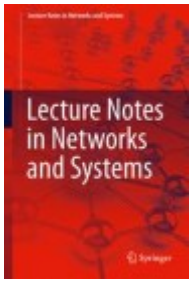


Book series

Springer Lecture Notes in Networks and Systems

Electronic ISSN
2367-3389
Print ISSN
2367-3370

ICIST2025
International Conference on Information Society and Technologies
Kopaonik, Serbia 9-12 March, 2025

15th International Conference on Information Society and Technology (ICIST),

2025

Springer Lecture Notes in Networks and Systems

ICIST 2025 Conference program

Tuesday 11th of March 2025

PAPER:

80 **Named Entity Recognition for Serbian Legal Documents: Design, Methodology and Dataset Development**

AUTHORS:

Vladimir Kalušev (The Institute for Artificial Intelligence Research and Development of Serbia)*;

Branko Brkljač (Faculty of Technical Sciences, University of Novi Sad)

Named Entity Recognition for Serbian Legal Documents: Design, Methodology and Dataset Development

Vladimir Kalušev[1][0009-0005-8851-5790] and Branko Brkljač[2][0000-0001-7932-6676]

[1] Visual Computing & Perception Group, The Institute for Artificial Intelligence Research and Development of Serbia, Novi Sad, Republic of Serbia
vladimir.kalusev@ivi.ac.rs
[2] Dept. of Power, Electronic and Telecommunication Engineering, Faculty of Technical Sciences, University of Novi Sad, Novi Sad, Republic of Serbia
brkljacb@uns.ac.rs

Abstract. Recent advancements in the field of natural language processing (NLP) and especially large language models (LLMs) and their numerous applications have brought research attention to design of different document processing tools and enhancements in the process of document archiving, search and retrieval. Domain of official, legal documents is especially interesting due to vast amount of data generated on the daily basis, as well as the significant community of interested practitioners (lawyers, law offices, administrative workers, state institutions and citizens). Providing efficient ways for automation of everyday work involving legal documents is therefore expected to have significant impact in different fields. In this work we present one LLM based solution for Named Entity Recognition (NER) in the case of legal documents written in Serbian language. It leverages on the pre-trained bidirectional encoder representations from transformers (BERT), which had been carefully adapted to the specific task of identifying and classifying specific data points from textual content. Besides novel dataset development for Serbian language (involving public court rulings), presented system design and applied methodology, the paper also discusses achieved performance metrics and their implications for objective assessment of the proposed solution. Performed cross-validation tests on the created manually labeled dataset with mean F_1 score of 0.96 and additional results on the examples of intentionally modified text inputs confirm applicability of the proposed system design and robustness of the developed NER solution.

Keywords: Named Entity Recognition (NER), legal documents, BERT, language model, NER4Legal_SRB.

1 Introduction

Named Entity Recognition (NER) represents identification and classification of named entities in certain text or document, where named entities are typically noun phrases or predefined categories that refer to some specific object, person, places, dates, or other domain specific entities [1]. These tools are often used for preprocessing or analysis of text documents for the purpose of information extraction, document tagging, retrieval

or search. Thus, the possible use cases for NER tools are very diverse and depend on specific text domain and application. In the context of legal documents, possibility to automatically extract structured information about involved parties (places, reference numbers, court names, dates, laws, official gazette, money amounts, etc.) allows precise archiving, design of efficient search engines and question answering (QA) tools. Since such information is usually the most significant in the document, extraction of key entity values also makes the text summarization, document classification and sentiment analysis easier to perform. Some of the examples of contemporary NER tools specifically targeting legal documents are described in [2] for Turkish, [3] for German, and in [4]–[6] for English language, demonstrating significant interest in this application area. Since Serbian language is still considered as low-resource in context of LLM development and different downstream applications. Thus, NER tools for legal documents in Serbian are still rare and uncommon among practitioners.

In order to overcome existing challenges, and contribute towards democratization of LLM technology and its proliferation among Serbian speaking community, we demonstrate design and development of one specific NLP downstream application that leverages on fine tuning of pre-trained model (PTM). Proposed NER for Serbian legal documents leverages on the BERT type PTM [7], which was previously developed for Serbian and other south Slavic languages by [8]. Presented solution and reported results confirm applicability of proposed design for PTM task adaptation in the case of low-resource languages, and especially in the domain of official legal texts. Therefore, it is expected that presented methodology will motivate further development of similar language tools for Serbian language. According to the conducted experiments on the annotated dataset consisting of 75 unique appellate court rulings from standard legal practice, the proposed NER model achieves mean recognition accuracy of 0,99 and individual recognition F_1 measures in the range between 0,89 and 0,99 over 15 NER categories corresponding to 8 unique named entity (NE) types characteristic for such legal documents. Besides cross-validation, additional tests involving noised textual data confirm robustness of NER model and its applicability for the specific task. The NER4Legal_SRB model parameters and proposed dataset are freely available in the following repository: https://huggingface.co/kalusev/NER4Legal_SRB.

The rest of the paper is organized as follows, in Section 2 we provide an overview of NER design principles and contemporary approaches involving PTM and different NLP representation learning techniques. In Section 3 are described labeling process and created legal texts dataset, as well as NER model architecture and PTM adaptation to downstream NER task by low-resource fine-tuning over developed dataset. Section 4 presents experimental results and their discussion. Finally, Section 5 indicates directions for future work.

2 Related work

There are different design approaches for NER tools, which are trying to capture intricate language context in order to extract and recognize specific entity values. Entity meaning is usually tied to its surrounding context, which makes the rule based [9], [10],

and dictionary based methods [11] less favorable in contrast to machine learning (ML) based approaches [4], [12], [13] or hybrid architectures [14], [15].

