Multi-label Classification of Croatian Legal Documents
Using EuroVoc Thesaurus

Frane Šarić∗, Bojana Dalbelo Bašić†, Marie-Francine Moens†, Jan Šnajder∗

∗University of Zagreb, Faculty of Electrical Engineering and Computing, Unska 3, 10000 Zagreb, Croatia
frane.saric@gmail.com, {bojana.dalbelo, jan.snajder}@fer.hr
†Department of Computer Science, KU Leuven, Celestijnenlaan 200A, Heverlee 3001, Belgium
sien.moens@cs.kuleuven.be

Abstract
The automatic indexing of legal documents can improve access to legislation. EuroVoc thesaurus has been used to index documents of the European Parliament as well as national legislative. A number of studies exists that address the task of automatic EuroVoc indexing. In this paper we describe the work on EuroVoc indexing of Croatian legislative documents. We focus on the machine learning aspect of the problem. First, we describe the manually indexed legislative documents collection, which we make freely available. Secondly, we describe the multi-label classification experiments on this collection. A challenge of EuroVoc indexing is class sparsity, and we discuss some strategies to address it. Our best model achieves 79.6% precision, 60.2% recall, and 68.6% F1-score.

1. Introduction
Semantic document indexing refers to the assignment of meaningful phrases to a document, typically chosen from a controlled vocabulary or a thesaurus. Document indexing provides an efficient alternative to traditional keyword-based information retrieval, especially in a domain-specific setting. As manual document indexing is a very laborious and costly process, automated indexing methods have been proposed, ranging from the early work of Buchan (1983) to the more recent system by Montejo Ráez et al. (2006). The practical value of indexing legal documents has long been recognized. Acknowledging this fact, the EU has introduced EuroVoc (Hradilova, 1995), a multilingual and multidisciplinary thesaurus covering the activities of the EU, used by the European Parliament as well as the national and regional parliaments in Europe. The EuroVoc thesaurus contains 6797 indexing terms, so-called descriptors, arranged into 21 different fields. The thesaurus is organized hierarchically into eight levels: levels 1 (fields) and 2 (microthesauri) are not used for indexing, while levels 3–8 contain the descriptors. The EuroVoc thesaurus exists in 23 languages of the EU.

In this paper we describe the work on EuroVoc indexing of Croatian legislative documents. Most of this work has been carried out within the CADIAL (Computer Aided Document Indexing for Accessing Legislation) project, in collaboration with the Croatian Information-Documentation Referral Agency (HIDRA). The overall aim of the CADIAL project was to enable public access to legislation. To this end, a publicly accessible semantic search engine has been developed. Furthermore, a computer-aided document indexing system eCADIS has been developed to speed up semantic document indexing. For more details about the CADIAL project, see (Tadić et al., 2009).

The focus of this paper is the machine learning aspect of the problem. Namely, EuroVoc indexing is essentially a multi-label document classification task, which can be addressed using supervised machine learning. The contribution of our work is twofold. First, we describe a new, freely available and manually indexed collection of Croatian legislative documents. Secondly, we describe EuroVoc multi-label classification experiments on this collection. A particular challenge associated with EuroVoc indexing is class sparsity, and we discuss some strategies to address it. Another challenge, as noted by Steinberger et al. (2012), is that document classification is generally more difficult for Slavic languages due to morphological variation, and we also consider ways to overcome this. Although we focus specifically on EuroVoc indexing of documents in Croatian language, we believe our results may transfer well to other languages with similar document collections.

2. Related Work
Most research in supervised learning deals with single label data. However, in many classification tasks, including document and image classification tasks, the training instances do not have a unique meaning and therefore are associated with a set of labels. In this case, multi-label classification (MLC) has to be considered. The key challenge of MLC is the exponentially-sized output space and the dependencies among labels. For a comprehensive overview, see (Zhang and Zhou, 2013; Tsoumakas and Katakis, 2007).

