Prediction of Medical Codes from Clinical Text using Transfer Learning

Yeow Long, Chua
University of Illinois at Urbana-Champaign
ylchua2@illinois.edu

Abstract

Clinical notes are text documents created by clinicians to document/record each patient encounter. It contains medical codes which effectively describes the diagnosis and treatment. Manually annotating these medical codes can be very labor intensive and prone to human error and when medical codes are omitted, it becomes harder for others to understand the specific rationale behind certain specific diagnosis and treatment. As such, NLP (Natural Language Processing) techniques have been used to automate this labelling process. In this work, we introduce the use of pre-trained language representation models such as BERT (Bidirectional Encoder Representations from Transformers) which is simple but powerful and used to obtain state-of-the-art results for various NLP tasks without modifying the BERT architecture. Our method utilises the pre-trained BERT model and fine-tunes it with the additional attentional convolutional network layer to create state-of-the-art(I hope) models to predict medical codes from clinical text. The model is accurate and achieves a precision of xx(yet to obtain results) and a Micro-F1 of xx which performs better than the prior state of the art.

1.Introduction

Clinical notes are text documents created by clinicians during patient encounters and are typically recorded together with a set of metadata codes from the ICD (International Classification of Diseases), a standardised way of recording diagnosis and procedures performed during each patient encounter. These ICD codes have vast number of use cases, from prediction of patient state to billing (Avati et al., 2017, Choi et al., 2016; Ranganath et al., 2015; Denny et al., 2010). As manual coding is very labor intensive and prone to error, there were attempts to study automatic coding with limited success (de Lima et al., 1998). This task is very challenging mainly due to two reasons. First, there are over 15,000 codes in the ICD-9 taxonomy and over 140,000 codes in the newer ICD-10 taxonomies (World Health Organization, 2016). Second, clinical notes also contain alot of information and these includes irrelevant information, spelling errors and a large medical vocabulary. These reasons combined make the prediction of ICD codes from clinical notes a challenging tasks for computers and humans alike(Birman-Deych et al., 2005).

In this paper, we introduce the use of pre-trained language representation models such as BERT which have shown to be effective for improving many NLP tasks (Dai and Le, 2015; Peters et al., 2018a; Radford et al., 2018; Howard and Ruder, 2018). There are two steps in the BERT framework: pre-training and fine-tuning. First, during the pre-training process, the model is pre-trained on unlabelled data over various NLP tasks. Second, for fine-tuning, the BERT model is first initialised using the pre-trained parameters and all these parameters are fine-tuned using the preprocessed MIMIC dataset for the specific task of medical codes prediction. Our model design is motivated by the success of transfer learning and the use of pre-trained models to achieve superior performance in various NLP tasks. We evaluate our approach on MIMIC-III dataset (Johnson et al., 2016), an open dataset of ICU medical records. We utilise the records which consists of clinical notes and codes which describes the patients diagnosis and procedures. Our approach (I hope) outperforms previous results on medical code prediction using the MIMIC-III dataset. As we consider the use of this work in a decision support environment, the system needs to be able to explain why it predicted each code and these considerations motivate the use of per-label attention mechanism (James Mullenbach, 2018) which provides justifications and texts for each prediction in the form of text snippets from the clinical notes.

2.Approach

We treat the ICD code prediction problem as a multi-label text classification problem (McCallum, 1999). Let L denote the set of ICD-9 codes; the labelling process for each instance i is to determine the $y_{i,l} \in \{0, 1\}$ for all $l \in L$. We will utilise a pre-trained BERT model and add a convolutional layer at the end to obtain a base representation of the text (Kim, 2014) and make $|L|$ binary predictions. We also make use of an attention mechanism to select parts of the documents that are most relevant for each possible code. These attention weights are then applied to the base representation and a sigmoid transformation is applied to obtain the likelihood for each code. We also employ a regulariser to ensure that similar codes have similar parameters and textual descriptions.

2.1.Pre-trained BERT architecture

The BERT model architecture is a multi-layer bidirectional Transformer encoder based on the implementation of (Vaswani et al., 2017) and since our implementation is similar to the original, we will omit an comprehensive description of the model architecture and refer readers to (Vaswani et al., 2017) and "The Annotated Transformer".

2.2.Fine-tuning BERT: Convolutional Attention Architecture

On top of the BERT model, we have pre-trained embeddings for each entry of the clinical notes which are horizontally concatenated into a matrix of the form $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N]$, where N is the length of the document. We make use of a convolutional

filter to combine the word embeddings and at each step, we compute

$$H_n = g(W_c * x_{n:n+k-1} + b_c)$$

where * denotes the convolution operator, g denotes the element-wise non-linear transformation and b_c denotes the bias. We pad the input in order to achieve a resulting matrix of size d_c by N.