Traditional NER methods were typically relying on handcrafted features capturing short-distance relations between the words in the sequence, and lacking the ability to consider bidirectional word relationships. As a result, they often fell short when dealing with complex linguistic scenarios where an entity's meaning reflects the surrounding context. Supervised NER solves a multi-class classification problem or a sequence labeling task, where each of the labeled training samples is represented by the corresponding feature vector, and the corresponding ML model is used to recognize new samples from the text. Depending on the classification model, there had been various learning approaches mainly based on sequence modelling capabilities of Hidden Markov Models (HMMs) [16], Conditional Random Fields (CRFs) [17], [18]. Such approaches were usually relying on fixed word embeddings and limited length observation windows over tokens (a single words or subword units in the input text) for feature engineering, as well as decision trees [19] or a set of binary Support Vector Machines (SVMs) [20] on the part of the learning task. A typical example of semi-supervised sequence modeling approach for NER is [21], where the K-Nearest Neighbors (KNN) classifier is used for pre-labeling of tweet data, after which the CRF model is applied in the sequential manner in order to produce the final predicted labels sequence.

In order to capture non-trivial long-distance dependencies in word or token sequences, neural networks capable of processing variable length inputs, like the recurrent neural network (RNN) and the long-short term memory (LSTM) units with the forget gate, were applied to NER classification tasks, which brought a significant performance improvement over the previous approaches [21]. Most recently, the concepts of bi-directional LSTMs and CNNs that learn both character- and word-level features were further improved with the introduction of pre-trained transformer based bi-directional representations provided by BERT type [7] language models. Such contextualized language-model embeddings, comprising of token position, segment and token embedding are usually characterized as hybrid representations.

The key for development of cost effective solutions for different NLP tasks is ability to exploit learned representations of input data (inherently learned by LLM pre-training) and perform low-resource model adaptation in domain specific downstream tasks. It was made possible by recent advancements in self-supervised training of LLM architectures that are designed in the style of encoder, decoder or encoder-decoder deep neural networks (DNNs). BERT [7] or bidirectional encoder representations from transformers are particularly well suited for NER task due to self-attention mechanism, which means that the encoder considers the entire context (e.g. in total up to 512 tokens for sentence, or multiple sentences in the paragraph) when predicting the category for a specific token, including observations from the past and future (i.e. both preceding and following tokens), due to its bidirectional training and structure. On the other hand, decoder type PTMs, like GPT [22], are generally considered as less suitable for NER and similar NLP tasks like sentiment analysis and masked word prediction, due to unidirectional structure of decoder type PTMs that is well suited for word prediction and

NLP tasks involving text generation, like text summarization, text completition or machine type translation. This was made possible by proposal of various learning strategies that have significantly improved representation learning by exploiting vast corpora of unannotated data. These include word-level objectives like causal language modeling (CLM) in [22], masked language modeling (MLM) in [7] and its span-level generalization in [23], replaced (token detection) language modeling (RLM) in [24], or denoising language modeling (DLM) in [25]. Similarly, their sentence-level counter parts like next sentence prediction (NSP) in [7], sentence order prediction (SOP) in [26] or sentence contrastive learning (SCL) in [27]. However, it should be noted that PTM performance varies depending on the type of downstream task, as well as the implementation, as suggested by [28], where it was shown that BERT type [7] baseline performs better in comparison to ALBERT model [26] on NER tasks, despite improvements that were brought by [26] over [7] (e.g. lower memory consumption and increased training speed, without the NSP strategy [29]).

When it comes to PTM based solutions for Serbian NLP, besides BERTić [8] and its derivatives for QA [30] and NER [31], notable works relying on other PTMs also include learned embedding models proposed in [32] and [33]. Significant efforts were also put into Serbian specific NER solutions proposed in [34], [35], while [36] considers the problem of using Serbian specific BERT based PTMs instead of multilingual BERTs [7], [37] or south Slavic BERT models [8]. As pointed out by [38], in the recent period there have been several attempts of developing Serbian specific PTMs like the [39] PTM based on RoBERTa architecture [29]. However, when it comes to downstream tasks, according to [38] and [36], NER solutions based on Serbian specific PTMs achieve similar performance to NER models fine-tuned on BERTić [8], as measured by NER experiments involving seven entities: demonyms (DEMO), professions and titles (ROLE), works of art (WORK), person names (PERS), places (LOC), events (EVENT) and organizations (ORG); defined in SrpELTeC dataset proposed in [35].

3 Methods and data

Specific challenges for NER in legal documents usually stem from regulatory and compliance complexities, which come from the unique set of laws and regulations in each country, besides the language barrier that can be regarded as significant problem in the case of PTMs adaptation for languages with low NLP resources. From the literature it is known that the pre-training of BERT type models on the domain specific corpora (by adaptation or from scratch) can bring certain performance gains, like in the case of Legal-BERT [4]. However, in practice such approach can be unfeasible for resource constrained solution designs. Similarly, our approach is more baseline in comparison to [5], where the Legal-LUKE model was trained from the beginning to solve MLM and NER at the same time (predict both words and entities, i.e. achieve legal-contextualized and entity-aware representations).

In this work we have decided to rely on the existing PTM proposed in [8] and focus on the development of small scale dataset of legal texts that would allow us to demonstrate feasibility of fine-tuning BERTić [8] model for the specific downstream task of

recognizing 8 named entities (NEs) in Serbian legal documents. Defined NEs include: names of the courts (COURT), calendar dates (DATE), final rulings on the matter at hand (DECISION), names and abbreviations of written laws issued by the government (LAW), money values (MONEY), names of the announcements regarding new laws, amendments, and regulations exclusively from the official gazette of the Republic of Serbia (OFFICIAL GAZETTE), full names of the persons and anonymized abbreviations (NAME), alphanumeric designations of the court rulings (REFERENCE). The number and type of defined NEs is similar to legal NER solution described in [2].

Adopted approach can be regarded as similar to methods proposed in [40], [41], in the sense that the goal was to design effective NER system with the small amount of training data for the fine-tuning of the existing PTM. However, in comparison to [40] proposed dataset is fully annotated, and we avoid training on partially annotated legal documents.

3.1 Dataset development

Legal texts, and especially court rulings, are known to contain long, nested and syntactically complex sentences, which are made of formal and domain specific language, with presence of complex abbreviations and cross-referencing on other documents. Although the process of digitization is ubiquitous, including optical character recognition (OCR) of the old archives, there is still a lack of annotated legal datasets in Serbian language, and a challenge of constantly evolving legal frameworks. Additional challenge is also that NEs in legal documents can contain synonyms, abbreviations, and misspellings.

