Summer Research

Author: Tengfei Cui

Supervisor: Lequan Yu

Content:

1.Introduction:

Clinical notes generated by healthcare professionals are parts of electronic health records and provide an essential source for intelligent healthcare applications (Zhang et al., 2020). Medical code prediction aims to assign standard medical codes to each clinical document, which requires professional medical knowledge and is usually costly and error-prone (Hsia et al., 1988; Stanfill et al., 2010). The International Classification of Diseases (ICD) system, as the most used coding system, provides a global standard for reporting diseases and health conditions.

The development of deep learning and natural language processing (NLP) can replace the conventional manual code assignment. Pretrained language models such as BERT (Devlin et al., 2019) learn contextualized text representation and have started a new era in NLP. NLP applications benefit from large-scale pretraining on massive corpora, and universal language representations from pretrained language models have been successfully utilized in downstream tasks via transfer learning. In the field of clinical NLP, incorporating pretrained contextualized language models to encode lengthy clinical notes for large-scale medical code prediction has not been well-studied.

Understanding medical text is a long-lasting research problem. We study an essential task – medical code prediction, assigning medical codes (ICD-9) to clinical notes. Many pretrained models have been published for medical NLP, such as BERT-base-uncased, Clinical-BERT, BioBERT and Clinical-Longformer. We study the usefulness and relative merits of these different pretrained models and suggest improvements to the neural network architecture to improve performance with long notes: a hierarchical model for longer notes and data augmentation. We hope to improve the performance of pre-trained models without dramatically increasing computation resource. And because the limit of computation resource, we choose the top-50 codes (see section 3) to predict. You could easily change my code to predict all medical codes.

Despite our careful attempts, we nevertheless find that fine-tuning pretrained models performs worse than carefully training conventional neural architectures. Hence, our results provide practical guidance for building medical information management systems: pretrained models offer a convenient plug-and-play solution; however, training robust existing standard models offers an appealing practical alternative with good performance in practice.
Our contributions are as follows:

- We propose some new hierarchical BERT architecture, attempt some data augmentation techniques, and conduct a comprehensive quantitative study to investigate the improvement.
- Use Clinical-Longformer to do the task and compare the result with the performance of other models to study the effect of spares attention mechanism

2.Related Work:

Rule-based and machine learning-based methods have been studied for diagnosis code assignment from clinical notes (Medori and Fairon, 2010; Perotte et al., 2014). Perotte et al. (2014) proposed an SVM-based classification algorithm with a flat and hierarchy-based classifier. Recently, the research trend turns to deep neural networks. Convolutional neural networks are one popular category with many model architecture proposed, including CAML that applies CNNs and a label-wise attention mechanism (Mullenbach et al., 2018), MultiResCNN that uses residual connection (Li and Yu, 2020) and DCAN that utilizes dilated convolutions (Ji et al., 2020a). Recurrent neural networks are also extensively studied to capture sequential dependency in clinical notes. Such recurrent models include AttentiveLSTM (Shi et al., 2017), HA-GRU (Baumel et al., 2018) and tree-of-sequences LSTM network (Xie and Xing, 2018) Attention mechanism for matching important diagnosis snippets is widely integrated into CNN- and RNN-based models (Shi et al., 2017; Dong et al., 2020). CAML (Mullenbach et al., 2018) introduced a label-wise attention mechanism to learn label-aware document representations.

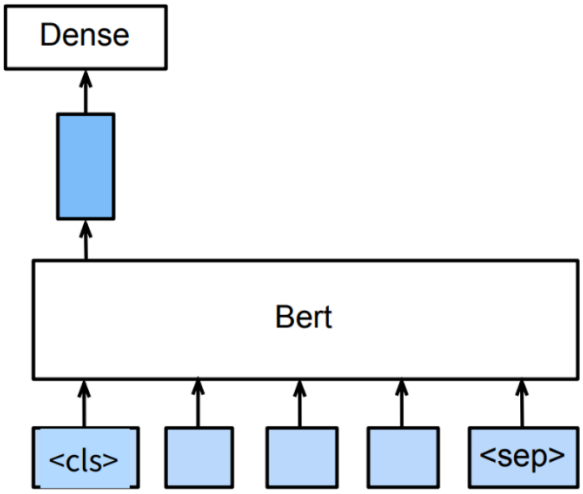
The challenging of our project is that transformer models, such as BERT and Clinical-BERT, cannot read long text (more than 512 tokens). Although BERT and its varieties perform well on many tasks, they cannot beat CNN-based models in the medical code prediction. Because of the self-attention mechanism and limited computation resource, most BERT models and its varieties can take at most 512 words at one time. But, a large percentage of data have more than 1000 words and BERT models have to omit many valuable information. Therefore, the goal of our project is to overcome this difficulty. We hope to let BERT models have access to more information without dramatically increasing our computation recourse.