EuroVoc indexing can be considered a large-scale MLC problem. Mencía and Fürnkranz (2010) describe an efficient application of MLC in legal domain, where three types of perceptron-based classifiers are used for EuroVoc indexing of EUR-Lex data. The most common approach to cope with large-scale MLC is to train a classifier for each label independently (Tsoumakas et al., 2008). Boella et al. (2012) use such an approach in combination with a Support Vector Machine (SVM) for EuroVoc MLC of the legislative document collection JRC-Acquis (Steinberger et al., 2006). Steinberger et al. (2012) present JEX, a tool for EuroVoc multi-label classification that can fully automatically assign EuroVoc
descriptors to legal documents for 22 EU languages (excluding Croatian). The tool can be used to speed up human classification process and improve indexing consistency. JEX uses a profile-based category ranking technique: for each descriptor, a vector-space profile is built from the training set, and subsequently the cosine similarity between the descriptor vector profile and the document vector representation is computed to select the k-best descriptors for each document. Daudaravicius (2012) studies the EuroVoc classification performance on JRC-Acquis on three languages of varying morphological complexity – English, Lithuanian, and Finish – as well as the influence of document length and collocation segmentation. Whether linguistic preprocessing techniques, such as lemmatization or POS-tagging, can improve classification performance for highly inflected languages was also investigated by Mohamed et al. (2012). Using JRC JEX tool on parallel legal text collection in four languages, they showed that classification can indeed benefit from POS tagging.

3. Croatian Legislative Document Collection

3.1. Collection Description

The document collection we work with is the result of the CADIAL project and consists of 21,375 legislative documents of the Republic of Croatia published before 2009 in the Official Gazette of the Republic of Croatia (Narodne Novine Republike Hrvatske). The collection includes laws, regulations, executive orders, and law amendments. The collection has been manually indexed with descriptors from EuroVoc and CroVoc. The latter is an extension of EuroVoc compiled by HIDRA, consisting of 7720 descriptors covering mostly names of local institutions and toponyms. Overall, the combined EuroVoc-CroVoc thesaurus consists of 14,547 descriptors.

The manual indexing was carried out in two rounds. In the first round, carried out before 2007, a total of 9225 documents were manually indexed. This part of the collection was used to train the machine learning-based indexer eCADIS. In the second round, carried out from 2007 onward, additional 12,510 documents were indexed (1187 in international treaties, 7129 law amendments, and 4194 additional laws, regulations, and executive orders). To speed up the procedure, in this round the eCADIS indexer was used as a starting point for manual indexing. Subsequently, each document has been manually inspected and the descriptors were revised where necessary. Also, descriptors from the first round were checked and some were revised.

The law amendments have not been indexed, as they in-herit the descriptors of the main regulation they refer to. We therefore did not consider law amendments in our experiments. The final collection that we use consists of 13,205 manually indexed documents, which amounts to 332K unique words and 39.9M tokens. The average document size is about 3K words. We refer to this collection as the NN13205 collection.

3.2. Indexing Principles and Quality

The NN13205 collection was indexed by five professional documentalists according to strict guidelines established by HIDRA. The main principle was to choose descriptors that are likely to match the end-users’ information needs. This transferred to two criteria: specificity and completeness. Specificity means that the most specific descriptors pertaining to document content should be chosen. The more general descriptors were not chosen, as they can be inferred directly from the thesaurus. Completeness means that the assigned descriptors must cover all the main subjects of the document. Essentially, the indexing followed the guidelines set by (ISO, 1985), the UNIMARC guidelines, and best practices developed in HIDRA.

At first sight, the specificity criterion might seem to imply that only the leaf descriptors are assigned to the documents. However, this is not the case, as sometimes the lower levels lack the suitable descriptor. In these cases, the indexers had to back off to a more general descriptor. Consequently, if a document is best described with a number of descriptors, some of them will be more general than the others. In fact, this happens in 23.7% of documents in the NN13205 collection. Note that this effectively introduces extra semantics: although a more specific descriptor implies all the more general ones, explicit assignment of a more general descriptor indicates that the more specific descriptors are not informationally complete for the document.

