Research interest in this field has been growing in the last years. Several learning algorithms were applied, such as decision trees [Tong & Appelbaum, 1994], linear discriminant analysis and logistic regression [Schütze *et al.* , 1995], the naïve Bayes algorithm [Mladenić & Grobelnik, 1999] and Support Vector Machines (SVM)[Joachims, 2002].

[Joachims, 2002] says that using SVMs to learn text classifiers is the first approach that is computationally efficient and performs well and robustly in practice. There is also a justified learning theory that describes its mechanics with respect to text classification.

Text classification is also an important task in the legal domain. In fact, most of the legal information is stored as text in a quite unstructured format and it is important to be able to automatically classify these texts into a predefined set of concepts.

In this domain, much work has been done in data and text analysis tasks. For instance, [Wilkins & Pillaipakkamnatt, 1997] used decision trees to extract rules to estimate the number of days until the final case disposition; [Zeleznikow & Stranieri, 1995] developed rule based and neural networks legal systems; [Borges *et al.* , 2003] used neural networks to model legal classifiers; [Thompson, 2001] proposed a framework for the automatic categorisation of case laws; [Schweighofer & Merkl, 1999, Schweighofer *et al.* , 2001] described the use of self-organising maps (SOM) to obtain clusters of legal documents in an information retrieval environment and explored the problem of text classification in the context of the European law; [Liu *et al.* , 2003] described classification and clustering approaches to case-based criminal summaries and [Brüninghaus & Ashley, 2003, Bruninghaus & Ashley, 1997] described also related work using linear classifiers for documents.

Aiming to design a system to prior case retrieval (find prior cases that belong to the appellate chain of the current case), [Al-Kofahi *et al.* , 2001] integrated information extraction, information retrieval and machine learning techniques. They used SVMs to rank prior case candidates according to their likelihood of being true priors.

However, in these research work the relevance of linguistic information in legal text analysis tasks is not studied in detail. In our work, the application of SVMs to the problem of legal text classification is described and an evaluation of the relevance of linguistic information is performed.

On previous work, we evaluated the SVM performance compared with other Machine Learning algorithms [Gonçalves & Quaresma, 2003], such as decision trees and naïve Bayes. We could conclude that decision trees and SVM outperforms naïve Bayes and that, even with similar performance, the model building time is much shorter with SVMs than with decision trees. In [Silva *et al.* , 2004], the application of linguistic information to the preprocessing phase of text mining tasks was studied for the Brazilian Portuguese language.

In this work, we apply a linear SVM to a legal Portuguese text base, the Portuguese Attorney General's Office dataset – PAGOD [Quaresma & Rodrigues, 2003], performing a thorough study on several preprocessing techniques such as feature reduction, feature subset selection and term weighting.

The relevance of using some linguistic information, such as lemmatisation and part-of-speech tags (POS), to reduce the number of features is studied in detail and we show that it is possible to strongly reduce the number of features and the complexity of the legal text classification problem without loosing accuracy.

In Section 2, a brief description of the Support Vector Machines theory is presented, while in Section 3 the PAGOD dataset is characterised. Section 4 describes our experimental setup and Sections 5 and 6 the experiments made. Conclusions and future work are pointed out in Section 7.

2. SUPPORT VECTOR MACHINES

Support Vector Machines, a learning algorithm introduced by Vapnik and coworkers [Cortes & Vapnik, 1995], was motivated by theoretical results from the statistical learning theory. It joins a kernel technique with the structural risk minimisation framework.

Kernel techniques comprise two parts: a module that performs a mapping from the original data space into a suitable feature space and a learning algorithm designed to discover linear patterns in the (new) feature space. These stages are illustrated in Figure 2.

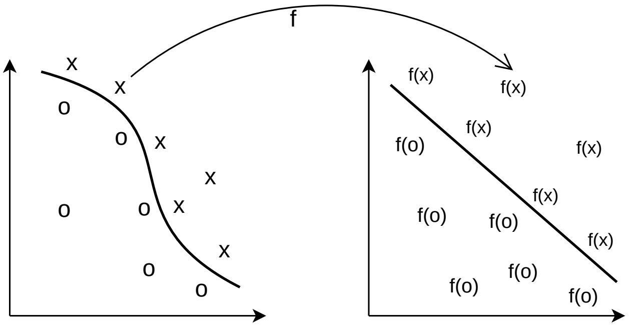

Figure 1: Kernel function: The nonlinear pattern of the data is transformed into a linear feature space.

The *kernel function*, that implicitly performs the mapping, depends on the specific data type and domain knowledge of the particular data source.

The *learning algorithm* is general purpose and robust. It's also efficient, since the amount of computational resources required is polynomial with the size and number of data items, even when the dimension of the embedding space (the feature space) grows exponentially [Shawe-Taylor & Cristianini, 2004].