3.Method:

we develop fine-tuning with different architectures, including a fully-connected classifier, a hierarchical classifier with an extra transformer / CNN layer and a hierarchical classifier trained by augmented data. The details of each model are discussed below:

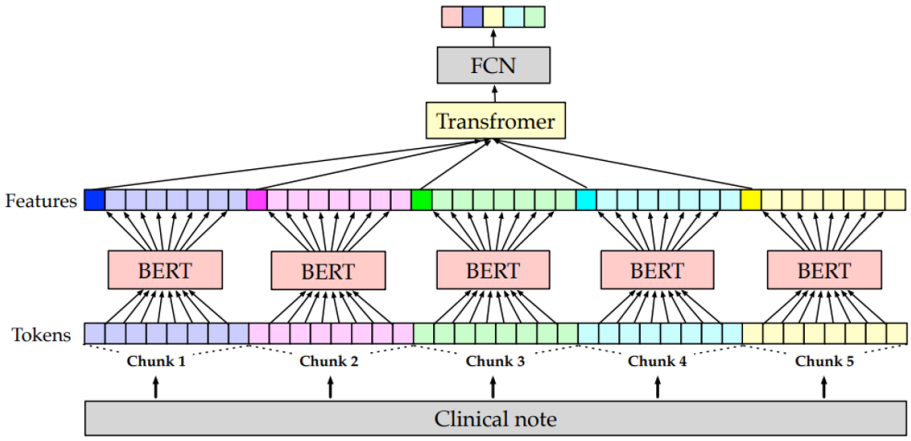
3.1 BERT + FCN

Firstly, let's look at how bert models perform without any other techniques. Here we use a common pipeline, a BERT model to encode the processed data and then connected to a full connection network for classification.



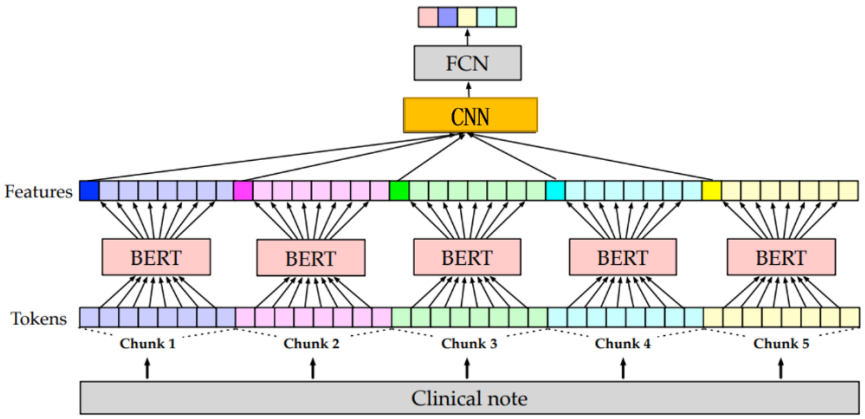
3.2 HierBERT + Transformer

A natural idea to improve the maximum length of BERT is to use different BERT models and let them encode different parts of inputs separately. We call this strategy as hierarchical structure. And then we could combine the features extracted by BERT models by a transformer layer and send the features to a full connected network (FCN). The pipeline is as shown below:



3.3 HierBERT + CNN / DPCNN

In the last part, we use a transformer encoder to deal with the combined features extracted by separated BERT models. Now, we try to replace the transformer encoder by a simple CNN network and a DPCNN network.



The reason we make this try is that CNN-based models perform outstandingly on this task. This might because compared with transformer models, such as BERT , which are good at comprehend the meaning of text, CNN-based models are more likely focus on some key phrases useful for the prediction. So we hope to make the use of the strength of both neural network.