As a means of quality control, indexing has undergone periodic revisions to ensure consistency. This was done either by inspecting all documents indexed with the same descriptor or by inspecting groups of topically related documents. Unfortunately, no methodology was established to measure the inter-annotator agreement; in particular, no document was ever indexed by more than a single documentalist.

As a consequence, we cannot estimate the overall quality of manual annotation using inter-annotator agreement as a proxy. Furthermore, the lack of inter-annotator estimate is troublesome from a machine learning perspective because it prevents us to establish the ceiling performance for a machine learning model on this task.

3.3. Indexing Statistics

In total, 3951 different EuroVoc descriptors were used to index the 13,205 documents. Indexers typically assigned up to 10 different descriptors to each document. The total number of descriptor assignments is 48,123, which amounts to 3.6 descriptors per document (see Fig. 1a). From a machine learning perspective, the major problem of NN13205 is that it is sparsely labeled. Out of 3951 descriptors assigned, 1078 were assigned to a single document and 2867 were assigned to less than ten documents, as shown in Fig. 1b. For comparison, the Reuters news stories corpus RCV1 (Rose et al., 2002), the benchmark collection for document classification, contains as much as 30K documents and only 100 indexing terms.

It is also interesting to compare our indexing statistics against that of the JRC-Acquis corpus (Steinberger et al., 2006). The statistics suggest that the class sparsity problem is more pronounced for the NN13205 than for the JRC-Acquis. For any single language, the JRC-Acquis has

5 Available under CC BY-NC-SA 3.0 from http://takelab.fer.hr/data/nn13205
approximately 2.5 times more documents than NN13205. While NN13205 documents have anywhere from 1 to 36 assigned descriptors (avg. 3.6), JRC-Acquis documents have from 2 to 17 assigned descriptors (avg. 5.4). The total number of different descriptors in the JRC-Acquis ranges from 3584 to 4234, depending on the language.

4. Classification Experiments

4.1. Classification Model

EuroVoc indexing is essentially a hierarchical MLC problem, since each document may be assigned several descriptors. As noted in Section 2, the simplest way to address an MLC problem is to frame it as a binary classification problem, by training a separate classifier for each label. This is the approach we adopt here.

A variety of classifiers can be used for text classification. We use a Support Vector Machine (SVM) (Joachims et al., 1998), which has shown to be competitive on a wide range of classification tasks, including text classification. We use the LIBLINEAR (Fan et al., 2008) implementation, and the particular model we use is the L2-regularized L2-loss SVM. Note that we use a linear kernel, since the high number of features typical for text classification problems usually implies linear separability.

To train the binary SVM classifiers, we adopt the one-vs-rest scheme: we train a separate classifier for each EuroVoc descriptor, using documents indexed with that descriptor as the positive instances and all other documents as the negative instances. If the classifier output exceeds a certain threshold, then the descriptor is assigned to the document, otherwise it is not. For improved accuracy, we additionally optimize the threshold of each individual model using the SCutFBR.1 method proposed by Yang (2001).

Another aspect that we do not explicitly consider here is hierarchy. EuroVoc indexing could be cast as a hierarchical classification problem, which has been extensively studied in the literature. A technique using combination of Bayes with SVM classifiers proposed by Cesa-Bianchi et al. (2006) shows good results, although the advantage is not so clear on real data sets. Most hierarchical models permit only leaf labels to be assigned to instances. Models have been proposed, such as the one by Sun and Lim (2001), that allow also the inner nodes to be used as labels, which is what would be required in our case because of the indexing principles used for the NN13205 collection (cf. Section 3.2). We leave the issue of hierarchical classification for future work.

4.2. Experimental Setup

To obtain reliable error estimates and to prevent overfitting the model, we used a 5×3 nested cross-validation for model selection. Because of the large number of classifiers involved, for each model we consider only three values (1, 10, and 100) for the regularization parameter C.

