Fine-tuning for Named Entity Recognition in EHR with Meta Pseudo Label

By F.Y. Huang*

Supervisor: Dr. Y.H. Deng

Statistics Program, Faculty of Science and Technology, HongKong Baptist University

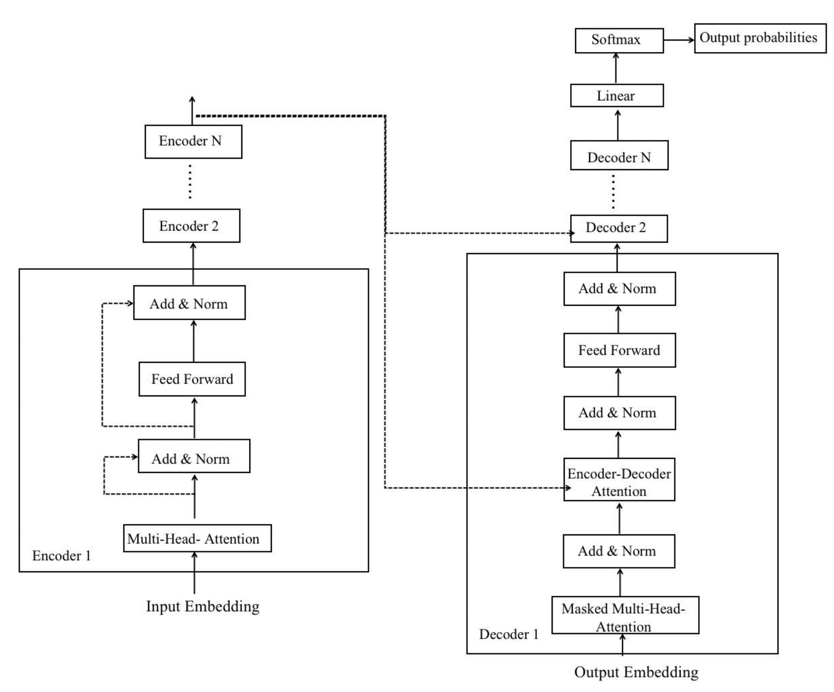
Abstracts

Electronic medical records (EMRs) contain unstructured data that lack a set structure, making it challenging to perform statistical analysis and other studies. Named Entity Recognition (NER) is a crucial step in analyzing medical knowledge within EMRs. However, training a pre-existing NER model based on a specific corpus can be time-consuming. To address this issue, this paper proposes a novel semi-supervised learning knowledge distillation method, known as Meta Pseudo Label, to fine-tune an NER model for EMR data. Compared to the traditional Pseudo Labels approach, there is an additional Meta modelling process. The traditional Pseudo Labels based distillation method is based on a pre-trained Teacher model, using the pseudo labels provided by the Teacher model as the Target of the Student model for training. On the other hand, MPL help optimize the Teacher model by using the Student model's loss on labelled data.MPL also differ from other semi-supervised models as its Teacher model is not updated by the exponentially weighted moving average (EMA) method but by the gradient method. Furthermore, the study aims to map the diverse ways of expressing clinical concepts by medical students in clinical patient notes to standard clinical concepts. This approach provides a more efficient and accurate means of analyzing and utilizing EMR data, thereby improving patient care and outcomes.

Keywords: Natural Language Processing, Knowledge Distillation, Fine-tune, Named Entity Recognition, Meta Pseudo Label

Introduction

NER is a subfield of NLP, which aims to identify and classify named entities, such as locations and dates. NER plays a vital role in various applications, and it is the primary step in electronic medical record text mining and information extraction research. In NER, the text is typically treated as a sequence labelling problem. Each word will be assigned a label indicating its type. Because the meaning of a word depends on the words that come before and after it in a sentence, the task of NER typically involves defining a probability distribution over the possible sequences of labels. An optimization algorithm is used to find the sequence that maximizes the probability.