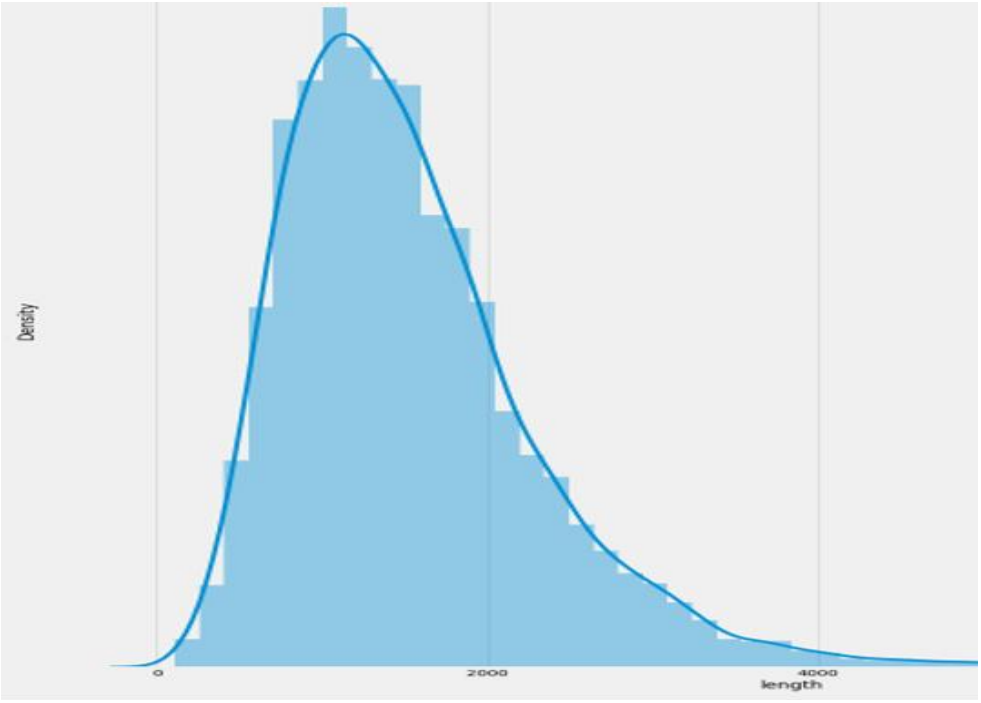
3.4 HierBERT + Data Augmentation

Apart from changing the design of models, we could also do the data augmentation to improve the performance. In fact, due to the limit of computational resource, we could at most use 3 different BERT models to take the input text, which means that the hierarchical structure discussed above could only read about 512*3 = 1536 words. Besides due to the rule of tokenization of BERT, one

word may be divided into different parts for models, the number of words our hierarchical model could deal with would decrease to about 1200 words. From the distribution of the length of input text, it seems that our model could still not make full use of data. So, we implement the following data augmentation:

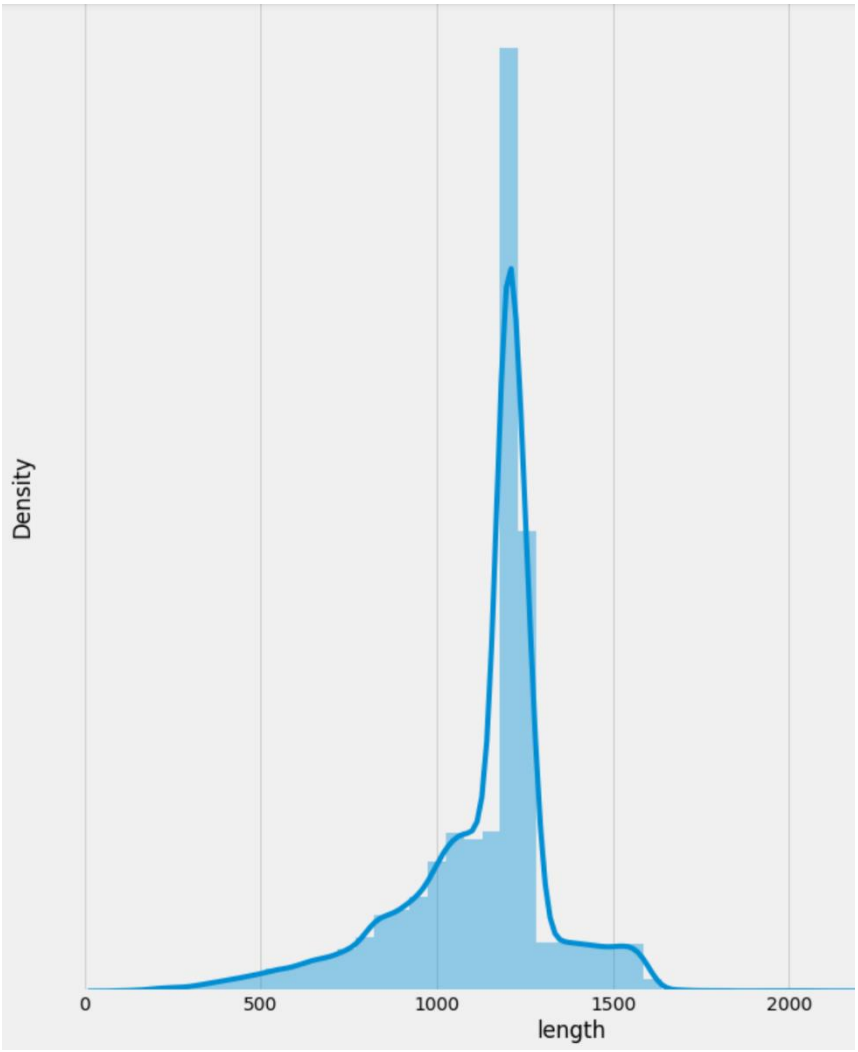
- In training process, if the length of input text is larger than 1800, we use an extractive text summarization model (**Roberta-base-extsum** from **transformersum**) to reduce the length of text about 1200
- In training process, if the length of input text is larger than 1600 , we split the text into multiple text chunks , so that each of the length of text chunk is about 1000 words . If the length of some text chunk is has to below 800, we use the raw text to complement the text chunks, so that each of text chunks has about 1000 words .
- In testing process, if the length of text input is larger than 1600, we also use extractive text summarization model to reduce the input. In the following, we will compare the performance of models by different test input text . (The original one is just to truncate the input when the length is over 1600 , the newer one is to summarize the input text .) Since we do not sure whether the summarization of test data could improve the performance, we do use both of data (raw text data and summarized data, which will be called **DA1** and **DA2** in the following part) .

