Experimental Protocol for XCS224U Project *gNER*

Matthias Droth
matthias.droth@gmail.com

Martin Nigsch
martin@nigsch.eu

Vasco Ribeiro
vascosousaribeiro@yahoo.com

Abstract

This document provides an overview over what we are trying to achieve with our xcs224u project *gNER*, how we intend to achieve it, what data we plan to use, and what models we are going to implement. We also clarify our motivation, progress, remaining tasks, and challenges we foresee.

1 Hypotheses

We hypothesize that conditional random fields (CRF) algorithms (Lafferty et al., 2001; McCallum and Li, 2003; Sutton and McCallum, 2012) perform on par with deep learning algorithms when trained only on a relatively small named entity recognition (NER) dataset. We hypothesize that recurrent neural networks (RNNs) (Tutschku, 1995; Hochreiter and Schmidhuber, 1997) and transformers (Vaswani et al., 2017; Rush, 2018; Devlin et al., 2019) outperform CRF when grounding on a large dataset is included before retraining on the small target dataset.

2 Datasets

We plan to employ two main datasets:

1. A relatively small target dataset of *NER-annotated* real estate offerings in Austria. The labels in this dataset have been obtained by applying prodigy's NER annotation tool to a small fraction of a roughly 25 times larger corpus of unlabeled real estate offerings. As a consequence, the target dataset can be enhanced to as many fully labeled instances as in the corpus.

2. The German language dataset of PAWS-X by Yang et al. (2019) for grounding.

3 Metrics

A single real score in $[0, 1]$ is useful for unambiguous ranking of models. We are going to balance precision and recall by using an F_β score. The averaging over different classes shall take the support of each class into account. This allows us to compare our different models and in particular, to determine whether deep learning models for *gNER* benefit from grounding on a large dataset.

In order to convey a rich picture of model performance, we will go beyond singular scores and also show confusion matrices and receiver operating characteristic (ROC) curves.

We also want to point out that not only the model architecture but also the cost function used for calculating gradients during training impacts the performance as different losses tend to have different biases towards a given metric (Li et al., 2020). Since sklearn-crfsuite inherits from scikit-learn, which only allows changing the loss function by modifying the library's source code, we refrain from exploring this approach for CRFs but intend to do so for RNNs or transformers.

4 Models

A description of the models that you'll be using as baselines, and a preliminary description of the model or models that will be the focus of your investigation.

As an overall baseline, we use a custom model that always predicts the most common label in the training data, as motivated by the lecture *Simple baselines* in module 8 of xcs224u.

In order to test our hypothesis, we need to implement CRF-, RNN-, and transformer-based models. For each of these model classes we will use a very basic but properly implemented version as a baseline, each of which we expect to be at least on par with the overall baseline. Then, we will try to increase the performance in each model class by tweaking the cost function as well as other hyperparameters and by grounding.

5 General reasoning

Due to the larger number of trainable parameters deep models like deep RNNs and transformers are capable of encoding more complicated relations than shallow models like CRFs. To avoid overfitting, deep models typically require large amounts of training data. Since our target dataset consists of only 140 fully labeled sequences, deep RNNs and transformers are likely to overfit on training data and might fail to outperform CRFs on unseen data.

Adding grounding on a large dataset as a pre-training step allows *deep models in particular* to encode the relations of natural language better. Therefore, we expect that grounding will boost performance of deep RNNs and transformers beyond the performance of CRFs.

6 Summary of progress so far

What we have done:

- We have built a corpus of text data by scraping a newspaper website for information on concluded real estate transactions. This resulted in some 3,400 short texts, 140 of which we have already been fully annotated. Annotation followed a hybrid strategy with recourse to gazetteers (e.g. for the self-contained set of distinct towns in the Voralberg region of Austria), regex (e.g. the numerical price "gesamtpreis" field) and manual annotation (using a third-party package called prodigy).

- We have implemented an overall baseline (see Sec. 4) in the form of a model that always predicts the most prominent class in the training data. Its class support averaged F_1 score on the test fraction of our dataset is 0.716.

- We have also implemented a specific baseline for CRF models where we approach our problem as a typical Named-Entity Recognition problem with the model being tasked with learning parameters for label-observation features in addition to label-label transitions and label-word emissions. Its class support averaged F_1 score on the test fraction of our dataset is 0.914.

- Finally, we have also attempted to find the best version of the CRF model class by searching the hyperparameter space with scikit-learn's *RandomizedSearchCV* class. The best model candidate has been refitted on the entire training data. Its class support averaged F_1 score on the test fraction of our dataset is 0.972.

What we still have to do:

- We still need to implement dedicated baselines for deep RNNs and for transformers.

- Omitting grounding, we still need to find the best version of a specific deep RNN model class (e.g. LSTM) as well as the best version of a specific transformer model class (e.g. BERT).

- We still need to explore the PAWS-X dataset and use if for grounding our deep RNN and transformer models.

- As for metrics, we still need to show confusion matrices and receiver operating characteristic curves.

- We would also like to explore the interplay of loss functions and metrics.

Obstacles or concerns that might prevent our project from coming to fruition:

- The German language dataset of PAWS-X is relatively large. Grounding a transformer on this dataset might slow down our development process significantly.

- The German language dataset of PAWS-X might not provide the right kind of grounding and consequently fail to improve performance on our task.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Xiaoya Li, Xiaofei Sun, Yuxian Meng, Junjun Liang, Fei Wu, and Jiwei Li. 2020. Dice loss for data-imbalanced NLP tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 465–476, Online. Association for Computational Linguistics.

Andrew McCallum and Wei Li. 2003. 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*, pages 188–191.

Alexander Rush. 2018. The annotated transformer. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 52–60, Melbourne, Australia. Association for Computational Linguistics.

Charles Sutton and Andrew McCallum. 2012. An introduction to conditional random fields. *Foundations and Trends® in Machine Learning*, 4(4):267–373.

K. Tutschku. 1995. Recurrent multilayer perceptrons for identification and control: The road to applications.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A Cross-lingual Adversarial Dataset for Paraphrase Identification. In *Proc. of EMNLP*.