In order to most efficiently cover all types of legal documents in which previously described 8 NEs appear, we have decided to exclusively focus on currently available examples of Serbian judicial practice, which are available from the official website [42] of the Ministry of Justice. Such decision was motivated by the fact that this repository contains official public documents (court rulings) that were carefully chosen by the repository editors in order to provide representative samples of Serbian judicial practice in appellate courts. Since these rulings, Fig. 1a, are mostly uniform in structure and cover appellations from municipal and high courts in Serbia, present NEs are quite diverse in both their values and context appearances. In contrast to Legal-BERT [4], we have not considered official law texts, which e.g. do not contain person names and court ruling references, while other NEs usually do not appear in the similar context as in the court rulings. In total 75 documents addressing non-economic damages were collected from [42].

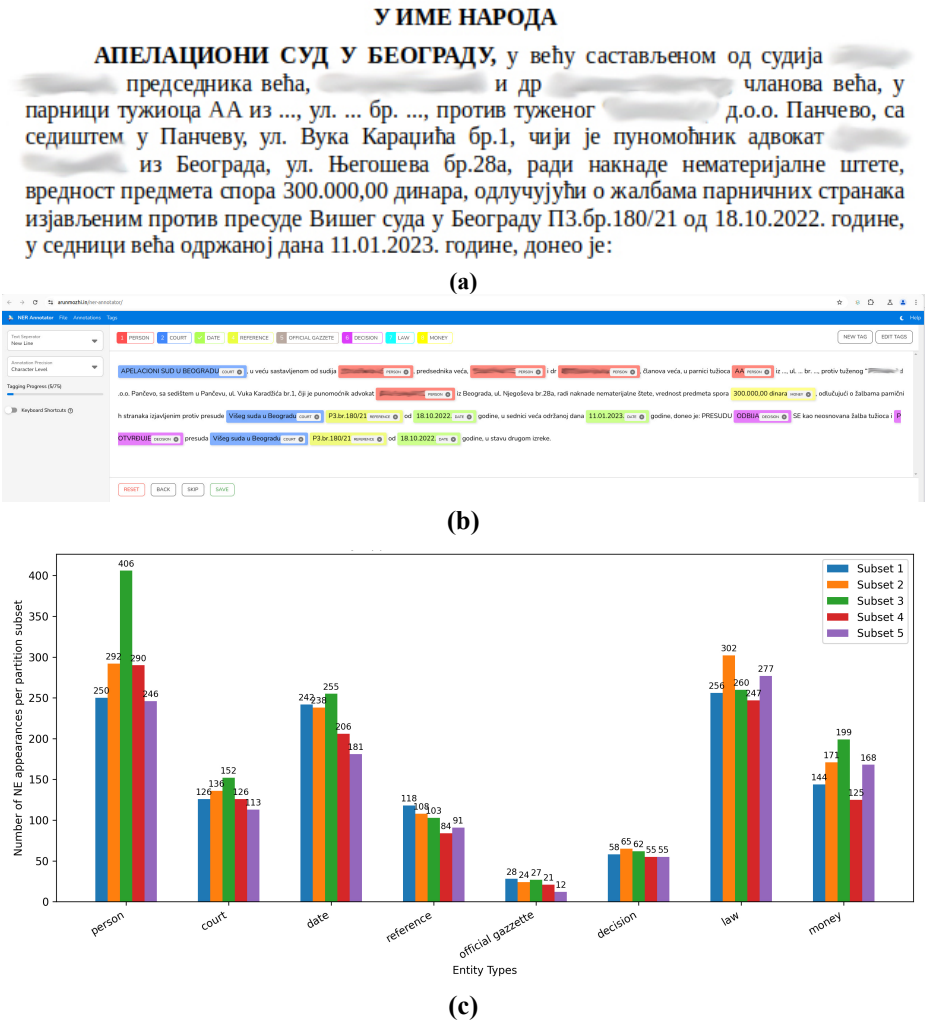


Fig. 1. (a) An illustration of original court ruling in Cyrillic script; (b) annotation process with BIO scheme; (c) number of NE types appearances per each cross-validation subset (random sampling procedure is described in Algorithm 1)

Since Serbian language can be written in both Cyrillic and Latin alphabet, and while BERTić [8] PTM is producing more tokens in tokenization process of Cyrillic texts, for the purpose of NER design demonstration we have decided to perform transliteration of original documents from Cyrillic to Latin script. Since BERT type models are usually accepting up to 512 token length for encoder input, original documents were processed into one sentence per line textual files. Such data were manually annotated by using the tool available in [43], Fig. 1b, with character level annotation precision (entity selection without white-spaces at the ends and punctuation marks) and outputs saved in JSON file format.

In the case of entities that consist of several words, there are different strategies for NEs annotation [44], e.g. IO, BIO (or IOB), IOE, IOBES, IE, BIES (B(begin)- and E(end)-, mark the first and the last token of an entity, S(single)- single token entity, while I(inside)- and O(outside)- mark the tokens within and outside of the entities, respectively). It means that the number of instances of O category (non-entity words) dominates the dataset. Although IO scheme was found to outperform others [45], it comes with disadvantage of not being able to distinguish between consecutive NEs, while the most common BIO performs similarly to others and was therefore applied, Fig. 1b. Note that sentences without any NE were not taken into final annotated dataset. Thus, annotation of 75 legal documents resulted in total 2172 sentences containing NEs, i.e. 758012 characters which have produced 183543 tokens after applied WordPiece tokenization [46], for which the same [47] tokenizer implementation as in selected PTM [8] was utilized. Regarding the NE distribution, there were: 1484 ”person”, 653 ”court”, 1122 ”date”, 504 ”reference”, 112 ”official gazette”, 295 ”decision”, 1342 ”law”, and 807 ”money” appearances, in total 6319. After application of BIO annotation scheme, instead of initial 8+1 entity types (8 described NEs and additional O-type) multi word NEs in the dataset produced in total 14+1 classification categories (”date” and ”decision” NE do not produce an I-type token label).