Before the data augmentation, the distribution of the length of training text data is :



x-axis is the length of input text , y-axis is the density of the distribution. It could be seen that a majority of input text have about 1000 – 2500 words.

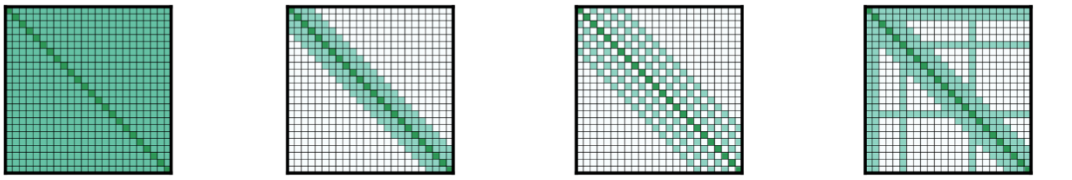
After the data augmentation, the distribution become:



x-axis is the length of input text , y-axis is the density of the distribution. It could be seen that now most of text input has about 1200 words, which is in the range of input length BERT models could tokenize and analyze.

3.5 Clinical-Longformer

Apart from the techniques, we could also use some varieties of BERT, such as Longformer and Big-Bird, which change the self-attention mechanism to a parser attention strategy and therefore reduce the computation cost . So such models could take into text over 4000 words.
Below is the sparse attention mechanism of Longformer:



(a) Full n^2 attention (b) Sliding window attention (c) Dilated sliding window (d) Global+sliding window

3.6 Experimental Setup:

Dataset:

The **MIMIC** database is available by request via this website https://mimic.physionet.org. We use the "**noteevents**" table in the latest release of version 1.4 with a total of 58,576 hospital admissions. Specifically, free-text discharge summaries are extracted. Two settings are shown below, and we only use the former one because of the limited computation resource. One uses the top 50 frequent labels, while another uses the full set of ICD codes. Table below shows a statistical summary of two sets of MIMIC data.

Dataset	Train	Dev.	Test	Labels
MIMIC-III top-50	8,066	1,573	1,729	50
MIMIC-III full	47,723	1,631	3,372	8,921

Preprocessing:

For the row data to processed data, we use the code from the paper, *Explainable Prediction of Medical Codes from Clinical Text*. Please find the repository of the paper by paper with code, and find out the notebook used to clean dataset .

For the processed data to augmented data, Please find my code of this project (https://github.com/ITCUI-XJTLU/Medical_Code_Prediction-), and try to run the notebook *Data_Augmentation_sub1.ipynb.* You might need to run the notebook in the google colab environment and change the file path if you want to use the code.

Training:

Every model will be trained about 10 – 12 epochs. Since we find that it is usually need about 8 epochs to converge the pretrained model.

4. Result :

In the following, we use to pre-trained model, BERT-base-uncase and Clinical-BERT to do the task and use the above techniques to improve their performance.

4.1 BERT + FCN

	F-1 (macro)	F-1 (micro)	AUC (macro)	AUC (micro)
BERT-base-uncase	0.383	0.482	0.807	0.848
Clinical-BERT	**0.460**	**0.541**	**0.821**	**0.859**

It appears that the performance of Clinical-BERT is much better than BERT. It because that Clinical-BERT has been pre-trained buy several medical corpus, including MIMIC III and PubMed. Now, we will try to improve their performance

4.2 HierBERT + Transformer

	F-1 (macro)	F-1 (micro)	AUC (macro)	AUC (micro)
BERT-base-uncase	0.383	0.482	0.807	0.848
Clinical-BERT	0.460	0.541	0.821	0.859
HierBERT-Trans	0.395	0.449	0.770	0.810
HierClinical-Trans	**0.496**	**0.542**	**0.834**	**0.865**

For BERT, although the model could take longer input text, three important scores (f1 micro, auc macro, auc micro) decrease about 3%. It is a little strange. It might because that the most valuable data is among the former part of input or when data read more data, there are more noise.

For Clinical-BERT, the improvement is obvious. F-1 (macro) increase 3%, it is a huge progress. AUC (macro) and AUC (micro) scores increase 1%. It seems that longer input texts is helpful for the Clinical-BERT model to do the prediction.