Four key aspects of the approach can be highlighted as follows:

- Data items are embedded into a vector space called the feature space.
- Linear relations are discovered among the images of the data items in the feature space.
- The algorithm is implemented in a way that the coordinates of the embedded points are not needed; only their pairwise inner products.
- The pairwise inner products can be computed efficiently directly from the original data using the kernel function.

The *structural risk minimisation* (SRM) framework creates a model with a minimised VC (Vapnik-Chervonenkis) dimension. This developed theory [Vapnik, 1998] shows that when the VC dimension of a model is low, the expected probability of error is low as well, which means good performance on unseen data (good generalisation).

SVM can also be derived in the framework of the regularisation theory instead of the SRM one. The idea of regularisation, introduced by Tikhonov and Arsenin [Tikhonov & Arsenin, 1977] for solving inverse problems, is a technique to restrict the (commonly) large original space of solutions into compact subsets.

3. DATASET DESCRIPTION

The working dataset (PAGOD – Portuguese Attorney General's Office Decisions), has 8151 legal documents and represents the decisions of the Portuguese Attorney General's Office since 1940. It is written in the European Portuguese language, and delivers 96 MBytes of characters. All documents were manually classified by juridical experts (from the Attorney General's Office) into a set of categories belonging to a taxonomy of legal concepts with around 7000 terms.

Each document was classified into multiple categories so, we are in presence of a multi-label classification task. This kind of problem is usually solved by splitting the original classification task into a set of binary ones and considering each one independently [Nigam *et al.* , 2000] [Joachims, 1998].

A preliminary evaluation showed that, from all potential categories, only 801 had ten or more documents assigned to it; from all available documents (8151), only 6773 contained at least one word on all experiments. For these documents, we found 68886 distinct words, and averages of 1592 words and 362 distinct words per document.

Figure 3 shows an histogram of the number of documents assigned to the 50 most used categories (the ones with more than 75 documents assigned to it).

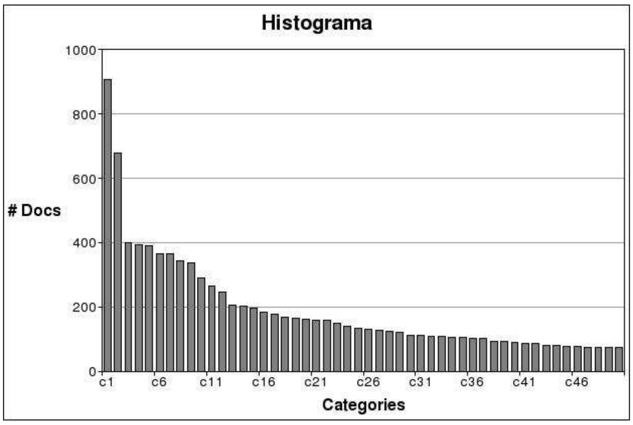

Figure 2: Number of documents asssigned to each of the 100 most used categories.

We studied the 10 most used categories (from here named the top ten categories). Table 1 shows them along with the number of documents belonging to each one. As can be depicted, there is a disparity on the abstraction level between some of these categories.

category	id	# docs
pensão por serviços excepcionais / exceptional services pension	c_1	906
deficiente das forças armadas / army injured	c_2	678
prisioneiro de guerra / war prisoner	c_3	401
estado da Índia / India state	c_4	395
militar / military	c_5	388
louvor / praise	c_6	366
funcionário público / public officer	c_7	365
aposentação / retirement	c_8	342
competência / competence	c_9	336
exemplar conduta moral e cívica / exemplary moral and civic behaviour	c_{10}	289

Table 1: PAGOD's top ten categories: label and number of documents.

4. EXPERIMENTAL SETUP

This section presents the experimental setup of our study: the learning tool chosen and how did we represent a document and measured learners' performance.

The linear SVM was run using the WEKA [Witten & Frank, 1999] software package from New Zealand's Waikato University, with default parameters (complexity parameter equal to one and normalised training data) performing a 10-fold cross-validation procedure.

WEKA is a collection of machine learning algorithms for data mining tasks. It contains tools for data preprocessing, classification, regression, clustering, association rules, and visualisation. It trains a support vector classifier implementing John Platt's sequential minimal optimisation algorithm [Platt, 1998] and executes a stratified cross-validation procedure.

Cross-validation (CV) is a model evaluation method. The original dataset is divided into k subsets (in this work, $k = 10$), each one with (approximately) the same distribution of examples between categories as the original dataset (stratified CV). Then, one of the k subsets is used as the test set and the other k-1 subsets are put together to form a training set; a model is built from the training set and then applied to the test set. This procedure is repeated k times (one for each subset). Every data point gets to be in a test set exactly once, and gets to be in a training set $k - 1$ times. The variance of the resulting estimate is reduced as k is increased.

To represent each document we chose the bag-of-words approach, a *vector space model* (VSM) representation: each document is represented by the words it contains, with their order and punctuation being ignored. From the bag-of-words we removed all words that contained digits.

