KDDIE at SemEval-2022 Task 11: Using DeBERTa for Named Entity Recognition

Caleb Martin, Huichen Yang, and **William Hsu**
Computer Science of Kansas State University
Manhattan, Kansas 66502
{calebjm288, huichen, bhsu}@ksu.edu

Abstract

In this work, we introduce our system to the SemEval 2022 Task 11: Multilingual Complex Named Entity Recognition (MultiCoNER) competition. Our team (KDDIE) attempted the sub-task of Named Entity Recognition (NER) for the language of English in the challenge and reported our results. For this task, we use transfer learning method: fine-tuning the pre-trained language models (PLMs) on the competition dataset. Our two approaches are the BERT-based PLMs and PLMs with additional layer such as Condition Random Field. We report our finding and results in this report.

1 Introduction

In today's world there is an ever-growing supply of unstructured data, a lot of it in the form of free text. Named Entity Recognition (NER) is a process that seeks to gather information from free text, by extracting and labeling the named entities. SemEval 2022 Task 11 is a competition where NER systems are trained and then they compete against each other for the best scores (Malmasi et al. (2022b)). The task is split into 13 tracks: one for each of 11 different languages, a multi-lingual track, and a mixed language track (Malmasi et al. (2022b)). In this paper we will be focusing on track 1, which is monolingual English NER. For this SemEval Task participants were asked to have their system both identify named entities and then label them with one of the six provided labels. The six labels to be used were person, location, group, corporation, product, and creative work (Malmasi et al. (2022b)). Figure 1 shows an example of these words being tagged with their appropriate labels.

For this paper our strategy involved trying various transformer models from the HuggingFace library (Wolf et al. (2020)) and training them on the training data provided. We then tried adding a conditional random field layer to some different models to see if that would improve the scores that we received. With all the models we trained we also fine-tuned many different training parameters to obtain the best scores possible. We tried using different number of epochs, different learning rates, different batch sizes, and changing many additional parameters.

Figure 1: Example of labeling named entities in a sentence using the SemEval 2022 Task 11 tagging scheme. These two sentences are taken directly from the SemEval 2022 Task 11 training set (Malmasi et al. (2022a)).

In the research we have done leading up to this paper we have learned many things. One is that adding a CRF layer to a BERT model (Devlin et al. (2018)) helps its performance, but adding a CRF layer to a DeBERTa model (He et al. (2021)) doesn't help and actually hurts its performance for NER. We also learned through our testing that a DeBERTa model (He et al. (2021)) is one of the best transformer models for NER, specifically on the SemEval 2022 Task 11 dataset (Malmasi et al. (2022a)).

2 Related Work

There are many challenges that can make NER extremely difficult. In Meng et al. (2021) they explain that named entity recognition is especially difficult in situations with low-context or in scenarios where the named entities are exceptionally complex and unique. Also as stated in Li et al. (2020) NER requires well annotated data and lots of it. This a major obstacle because annotating data can be extremely time consuming and expensive.

Some older approaches to NER include rule-

Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 1531 - 1535

based systems and unsupervised learning systems, however both of these fall short of the performance of modern feature-based supervised methods, which often use transformer models (Li et al. (2020)). According to Li et al. (2020) using pre-trained transformer models with other possible layers and then fine tuning these transformers is becoming the standard for NER. Within this paper we will be using pre-trained transformer models to help us achieve the best results.

Another related work (Souza et al. (2019)) shows the use of a BERT transformer model (Devlin et al. (2018)) for named entity recognition. They also added a conditional random field (CRF) layer on top the BERT model which ended up improving results. This paper (Souza et al. (2019)) shows that both BERT is good for NER and that adding a CRF layer can also help a transformer model perform better.

3 System Overview

For this paper our main strategy will be fine-tuning some large feature-based transformer models on the training data provided. We used the HuggingFace library (Wolf et al. (2020)) to download and train the transformer models. To train our models we used the HuggingFace default optimizer called AdamW. For the most part, we left the AdamW optimizer parameters at the default values. Also, for all of the models we used a linear learning rate scheduler with zero warm up steps.

3.1 BERT

For this competition we experimented with several different systems and methodologies. The first and most basic was to train a BERT model on the data provided (Devlin et al. (2018)). To do this we took a BERT-large-uncased model and trained it for 5 epochs at a learning rate of 2e-5 on the competition training data. With this model we received a precision of 0.870, a recall of 0.811, and a F1-score of 0.839.

3.2 BERT-CRF

Next, we tried improving the score of the BERT model (Devlin et al. (2018)), by adding a conditional random field (CRF) layer after the BERT model. Following the work in (Yang and Hsu (2021)), using the CRF layer should help the model learn the specific parameters of the task, and thus increase the score. We trained this model for 6 epochs at a learning rate of 2e-5 and with a weight decay of 0.01. With this model we received a precision of 0.851, a recall of 0.862, and a F1-score of 0.856.