4.3 HierBERT + CNN / DPCNN

	F-1 (macro)	F-1 (micro)	AUC (macro)	AUC (micro)
BERT-base-uncase	0.383	0.482	0.807	0.848
Clinical-BERT	**0.460**	**0.541**	0.821	**0.859**
HierBERT-CNN	0.437	0.523	**0.825**	**0.859**
HierClinical-CNN	0.338	0.412	0.761	0.809
HierBERT-DPCNN	0.334	0.424	0.770	0.819
HierClinical-DPCNN	0.416	0.509	0.817	0.853

For BERT , it is surprising that when we replace transformer blocks by a simple CNN network, there is a huge improvement . F-1 (macro) and F-1 (micro) score increase 5% and 4% respective, while AUC scores (macro and micro) grows 1% and 2%. We do not know why this happen. Why CNN layer could extract more important features than transformer blocks. Besides, since CNN layer could excellently deal with the features from BERT, it is supposed that DPCNN could do the work better , since it use residual network and therefore could catch longer n-gram words. But this idea does not happen. Contrary to useful for the prediction task, BERT + DPCNN performs worse that the origin model. We have not idea about this phenomenon.

For Clinical-BERT , there is a different story. Although CNN layer could increase the performance of BERT , it does not work for Clinical-BERT. It seems that transformer is more suitable for Clinical-BERT. It is an interesting finding. In the future, if you use different pretrained-models , you may need to change corresponding components to cooperate with them .

4.4 HierBERT + Data Augmentation

	F-1 (macro)	F-1 (micro)	AUC (macro)	AUC (micro)
BERT-base-uncase	0.383	0.482	0.807	0.848
Clinical-BERT	0.460	0.541	0.821	0.859
HierBERT-CNN-DA 1	0.469	0.540	**0.832**	**0.869**
HierBERT-CNN-DA 2	0.453	0.516	0.825	0.861
HierClinical-Trans-DA1	0.500	0.543	0.831	0.867
HierClinical-Trans-DA 2	**0.506**	**0.545**	**0.832**	0.867

For BERT, DA1 is more useful, which means that model prefer to read the origin test data to do the prediction. With the augmented training data and summarized test data, four scores increase1% to 1.5% on average. But the performance of DA2 is not evident. It might because that CNN layer in

the training process has already learn some valuable information of words after 512 , so the model could just use former 512 words of test data to do the prediction. The summary of test data might include more noise. This needs more evidence to demonstrate.

For Clinical-BERT, DA2 is more useful. But this increasement of performance is within 0.5% , it is not huge. At least the augmented data does do influence the performance.

4.5 Clinical-Longformer

	F-1 (macro)	F-1 (micro)	AUC (macro)	AUC (micro)
Clinical-Longformer	0.508	0.594	0.869	0.890

We have to admit that the performance of Clinical-Longformer is much better than any models we have designed. It is not surprised, because Longformer is the model that was specifically designed for dealing with long document and that has already pre-trained on MIMIC III dataset. But , if you want to use this excellent models, you have to keep in mind that the computation cost is also much larger than any model mentioned above . Some good things and some bad things.

Now, let's look at the all result:

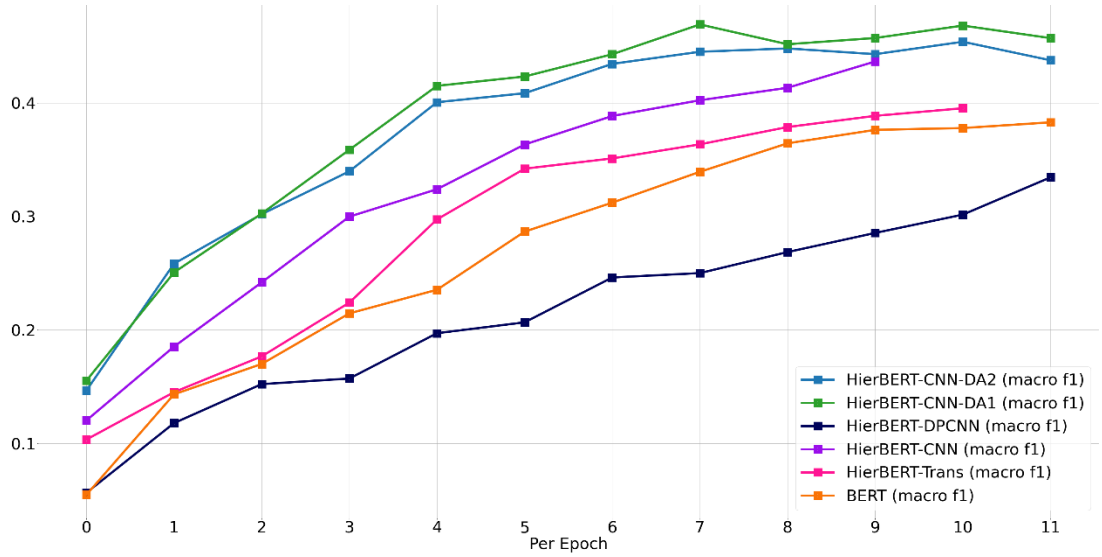
	F-1 (macro)	F-1 (micro)	AUC (macro)	AUC (micro)
BERT-base-uncase	0.383	0.482	0.807	0.848
Clinical-BERT	0.460	**0.541**	0.821	0.859
HierBERT-Trans	0.395	0.449	0.770	0.810
HierClinical-Trans	0.496	**0.542**	**0.834**	0.865
HierBERT-CNN	0.437	0.523	0.825	0.859
HierClinical-CNN	0.338	0.412	0.761	0.809
HierBERT-DPCNN	0.334	0.424	0.770	0.819
HierClinical-DPCNN	0.416	0.509	0.817	0.853
HierBERT-CNN-DA 1	0.469	**0.540**	0.832	**0.869**
HierBERT-CNN-DA 2	0.453	0.516	0.825	0.861
HierClinical-Trans-DA1	**0.500**	**0.543**	0.831	**0.867**
HierClinical-Trans-DA 2	**0.506**	**0.545**	0.832	**0.867**

