Semi-supervised Learning in Named Entity Recognition

Peining Zhang
Rutgers University
New Brunswick, NJ 08854
pz129@rutgers.edu

Abstract

In the past decades, we have witnessed the rapid development of deep neural networks. The training of the modern, deep neural networks, relies heavily on large labeled data sets. Yet, collecting labeled data is expensive for many learning tasks because it necessarily involves expert knowledge. Semi-supervised learning has proven to be a powerful approach for leveraging unlabeled data to mitigate the reliance on large labeled datasets. There has been many methods and models to implement semi-supervised learning on deep neural networks. Named entity recognition(NER) is one of the most common natural language processing tasks involving linguistics, whose labeling is relatively more expensive than classification. This work attempts to introduce the application of the modern mainstream algorithms of the semi-supervised learning mainly in classification problems, to NER, a sequence labeling problem.

1 Problem

As the increasingly ability to train larger and deeper neural networks, our current AI models get substantial progress. More and more techniques have been proposed to train on larger data sets. However, due to the higher labeling cost, it is difficult for some problems to obtain a larger data set. The pre-trained models that have recently emerged in the NLP field have given this approach greater feasibility. Our approach generally uses pre-trained models to fine-tune a small amount of labeled data and more unlabeled data.

In our project, we decided to first fine-tine the ALBERT[Lan+19] on the PeopleDaily1998 and CoNLL03 NER dataset. And we plan to introduce the new training paradigm called "**MixMatch**" [Ber+19]. It is said to be a good training paradigm which obtains state-of-the-art results on all standard image benchmarks.

What we need to mention here is that the paper of MixMatch only test on four computer vision dataset(CIFAR-10 and CIFAR-100[KH+09], SVHN[Goo+13], and STL-10). So we would refer another semi-supervised method "**Unsupervised Data Augmentation**" [Xie+19], which finished its experiments on IMDb dataset, a text classification task.

2 Goal

For the supervised learning as a baseline, We have done some works on fine-tuning ALBERT on PeopleDaily1998 dateset, and it gets 0.947 F-score on the test set. When we reduce the size of the dataset to 10%, the F-score sharply decrease to 0.908, and 0.886 on 3% of the dataset. Our goal is to narrow the gap bewteen before and after the labeled data set becomes smaller.

Preprint. Under review.

Our task would have two kinds of baselines: the Supervised result and the semi-supervised results by other methods.

3 Related Work

3.1 Named Entity Recognition

NER was known as a significant component of Information Extraction, and has become important for many down streaming natural language processing applications. NER systems are often used as the first step in information retrieval, question answering, co-reference resolution, topic modeling, etc. Recently, starting with [Col+11], NER systems based on neural networks have become increasingly popular due to the minimal requirements of feature engineering, which contributes to a higher domain independence. The LSTM architecture [Lam+16] has been commonly used in NER task, which use LSTM to extract word-level features. lee2013pseudo Recent works explored contextual embedding extracted from language models, [WMR+20] introduces the pretraining model of BERT to the task of NER. ALBERT, a pretraining model replacing BERT with much less parameter by parameter sharing, is proposed in [Lan+19] to make the fine-tuning of the downstream model more easier. Our model use ALBERT as the pretraining model and feature extractor, and follows with a linear layer to get the predicted entities classes.

3.2 Semi-supervised Learning

Semi-supevised learning (SSL) seeks to alleviate the demand of labeled data by allowing the model to leverage unlabeled data. Pseudo label is a simple and efficient semi-supervised learning method introduced in [Lee13], which shows that it is equivalent to Entropy Regularization. In the work of Mixmatch[Ber+19], they sharpened the average of the prediction of the augmented data, as the pseudo label. And mix them up as [Zha+17] with the labeled data to increase the reliability of the pseudo label. In our work, we use a confidence based method to generate the pseudo labels, and also the sharpened average of prediction for the augmented data.