To measure learner's performance we analysed precision, recall and the F_1 measures [Salton & McGill, 1983] of the positive class. These measures are obtained from contingency table of the classification (prediction *vs.* manual classification). For each performance measure we calculated the micro- and macro-averaging values of the top ten categories.

Precision is the number of correctly classified documents (true positives) divided by the number of documents classified into the class (true positives plus false positives).

Recall is given by the number of correctly classified documents (true positive) divided by the number of documents belonging to the class (true positives plus false negatives).

F_1 is the weighted harmonic mean of precision and recall and belongs to a class of functions used in information retrieval, the F_β*-measure*. F_β can be written as follows

$$F_\beta(h) = \frac{(1+\beta^2)prec(h)rec(h)}{\beta^2 prec(h) + rec(h)}$$

Macro-averaging corresponds to the standard way of computing an average: the performance is computed separately for each category and the average is the arithmetic mean over the ten categories.

Micro-averaging does not average the resulting performance measure, but instead averages the contingency tables of the various categories. For each cell of the table, the arithmetic mean is computed and the performance is computed from this averaged contingency table.

All significance tests were done regarding a 95% confidence level.

5. IR PREPROCESSING EXPERIMENTS

We considered three classes of preprocessing experiments: feature reduction/construction, feature subset selection and term weighting.

Most of these experiments are common in the IR community and the best setups are known for the English language. Nevertheless, and since we are working with Portuguese written texts from a specific area – the legal one, we decided to make a wide range of experiments to validate/discover the "best" setup.

5.1 Experiments

For each class of preprocessing experiments we considered several possible setups.

5.1.1 Feature Reduction/Construction

On trying to reduce/construct features we used some linguistic information: we applied a Portuguese stop-list (the set of non-relevant words such as articles, pronouns, adverbs and prepositions) and POLARIS, a lexical database [Lopes *et al.* , 1994], to generate the lemma for each Portuguese word.

Stemming and lemmatisation are not quite the same thing: while stemming cuts each word transforming it into its radical, lemmatisation reduces the word to its canonical form. For example, the canonical form of "driven" is "drive" while its stem is "driven".

Except for the irregular verbs, stemming and lemmatisation generate the same "word" for most English words. For the Portuguese language, this is not true: most lemmatised words are different from the stemmed ones.

We made three different experiments:

- rdt_1: consider all words of the original documents (except, as already mentioned, the ones that contained digits)
- rdt_2: consider all words except the ones that belong to the stop-list
- rdt_3: consider all words (except the ones that belong to the stop-list) transformed into its lemma

5.1.2 Feature Subset Selection

For the feature subset selection we used a filtering approach, keeping the features that receive higher scores according to different functions:

- scr_1: *term frequency.* The score is the number of times the feature appears in the dataset; only the words occurring more frequently are retained;
- scr_2: *mutual information.* It evaluates the worth of an attribute by measuring the mutual information with respect to the class. Mutual Information, $I(C;A)$, is an Information Theory measure [Cover & Thomas, 1991] that ranks the information received to decrease the uncertainty. The uncertainty is quantified through the Entropy, $H(X)$.
- scr_3: *gain ratio* – $GR(A,C)$. The worth is the gain ratio with respect to the class. Mutual Information is biased through attributes with many possible values. Gain ratio tries to oppose this fact by normalising mutual information by the feature's entropy.

Mutual information and *gain ratio* are defined in terms of the probability function $p(x)$ where C is the class and A is the feature. $H(C|A)$ is the class entropy when we know the feature's value. These quantities are defined by the following expressions:

$$H(X) = -\sum_x p(x)\log_2 p(x)$$

$$\begin{aligned} I(C;A) &= H(C) - H(C|A) \\ &= -\sum_c p(c)\log_2 p(c) + \\ &\quad + \sum_a p(a)\sum_c p(c|a)\log_2 p(c|a) \end{aligned}$$

$$GR(A,C) = \frac{I(C;A)}{H(A)}$$

For each filtering function, we tried different threshold values. The threshold is the number of times the feature appears in all documents. We performed experiences for thr_1, thr_{50}, thr_{100}, thr_{200}, thr_{400}, thr_{800}, thr_{1200} and thr_{1600}, where thr_n means that all words appearing less than n are eliminated.

For each threshold, we used the number of features selected as the number the features retained for each scoring function. Table 2 shows the number of features obtained for each threshold value. The last two rows show, per document, the average number of all (avg_{all}) and distinct ($avg_{distinct}$) features.

5.1.3 Term Weighting

Term weighting techniques usually consist of three components: the document component, the collection component and the normalisation component. In the final feature vector x, the value x_i for word w_i is computed by multiplying the three components.