3.2 Model development

Adopted PTM was pre-trained using ELECTRA framework [24], i.e. replaced token detection language modeling (RLM), which is characterized by the following optimization objective:

$$\mathcal{L}_{PTM} = \mathcal{L}_G + \Lambda\mathcal{L}_D \tag{1}$$

where $\mathcal{L}_G$ denotes the standard MLM pre-training of generative BERT encoder *G*, while $\mathcal{L}_D$ is the objective of the RLM specific discriminator *D*. If we denote with $\widehat{x_t}$ masking of token *t* in MLM, and with $\widetilde{x_t}$ token replacement with the word that was generated by *G* in RLM, then the corresponding objectives can be defined by probabilities P_G and P_D, which correspond to probability of generator *G* correctly predicting the masked word x_t in sequence $\mathbf{x}$, or probability of discriminator *D* correctly detecting that the original word x_t was replaced by the generated $\widetilde{x_t}$:

$$\mathcal{L}_G = -\sum_{t\in M} y_t \log P_G\left(x_t \mid \widehat{x_t}\right) \tag{2}$$

$$\mathcal{L}_D = -\sum_{t=1}^{T}[y_t \log P_D(y_t = 1 \mid \widetilde{x_t})] + (1 - y_t)\log P_D(y_t = 0 \mid \widetilde{x_t}) \tag{3}$$

where $y_t = 1$ indicates the word replacement or masking operation.

We note that *G* is not trained in an adversarial fashion, characteristic for generative adversarial networks (GANs), but separately from *D*, while $\mathcal{L}_D$ contains two terms encompassing discriminator decisions in both cases (when the original word x_t in the sequence is present at the input of *D*, as well as when it is replaced by the MLM generator *G*), which increases the number of tokens *T* contributing to model update. This approach makes training more efficient, as the PTM learns from every token rather than

just masked ones. Pre-trained discriminator *D* is then fine-tuned on sentences from legal documents for token classification task with 15 unique labels (categories) corresponding to described NER in Section 3.1.

3.3 Experimental setup

In order to effectively perform and assess proposed downstream adaptation of PTM, the following statistical crossvalidation procedure was used. Initially created dataset with imbalanced number of tokens per NE types was first randomly shuffled at document level into five disjunct subsets, i.e. random 5-fold cross-validation partition, as shown in Fig. 1c. It is achieved by iterative clustering procedure described in Algorithm 1, consisting of grouping of documents with similar NE distributions:

Algorithm 1

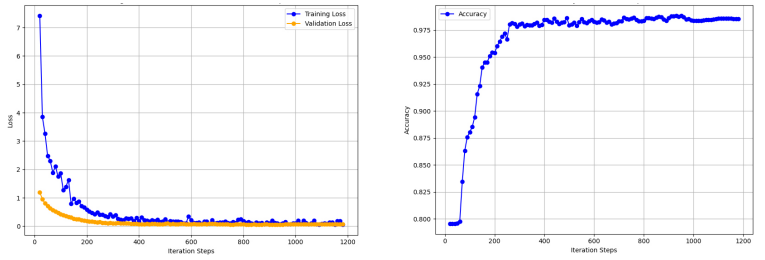
procedure TRAINING/TEST SET RANDOM PARTITION
  Initialization: *$\mathcal{K}$ annotated documents, with $\mathcal{C}$ NE types (categories); set number of partition $\mathcal{P}$ subsets K, $\mathcal{P} = \{S_1, ..., S_K\}$*
  Iterative procedure:
  # 1: *Assign feature vector f_i to each document D_i, reflecting the NE content in labeled sentences, e.g. normalized histogram of NE appearances in D_i*
  # 2: *Group D_i, $i = 1..\mathcal{K}$, into K subsets $\tilde{S}_k$ based on similarity in feature space $f_i \in \mathbb{R}^{\mathcal{C}}$, e.g. by K-means clustering with L_p distance*:
    $\min_{\{\tilde{S}_1, \tilde{S}_2, ..., \tilde{S}_K\}} \sum_{k=1}^{K} \sum_{f_i \in \tilde{S}_k} \|x_i - \mu_k\|_p^p$
  # 3: *By sampling without replacement, randomly select one document per each $\tilde{S}_k$ and assign that document to final partition subset S_k. If some of $\tilde{S}_k$ become empty, continue the iterative procedure over remaining ones until all D_i, $i = 1..\mathcal{K}$ are assigned to some $S_k \in \mathcal{P}$.*
  Final Deduplication: *Remove duplicates, i.e. identical sentences that appear in all documents in the same partition subset S_k (e.g. name of the court in the heading)*
end procedure

Although random partition of original dataset is performed on document level, final training/test subsets are made only of one sentence per line entries from corresponding documents. Applied procedure with parameters $\mathcal{K}$=75, *C*=9, *K*=5, L1 distance and NE type histograms as feature vectors fi resulted in partition shown in Fig. 1c. Class ”O” of tokens denoting words outside 8 predefined NEs expectedly had several orders of magnitude higher number of instances and therefore was not shown on the same diagram in Fig. 1c, although it was taken into account for fi computation. Note that proposed randomization procedure did not consider final 14+1 categories of NEs that are obtained after text tokenization, as described in Section 3.1.

3.4 Model training process

Model performance was measured by 5 independent experiments involving selection of 4 partition subsets for training and 1 for test. Production model parameters were obtained by training over all available data. Model training convergence over one of the cross-validation folds is illustrated in Fig. 2.