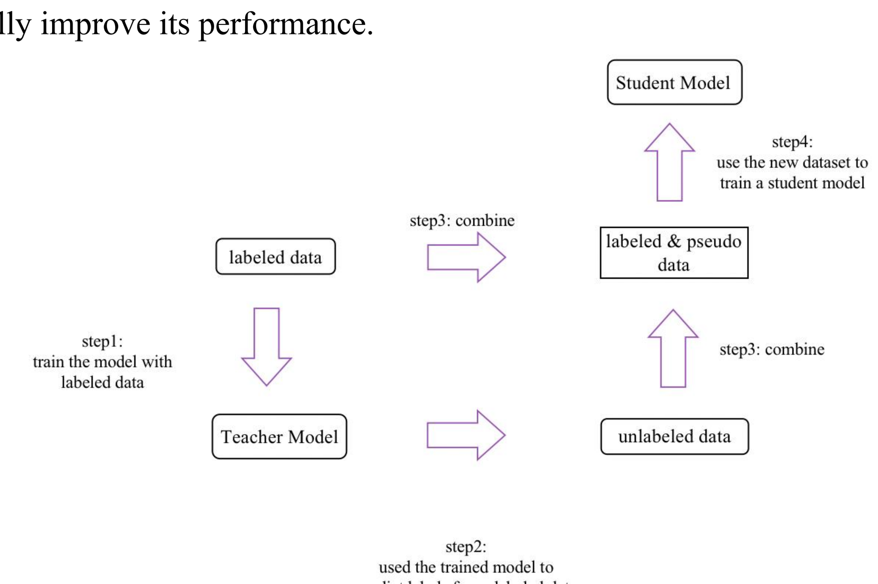


There are many machine learning and deep learning methods used in NER. BERT is the most popular among them. BERT is a pre-trained language model trained using a self-supervised learning method called masked language modelling. However, using a pre-trained model like BERT on a new task or dataset may not always produce the best results. This is because the pre-trained model has already been trained on a large amount of data and has learned certain features and patterns that may not be relevant to the new task or dataset. Therefore, Fine-tuning, a technique that adapts a pre-trained model to a specific dataset, is proposed. Fine-tuning involves adding additional layers to the pre-trained model and training it on the new dataset, allowing one to leverage the knowledge from the pre-trained model and adapt it to the specific characteristics of the new dataset.

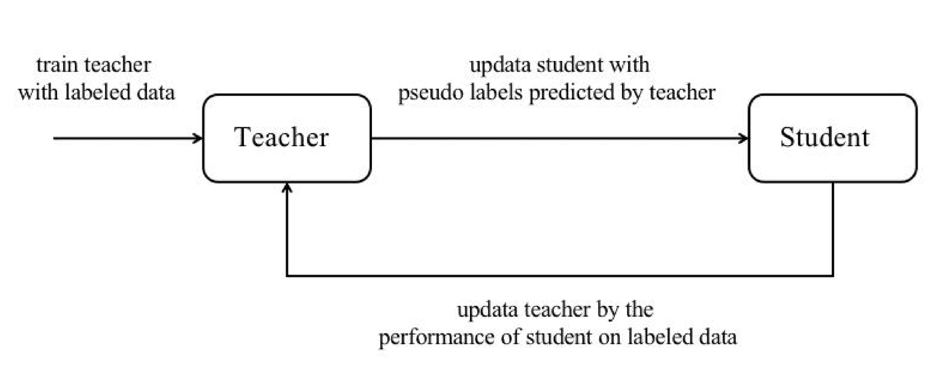
Methods and Experiment

Methodology

Pseudo-labelling, a method of fine-tuning, is a semi-supervised learning technique that can be used to improve the performance of machine learning models. It involves using a trained model, denoted as the Teacher model, to make predictions on a set of unlabeled data and then using the predicted labels as the "pseudo labels" for the unlabeled data. The labelled and pseudo-labelled data are combined to train a new student model. Pseudo-labelling can be particularly useful when relatively few labelled examples are available, as it allows the model to learn from a larger dataset and potentially improve its performance.



Despite the superior performance of the pseudo-labelling method, it also has a significant drawback. If the pseudo-labelling is inaccurate, the student model has to learn from the incorrect data. As a result, the final trained student model may not be much better than the teacher model. This drawback is also known as the confirmation bias problem of pseudo-labelling. To address this issue, the Teacher model needs to correct for bias through the effect of its pseudo-labels on the Student model, which is exactly Meta Pseudo model (MPL).



Experiment

A dataset from the USMLE Clinical Skills Examination is used, which requires converting the diverse expressions of certain concepts found in clinical patient notes written by medical students into standard clinical concepts. There are 14300 labelled data and 42146 unlabelled data, randomly choosing 70% of labelled data as the training set, and the rest would be the testing set.

Algorithm 1 Meta Pseudo Label Algorithm

1: Input labelled dat x_l, y_l and unlabelled data x_u

2: Initialize $\theta_T^{(0)}$ and $\theta_S^{(0)}$, the parameters for Teacher model and Student model

3: **for** episode $= 0$ to N **do**

4: Generate pseudo label from Teacher model $\hat{y}_l \sim P(\cdot|x_u, \theta_T)$

5: Update student model using $\hat{y}_l$

$$\theta_S^{(t+1)} = \theta_S^{(t)} - \eta_S \nabla_{\theta_S} \mathrm{CE}\left(\hat{y}_u,\, S\left(x_u; \theta_S\right)\right|_{\theta_S=\theta_S^{(t)}}$$

6: Compute feedback coefficient and gradient for teacher model based on the the cross-entropy of student on labelled data

$$h = \eta_S \cdot \left(\left(\nabla_{\theta'_S} \mathrm{CE}\left(y_l, S\left(x_l; \theta_S^{(t+1)}\right)\right)\right)^{\top} \cdot \nabla_{\theta_S} \mathrm{CE}\left(\widehat{y}_u, S\left(x_u; \theta_S^{(t)}\right)\right)\right)$$

$$g_T^{(t)} = h \cdot \nabla_{\theta_T} \mathrm{CE}\left(\widehat{y}_u, T\left(x_u; \theta_T\right)\right)\Big|_{\theta_T=\theta_T^{(t)}}$$