Document component captures statistics about a particular term in a particular document. Its basic measure is the *term*

	rdt_1	rdt_2	rdt_3
thr_1	68886	68688	42423
thr_{50}	9479	9305	5983
thr_{100}	6439	6275	4413
thr_{200}	4238	4085	3147
thr_{400}	2578	2440	2115
thr_{800}	1515	1390	1332
thr_{1200}	1076	962	956
thr_{1600}	831	724	743
avg_{all}	1592	802	768
$avg_{distinct}$	362	277	215

Table 2: Number of features for each threshold value and feature construction/reduction combination.

frequency – $TF(w_i, d_j)$. It is defined as the number of times word w_i occurs in document d_j.

The collection component assigns lower weights to terms that occur in almost every document of a collection. Its basic statistic is the *document frequency* – $DF(w_i)$, *i.e.* the number of documents in which w_i occurs at least once.

The normalisation component adjusts weights so that small and large documents can be compared on the same scale.

We made experiments for the following combination of components:

- wgt_1: *binary* representation. Each word occurring in the document has weight 1; all others have weight 0. The resulting vector is normalised to unit length.
- wgt_2: *raw term frequencies.* It's $TF(w_i, d_j)$.
- wgt_3: *normalised term frequencies.* It's $TF(w_i, d_j)$ normalised to unit length.
- wgt_4: *TFIDF representation.* It's $TF(w_i, d_j)$ multiplied by $log(N/DF(w_i))$ where N is the total number of documents. The quantity is normalised to unit length.

These experiments can be represented graphically in a 4-dimensional space. First we have a three dimension space with one axis for feature reduction/construction, feature subset selection and term weighting. In each axis there are three or more possible values representing different experiments. The feature subset selection axis is then "subdivided" in another two: the scoring function and the threshold value. Figure 5.1.3 shows one of the possible experiments.

We performed experiences for all combinations of feature reduction/construction, scoring function and term weighting (rdt_1, rdt_2 and rdt_3; scr_1, src_2 and src_3; wgt_1, wgt_2, wgt_3 and wgt_4) and all the already presented threshold values, totalling a number of 288 experiments.

5.2 Results

Now, we present and discuss the results obtained for this set of experiments.

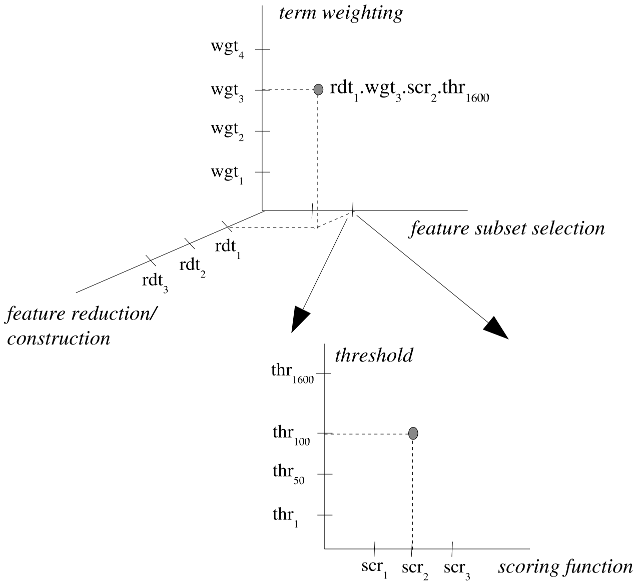

Figure 3: Graphical representation of the IR experiments.

Table 3 presents the minimum, maximum, average and standard deviation of all experiments (precision, recall and F_1 micro- and macro- averages for the top ten categories).

	micro			*macro*		
	Prec	*Rec*	F_1	*Prec*	*Rec*	F_1
min	.667	.407	.560	.580	.325	.386
max	.953	.714	.763	.903	.632	.667
avg	.852	.634	.722	.723	.535	.581
stdev	.052	.071	.038	.047	.079	.073

Table 3: Minimum, maximum, average and standard deviation of micro- and macro- precision, recall and F_1 measures.

While precision reached values above 0.9, the recall values were lower. These values could be explained by the fact that we are in presence of a highly imbalance dataset since, for example, from all almost 7000 documents just 906 belong to the most common category (see Table 1) and, as referred in [Japkowicz, 2000], it can be a source of bad results.

Table 4 presents the number of experiments that have no significance difference with respect to the best one. We also present the distribution of these "best" experiments on each set of IR experiments. For example, for the macro-F_1 measure we have 26 "best" experiments. From these, 16 belong to the rdt_2 setup and 10 to the rdt_3 one.

