Named Entity Recognition

NLP - Final defense

Jure Bevc, Anže Dežman, Robert Modic

Abstract

In this report we worked on solving named entity recognition (NER) problem in Slovenian language. For this purpose we built solutions based on existing two models: *Stan-ford Named Entity Recognizer model* and *DeepPavlov BERT based model*. The models were trained using the ssj500k dataset and with different training parameters. In the beginning we used only *word* and *NER tag*, to which we later added *morphosyntactic tag*.

1 Introduction

Named entity recognition (NER) is a sub-field of natural language processing, where we seek to process text so that we locate and classify named entities into predefined categories. This has been done through hand crafted grammar-based techniques and through machine learning models. While state-of-the-art solutions for English are reaching near-human performance, we will attempt to implement a NER system for Slovenian language using the ssj500k dataset [5]. To do this, we have researched existing approaches in terms of how they are implemented and evaluated. This report summarizes our findings and approaches in the field of NER.

2 Related work

Since many approaches have been developed to solve the problem of named entity recognition, we started looking at surveys that were done on NER systems. One such survey covers fifteen years of research in the NER field [6], where the authors describe the developed machine learning techniques and also review features and model evaluation.
Another survey was done on deep learning models, which have only recently been studied and developed [9]. In this survey they present architectures for NER and compare them to previous approaches.

From these surveys we found out that most of the popular and recent approaches use some variant of recurrent neural networks (RNN). An efficient and lightweight architecture is presented in the paper [7], where they have drastically reduced the amount of training data, while still achieving close to state-of-the-art results.

Besides neural networks we have also looked at other papers, which describe conditional random fields or Hidden Markov Models as described in [10], where they also achieved great performance. We also reviewed an article published on NER in Slovene [11], where they used conditional random fields and the same ssj500k dataset as we were planning to use.

3 Implementations

For the task of Named entity recognition of the Slovene language we decided to evaluate the *Stanford Named Entity Recognizer model* [4] and the *DeepPavlov BERT based model* [3]. We trained them using the ssj500k [5] dataset.

3.1 Dataset preprocessing

We used the *minidom* [8] Python library to write an *XML-Parser* class, with functions to allow extraction of all words with their named entity sub-type.

In ssj500k dataset the named entities are marked with the <*seg*> tag, sub-type provided in the "sub-type" argument, "PERSON" or "LOCATION" for example. This tag can contain one or more <*w*> tags, denoting words, that are part of a named entity. However, not every named entity in the dataset was marked with the <*seg*> tag. We only used the part of the dataset that had the named entities marked.

Each <*w*> node tag also had a morphosyntactic tag written in its *ana* attribute. When parsing these, every hyphen inside its value was removed, to not cause problems during learning.

We wrote two named entity extraction methods, one for the *Stanford model* and one for *DeepPavlov BERT based model*. They both return the same elements, just in different format.

3.2 DeepPavlov BERT based model

For the base BERT model we used the *BERT-base, multilingual, cased, 12-layer, 768-hidden, 12-heads, 180M parameters* model provided by DeepPavlov[1]. The configuration we used is the default *bert_config.json* configuration, that comes with the base model. It is written bellow:


```
{
  "attention_probs_dropout_prob": 0.1,
  "directionality": "bidi",
  "hidden_act": "gelu",
  "hidden_dropout_prob": 0.1,
  "hidden_size": 768,
  "initializer_range": 0.02,
  "intermediate_size": 3072,
  "max_position_embeddings": 512,
  "num_attention_heads": 12,
  "num_hidden_layers": 12,
  "pooler_fc_size": 768,
  "pooler_num_attention_heads": 12,
  "pooler_num_fc_layers": 3,
  "pooler_size_per_head": 128,
  "pooler_type": "first_token_transform",
  "type_vocab_size": 2,
  "vocab_size": 119547
}
```


The *ssj500k* dataset was first preprocessed with our parser described in section 3.1, output of new-line separated sentences was first randomly split into a training set, validation set and test set with the ratios of 0.8, 0.1 and 0.1, as instructed in the documentation[2]. These sets were written to files *test.txt*, *train.txt* and *valid.txt* in CoNLL-2003 [2] like format, as shown below.

```
# Training with NER tags only:
<word> <NER tag>
# Training with NER and morphosyntactic tags:
<word> <morphosyntactic  tag> <NER tag>
```

The base model was then trained on preprocessed data. First it was used as a base for training with *NER* tags only and then for the training with both the *NER* and the *morphosyntactic tags*, producing two trained models. The training for both models lasted about 12 hours, or 5 epochs. Both models were then evaluated on their *test set*, results are shown in section 4. Because of the time difference between model training, their sets were different, so the results may not represent the true state of the models.

[1] http://files.deeppavlov.ai/deeppavlov_data/bert/multi_cased_L-12_H-768_A-12.zip

[2] http://docs.deeppavlov.ai/en/master/features/models/ner.html#training–data

3.3 Stanford model

The testing and training datasets that were used for the DeepPavlov BERT model were also used to evaluate the Stanford Named Named Entity Recognizer model [4]. This model is based on a Conditional Random Field (CRF) sequence model and it comes with many feature extractors for Named Entity Recognition, which can be configured through the NER properties file.

