# Named Entities Recognition with Transfer Learning

Yuchuan Fu (yf2127), Xiaohan Xue (xx715)

## *Abstract*

**Named Entities Recognition (NER) is a crucial information extraction task that is usually the first step for many Natural Language Processing applications. In this paper, we designed a deep learning pipeline to identify and classify the CONLL2003 dataset into their corresponding tags, thus extracting named entities such as person's name, location and date etc to obtain useful information from unstructured news texts. In particular, without constructing hand-crafted features, we adopted the BiLSTM based model as our baseline model and resort to BERT based models with transfer learning to improve the existing solutions. Then we compared their performances, and our results highlight the improvements of using a pretrained BERT_news model from a simple BERT based model and can show possible adaptations that could have further improvements .**

## 1. Project Description

Named Entity Recognition (NER) is an important subtask for Natural Language Processing. The goal is to be able to locate and classify the Named Entities (NE) to their appropriate types so that we can gather information from multiple, unstructured large pieces of texts, and then to produce a structured representation of relevant information by means of machine learning techniques. NE mainly consist of two categories, Generic named entities and Domain-specific named entities. Generic NE includes names of persons, locations, organizations, phone numbers, and dates, while Domain-specific NE are usually professional vocabularies. For instance, in a biology domain, names of proteins, enzymes, organisms, genes are all examples of NE. The final goal of NER tasks is to first organize information in a way such that it is useful to people. Secondly, NER is exploited to transform text to semantically precise information that allows further inference by machine learning algorithms in downstream applications.

NE was first introduced in the Message Understanding Conferences (MUC) in the U.S. in 1990's, which influenced subsequent IE research. Then the first NER task was organized by Grishman and Sundheim (1996) in the 6th MUC and early NER systems were rule-based and depended on handcrafted rules and lexicons. Now, it has been more popular for novel NER systems mainly constructed by neural networks that require minimal feature engineering after Collobert et al., 2011 had proposed "deep architecture" sequence models including

NER tasks. However, regardless of the types of NER techniques, either rule-based or the feature-based supervised learning techniques takes considerable training time for feature extractions and fine-tuning the rules to any new dataset independently. Even for the deep architectures that have yielded state-of-the-art performances without minimal feature engineering or even labels for different datasets, it may still be slow and thus inefficient to mass deploy neural network based NER systems in practical settings. Here is where transfer learning comes in. With transfer learning, one can efficiently save training time, usually resulting in better performance of neural networks, which will subsequently benefit downstream applications of NER.

## 2. Approach

Our solution approach is to first start with a BiLSTM+CNN model as our baseline model, and then apply transfer learning by using the pretrained RoBERTa_news model along with BERT and RoBERTa models.

Our solution will largely benefit from reducing training time by using a BERT based model instead of BiLSTM. Also training accuracy is further improved by introducing RoBERTa, which removes the Next Sentence Prediction Task.

**Baseline Model - BiLSTM + CNN**

We have referenced the paper Chiu and Nichols (2016) and the code from the Github Repository by Maximilian Hofer. Feature-based supervised learning approaches mostly achives highest performance by applying Conditional Random Field (CRF), Support Vector Machines (SVM), or perceptron models but they require careful hand-crafted features. Chiu and Nichols (2016)'s paper proposed an effective neural network model that adds a Convolutional neural network (CNN) to learn important character-level features from word embeddings trained on large quantities of unlabelled text. We used pre-trained GloVe (Pennington et al., 2014) to convert our text into embeddings that have been trained on Wikipedia.

One of the important challenges in NER tasks is rare words that are unseen in training dataset and thus word embeddings are poorly trained on. Therefore, explicit character level features such as prefix and suffix can be useful to learn such information. CNN can be employed to model character-level information which could solve this issue. Then, we employed a stacked Bi-directional Recurrent Neural Network with Long Short-Term Memory (BiLSTM) units to transform word features into named entity tag scores.

The word embedding learns word-level features and CNN could induce character level features. Words and characters that are discrete are transformed into continuous vector representations to model the distribution. After that, the vector representations are then concatenated and fed into a forward LSTM network and a backward LSTM network. As NER is a sequential labelling task, the context on both sides of a word can be effectively taken into account by the BiLSTM network, which solves the limitation of a feed-forward model that only considers the preceding context.