For the feature reduction/construction experiments one can say that removing the stop-words and/or doing lemmatisation is beneficial for the classification. For the feature subset selection experiments, the term frequency and the mutual information functions are better than the gain ratio one and the thr_{400} threshold is the biggest one that produces good results. For the term weighting experiments, the normalised term frequencies experiments are the ones

	micro			*macro*		
	Prec	*Rec*	F_1	*Prec*	*Rec*	F_1
best	4	20	55	1	13	26
rdt_1	0	1	7	0	0	0
rdt_2	2	7	29	0	5	16
rdt_3	2	12	19	1	8	10
src_1	0	18	21	0	11	17
src_2	0	1	25	0	1	5
src_3	4	1	9	1	1	4
wgt_1	0	4	22	0	2	7
wgt_2	4	0	0	1	0	0
wgt_3	0	11	25	0	9	13
wgt_4	0	5	8	0	2	6
thr_1	0	3	18	0	3	12
thr_{50}	0	3	2	0	2	2
thr_{100}	0	3	2	0	1	3
thr_{200}	0	6	3	0	4	4
thr_{400}	0	5	9	0	3	4
thr_{800}	0	0	8	0	0	1
thr_{1200}	2	0	7	0	0	0
thr_{1600}	2	0	6	1	0	0

Table 4: Number of experiments belonging to the set of best results for micro- and macro- precision, recall and F_1 measures.

with better results.

Table 5 presents the micro- and macro- precision, recall and F_1 values for the "best" setups just referred (for wgt_3 and thr_{400}). The bold faced values have no significance difference with the best one obtained (for each measure).

	micro			*macro*		
	Prec	*Rec*	F_1	*Prec*	*Rec*	F_1
$rdt_2.scr_1$	.810	**.709**	**.756**	.711	**.626**	**.661**
$rdt_2.scr_2$	.843	.682	**.754**	.732	.590	**.633**
$rdt_3.scr_1$	.815	**.714**	**.761**	.717	**.632**	**.667**
$rdt_3.scr_2$	.850	.679	**.755**	.728	.585	.626

Table 5: Micro- and macro- precision, recall and F_1 measures for the "best" setups.

Since the mutual information scoring function appears less in the set of "best" values (Tables 4 and 5) we can chose the term frequency scoring function as the best one.

Table 6 shows the precision, recall and F_1 measures for 5 studied categories for the term frequency scoring function, the normalised term frequencies weighting scheme and the thr_{400} threshold value ($src_1.wgt_3.thr_{400}$), with the use the stop words (rdt_2) and lemmatisation (rdt_3).

There is a big disparity in the values obtained between different categories. For the c_1 and c_2 ("exceptional services pension" and "army injured"), the values for the three measures are very good (almost one), while for the others that is not true. c_5 and c_8 ("military" and "retirement") present values near 0.5 for precision, recall and F_1 and c_7 ("public officer") has an especially bad recall (less than 0.3). c_3, c_4 and c_{10} ("war prisoner", "India state" and "exemplary moral and civic behaviour", respectively) have values similar to c_1 and c_2; c_6 ("praise") has also a similar behaviour with its values a little bit smaller (around 0.8); c_9 ("competence") have values similar to c_7.

	rdt_2			rdt_3		
category	*Prec*	*Rec*	F_1	*Prec*	*Rec*	F_1
c_1	.970	.963	.966	.968	.966	.967
c_2	.988	.969	.978	.984	.968	.976
c_5	.564	.506	.534	.592	.557	.574
c_7	.434	.288	.346	.413	.271	.327
c_8	.597	.402	.481	.630	.400	.490

Table 6: Precision, recall and F_1 measures for the "best" setups and for 5 categories.

With this values, one can conclude that some categories are *easier* to learn than others. In a way, this *difficulty* is not a surprise if we look at the categories' description: in fact, a document that concerns an "army injured" or a "war prisoner" should be more easy to classify than one that speaks of a "public officer" or "competence", since the abstraction level of the latter is much higher than the former.

Another likely problem is that the classification of the documents was not made by a a single group of people but, instead, by several people along the years and, since there exists overlapping between the set of possible categories, some documents that could be classified on the same group belong, in fact, to different categories.

6. PART-OF-SPEECH TAG EXPERIMENTS

We used the PALAVRAS [Bick, 2000] parser (that performs the syntactic analysis of the documents), to obtain the part-of-speech (POS) tags. This parser was developed in the context of the VISL (Visual Interactive Syntax Learning) project in the Institute of Language and Communication of the University of Southern Denmark[1].

The parser's output is the syntactic analysis of each phrase and the POS tag associated with each word. For example, the morphological tagging of the phrase "O Manuel ofereceu um livro ao seu pai. /Manuel gave a book to his father." is:

```
o [o] <artd> <dem> DET M S
Manuel [Manuel] PRP M S
ofereceu [oferecer] V PS 3S IND VFIN
um [um] <quant> <arti> DET M S
livro [livro] N M S
a [a] <prep>
o [o] <artd> <dem> DET M S
seu [seu] <pron-det> <poss> M S
pai [pai] N M S
```

This Portuguese parser is robust enough to always give an output even for incomplete or incorrect sentences, which

[1] http://www.visl.sdu.dk/

might be the case for the type of documents used in text classification, and has a comparatively low percentage of errors (less than 1% for word class and 3-4% for surface syntax) [Bick, 2003].