And below this the performance of the two classical CNN-Based model:

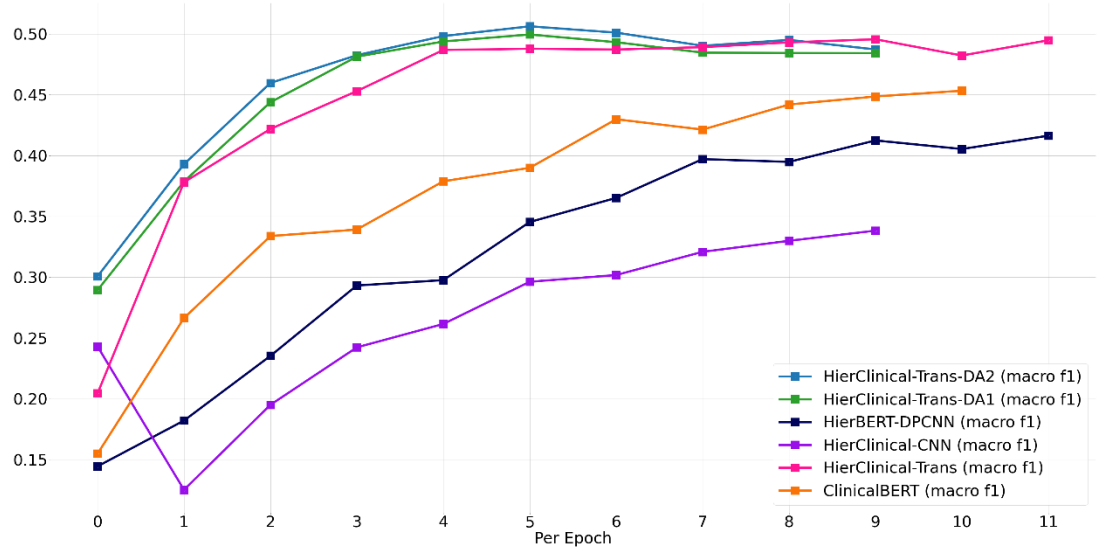
	F-1 (macro)	F-1 (micro)	AUC (macro)	AUC (micro)
CNN (Kim, 2014)	0.576	0.625	0.876	0.907
MultiResCNN (Li and Yu, 2020)	0.606	0.670	0.899	0.928

4.6 Visualize F-1 score and AUC score:

Now , let's look at the growing trend of six techniques.
Below is the change of F-1 score of BERT model with six techniques:



Below is the change of F-1 score of Clinical-BERT model with six techniques:



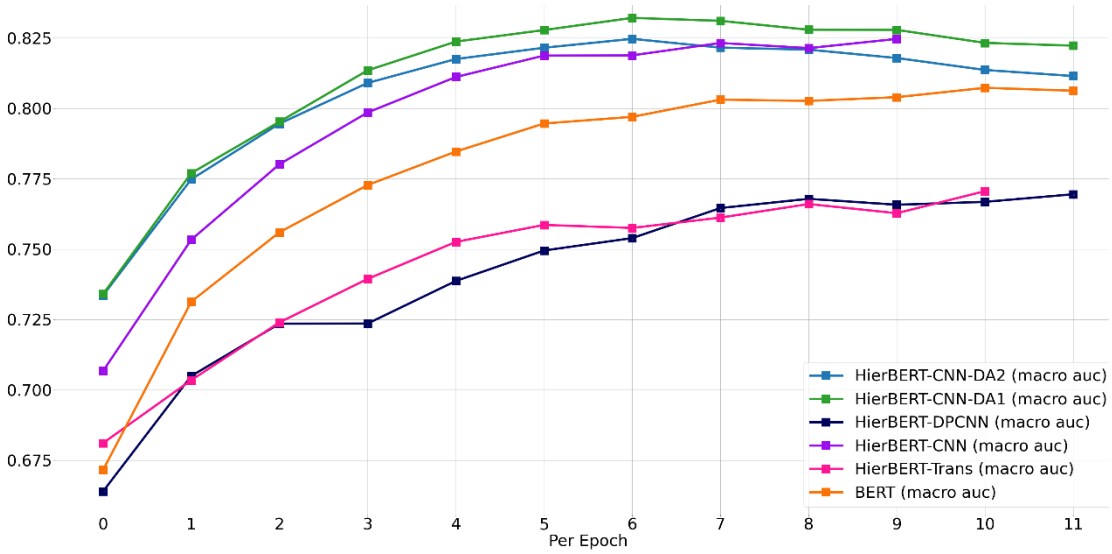
Based on F-1 score of BERT and Clinical-BERT, we have find that :