3.3 DeBERTa-Large

The third model was created using a DeBERTa pre-trained language model. DeBERTa is a BERT (Devlin et al. (2018)) based transformer model that uses disentangled attention and enhanced decoding to improve performance (He et al. (2021)). Within DeBERTa they used a two-vector approach where they split the position encoding and the token encoding into two separate vectors. This allowed the attention layers to be disentangled and learn from each encoding vector separately. They also used enhanced decoding where they give the model both the relative word positions within the sentence and their absolute positions. These improvements allow DeBERTa to outperform BERT in many scenarios (He et al. (2021)). It achieves state of the art scores in many Natural Language Processing tasks including NER (He et al. (2021)).

For this paper we took a DeBERTa model and trained it on the SemEval 2022 Task 11 training data. We got the best results when training the DeBERTa large model for 5 epochs with a learning rate of 2e-5. With this model we received a precision of 0.870, a recall of 0.872, and a F1-score of 0.871.

3.4 DeBERTa-XLarge

This model is similar to the DeBERTa Large except with more parameters. This model has twice the number of layers and parameters. We also got the best results when training the DeBERTa xlarge model for 5 epochs with a learning rate of 2e-5. With this model we received a precision of 0.868, a recall of 0.876, and a F1-score of 0.872.

3.5 DeBERTa-CRF

For this model we decided to follow in the ideas of the BERT-CRF model (Yang and Hsu (2021)) and try to add a CRF layer to our DeBERTa large model (He et al. (2021)). We thought since the CRF layer improved the score of the BERT model that it might do the same for a DeBERTa model. We made a DeBERTa-CRF model and trained it on the task training data for 5 epochs with a starting learning rate of 5e-5 and a weight decay of 0.01. Unfortunately, this model performed quite poorly with a final score precision of 0.820, a recall of

0.830, and a F1-score of 0.825 on the validation dataset (Malmasi et al. (2022a)).

4 Experimental Setup

The data that the organizers provided was in text files in CoNLL format (Malmasi et al. (2022a)). For the English language there was 15300 training examples and 800 validation examples (Malmasi et al. (2022a)). All of the examples provided were in BIO format. BIO is a scheme where all words at the beginning of a named entity are labeled with a B, all the words inside an entity are labeled with a I, and all words outside an entity are labeled with an O. In total that means there are 13 possible tags: O, B/I-PER, B/I-LOC, B/I-GRP, B/I-CORP, B/I-PROD, and B/I-CW (Malmasi et al. (2022b)). The following is the meaning of the abbreviations: PER is person, LOC is location, GRP is group, CORP is corporation, PROD is product, and CW is creative work.

To process that data file first we split the text into each example and then split each example into a list of tokens and labels. Then we mapped each label to a specific number to represent it, 0-12. Finally, from these lists of lists we created a Hugging Face dataset (Wolf et al. (2020)). We left the data sets in the default train/eval splits of 15300 training examples and 800 validation examples.

To evaluate our models, we used the metrics macro precision, recall, and f1-score. Precision deals with how accurate the model is when it does predict a label. Recall corresponds to how good the model is at predicting labels compared to the total number of actual entities. Macro f1-score is the harmonic mean of these two, and gives the best single number representation of how well the model is performing. To calculate these, we used a python library called seqeval (Nakayama (2018)). To use seqeval we simply provide it a list of the predicted values and a list of the ground truth values. The equations seqeval (Nakayama (2018)) uses are shown below with tp being true positives, fp being false positives, and fn being false negatives:

$$Macro\ Precision = \frac{tp}{tp + fp}$$

$$Macro\ Recall = \frac{tp}{tp + fn}$$

$$Macro\ F1 = \frac{2 * Precision * Recall}{Precision + Recall}$$

5 Results

Based on the results shown in Table 1 the DeBERTa-XLarge model (He et al. (2021)) performed the best. The BERT-CRF model was significantly better than the BERT model (Devlin et al. (2018)), but it still wasn't better than the DeBERTa models. As expected, the DeBERTa-XLarge model slightly outperformed the DeBERTa-Large model since it had more parameters. The DeBERTa-CRF model performed much worse than we had hoped and was worse than even the base BERT model.

Model	Precision	Recall	F1
BERT	**0.870**	0.811	0.839
BERT-CRF	0.851	0.862	0.856
DeBERTa-Large	**0.870**	0.872	0.871
DeBERTa-XLarge	0.868	**0.876**	**0.872**
DeBERTa-CRF	0.820	0.830	0.825

Table 1: Summary of the scores of all the models tested in this paper. All the scores are from testing on the SemEval 2022 Task 11 validation data set (Malmasi et al. (2022a)).

Interestingly the BERT base model (Devlin et al. (2018)) was tied for the highest precision score even though its recall and F1 scores were relatively low. This means that when it did make a prediction the BERT model was the most accurate at labeling that entity. Out of all the models tried the DeBERTa-XLarge model (He et al. (2021)) ended up as the best scoring model overall and had the highest recall score.