We evaluate the classifiers in terms of commonly used performance measures: precision (P), recall (R), and the F1-score (the harmonic mean of P and R). Because we deal with multiple classes, we calculate the micro-average of these measures. We additionally calculate the macro-averaged F1-score (F1-score averaged over descriptors), which is more sensitive to the performance of the model on sparse classes. Note that micro P, micro R, and micro F1-score generally differ from each other because this is a multi-label problem, unlike in a multi-class (one-class-per-instance) classification problem.

As noted by Lewis et al. (2004), class sparseness raises the issue of how to compute the F1 score on under-represented classes. This has a significant impact on the overall result because NN13205 has many such classes. Stratified sampling is not an option here because the collection is sparsely multi-labeled. Instead, we decided to average the performance metrics over classes with one or more positive test instances, as proposed by Lewis et al. (2004). If, for a given descriptor, only documents from the test set are indexed with it, then a model for this descriptor cannot be trained and the F1-score is set to 0 for that descriptor. Note that this is a more realistic evaluation setting than averaging over classes with one or more positive training examples.

It should be noted that other evaluation schemes are applicable in our setting, such as the multi-label classification evaluation (e.g., Tsoumakas and Katakis (2007)) and hierarchical classification evaluation (e.g., category-similarity measures proposed by Sun and Lim (2001)). We leave this line of research for future work.

---

6 Subsampling negative instances, typically used to balance the classes, did not improve the performance.
4.3. Preprocessing

Prior to constructing the feature vector, we remove from each document the stop words using a manually compiled list of 2000 inflected stop words (conjunctions, prepositions, pronouns, numbers, etc.). The large number of features often poses an efficiency problem in text classification. This also applies to EuroVoc classification, where a large number of models has to be trained. To make training more efficient, we decided to employ a feature selection procedure (Yang and Pedersen, 1997). Preliminary experiments have indicated that we can discard 10% of features as being more efficient, we decided to employ a feature selection procedure. For lemmatization, we use an automatically acquired inflectional lexicon of Croatian compiled by Šnajder et al. (2008). For stemming, we use the rule-based inflectional stemmer developed by Ljubešić et al. (2007). Lemmatization is a more accurate technique than stemming, which also takes into account the homographs by normalizing them to several lemmas. Morphological normalization reduces the number of features to ~190K with lemmatization and ~170K with stemming, which amounts to a reduction of about 29% and 37%, respectively.

4.4. Baseline Results

We first evaluate a model trained on the complete NN13205 collection, utilizing 3405 classifiers, one for each descriptor used. The results are summarized in Table 1. Expectedly, macro F1-score is lower than micro F1-score because the performance on sparse categories is generally lower. For the same reason, the recall is substantially lower than precision because the model generally fails to assign the rarely used descriptors. Morphological normalization improves the overall performance (4% relative performance improvement in macro F1-score), although it decreases precision. Lemmatization and stemming seem to be equally effective. In all subsequent experiments, we use lemmatization. As noted earlier, EuroVoc classification is known to suffer from class sparsity. To account for this, Steinberger et al. (2012) discard the descriptors that were assigned less than four times in JRC-Acquis. To gain an insight into how class sparsity affects the performance on NN13205 collection, we also discard the rarely used descriptors and re-train the model. We experiment with a cut-off threshold ranging from 2 (descriptor has to be assigned to at least two documents) to 10 (descriptors have to be assigned to at least ten documents). The results are shown in Table 2. The recall increases proportionally to the cut-off threshold, while precision increases only marginally. When only the descriptors assigned to ten or more documents are considered, micro recall improves by 6.5 percent points, resulting in a relative improvement of macro F1-score of almost 25%.

It is perhaps interesting to compare our results to that of Steinberger et al. (2012), obtained on the JRC-Acquis corpus. Steinberger et al. use a documents-per-descriptor cut-off of 4, but always assign six descriptors per document, while we assign descriptors independently of the other descriptors, based on the classifier output and the threshold. As they computed a non-standard variant of the F1-score, we computed the modified F1-score in the same way for the sake of comparison. The modified F1-score on JRC-Acquis varies from 44.2% to 54.4% depending on the language. The modified F1-score at NN13205 with a cut-off of four is 60.8%. Note, however, that this comparison is for indicative purposes only, as the collections are different.