7: Compute the gradient of teacher model on labelled data

$$g_{T,supervised}^{(t)} = \nabla_{\theta_T} CE\left(y_l, T\left(x_l; \theta_T\right)\right)\Big|_{\theta_T=\theta_T^{(t)}}$$

8: Compute the the gradient of teacher model on the UDA loss with unlabelled data

$$g_{T,\mathrm{UDA}}^{(t)} = \nabla_{\theta_T} CE\left(StopGradient\left(T\left(x_l\right); \theta_T\right), T\left(RandAugment\left(x_l\right); \theta_T\right)\right)\Big|_{\theta_T=\theta_T^{(t)}}$$

9: Update the parameter of teacher model

$$\theta_T^{(t+1)} = \theta_T^{(t)} - \eta_T \cdot \left(g_T^{(t)} + g_{T,supervised}^{(t)} + g_{T,\mathrm{UDA}}^{(t)}\right)$$

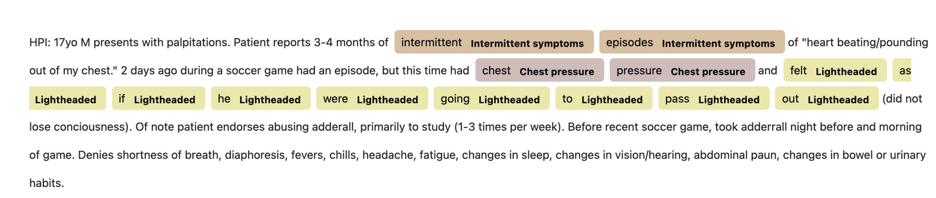
10: **end for**

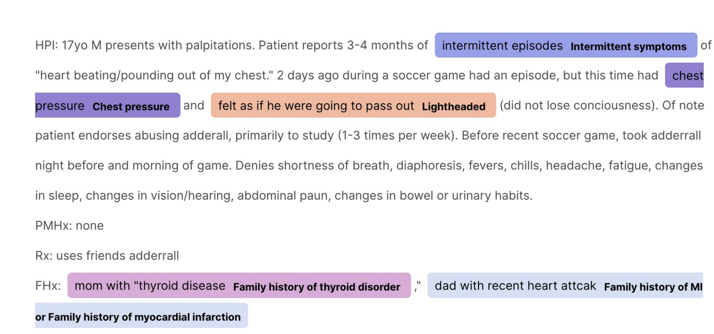
The problem's final output is the starting and ending locations of the target sentence, with label 1 for words within the target sentence and label 0 for terms that are not part of the answer. That is, the problem can be considered a classification problem. Therefore, in the baseline, a fully connected layer is directly used after the pre-trained model BERT-base to get the results.

The effect of MPL will beverified by comparing the results of baseline, fine-tuning with Bi-LSTM, and fine-tuning with MPL.

Results and Discussion

This is an example of the final extraction result. We can see that it extracts and maps all the sentences related to various features to the standard expression.





Here is the F1-score of these three models, Baseline model, Baseline model with Bi-LSTM layer, and Baseline model with fine-tuning.

Model	F1-score
Baseline	0.7922
Baseline+Bi-LSTM	0.7838
Baseline+Fine-tuning	0.8651

It can be seen that the F1-score decreases by about 0.01 compared to the baseline after adding the Bi-LSTM layer, which may be because the addition of before and after clause features weakens the model's learning of the current clause features, thus affecting the final classification results. Adding a Bi-LSTM layer may give different results depending on other datasets' features. Positive or negative ones are possible. As for MPL, fine-tuning the data with labelled and unlabelled data that can effectively update the model according to the dataset's characteristics. Therefore, it has better results. The experimental results prove this, and the MPL brings nearly 0.1 improvements.

Conclusion

The results of the experiments presented in this paper suggest that MPL is an effective method for fine-tuning pre-trained models on a specific dataset. However, implementing MPL takes less time and data than retraining a pre-trained model. Furthermore, it will not introduce indeterminable effects on the model, as can be true when modifying the neural network architecture by adding Bi-LSTM layers. This is because MPL works by leveraging the characteristics of the data itself rather than directly altering the model's underlying structure.

References

- Hinton, G., Vinyals, O., & Dean, J. (2015). *Distilling the knowledge in a neural network.* arXiv preprint arXiv:1503.02531, 2(7).
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). *Attention is all you need.* Advances in neural information processing systems, 30.
- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). *Bert: Pre-training of deep bidirectional transformers for language understanding.* arXiv preprint arXiv:1810.04805.
- Pham, H., Dai, Z., Xie, Q., & Le, Q. V.(2021). *Meta pseudo labels.* In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 11557-11568).