For model implementation was used Python programming language and cloud computing platform with one A100-SXM4 GPU device and 40GB of memory. Each of the 5 training sessions consisted of selecting 4 out of 5 partition subsets and random splitting of their labeled sentences into temporary training and validation sets in the ratio 9:1, respectively. After initial pre-study involving AdamW optimizer, transformer DNN model parameters were optimized using learning rate of 2e-5, weight decay 0.01, fp16 precision, 4 gradient accumulation steps, number of training epochs 6, and the best model selection based on the mean validation set F1 score. Since the input sequence length was limited to 512 tokens, training batch size was deliberately set to 2 in order to avoid any possibility of exceeding the maximum input length in case of long and nested sentences in legal documents, which can be regarded as suboptimal solution. Number of labeled NE training samples varied between 1143 and 1464 per training session. Since the total number of the labeled sentences was 2172, approximately 90% of 1738 was used in each training session. This corresponds to ≈ 1560 sentences for each training epoch, with the effective batch size of 8 (due to 4 gradient accumulation steps), which leads to around ≈ 195 training steps per epoch, i.e. ≈ 1172 iterations. A more detailed illustration of the training process over each of the 15 NER model output categories (BIO labels) is shown in Fig. 3. The number of steps in Fig. 2 and Fig. 3 is the same, but in general varies between the experiments.



(a) **(b)**

Fig. 2. (a) Model optimization loss, and (b) mean accuracy over training iteration steps.

We note that precision is measured as the percentage of labeled NEs found by the NER system that are correct, while the recall is the percentage of labeled NEs present in the corpus that are found by the system. A NE was considered as the correct only if it was an exact match of the corresponding entity in the data file. Individual F1 measures over each of the classes during training are shown in Fig. 4.

Final production model had 450 MB in size, 110 M trainable parameters and its training using all available data lasted under ten minutes on the given hardware. During training total memory reserved on the GPU did not exceed 4 GB, while the initially allocated memory was around 1.4 GB in size.

Unlike traditional approaches, BERT introduces a bidirectional context representation. It simultaneously considers words to the left and right of each word in a sentence, enabling a more nuanced understanding of language. This means when analyzing a word, BERT examines both preceding and following words, capturing intricate linguistic dependencies.

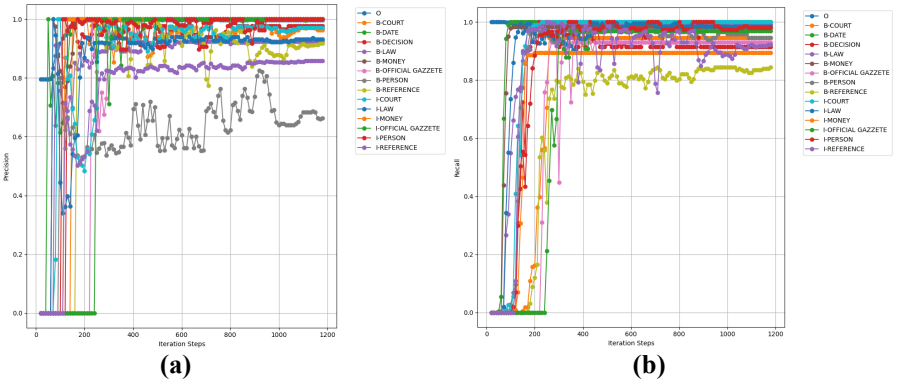


Fig. 3. (a) Precision and (b) recall curves for each of 15 NER output classes (categories) over training iteration steps.

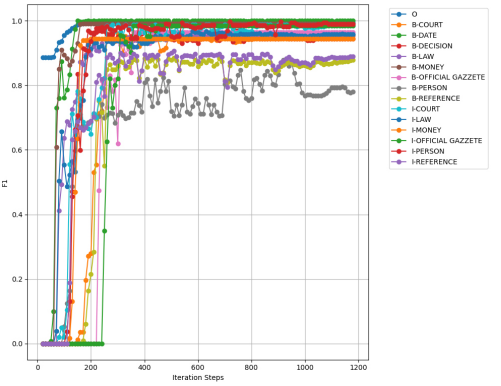


Fig. 4. F1 measure per each output class vs training iterations.

While traditional NER models typically use generic semantic categories like Location, Organization, and Person, legal domain NER requires recognition of domain-specific entities such as judges, courts, and court decisions.

4 Results and discussion

Proposed NER solution was extensively tested using objective performance measures, as well as by additional model analysis involving human judgment of NER quality and effectiveness. Model efficiency was not specifically analyzed, but the model size was such that it was easy to run the production model on local notebook machine. Subjective evaluation was done in order to test developed solution on out of sample sentences and investigate model robustness against different kinds of noisy inputs. More details regarding obtained results and their analysis is provided in the following.

4.1 Cross-validation performance

Detailed statistical cross-validation of model performance was performed according to procedure outlined in Section 3.4., which consisted of estimation of accuracy (Acc), precision (Prec), recall (Rec) and F1 measure. Besides individual metrics computed for each NER category ($C = 15$ output classes), their statistical averages were also reported, the last row in Table 1. All metrics were computed based on results of 5 independent test experiments, in which there was no overlap between the legal documents in the training and test sets. Individual NER results are aggregated and shown in accuracy assessment matrix in Fig. 5.

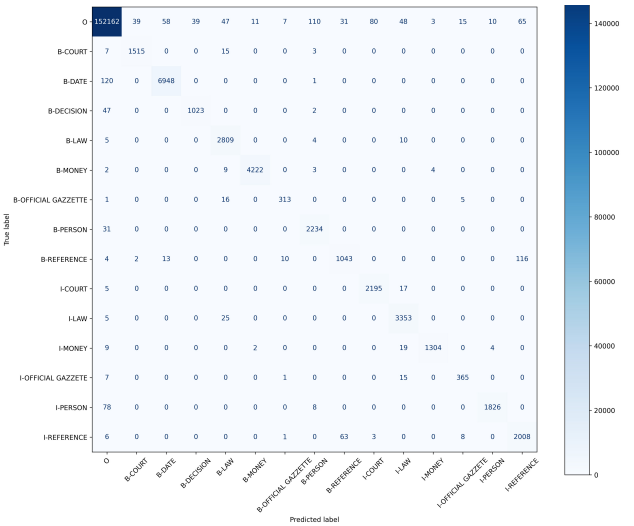


Fig. 5. Accuracy assessment matrix

The average values in Table 1 were computed by macro averaging, in which the metrics were first calculated individually for each class based on values in Fig. 5 and then averaged across all classes. Alternative aggregate metrics would be the ones obtained by micro averaging that would compute number of true positives (TP), true negatives (TN) and false negatives (FN) across all classes and then compute aggregate metrics by their averaging. However, this approach was not pursued since it would lead to more optimistic aggregate metrics due to bias that would be introduced by class ”O” with several orders of magnitude more samples. Similarly, also holds for Acc, which is also known to have overly optimistic value in case of large number of negative samples that are correctly classified. Thus, based on results in Fig. 5 the reported metrics were computed by following expressions:

$$\overline{Prec} = \frac{1}{C}\sum_{i=1}^{C}\frac{TP_i}{TP_i + FP_i} \tag{4}$$

$$\overline{Rec} = \frac{1}{C}\sum_{i=1}^{C}\frac{TP_i}{TP_i + FN_i} \tag{5}$$

$$\overline{F1} = \frac{1}{C}\sum_{i=1}^{C}\frac{2 \cdot Precision_i \cdot Recall_i}{Precision_i + Recall_i} \tag{6}$$

$$Acc = \frac{1}{C}\sum_{i=1}^{C}\frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i} \tag{7}$$

Table 1. Performance metrics: per class and average values

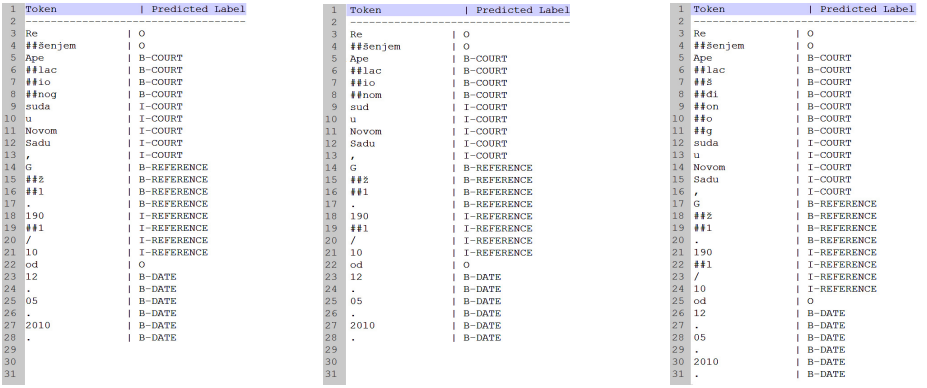
Class	*Recall*	*Precision*	*Accuracy*	F_1
O	0.99	0.99	0.99	0.99
B-Court	0.98	0.97	0.99	0.97
B-Date	0.98	0.98	0.99	0.96
B-Decision	0.95	0.96	0.99	0.95
B-Law	0.99	0.96	0.99	0.97
B-Money	0.99	0.99	0.99	0.99
B-O.Gazette	0.93	0.94	0.99	0.93
B-Person	0.98	0.94	0.99	0.96
B-Reference	0.87	0.91	0.99	0.89
I-Court	0.99	0.96	0.99	0.97
I-Law	0.99	0.96	0.99	0.97
I-Money	0.97	0.99	0.99	0.98
I-O.Gazette	0.94	0.92	0.99	0.93
I-Person	0.95	0.99	0.99	0.97
I-Reference	0.96	0.91	0.99	0.93
Average	**0.97**	**0.96**	**0.99**	**0.96**

4.2 Model robustness

After successful cross validation experiments which have confirmed initial research hypothesis that it is possible to create effective domain specific NER solution for Serbian language with small amount of resources, we have also investigated solution's robustness. E.g. in Fig. 6 are shown some illustrations of NER outputs (token level classification decisions), which are correct despite the presence of text errors and misspellings.

4.3 Results analysis

Values in Table I suggest that model in general performs well, although slightly weaker on "B-Reference" and "I-Reference" classes. It could be expected due to diverse nature of "reference" NE values, which can contain various alphanumeric characters, spacings, syntax that varies among different courts and usually consist of character sequences that do not generally correspond to any word from the language. On the other hand, the proposed dataset had relatively limited corpus of annotated examples (e.g. 504 "reference" NE), implying that the more complex NEs would be harder to learn by limited amount of training data. Slightly better are the results for "official gazette" NE,



Token	Predicted Label
Re	O
##šenjem	O
Ape	B-COURT
##lac	B-COURT
##io	B-COURT
##nog	B-COURT
suda	I-COURT
u	I-COURT
Novom	I-COURT
Sadu	I-COURT
,	I-COURT
G	B-REFERENCE
##ž	B-REFERENCE
##1	B-REFERENCE
.	B-REFERENCE
190	I-REFERENCE
##1	I-REFERENCE
/	I-REFERENCE
10	I-REFERENCE
od	O
12	B-DATE
.	B-DATE
05	B-DATE
.	B-DATE
2010	B-DATE
.	B-DATE

(a)

Token	Predicted Label
Re	O
##šenjem	O
Ape	B-COURT
##lac	B-COURT
##io	B-COURT
##nom	B-COURT
sud	I-COURT
u	I-COURT
Novom	I-COURT
Sadu	I-COURT
,	I-COURT
G	B-REFERENCE
##ž	B-REFERENCE
##1	B-REFERENCE
.	B-REFERENCE
190	I-REFERENCE
##1	I-REFERENCE
/	I-REFERENCE
10	I-REFERENCE
od	O
12	B-DATE
.	B-DATE
05	B-DATE
.	B-DATE
2010	B-DATE
.	B-DATE

(b)

Token	Predicted Label
Re	O
##šenjem	O
Ape	B-COURT
##lac	B-COURT
##š	B-COURT
##đi	B-COURT
##on	B-COURT
##o	B-COURT
##g	B-COURT
suda	I-COURT
u	I-COURT
Novom	I-COURT
Sadu	I-COURT
,	I-COURT
G	B-REFERENCE
##ž	B-REFERENCE
##1	B-REFERENCE
.	B-REFERENCE
190	I-REFERENCE
##1	I-REFERENCE
/	I-REFERENCE
10	I-REFERENCE
od	O
12	B-DATE
.	B-DATE
05	B-DATE
.	B-DATE
2010	B-DATE
.	B-DATE