4.5. Addressing Class Sparsity

Discarding rarely used descriptors does not really address the issue of class sparsity but rather avoids it. The question arises how to practically address this issue. An intuitive approach is to rely on the hierarchical nature of the EuroVoc thesaurus. We experiment with three such techniques.

Descriptor lifting. The first technique is simply to lift the descriptors up the taxonomy tree. We replace all descriptor assignments with the corresponding microthesauri or fields, i.e., we effectively lift the descriptors to the second or first level of the EuroVoc thesaurus. The results are shown in Table 3. Expectedly, lifting to level 2 substantially improves the recall (cf. Table 1), while precision remains unaffected, suggesting that most false positive assignments occur within microthesauri. Lifting to level 1 improves re-

---

Table 1: Performance on the complete NN13205 collection

<table>
<thead>
<tr>
<th>Features</th>
<th>Micro P</th>
<th>Micro R</th>
<th>Micro F1</th>
<th>Macro F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Words</td>
<td>82.6</td>
<td>56.5</td>
<td>67.1</td>
<td>45.9</td>
</tr>
<tr>
<td>Lemmas</td>
<td>80.7</td>
<td>58.8</td>
<td>68.0</td>
<td>47.8</td>
</tr>
<tr>
<td>Stems</td>
<td>80.2</td>
<td>58.7</td>
<td>67.8</td>
<td>47.9</td>
</tr>
</tbody>
</table>

Table 2: Performance with documents-per-descriptor cut-off

<table>
<thead>
<tr>
<th>Cut-off</th>
<th>Micro P</th>
<th>Micro R</th>
<th>Micro F1</th>
<th>Macro F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>80.7</td>
<td>58.8</td>
<td>68.0</td>
<td>47.8</td>
</tr>
<tr>
<td>3</td>
<td>80.6</td>
<td>59.5</td>
<td>68.4</td>
<td>50.0</td>
</tr>
<tr>
<td>4</td>
<td>80.6</td>
<td>60.2</td>
<td>68.9</td>
<td>52.2</td>
</tr>
<tr>
<td>5</td>
<td>80.6</td>
<td>60.9</td>
<td>69.4</td>
<td>54.1</td>
</tr>
<tr>
<td>6</td>
<td>80.6</td>
<td>61.5</td>
<td>69.8</td>
<td>55.7</td>
</tr>
<tr>
<td>7</td>
<td>80.6</td>
<td>61.9</td>
<td>70.0</td>
<td>56.4</td>
</tr>
<tr>
<td>8</td>
<td>80.7</td>
<td>62.3</td>
<td>70.3</td>
<td>57.3</td>
</tr>
<tr>
<td>9</td>
<td>80.9</td>
<td>62.8</td>
<td>70.7</td>
<td>58.7</td>
</tr>
<tr>
<td>10</td>
<td>81.1</td>
<td>63.3</td>
<td>71.9</td>
<td>59.5</td>
</tr>
</tbody>
</table>

---

7We base this assessment on the analysis of JEX source code.
add descriptors to the original document collection:
upping set. We experimented with two schemes, both of which
(smoothened) to cover some classes not present in the train-
to every node in the class hierarchy can be redistributed
to combat class sparsity transforms the training set in a way
Descriptor expansion. The other technique we tried out
based on fields or microthesauri.
search, in which the retrieved documents could be grouped
such coarse-grained EuroVoc classification, such as faceted
down the search. Other applications could also benefit from
such coarse-grained EuroVoc classification, such as faceted
search, in which the retrieved documents could be grouped
based on fields or microthesauri.