- All models except HierBERT + DPCNN could increase training process and the increase is also evident.
- Combined with data augmentation, HierBERT + CNN could coverage faster .
- Compare with origin Clinical-BERT , HierClinical-BERT-CNN and HierClinical-BERT-DPCNN , the increased performance of Clinical-transformer is very obvious.
- Clinical-BERT-Transformer is much lower than the same model with augmented data ,

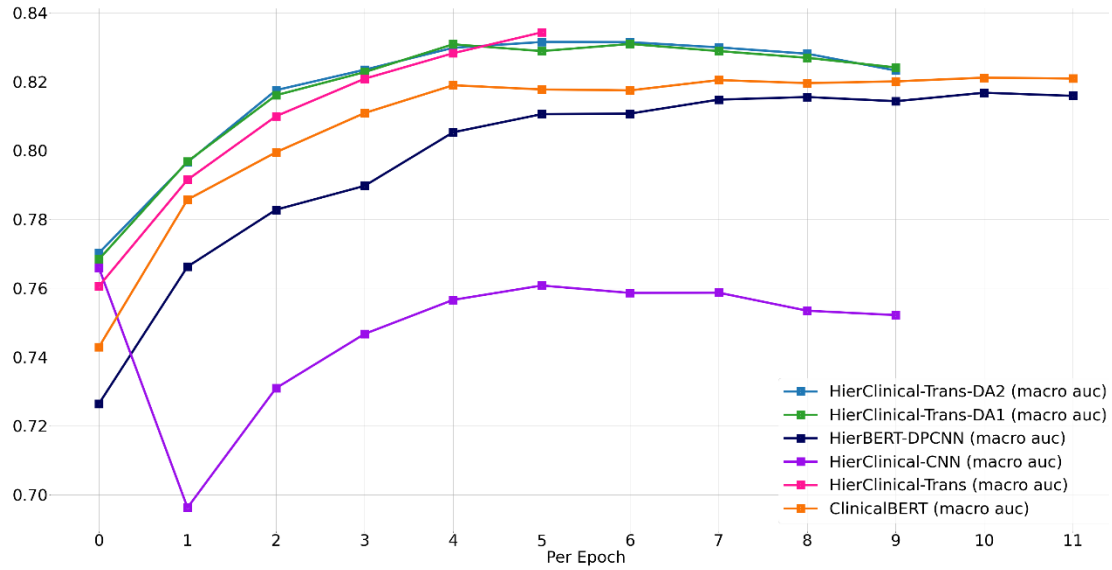
but it grow quickly . Within 3 epochs, the performance of three models are closed .

- Data augmentation is not very useful for Clinical-BERT .

Now, let's look at the change of AUC scores in the training process:
Below is the change of AUC score of BERT model with six techniques:



Below is the change of AUC score of Clinical-BERT model with six techniques:



Based on AUC scores of BERT and Clinical-BERT models, we could find that:

- DPCNN and Transformer blocks would destroy the origin model performance . The positive effect is obvious in this graph.
- Data augmentation could also increase the courage of auc scores. But it seems to be not useful to improve the performance of Clinical-BERT on this task.

5. Discussion:

- It seems that the hierarchical structure does improve the performance. But the positive effect to different models varies. Clinical-BERT are more suitable to this strategy while BERT model does not have better performance by this way.
- For BERT models, using simple CNN layers to deal with the hidden features extracted by separated BERT models is more useful.
- For Clinical-BERT models, using transformer layers is a better choice. (we do not know the reasons, welcome you to find the answer)
- For data Augmentation, BERT model needs this technique. This might because BERT-base-uncased model is not pretrained on MIMIC III dataset , so more slightly different models could exactly improve the performance. As for Clinical-BERT, there is no significate difference between the origin model and the model with augmented data.