The possible morpho-syntactic tags are:

adjective –	`ADJ`
adverb –	`ADV`
article –	`DET`
conjunction –	`CONJ`
interjection –	`IN`
noun –	`N`
numeral –	`NUM`
preposition –	`PREP`
pronoun –	`PRON`
proper noun –	`PRP`
verb –	`V`

Note that Portuguese is a rich morphological language: while nouns and adjectives have 4 forms (two *genders* – male and female and two *numbers* – singular and plural), a regular verb has 66 different forms (two *numbers*, three *persons* – 1st, 2nd and 3rd and five *modes* – indicative, conjunctive, conditional, imperative and infinitive, each with different number of *tenses* ranging from 1 to 5).

6.1 Experiments

Having as a baseline the best setup obtained on the previous sections (here named *base*), we present the characteristics of the dataset in study now. We generated models for the following conjunction of POS tags experiments: `NN`, `VRB`, `NN+ADJ`, `NN+PRP`, `NN+VRB`, `NN+ADJ+PRP`, `NN+ADJ+VRB` and `NN+PRP+VRB`.

Table 7 shows the number of features and the averages per document (of all and distinct features) obtained for each POS tag experiment for original words (rdt_2) and their lemmas (rdt_3).

	$features$		avg_{all}		$avg_{distinct}$	
	rdt_2	rdt_3	rdt_2	rdt_3	rdt_2	rdt_3
`NN`	1168	1026	437	424	126	110
`VRB`	601	542	212	184	120	76
`NN+ADJ`	1535	1349	559	540	175	148
`NN+PRP`	1329	1165	547	514	149	130
`NN+VRB`	1752	1533	638	598	237	179
`NN+ADJ+PRP`	1679	1473	668	630	196	166
`NN+ADJ+VRB`	2122	1855	759	714	285	216
`NN+PRP+VRB`	1917	1669	747	688	260	198
base	2440	2115	802	768	277	215

Table 7: Number of features and averages per document (all and distinct) for each POS experiment.

6.2 Results

For each experiment, we, once again, analysed precision, recall and F_1 measures and calculated the micro- and macro-averaging of the top ten categories. Tables 8 and 9 show these values for each experiment. Once again, the bold faced figures are the "better" ones (for each measure) with no significant difference.

	micro			*macro*		
	Prec	*Rec*	F_1	*Prec*	*Rec*	F_1
`NN`	.887	.753	**.814**	**.795**	.694	.728
`VRB`	**.926**	.683	.786	.701	.611	.642
`NN+ADJ`	.871	.771	**.818**	**.793**	.716	**.745**
`NN+PRP`	.879	.770	**.821**	**.801**	.715	**.746**
`NN+VRB`	.850	.772	.809	.775	.719	.742
`NN+ADJ+PRP`	.862	**.777**	**.817**	.788	**.724**	**.749**
`NN+ADJ+VRB`	.842	**.770**	.809	.776	**.726**	**.747**
`NN+PRP+VRB`	.845	**.776**	.809	.776	**.723**	**.745**
base	.810	.709	.756	.711	.626	.661

Table 8: The *words* setup – precision, recall and F_1 micro- and macro averaging values for each POS experiment.

	micro			*macro*		
	Prec	*Rec*	F_1	*Prec*	*Rec*	F_1
`NN`	.888	.748	.812	**.791**	.690	.721
`VRB`	**.924**	.680	.783	.703	.610	.645
`NN+ADJ`	.880	.762	**.817**	**.796**	.705	.739
`NN+PRP`	.879	.763	**.817**	**.795**	.708	.738
`NN+VRB`	.858	.772	**.813**	.786	.719	**.747**
`NN+ADJ+PRP`	.866	**.783**	**.822**	**.795**	**.731**	**.754**
`NN+ADJ+VRB`	.856	**.783**	**.818**	.790	**.731**	**.756**
`NN+PRP+VRB`	.854	**.784**	**.818**	.786	**.733**	**.754**
base	.815	.714	.761	.717	.632	.667

Table 9: The *lemma* setup – precision, recall and F_1 micro- and macro averaging values for each POS experiment.

As can be noticed in the figures, using certain POS tags it's possible to obtain better values than using all words (the *base* experiment). Apart from using just the verbs – `VRB` (with the words or their lemmas) the experiments were not worse than the *base* one.

The `VRB` experiment produces high precision values and very poor recall ones. The `NN` experiment generated better values than the base experiment, but even better values can be obtained by just adding `ADJ` or `PRP` words (with no significant differences between the two). On the other hand, using three categories doesn't improve the results obtained when using just `NN+ADJ` or `NN+PRP`.

From these both "winning" experiments, the one that has less features is `NN+PRP` with 1329 (rdt_2) and 1165 (rdt_3) against 1535 (rdt_2) and 1349 (rdt_3) of the `NN+ADJ` experiment.

As happend in the IR experiments, there is no significant difference between using the rdt_3 and rdt_3 setups.

Table 10 shows precision, recall and F_1 measures for the `NN+PRP` experiment for the same categories presented in the IR experiments (Table 6).