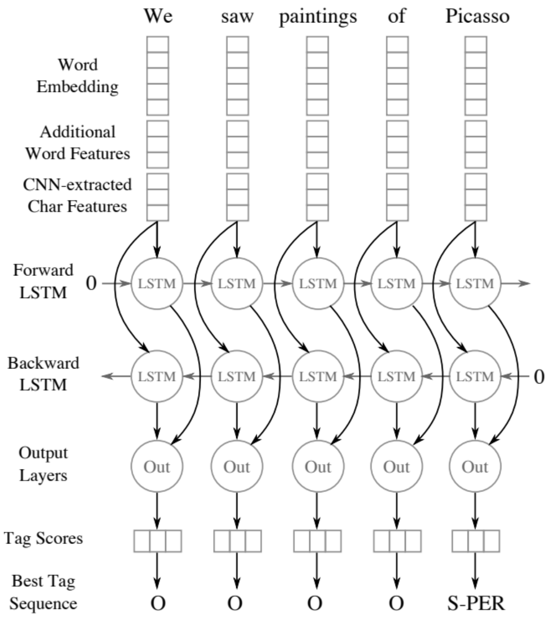

Figure 1: The (unrolled) BLSTM for tagging named entities. Multiple tables look up word-level feature vectors. The CNN (Figure 2) extracts a fixed length feature vector from character-level features. For each word, these vectors are concatenated and fed to the BLSTM network and then to the output layers (Figure 3).

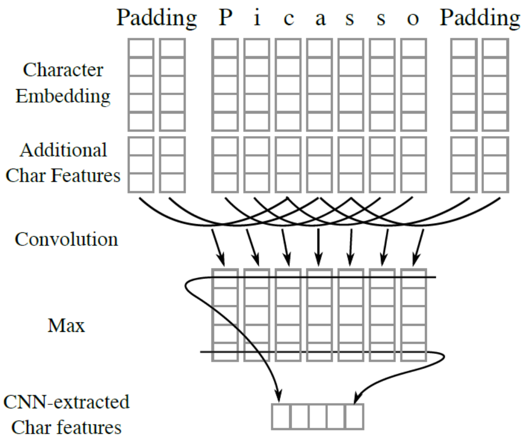

Figure 2: The convolutional neural network extracts character features from each word. The character embedding and (optionally) the character type feature vector are computed through lookup tables. Then, they are concatenated and passed into the CNN.

Figure1: Structure of Bi-LSTM + CNN

The output of each network at each time step is passed to a linear layer with softmax activations into log-probabilities for each tag category. These two vectors are then simply added together to produce the final output.

### BERT-base

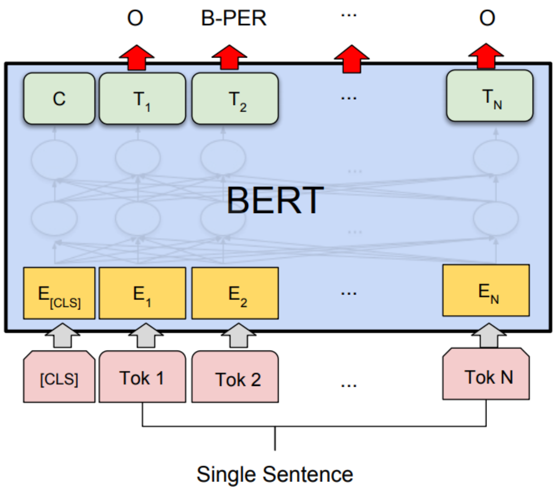

Figure2: Structure of BERT

BERT uses masked language models to enable pre-trained deep bidirectional representations. For a given token, its input representation is composed by summing the corresponding position, segment and token embeddings. The model we use is bert-base-cased, which is a case-sensitive model, meaning that it makes a difference between english and English. It includes two subtasks during pre-training, the masked language modeling and next sentence prediction. With the architecture of BERT, when we use it for downstream tasks, we only need to make changes in the last output layer.

### RoBERTa-base

RoBERTa was proposed as the author states, the subtask of Next Sentence Prediction in

BERT is useless. Thus during pre-training, only masked language modeling was conducted in order to improve upon BERT on downstream tasks. We applied the RoBERTa-base model as another baseline, since the transfer learning model RoBERTa_news used RoBERTa instead of BERT to further pre-train.

### RoBERTa_news

The transfer learning model roberta_news was proposed in Gururangan and Suchin(2020)'s paper. They investigated whether it was still helpful to tailor a pretrained model to the domain of a target task. They experimented with four domains: BioMed, CS, News, Reviews. The author trained ROBERTA on each of the four domains for 12.5K steps, which amounts to a single pass on each domain dataset. Since our CoNLL-2003 dataset is also a news dataset, we think it's a great chance if we could implement this domain-trained bert model to improve upon vanilla roberta. In this paper, RoBERTa was further pre-trained on a corpus of 11.90M articles from RealNews, which has 6.66B tokens.