Descriptor expansion. The other technique we tried out
to combat class sparsity transforms the training set in a way
that incorporates information stored in the descriptor hier-
archy. The intuition is that the probability mass assigned
to every node in the class hierarchy can be redistributed
(smoothened) to cover some classes not present in the train-
ing set. We experimented with two schemes, both of which
add descriptors to the original document collection: upward expansion (adding parent descriptors all the way up to
the third level of the taxonomy) and downward expansion
(adding child descriptors to the immediately lower level).
Note that, since we work with taxonomic relations, upward expansion introduces no noise, while downward information
does. In the latter case, the intuition behind descriptor expansion is that human indexers are not always consistent
when deciding whether a parent or a child class should be
selected, thus adding new descriptors with smaller weights
to documents in the training set models this uncertainty in
a simple way. The decision whether to apply expansion
on a descriptor is done at the level of the whole collection,
by optimizing the F1-score of that descriptor on the valida-
tion set (within the nested cross-validation loop, cf. Section
4.2.).
The classification results with descriptor expansion tech-
niques are shown in Table 4. Upward expansion leads to
slight improvements in performance (cf. Table 1), while
downward expansion decreases the performance.

<table>
<thead>
<tr>
<th>Expansion</th>
<th>Micro P</th>
<th>Micro R</th>
<th>Micro F1</th>
<th>Macro F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Upward</td>
<td>79.6</td>
<td>60.2</td>
<td>68.6</td>
<td>48.0</td>
</tr>
<tr>
<td>Downward</td>
<td>72.7</td>
<td>57.2</td>
<td>64.0</td>
<td>43.8</td>
</tr>
</tbody>
</table>

Table 4: Performance with descriptor expansion techniques

Recall optimization. The last technique we considered is
to optimize the threshold of each model to maximize the re-
call. As the above experiments have shown, low recall can
be traced down to low performance on sparse classes. By
inverse logic then, we hope to address the problem of class
sparsity by directly optimizing the recall. To this end, we
again optimize the threshold of each individual model using
the SCutFBR.1 method proposed by Yang (2001), only this
time we optimize the F2-score instead of F1-score. The F2-
score weights recall twice as much as precision. The results
are shown in Table 5, alongside the previous results with
F1-score optimization. F2-score optimization improves the
recall by almost 5 percent points, however it decreases the
precision by over 10 percent points. Overall, the macro F2-
score gets improved by 1.1 percent points.

5. Conclusion
We have described the work on multi-label classification
of Croatian legislative documents with the descriptors
from EuroVoc. We presented NN13205, a manually
indexed document collection of Croatian legislative docu-
ments, which is now freely available. We performed sev-
eral multi-label classification experiments on this collec-
tion. We considered several techniques to address the class
sparsity problem. In particular, using upward expansion of
descriptors we were able to improve the performance of the
classifier, reaching 79.6% precision, 60.2% recall, and
68.6% micro F1-score.

Table 5: Performance with F2 (recall) optimization

<table>
<thead>
<tr>
<th>Objective</th>
<th>Micro P</th>
<th>Micro R</th>
<th>Macro F1</th>
<th>Macro F2</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>80.7</td>
<td>58.8</td>
<td>47.8</td>
<td>48.0</td>
</tr>
<tr>
<td>F2</td>
<td>70.1</td>
<td>63.6</td>
<td>47.6</td>
<td>49.1</td>
</tr>
</tbody>
</table>

There are a number of interesting directions for future
work. First, it would be useful to obtain an estimate of the
inter-annotator agreement on the NN13205. From a ma-
chine learning perspective, it would be interesting to con-
sider multi-label classification models, hierarchical classifi-
cation models, as well as combinations thereof, such as the
HOMER algorithm proposed by Tsoumakas et al. (2008).
Evaluation that takes into account multiple labels and hi-
erarchy could also be considered. Finally, an interesting
direction for future work are the methods for improving of
annotation quality based on semi-supervised active learn-
ing, perhaps along the lines of (Settles, 2011) and (Ragha-
van and Allan, 2007).

Acknowledgments. We thank the Croatian Information-
Documentation Referral Agency (HIDRA), now known as
the Digital Information Documentation Office of the Gov-
ernment of the Republic of Croatia, for their support and
for allowing to make NN13205 collection publicly avail-
able. Special thanks go to Maja Cvitaš and Neda Erceg for
their assistance and advice. We also thank all the partici-
pants involved in the CADIAL project.
6. References