Many recent works for semi-supervised learning use loss terms computed from unlabeled data and encourage the model to generalize better on these data, to improve the performance on the unseen data. In our work, we use pseudo labeling and also two loss term that limits the output probability has the desired properties.

4 Methodology

4.1 Data Augmentation

Data augmentation is a common regularization technique in supervised learning, which applies random transformations on inputs and keep the classes unchanged. For example, in the image classification problem, it is common to add noise to an input image or do some deformations. In our work, we use random insertion as the data augmentation, for our task limits most of the common augmentation methods in NLP, like back translation and random dropping. In those ways, we cannot determine the label for the augmented sentences, and even only dropping one word would make the sentence lose the start word of an entity, which causes the failure in recognizing the whole entity.
We add 5 random words in random place for each input sentences, and label the inserted words a special index, which is beyond the prediction. In the supervised training, we filtered the labels with a mask that the inserted words to be 0 and other to be 1, so it makes the inserted words no included in the training cross entropy loss.

4.2 Pseudo Labeling

We use the pseudo label as $q = argmax(\frac{1}{K}\sum_{i=1}^{K} P_{model}(y|u_k, \theta))$, where q is the guessed label of the unlabeled data, u_k is the kth augmentation of the unlabeled sample u, P_{model} is the model trained in supervised way. Here we use a different sharpen function argmax from the temperature based sharpen function $p_i^{\frac{1}{T}} / \sum_{j=1}^{L} p_j^{\frac{1}{T}}$ used in MixMatch, for the labeling is highly related in the sequence labeling task.

Labels	0.1	0.05	0.03	0.01
F-Score	0.858	0.863	0.866	0.876

Table 1: The semi-supervised training on 500 labeled data, and filter the unlabeled data with real labels in different thresholds

Labels	500	1500	5000	15000	50000
Supervised	0.834	0.887	0.908	0.930	0.947
Augment-Supervised	0.843	0.892	0.913	0.923	0.946
Conf semi-supervised	0.845	0.893	0.913	–	–
Our model	**0.865**	**0.903**	**0.925**	–	–

Table 2: F-score of different model on Evaluation Set

The confidence mentioned above is defined as $conf = mean_{t=1:T}(max_{i \in labels}(prob_{i,t}))$, which is predicted probability for the argmax labels.

4.3 Consistency Regularization

Data augmentation is a common technique to regularize in supervised learning, and we can also apply it here on unsupervised data to increase the consistency regularization. The loss term

$$||P_{model}(y|Augment(x);\theta) - P_{model}(y|Augment(x);\theta)||_2^2$$

enforces that different augmentation of a unlabeled example x should be classified the same. The loss term is used in the two supervised training part on labeled data and unlabeled data with guessed labels.

5 Experiment

5.1 Implementation details

We test the effectiveness of our model on different labeled data size on our datasets. In all experiments, we use ALBERT base model, with a linear model to the predicted logits and softmax to the probabilities. For the supervised model, we train about 500 steps on the small datasets to make the model converge well, and train more steps on large datasets to make sure all the samples are seen more than 2 times. In the labeling guessing part, we predict all samples in the unlabeled dataset for K(K=3) times, and compute their pseudo label. In the subsequent training, we mix the labeled and unlabeled data with pseudo labels, where the original labeled data is weighted 5 times in the loss function to ensure the model's accuracy on current labeled data. We trained the mixed dataset for 2 epoches, which is about 3000 steps, as we use the batchsize of 32, and we totally have about 50000 samples.

5.2 Supervised Learning Baseline

In the experiment, we found that data augmentation make limited gains on the result in the supervised case, which shows the effect and necessity of the semi-supervised learning. However, using the confidence to select the samples require us to verify that the higher confidence comes with higher accuracy label. In our experiment, we can observe that the statement is true. When we use the threshold 0.01 for the confidence, we filter 17% of the unlabeled data, and the average error rate in the unfiltered data is 0.009, when is is 0.076 in the filtered data. We also test the filter using the real label of the unlabeled data, and use the predicted labeled as the subsequent training. As shown in 1, the smaller the threshold is, the higher F-Score we can get in the subsequent training.