## 3. Solution Diagram

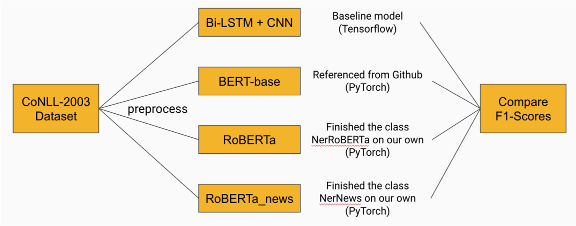
CoNLL-2003 Dataset
preprocess
Bi-LSTM + CNN
BERT-base
RoBERTa
RoBERTa_news
Baseline model (Tensorflow)
Referenced from Github (PyTorch)
Finished the class NerRoBERTa on our own (PyTorch)
Finished the class NerNews on our own (PyTorch)
Compare F1-Scores

Figure3: Solution Diagram

Figure 3 shows the architecture of our overall solution. Our dataset is the famous CoNLL-2003 dataset, and we applied four models: Bi-LSTM with CNN, BERT-base, RoBERTa, RoBERTa_news. For the first two models, we used referenced codes in other people's Github, and LSTM was using tensorflow and GloVe embeddings, while BERT models were using PyTorch. And finally we compare their F1-scores to evaluate their performances.

## 4. Implementation Details

For implementation, we used the computing resources of Courant, which offers five cuda services. We used cuda1 and cuda4 most of the time, which offers Two GeForce GTX TITAN Black (6 GB memory each) and Two GeForce GTX TITAN X (12 GB memory each) respectively.

In addition, for LSTM, we used Nadam optimizer with a fixed learning rate 0.0105. We choose 100 mini-batches and for each batch, keep the same number of tokens as the batch size from different numbers of sentences. We also applied a dropout layer to each LSTM layer to reduce the overfitting problem with a dropout probability of 0.5.

For BERT models, we used maximum total input sequence length of 128, training batch size of 32, evaluation batch size of 8, initial learning rate of 5e-5, and Adam optimizer.

## 5. Experiments

### 5.1 Dataset Description

|  | Articles | Sentences | Tokens |
|---|---|---|---|
| Training | 946 | 14987 | 203621 |
| Valid | 216 | 3466 | 51362 |
| Test | 231 | 3684 | 46435 |

Table1: Overview of CoNLL-2003 Dataset 1

|  | LOC | MISC | ORG | PER |
|---|---|---|---|---|
| Training | 7140 | 3438 | 6321 | 6600 |
| Valid | 1837 | 922 | 1341 | 1842 |
| Test | 1668 | 702 | 1661 | 1617 |

Table2: Overview of CoNLL-2003 Dataset 2

The CoNLL-2003 dataset has both English and German versions. The English version has a huge proportion of sports news with annotations in four entity types: ‘Person’, ‘Location’, ‘Organization’, and ‘Miscellaneous’. This dataset contains around 14k training samples, and over 3000 samples for both validation and test set.

## 5.2 Experimental Results

| Models | F1 score<br>(Valid set) | F1 score<br>(Test set) |
|---|---|---|
| Bi-LSTM + CNN | 0.9311 | 0.8910 |
| BERT-base | 0.9498 | 0.9125 |
| RoBERTa-base | 0.7592 | 0.7161 |
| RoBERTa_news | 0.8376 | 0.7843 |

Table3: Experiment results of four models

Table3 shows the experiment result of our four models. We trained the Bi-LSTM + CNN for 100 epochs, and the F1 score on the test set is close to 0.9. The BERT-base model was trained for 5 epochs, while RoBERTa-base and RoBERTa_news were both trained for 20 epochs.