For the official results we used our best model which was the DeBERTa-XLarge model (He et al. (2021)) and made predictions on the provided test data set (Malmasi et al. (2022a)). According to those official SemEval 2022 Task 11 metrics our model had a macro F1 score of 0.717 on the test data. This score put us in 16th place in the competition results.

5.1 Category Results

Table 2 displays the results of our best model, the DeBERTa-XLarge model, on each tag category. Person and location both have high scores with creative work and product being much lower. This makes sense because person and location had the most instances in the training data. Intuitively it also makes sense because person and location are generally simpler entities to identify.

Model	Precision	Recall	F1
Corporation	0.82	0.84	0.83
Creative Work	0.71	0.74	0.73
Group	0.80	0.88	0.84
Location	0.88	0.93	0.91
Person	**0.94**	**0.96**	**0.95**
Product	0.78	0.81	0.80

Table 2: The results of our best model the DeBERTa-XLarge model on each possible label. These scores are on from evaluating on the SemEval 2022 Task 11 validation data set (Malmasi et al. (2022a)).

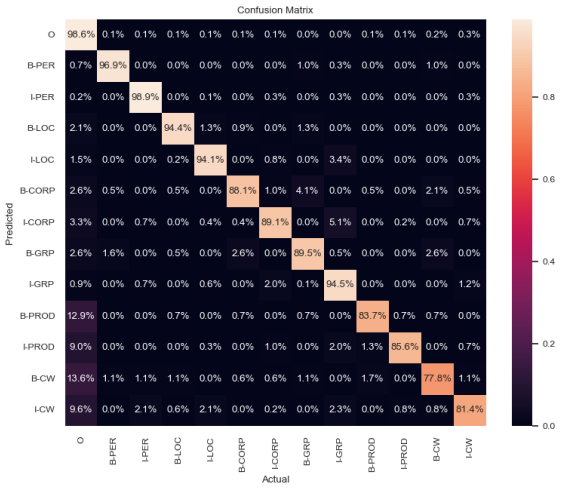

Figure 2: This figure shows a confusion matrix for the different entities that the DeBERTa-XLarge model had to identify. These scores are on from evaluating on the SemEval 2022 Task 11 validation data set (Malmasi et al. (2022a)).

5.2 Error Analysis

Figure 2 shows a confusion matrix for out best model, the DeBERTa-XLarge model. It shows a couple of key areas where our model is struggling. The matrix shows that while our model is doing well most of the time, there are a couple of labels it commonly confuses. For example it struggles distinguishing between group and corporation. As seen in the matrix when our model predicts B-CORP it is right 88.1% of the time and its most common error is B-GRP which it mistook for B-CORP 4.1% of the time. Logically this makes sense because those labels are quite semantically similar. Interestingly for the entities the model is the worst at predicting (product and creative work), the most common mistake is labeling them PROD or CW when in reality it isn't an entity and is O.

Another surprising error is the number of I or inside entities the model struggles with. It was expected that the model would figure out that the for example an inside entity like I-PER can not follow a start entity of a different type like B-LOC. In an ideal case the model shouldn't mistake an I entity with another I entity, it should just match the B entity. However there are cases in the output of this happening. This would be a good problem to fix in future work.

6 Conclusion

We tried many different types of models for this SemEval 2022 Task 11 competition. We tried training each of these models on the training data, a BERT model, a BERT-CRF model, a DeBERTa-Large model, a DeBERTa-XLarge model, and lastly a DeBERTa-CRF model. In the end the fine-tuned DeBERTa-XLarge model achieved the highest F1-score. We also found out that adding a CRF layer to a DeBERTa pre-trained model didn't help its performance at all.

For any future work or improvements on our current work, we would like to try adding some other layers to the DeBERTa model and work on fixing some commmon errors our model has. Since the DeBERTa model seems to be the best at NER, we would like to modify it in some way or modify the data in some way to help it perform better. With further adjustments and experimentation, we believe that the performance of our DeBERTa model could improve.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention.

Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. 2020. A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering*, 34(1):50–70.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022a. MultiCoNER: a Large-scale Multilingual dataset for Complex Named Entity Recognition.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022b. SemEval-2022 Task 11: Multilingual Complex Named Entity Recognition (MultiCoNER). In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Tao Meng, Anjie Fang, Oleg Rokhlenko, and Shervin Malmasi. 2021. GEMNET: Effective gated gazetteer representations for recognizing complex entities in low-context input. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1499–1512.

Hiroki Nakayama. 2018. seqeval: A python framework for sequence labeling evaluation. Software available from https://github.com/chakki-works/seqeval.

Fábio Souza, Rodrigo Frassetto Nogueira, and Roberto de Alencar Lotufo. 2019. Portuguese named entity recognition using BERT-CRF. *CoRR*, abs/1909.10649.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Huichen Yang and William Hsu. 2021. Named entity recognition from synthesis procedural text in materials science domain with attention-based approach. In *SDU@ AAAI*.