(c)

Fig. 6. Proposed NER in case of noisy inputs: (a) regular text, (b) and (c) text with the presence of errors and misspellings; in all cases the outputs are correctly classified

which according to Fig. 1c had even smaller corpus of 112 annotated samples over whole training/test set partition. However, NER model adaptation based on PTM was still successful, since these NEs had smaller variability in comparison to previously discussed "reference" type. Results of model training convergence in Fig. 3a reveal that improvements of model precision on "person" NER are relatively slow and underperforming in comparison to other NEs. On the other hand, recall values change in similar fashion to other NEs, Fig. 3b. This indicates that recognition of "person" NE is almost always successful in case of real or correct words corresponding to person names, but there is also a large number of misclassification or false positives. Although this NE is the most frequent in the created dataset, Fig. 1c, observed model behavior could be due to the fact that "person" NE also encompass name abbreviations in the form of initials (e.g. "AA", "CC", ...), which correspond to anonymized personal data in public court rulings. Thus, possible improvements could be directed towards such issues.

Potential limitations of the presented NER model are that it was trained on the limited corpus of annotated documents, which were selected from the specific domain addressing non-economic damages. In that sense, the style and context of NEs appearance could bias the model towards certain decision regions. Such challenges could be overcome by more diverse and comprehensive dataset, e.g. including different types of court rulings, but also official law documents from standard judicial practice. Regarding computational efficiency, the model is using 32-bit floating-point precision for numerical computation, which could be regarded as inefficient in case of large scale applications with high processing throughput and requirements for low operational costs.

5 Conclusions

In the presented work we have demonstrated design of novel NER system for Serbian legal documents and proposed novel domain specific dataset based on publicly available court rulings. Besides model development and conducted experiments, the paper

also addresses the methodology of successful adaptation of pre-trained language models for specific downstream NLP tasks. This is particularly important in case of languages and applications with low resources, in terms of NLP tools and specific training corpora. We hope that this work will stimulate further interest into this emerging topic, especially in the case of Serbian language.

ACKNOWLEDGMENT

The authors would especially like to acknowledge previous collaboration with other AI4Legal team members: attorney at law Sanja Kalušev and M.Sc. Milica Brković, with whom AI ACTA project proposal was made in 2023.
The second author would also like to acknowledge support by the Science Fund of the Republic of Serbia through the grant agreement no. 7449, project AI-SPEAK - "Multimodal multilingual human-machine speech communication", and the Ministry of Science, Technological Development and Innovation of the Republic of Serbia (Contract No. 451-03-137/2025-03/200156) for the support by the project “Scientific and artistic research work of researchers in teaching and associate positions at the Faculty of Technical Sciences, University of Novi Sad 2025” (No. 01-50/295).

References

1. Li, J., Sun, A., Han, J., Li, C.: A survey on deep learning for named entity recognition. IEEE Transactions on Knowledge and Data Engineering vol. 34(1), 50–70 (2020).
2. Çetindağ, C., Yazıcıoğlu, B., Koç, A.: Named-entity recognition in Turkish legal texts. Natural Language Engineering vol. 29(3), 615–642 (2023).
3. Darji, H., Mitrovic, J., Granitzer, M.: German BERT model for legal named entity recognition. In: Proceedings of the 15th International Conference on Agents and Artificial Intelligence (ICAART 2023) vol. 3, pp. 723–728. SCITEPRESS (2023).
4. Chalkidis, I., Fergadiotis, M., Malakasiotis, P., Aletras, N., Androutsopoulos, I.: Legal-BERT: The Muppets straight out of law school. In: Findings of the Association for Computational Linguistics: EMNLP 2020, pp. 2898–2904 (2020).
5. Jin, X., Wang, Y.: TeamShakespeare at SemEval-2023 Task 6: Understand legal documents with contextualized large language models. In: Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023), pp. 517–525 (2023).
6. Rajamanickam, D.: Improving legal entity recognition using a hybrid transformer model and semantic filtering approach. arXiv preprint arXiv:2410.08521 (2024).
7. Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of NAACL-HLT, vol. 1(2). Minneapolis, Minnesota (2019).
8. Ljubešić, N., Lauc, D.: BERTić – The transformer language model for Bosnian, Croatian, Montenegrin, and Serbian. In: Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing, pp. 37–42 (2021).
9. Alfred, R., Leong, L.C., On, C.K., Anthony, P.: Malay named entity recognition based on rule-based approach. Int. Journal of Machine Learning and Computing vol. 4(3) (2014).
10. Chiticariu, L., Li, Y., Reiss, F.: Rule-based information extraction is dead! Long live rule-based information extraction systems! In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 827–832 (2013).