As shown in 2, our model is significant higher than the confidence based semi-supervised learning model, which indicates the effect of the K augmentations, and the consistency regularization. It seems that the F-Score is just slightly higher than the supervised way, but as it's based on the pretraining model, such improvement is almost equals to the effect of 3 times amount of data, which has some practical significance. As the fine-tuning on the pretraining models usually only require the training

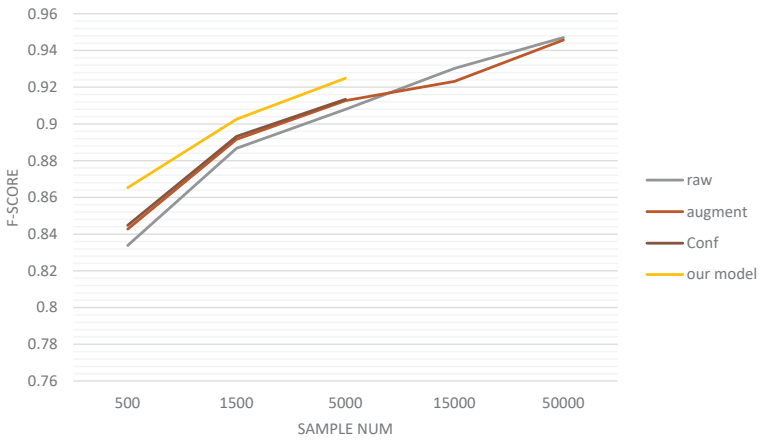

Figure 1: Comparing with the baselines, our model has a significant improvement on F-Score for different sample numbers

for not more than 5 epoches(less than 3 in our experiment among different labeled data size), the more computing resource cost in this process is acceptable.

6 Conclusion

We apply the modern semi-supervised learning techniques and pretraining model on the NER task, and get significant improvement on the F-Score. By using the data augmentation, pseudo labeling and consistency regularization, our model can get the competing result with only about 30% labeled data. Due to the limit of time and the limit flexibility of the ALBERT model, we believe our work can still have further progress by add more components.

References

References

[KH+09] Alex Krizhevsky, Geoffrey Hinton, et al. “Learning multiple layers of features from tiny images”. In: (2009).

[Col+11] Ronan Collobert et al. “Natural language processing (almost) from scratch”. In: *Journal of machine learning research* 12.Aug (2011), pp. 2493–2537.

[Goo+13] Ian J Goodfellow et al. “Multi-digit number recognition from street view imagery using deep convolutional neural networks”. In: *arXiv preprint arXiv:1312.6082* (2013).

[Lee13] Dong-Hyun Lee. “Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks”. In: *Workshop on challenges in representation learning, ICML*. Vol. 3. 2013, p. 2.

[Lam+16] Guillaume Lample et al. “Neural architectures for named entity recognition”. In: *arXiv preprint arXiv:1603.01360* (2016).

[Zha+17] Hongyi Zhang et al. “mixup: Beyond empirical risk minimization”. In: *arXiv preprint arXiv:1710.09412* (2017).

[Ber+19] David Berthelot et al. “Mixmatch: A holistic approach to semi-supervised learning”. In: *Advances in Neural Information Processing Systems*. 2019, pp. 5050–5060.

[Lan+19] Zhenzhong Lan et al. “Albert: A lite bert for self-supervised learning of language representations”. In: *arXiv preprint arXiv:1909.11942* (2019).

[Xie+19] Qizhe Xie et al. “Unsupervised data augmentation”. In: *arXiv preprint arXiv:1904.12848* (2019).

[WMR+20] Zihan Wang, Stephen Mayhew, Dan Roth, et al. “Extending Multilingual BERT to Low-Resource Languages”. In: *arXiv preprint arXiv:2004.13640* (2020).