## 5.3 Training Log

The evaluation metric we use was F1-score, and we used the function from this seqval

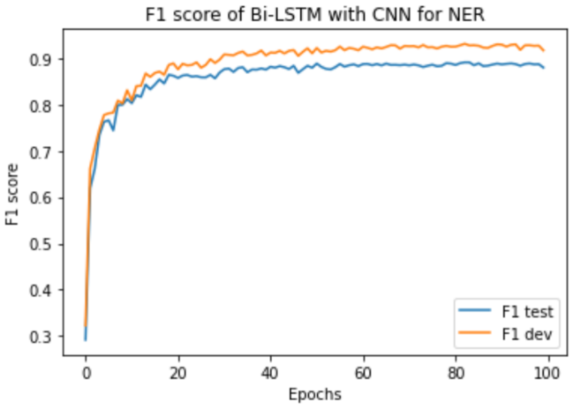

Figure4: Training log of Bi-LSTM + CNN

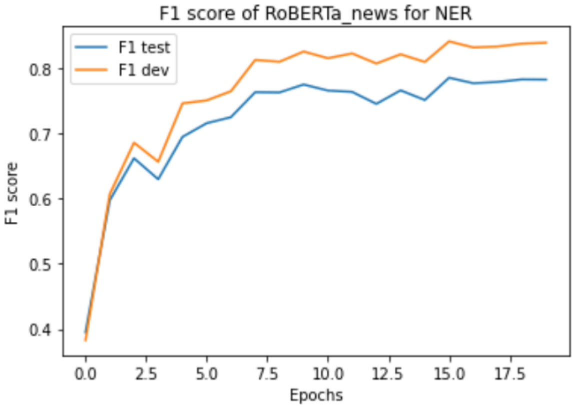

Figure5: Training log of RoBERTa_news

Fig 4 and 5 show the training logs of LSTM and RoBERTa_news models. The x-axis is Number of epochs, and the y-axis is the F1 score on the test set.

# 6. Observations

First, as shown in the table, BERT-base outperforms Bi-LSTM + CNN, showing a

2.4% increase in F1 score on the test set, which proves that BERT models can indeed capture better representation of sentences and make downstream predictions more accurate than LSTM.

However, for the remaining two RoBERTa models, our own-implemented version was far worse than that in other papers'(92.4%). We suspect it's because of other different parameter settings, and unpublished training tricks.

RoBERTa_news outperforms RoBERTa under the same configuration, a 9.5% increase of F1 score on the test set, which proves that pretraining on domain corpus indeed produces better performance on NER.

## 7. Conclusion

From the experiment results, we notice that compared with LSTM, BERT-base model gives higher accuracy on the test set, showing much more powerfulness of BERT. In the meantime, we only need to train BERT-base for 5 epochs, which would be time-saving compared with LSTM.

For RoBERTa-base and RoBERTa_news models, although these two models show worse performance, RoBERTa_news still outperforms RoBERTa-base under the same settings, which proves that pre-training on domain corpus is effective in improving the accuracy of BERT on NER.

As for the code structure, we only added one linear layer on top of RoBERTa_news to map from 768 hidden states to 12 classes, where more complicated ones may be useful. Other reasons for bad performance could be due to not comprehensive hyper parameter tuning and not good use of tricks about fine-tuning.

Additionally, we did not implement the famous CRF layers on top of our models due to time issues. These could have been implemented to study whether better performance would be achieved as most papers state.

## 8. Bibliography

1. Huang Z , Wei X , Kai Y . Bidirectional LSTM-CRF Models for Sequence Tagging[J]. Computer Science, 2015.

2. Chiu J , Nichols E . Named Entity Recognition with Bidirectional LSTM-CNNs[J]. Computer Science, 2015.

3. Devlin J , Chang M W , Lee K , et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[J]. 2018.

4. Liu Y , Ott M , Goyal N , et al. RoBERTa: A Robustly Optimized BERT Pretraining Approach[J]. 2019.

5. Gururangan S , A Marasović, Swayamdipta S , et al. Don't Stop Pretraining: Adapt Language Models to Domains and Tasks[J]. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020.

6. Li J , Sun A , Han J , et al. A Survey on Deep Learning for Named Entity Recognition[J]. IEEE Transactions on Knowledge and Data Engineering, 2020, PP(99):1-1.

7. Language-Independent Named Entity Recognition (II)

8. Wang, Mengqiu, and Christopher D. Manning. "Effect of Non-Linear Deep

Architecture in Sequence Labeling." https://www.aclweb.org/anthology/I13-1183.pdf.

9. Lee, Ji Young, et al. "Transfer Learning for Named-Entity Recognition with Neural Networks." 2017, https://arxiv.org/pdf/1705.06273v1.pdf.

10. Qiang Zhang, Yong Sun, Linlin Zhang, Yanfei Jiao, Yue Tian, Named entity recognition method in health preserving field based on BERT, Procedia Computer Science, Volume 183, 2021, Pages 212-220, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2021.03.010.