6. Conclusion:

This project presented a comprehensive quantitative analysis of medical code assignment from clinical notes using various pretrained models with BERT. To solve the problem of lengthy clinical note encoding, we developed many architectures, which are mainly hierarchical fine-tuning architecture with data augmentation and an additional transformer on top. Through intensive experiments, we found that the magic of BERT does not apply to the task of assigning ICD codes from clinical notes. In contrast, we found that a simple CNN trained from scratch can achieve superior predictive performance on frequent codes, achieving a new state of the art in the MIMIC-III top-50 dataset. This demonstrates how recent training strategies can improve old models. Our results furthermore suggest that medical code assignment algorithms should pay more attention to less frequent codes.

Improving the performance further is difficult , and we have to notice that the computation resource of BERT to do this terrible . Based on a recent review of medical code prediction, here are some advice for future compromising directions:

- Understanding the data better. Clinical notes is usually very long, we have tried to use some developed model to summarize them . But this is not enough. If hoping to enhance the performance, we have to know more about the data. How are the clinical notes made? What parts does clinical notes have? Which part is usually import for the prediction?
- Coding from heterogeneous, incomplete, and noisy sources . In this project , we only use the notetable oof MIMIC III dataset , but the most clinical data in MIMIC III dataset is numerical data , such as results of different experience of patients . Combining those data with discharged summary is promising . But we have to mention that the way of combining data is difficult.

Acknowledgements:

I am very appreciated that prof. Yu gives me a valuable opportunity to work with him this summer, I learned a lot from this project.

Firstly, I recognize the importance of data. In deep learning, every excellent work depends on sufficient good data. At first, we took a long time to find a satisfied dataset MIMIC III to do the medical codes prediction tasks. And then when I train pre-trained models, I find that the more you know your data, the better performance your model could achieve. Data is very important. We need to spend enough time to find data and get familiar with them. Secondly, I know more about how to do the research. At first, I do not know how to read papers and how to read them. With the help of Dr.YU, I find that there are many interesting and essential information I could get from papers and reading them is not difficult. Especially, when I try to understand papers with attempting running the project codes, everything becomes so cool. Thirdly, my coding ability grows much. Now I am familiar with multiple pretrained models on hugging face and use them easily.

In the end, if you are interested in my project, here are the code and trained-model you could use to represent my work:

Code:
https://github.com/ITCUI-XJTLU/Medical_Code_Prediction-

Trained models:
https://huggingface.co/cuitengfei/medical_code_prediction_BERT_clinicalbert

My email:
Tengfei.cui19@student.xjtlu.edu.cn

Reference:

T. Baumel, J. Nassour-Kassis, R. Cohen, M. Elhadad, and N. Elhadad. Multi-label Classification of Patient Notes: Case Study on ICD Code Assignment. In *Workshops at the Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Trans□formers for Language Understanding. In *NAACL-HLT*, 20

D. C. Hsia, W. M. Krushat, A. B. Fagan, J. A. Tebbutt, and R. P. Kusserow. Accuracy of Diagnostic Coding for Medicare Patients under the Prospective-payment System. *New England Journal of Medicine*, 318(6):352–355, 1988.

J. Medori and C. Fairon. Machine Learning and Features Selection for Semi-automatic ICD-9-CM Encoding. In *Proceedings of the NAACL HLT 2010 Second Louhi Workshop on Text and Data Mining of Health Documents*, pages 84–89. Association for Computational Linguistics, 2010.

J. Mullenbach, S. Wiegreffe, J. Duke, J. Sun, and J. Eisenstein. Explainable Prediction of Medical Codes from Clinical Text. In *Proceedings of NAACL-HLT*, pages 1101–1111, 2018.

S. Ji, E. Cambria, and P. Marttinen. Dilated Convolutional Attention Network for Medical Code Assignment from Clinical Text. In *3rd Clinical Natural Language Processing Workshop at EMNLP*, 2020a.

A. Perotte, R. Pivovarov, K. Natarajan, N. Weiskopf, F. Wood, and N. Elhadad. Diagnosis Code As□signment: Models and Evaluation Metrics. *Journal of the American Medical Informatics Association*, 21(2):231–237, 2014.

H. Shi, P. Xie, Z. Hu, M. Zhang, and E. P. Xing. Towards Automated ICD Coding Using Deep Learning. *arXiv preprint arXiv:1711.04075*, 2017.

M. H. Stanfill, M. Williams, S. H. Fenton, R. A. Jenders, and W. R. Hersh. A Systematic Literature Review of Automated Clinical Coding and Classification systems. *Journal of the American Medical Informatics Association*, 17(6):646–651, 2010.

P. Xie and E. Xing. A neural architecture for automated icd coding. In *Proceedings of the 56th Annual*

Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1066–1076, 2018.

D. Zhang, J. Thadajarassiri, C. Sen, and E. Rundensteiner. Time-Aware Transformer-based Network for Clinical Notes Series Prediction. In *Machine Learning for Healthcare Conference*, pages 566–588. PMLR, 2020.