11. Hanisch, D., Fundel, K., Mevissen, H.-T., Zimmer, R., Fluck, J.: ProMiner: Rule-based protein and gene entity recognition. BMC Bioinformatics vol. 6, pp. 1–9 (2005).
12. Geng, D.: Clinical name entity recognition using conditional random field with augmented features. In: CCKS 2017, pp. 61–68 (2017).
13. Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C.H., Kang, J.: BioBERT: A pre-trained biomedical language representation model for biomedical text mining. Bioinformatics vol. 36(4), 1234–1240 (2020).
14. Petasis, G., Vichot, F., Wolinski, F., Paliouras, G., Karkaletsis, V., Spyropoulos, C.D.: Using machine learning to maintain rule-based named entity recognition and classification systems. In: Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics, pp. 426–433 (2001).
15. Nastou, K., Koutrouli, M., Pyysalo, S., Jensen, L.J.: Improving dictionary-based named entity recognition with deep learning. Bioinformatics vol. 40(2), ii45–ii52 (2024).
16. Bikel, D.M., Miller, S., Schwartz, R., Weischedel, R.: Nymble: A high-performance learning name-finder. In: 5th Conf. on Applied Natural Language Processing, pp. 194–201 (1997).
17. Lafferty, J., McCallum, A., Pereira, F., et al.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: ICML, vol. 1(2), p. 3., MA (2001).
18. McCallum, A., Li, W.: Early results for named entity recognition with conditional random fields, feature induction, and web-enhanced lexicons. In: Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, pp. 188–191 (2003). https://aclanthology.org/W03-0430/, last accessed 2025/02/02.
19. Szarvas, G., Farkas, R., Kocsor, A.: A multilingual named entity recognition system using boosting and C4.5 decision tree learning algorithms. In: Discovery Science: 9th International Conference, DS 2006, Barcelona, Spain, October 7–10, 2006. Proceedings 9, pp. 267–278. Springer (2006).
20. McNamee, P., Mayfield, J.: Entity extraction without language-specific resources. In: Proceedings of the 6th Conference on Natural Language Learning, vol. 3, pp. 1–4 (2002).
21. Chiu, J.P., Nichols, E.: Named entity recognition with bidirectional LSTM-CNNs. Transactions of the Association for Computational Linguistics vol. 4, 357–370 (2016).
22. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. Advances in Neural Information Processing Systems vol. 33, 1877–1901 (2020).
23. Joshi, M., Chen, D., Liu, Y., Weld, D.S., Zettlemoyer, L., Levy, O.: SpanBERT: Improving pre-training by representing and predicting spans. Transactions of the Association for Computational Linguistics vol. 8, 64–77 (2020).
24. Clark, K., Luong, M.-T., Le, Q.V., Manning, C.D.: ELECTRA: Pre-training text encoders as discriminators rather than generators. In: Int. Conf. on Learning Representations (2020).
25. Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., Zettlemoyer, L.: BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 7871–7880 (2020).
26. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R.: ALBERT: A lite BERT for self-supervised learning of language representations. In: ICLR (2020).
27. Kim, T., Yoo, K.M., Lee, S.-g.: Self-guided contrastive learning for BERT sentence representations. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing vol. 1, pp. 2528–2540 (2021).
28. Ryu, M.: [Re] ALBERT: A lite BERT for self-supervised learning of language representations. https://openreview.net/forum?id=UkIQrHoru_J, last accessed 2025/02/02.

29. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: RoBERTa: A robustly optimized BERT pretraining approach. arXiv preprint arXiv:1907.11692 (2019).
30. Cvetanović, A.: BERTić-SQuAD-SR-Lat: A question answering neural network for Serbian language. https://huggingface.co/aleksahet/BERTic-squad-sr-lat, last accessed 2025/02/02.
31. CLASSLA - CLARIN Knowledge Centre for South Slavic Languages: BERTić model fine-tuned for the task of named entity recognition in Bosnian, Croatian, Montenegrin, and Serbian (BCMS) (2023). https://huggingface.co/classla/bcms-bertic-ner, last accessed 2025/02/02.
32. Milutin, S., Mihajlov, T., Studen, M.: SRBedding: Information retrieval embedding model for Serbian (2024). https://github.com/smartcat-labs/SRBedding, last accessed 2025/02/02.
33. Živanić, N.: Embedić: A group of new text embedding models finetuned for the Serbian language (2024). https://huggingface.co/djovak/embedic-large, last accessed 2025/02/02.
34. Perišić, O., Stanković, R., Milica, I.-N., Škorić, M., et al.: IT-SR-NER: Web services for recognizing and linking named entities in text and displaying them on a web map. Infotheca - Journal for Digital Humanities, pp. 61–77 (2023).
35. Todorović, B.Š., Krstev, C., Stanković, R., Nešić, M.I.: Serbian NER & Beyond: The archaic and the modern intertwined. In: Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021), pp. 1252–1260 (2021).
36. M. Ikonić Nešić, S. Petalinkar, M. Škorić, and R. Stanković.: BERT downstream task analysis: Named entity recognition in Serbian, in Lecture Notes in Networks and Systems (LNNS): Disruptive Information Technologies for a Smart Society, 14th Int. Conf. on Information Society and Technology (ICIST), Kopaonik, Serbia, M. Trajanović, N. Filipović, and M. Zdravković, Eds., vol. 860. Springer, pp. 333–347, (2024).
37. Conneau, A., Khandelwal, K., Goyal, N., et al.: Unsupervised cross-lingual representation learning at scale. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 8440–8451 (2020).
38. Škorić, M.: Novi jezički modeli za srpski jezik. Infoteka 24. arXiv preprint arXiv:2402.14379 (2024).
39. Društvo za jezičke resurse i tehnologije: JerTEH-355: Najveći BERT model specijalno obučen za srpski jezik (2024). https://huggingface.co/jerteh/Jerteh-355, last accessed 2025/02/02.
40. Scherbakov, V., Mayorov, V.: Finetuning BERT on partially annotated NER corpora. In: 2022 Ivannikov ISPRAS Open Conference (ISPRAS), pp. 86–91. IEEE (2022).
41. Košprdić, M., Prodanović, N., Ljajić, A., Bašaragin, B., Milošević, N.: From zero to hero: Harnessing transformers for biomedical named entity recognition in zero- and few-shot contexts. Artificial Intelligence in Medicine, vol. 156, 102970 (2024).
42. Ministry of Justice of the Republic of Serbia: Serbian judicial practices - official website (2025). https://www.sudskapraksa.sud.rs/sudska-praksa, last accessed 2025/02/02.
43. Mozhi, A.: NER annotator tool for word-level and character-level annotation (2024). https://github.com/tecoholic/ner-annotator, last accessed 2025/02/02.
44. Keraghel, I., Morbieu, S., Nadif, M.: Recent advances in named entity recognition: A comprehensive survey and comparative study. arXiv preprint arXiv:2401.10825 (2024).
45. Alshammari, N., Alanazi, S.: The impact of using different annotation schemes on named entity recognition. Egyptian Informatics Journal vol. 22(3), 295–302 (2021).
46. Wu, Y., Schuster, M., et al.: Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144 (2016).
47. HuggingFace: Fast state-of-the-art tokenizers optimized for research and production (2024). https://github.com/huggingface/tokenizers, last accessed 2025/02/02.