All available properties are documented in the NERFeatureFactory class [1] and since these can heavily impact the model performance, we present our properties in the following section:

```
useClassFeature=true
useWord=true
useNGrams=true
noMidNGrams=true
maxNGramLeng=6
usePrev=true
useNext=true
useSequences=true
usePrevSequences=true
maxLeft=1
useTypeSeqs=true
useTypeSeqs2=true
useTypeySequences=true
wordShape=chris2useLC
useDisjunctive=true
```

4 Results and discussion

4.1 BERT models' results

Tables 1 and 2 show the results of the DeepPavlov BERT models, the former trained only with *NER* tags and the latter trained with both *morphosyntactic tags* and *NER* tags.

NER tag	precision	recall	F1
LOC	0.90	0.92	0.91
MISC	0.56	0.50	0.52
ORG	0.79	0.76	0.77
PERSON	0.87	0.93	0.90
TOTAL	0.85	0.86	0.85

Table 1: DeepPavlov BERT NER tags only results.

NER tag	precision	recall	F1
LOC	0.84	0.90	0.87
MISC	0.62	0.50	0.55
ORG	0.64	0.68	0.66
PERSON	0.90	0.90	0.91
TOTAL	0.80	0.82	0.81

Table 2: DeepPavlov BERT NER and morphosyntactic tags results.

4.2 Stanford models' results

NER tag	precision	recall	F1
LOC	0.84	0.81	0.83
MISC	0.60	0.32	0.42
ORG	0.76	0.57	0.65
PERSON	0.76	0.88	0.82
TOTAL	0.77	0.73	0.75

Table 3: Results of the Stanford model trained on ssj500k dataset using only NER tags.

NER tag	precision	recall	F1
LOC	0.80	0.69	0.74
MISC	0.72	0.17	0.28
ORG	0.70	0.44	0.54
PERSON	0.73	0.84	0.78
TOTAL	0.74	0.63	0.68

Table 4: Results of the Stanford model trained on ssj500k dataset using morphosyntactic tags.

4.3 Total values of all models

model	precision	recall	F1
Štajner	0.63	0.59	0.61
Štajner + morph. tag	0.72	0.68	0.70
BERT	0.85	0.86	**0.85**
BERT + morph. tag	0.80	0.82	0.81
Stanford	0.77	0.73	0.75
Stanford + morph. tag	0.74	0.63	0.68

Table 5: total values for precision, recall and F1 score for all tested models summarised

4.4 Discussion

Both DeepPavlov BERT based models seem to be very good at distinguishing locations and persons. Low performance for MISC subtype can be attributed to low number of such samples in the ssj500k dataset. Overall the performance of the BERT model trained only on *NER* tags seems better than the one trained with *NER* and *morphosyntactic tags*, but as already stated in section 3.2 the evaluation sets were different, because of the time interval between model training, so the results may not be comparable.

We can observe that although the Stanford model achieved decent results in some categories, it generally performed worse than the DeepPavlov model. This model is closer to the results of the Štajner et. al. model with morphosyntactic tags. With added morphosyntactic tags the performance of the Stanford model seems to worsen, just like our BERT model.

We believe the reason for worse results in our models added morphosyntactic tags are due to added variability of possible true selections and the fact that there isn't enough data in the dataset to train on that variability.

5 Conclusion

As already stated in 3.1, the used ssk500j[5] dataset wasn't completely segmented, so we only had to use a portion of it. The dispersion of the named entity types was also very far from uniform, and of those, only "PERSON", "LOCATION" and "ORGANIZATION" major named entity types were present. If the rest of the dataset were to be completely segmented in the future, the models might be able to learn better how to distinguish certain named entities, perhaps other major entity types would be present in that portion. The portion of the dataset we didn't use, could also be used as an additional test set.

References

[1] Class nerfeaturefactory documentation.

[2] Language-independent named entity recognition (ii).

[3] M. Burtsev, A. Seliverstov, R. Airapetyan, M. Arkhipov, D. Baymurzina, N. Bushkov, O. Gureenkova, T. Khakhulin, Y. Kuratov, D. Kuznetsov, A. Litinsky, V. Logacheva, A. Lymar, V. Malykh, M. Petrov, V. Polulyakh, L. Pugachev, A. Sorokin, M. Vikhreva, and M. Zaynutdinov. Deeppavlov: Open-source library for dialogue systems. 07 2018.

[4] J. R. Finkel, T. Grenager, and C. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 363–370. Association for Computational Linguistics, 2005.

[5] S. Krek, K. Dobrovoljc, T. Erjavec, S. Može, N. Ledinek, N. Holz, K. Zupan, P. Gantar, T. Kuzman, J. Čibej, Š. Arhar Holdt, T. Kavčič, I. Škrjanec, D. Marko, L. Jezeršek, and A. Zajc. Training corpus ssj500k 2.2, 2019. Slovenian language resource repository CLARIN.SI.

[6] D. Nadeau and S. Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26, 2007.

[7] Y. Shen, H. Yun, Z. C. Lipton, Y. Kronrod, and A. Anandkumar. Deep active learning for named entity recognition. *arXiv preprint arXiv:1707.05928*, 2017.

[8] xml.dom.minidom. 20.7. xml.dom.minidom — minimal dom implementation — python 3.6.10 documentation.

[9] V. Yadav and S. Bethard. A survey on recent advances in named entity recognition from deep learning models. *arXiv preprint arXiv:1910.11470*, 2019.

[10] G. Zhou and J. Su. Named entity recognition using an hmm-based chunk tagger. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 473–480, USA, 2002. Association for Computational Linguistics.

[11] T. Štajner, T. Erjavec, and S. Krek. Named entity recognition in slovene text. *Slovenščina 2.0: empirical, applied and interdisciplinary research*, 1(2):58–81, Dec. 2013.