While c_7, which had the worse values in the first set of exper-

	rdt_2			rdt_3		
category	*Prec*	*Rec*	F_1	*Prec*	*Rec*	F_1
c_1	.975	.971	.973	.974	.969	.972
c_2	.984	.976	.980	.984	.970	.977
c_5	.609	.512	.556	.588	.499	.540
c_7	.492	.239	.322	.425	.179	.251
c_8	.709	.601	.651	.692	.607	.647

Table 10: Precision, recall and F_1 measures for the NN+PRP setup for 5 categories.

iments, improved (with significant differences) by the use of nouns and proper nouns (NN+PRP), c_8 presented worse significant values in all measures and c_1 in some of them. c_2 and c_5 had some better and some worse values in this experiment compared with the one with all words (IR experiment).

7. CONCLUSIONS AND FUTURE WORK

In this work the application of Support Vector Machines to the classification of European Portuguese legal documents was described and evaluated. Several information retrieval techniques were used to reduce, select and weight document words (features). Moreover, the use of part-of-speech information as a selection procedure was also studied.

It was possible to identify a good combination of all these factors obtaining, for the top ten categories, F_1 values of 0.821 (micro-averaging) and 0.746 (macro-averaging): the $rdt_2.scr_1.wgt_3.thr_{400}$.NN+PRP experiment. This means that it is a good approach to use only words tagged as nouns and proper nouns, ranked by the term frequency scoring function and normalised term frequency as the term weighting scheme.

Using the referred combination, it was possible to reduce the number of features from a total 68886 distinct words to 1329 and to increase the F_1 measure for the top 10 categories (using as the baseline the $rdt_1.scr_1.wgt_1.thr_1$ experiment) from 0.687 to 0.821 (micro-averaging) and from 0.531 to 0.746 (macro-averaging).

Using lemmatisation (rdt_3) and NN+ADJ generates similar results. While the latter generates more features than NN+PRP, the former generates less but is also more time-consuming, since we have to obtain the lemma for each word.

Using the mutual information (scr_2) scoring function also produces similar results to the ones obtained with the term frequency (scr_1) one, but the time consumed to rank the features through the mutual information function is not despising.

In conclusion, one can state that linguistic information, such as lemmatisation and part-of speech tags, improves SVM classifiers and strongly reduces the computational complexity of the task.

As future work, and in order confirm these results, we intend to make the same experiments with legal datasets written in other languages and with non-legal datasets. It will be important to evaluate if these results are binded to the Portuguese language and/or the legal domain.

On the other hand, some categories have quite good precision and recall measures while others have quite bad results. We believe these results may be explained by the existence of concepts with distinct levels of abstraction. For instance, we have very specific concepts, such as, "army injured", but we also have more generic ones, such as, "public officer". The classification of abstract concepts is more difficult and requires a more complex approach.

In order to cope with this difficulty, and aiming to develop better classifiers, we intend to address the document representation problem by trying more powerful representations than the bag-of-words, allowing us to use word order and syntactical and/or semantical information in the representation of documents. To achieve this goal we plan to use other kind of kernel such as the string kernel (see, for example, [Shawe-Taylor & Cristianini, 2004]).

8. ACKNOWLEDGEMENTS

We would like to thank the referees for all the useful comments made during the reviewing process.

9. REFERENCES

Al-Kofahi, A.K. Khalid, Vachher, A., & Jackson, P. 2001. A Machine Learning Approach to Prior Case Retrieval. *Pages 88–93 of: Proccedings of the 8th International Conference on Artificial Intelligence and Law – ICAIL'2001.*

Bick, E. 2000. *The Parsing System PALAVRAS – Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework.* Aarhus University Press.

Bick, E. 2003. A Constraint Grammar Based Question Answering System for Portuguese. *Pages 414–418 of: Proceedings of the 11th Portuguese Conference of Artificial Intelligence – EPIA'03.* LNAI Springer Verlag.

Borges, F., Borges, R., & Bourcier, D. 2003. Artificial Neural Networks and Legal Categorization. *Pages 11–20 of: Proccedings of the 16th International Conference on Legal Knowledge Based Systems.* IOS Press.

Bruninghaus, S., & Ashley, K. 1997. Finding factors: learning to classify case opinions under abstract fact categories. *Pages 123–131 of: Proccedings of the 7th International Conference on Artificial Intelligence and Law – ICAIL'1999.* ACM.

Brüninghaus, Stefanie, & Ashley, Kevin D. 2003. Predicting Outcomes of Case-Based Legal Arguments. *Pages 233–242 of: Proccedings of the 9th International Conference on Artificial Intelligence and Law – ICAIL'2003.*

Cortes, C., & Vapnik, V. 1995. Support-vector networks. *Machine Learning*, **20**(3), 273–297.

Cover, Thomas M., & Thomas, Joy A. 1991. *Elements of Information Theory*. Wiley Series in Telecomunication. New York: John Wiley and Sons, Inc.

Gonçalves, T., & Quaresma, P. 2003. A preliminary approach to the multilabel classification problem of Portuguese juridical documents. *Pages 435–444 of:* Moura-Pires, F., & Abreu, S. (eds), *Proceedings of the 11th Portuguese Conference on Artificial Intelligence – EPIA'2003*. LNAI 2902. Évora, Portugal: Springer-Verlag.

Japkowicz, N. 2000. The Class Imbalance Problem: Significance and Strategies. *Pages 111–117 of: Proceedings of the International Conference on Artificial Intelligence – AI'2000*, vol. 1.

Joachims, T. 1998. Text categorization with Support Vector Machines: Learning with many relevant features. *Pages 137–142 of: Proceedings of the 10th European Conference on Machine Learning – ECML'98*.

Joachims, T. 2002. *Learning to Classify Text Using Support Vector Machines*. Kluwer academic Publishers.

Liu, Chao-Lin, Chang, Cheng-Tsung, & Ho, Jim-How. 2003. Classification and Clustering for Case-Based Criminal Summary Judgement. *Pages 252–261 of: Proccedings of the 9th International Conference on Artificial Intelligence and Law – ICAIL'2003*.

Lopes, J.G., Marques, N.C., & Rocio, V.J. 1994. POLARIS: POrtuguese Lexicon Acquisition and Retrieval Interactive System. *Page 665 of: The Practical Applications of Prolog*. Royal Society of Arts.

Mladenić, D., & Grobelnik, M. 1999. Feature selection for unbalanced class distribution and naïve Bayes. *Pages 258–267 of: Proceedings of the 16th International Conference on Machine Learning – ICML'99*.

Nigam, K., McCallum, A.K., Thrun, S., & Mitchell, T. 2000. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, **39**(2/3), 103–134.

Platt, J. 1998. Fast Training of Support Vector Machines using Sequential Minimal Optimization. *In:* Schölkopf, B., Burges, C., & Smola, A. (eds), *Advances in Kernel Methods - Support Vector Learning*. MIT Press.

Quaresma, P., & Rodrigues, I. 2003. PGR: Portuguese Attorney General's Office Decisions on the Web. *Pages 51–61 of:* Bartenstein, Geske, Hannebauer, & Yoshie (eds), *Web-Knowledge Management and Decision Support*. LNCS/LNAI 2543. Springer-Verlag.

Salton, G., & McGill, M. 1983. *Introduction to Modern Information Retrieval*. McGraw-Hill.

Schütze, H., Hull, D., & Pedersen, J. 1995. A comparison of classifiers and document representations for the routing problem. *Pages 229–237 of: Proceedings of the 18th International Conference on Research and Developement in Information Retrieval – SIGIR'95*.

Schweighofer, E., & Merkl, D. 1999. A Learning Technique for Legal Document Analysis. *Pages 156–163 of: Proccedings of the 7th International Conference on Artificial Intelligence and Law – ICAIL'1999*. ACM.

Schweighofer, Erich, Rauber, Andreas, & Dittenbach, Michael. 2001. Automatic text representation, classification and labeling in European law. *Pages 78–87 of: Proccedings of the 8th International Conference on Artificial Intelligence and Law – ICAIL'2001*.

Shawe-Taylor, J., & Cristianini, N. 2004. *Kernel Methods for Pattern Analysis*. Cambridge University Press.

Silva, C.F., Vieira, R., Osorio, F.S., & Quaresma, P. 2004 (August). Mining Linguistically Interpreted Texts. *In: Proceedings of the 5th International Workshop on Linguistically Interpreted Corpora*.

Thompson, Paul. 2001. Automatic categorization of case law. *Pages 70–77 of: Proccedings of the 8th International Conference on Artificial Intelligence and Law – ICAIL'2001*.

Tikhonov, A.N., & Arsenin, V.Y. 1977. *Solution of Ill-Posed Problems*. Washington DC: John Wiley and Sons.

Tong, R., & Appelbaum, L.A. 1994. Machine learning for knowledge-based document routing. *In:* Harman (ed), *Proceedings of the 2nd Text Retrieval Conference*.

Vapnik, V. 1998. *Statistical learning theory*. NY: Wiley.

Wilkins, D., & Pillaipakkamnatt, K. 1997. The effectiveness of machine laerning techniques for predicting time to case disposition. *Pages 39–46 of: Proccedings of the 6th International Conference on Artificial Intelligence and Law – ICAIL'1997*. ACM.

Witten, I., & Frank, E. 1999. *Data Mining: Practical machine learning tools with Java implementations*. Morgan Kaufmann.

Zeleznikow, J., & Stranieri, A. 1995. The Split-up system: Integrating neural networks and rule based reasoning in the legal domain. *Pages 195–194 of: Proccedings of the 5th International Conference on Artificial Intelligence and Law – ICAIL'1995*. ACM.