After convolution, we reduce this matrix vector by applying pooling across the length of the document and select the maximum or average value at each row (Kim, 2014). We assign multiple labels for each entry of the clinical notes as it is very possible that a single clinical note entry can lead to multiple code predictions. As a result, we use a per-label attention mechanism, selecting the k-grams that is most relevant for each predicted label.

For each label prediction l, we compute the product $H^T u_l$ and pass the result through a softmax operator obtaining the distribution over text location in the document

$$\alpha_l = SoftMax(H^T u_l),$$

The attention vector is then used to compute the vector representation for each label

$$v_l = \sum_{n=1}^{N} a_{l,n}, h_n$$

We use max-pooling to compute a single vector v for all labels to serve as a baseline.

$$v_j = \max_n h_{n,j}$$

Given the vector representation, we compute the probability for all labels using a linear layer and sigmoid transformation.

$$\hat{y}_l = \sigma(\beta_l^T v_l + b_l)$$

where beta contains the prediction weights and b is the bias.

For the training procedure, we minimise the binary cross-entropy loss,

$$L_{BCE}(X, y) = -\sum_{l=1}^{L} y_l log(\hat{y}_l) + (1 - y_l) log(1 - \hat{y}_l)$$

plus the L2 norm of the model weights (Kingma and Ba, 2015).

2.3.Evaluation Metrics

In order to facilitate the comparison between different model, we will report various different metrics, with a focus on macro-averaged and micro-averaged F1 as well as area under the curve. Macro-averaged F1 are computed by averaging the F1-score computed per-label whereas micro-averaged F1 are computed by treating each text and code pair as a separate prediction. Other than F1-score, Micro-R scores will be reported as well.

$$Micro - R = \frac{\sum_{l=1}^{|L|} TP_l}{\sum_{l=1}^{|L|} TP_l + FN_l}$$

$$Macro - R = \frac{1}{|L|} \sum_{l=1}^{|L|} \frac{TP_l}{TP_l + FN_l}$$

Where TP denotes the true positive and FN denotes false negatives. Precision will also be computed. We also report precision at n denoted as 'P@n' which is the combined precision of the n highest scored labels and this is motivated by the use case as a decision support application where a user is presented with a fixed number of predicted codes to review. We will present precision with values 5 and 8 to compare with prior work (Vani et al., 2017; Prakash et al., 2017). We will also compute precision@15 which roughly corresponds to the average number of medical codes for each of the MIMIC-III discharge summaries.

3.Experimental Results

3.1.Text Pre-processing

In this work, we deal with a multi-label classification problem and before we build and train our model, we need to pre-process the data. We use the MIMIC-III dataset and outlined the following pre-processing steps similar to (James Mullenbach, 2018). Firstly, extract the discharge summaries from noteevents csv. Secondly, extract the ICD-9 codes from diagnoses_icd and procedures_icd. Thirdly, drop rows that contain empty values. Fourth, convert the codes into ICD-9 format and combined the diagnoses and procedures codes into one table. The codes should have a certain format with dots. Fifth, filter the discharge summaries by removing special characters and remove numeric only words. Sixth, tokenise the discharge summaries and transform them into an array of words. Seventh, filter out rows where there are no discharge summaries. Eighth, truncate the number of tokens per document to a maximum of 2500. Ninth, perform a train-test-split on the dataset based on the patient id. Finally, create a vocabulary and use word2vec to train the word embeddings.

3.2.Feature Extraction

We first need to create features from our preprocessed data from the earlier section, we convert the tokens for every clinical note summaries into 2-dimensional array of vectors using (Tomas Mikolov, 2013). In this embedding method, for every word we have a 1-dimensional representation of fixed depth with a specified vector size of 100. We will create a dictionary for each of the word in the clinical summaries and the labels. We also zero-pad the vector so as to ensure that the clinical summaries are of a fixed length. The result is a matrix with each row the different clinical summaries embeddings of fixed depth 100.

3.3.Comments

I'm still working on getting the baseline functional to ensure that the pre-processing steps are indeed valid. Here, I will focus on the results I hope I get by the end of this project of different architectures I intend to implement and test such as RNN, CNN (this two will serve as the baseline) and BERT fine-tuned with CAML(James Mullenbach, 2018). If there is time, I would want to validate the assumption of "important information is hidden in small snippets of text and for every label they are different" also from James Mullenbach.

4. Discussion

During the initial phase of the project, when I was just starting out, I totally missed the part on converting the diagnoses and procedure codes into the full format with dots. As a result, the inputs to the model was pretty erroneous and fortunately, I was able to discover it within a short amount of time after reading other papers and working on the pre-processing steps and a baseline model to benchmark results.

The results were pretty bad during this draft phase with the baseline model. First, the dimensionality of the labels were too high and very few labels dominated and the dataset is pretty

imbalanced. Following the approach of (James Mullenbach, 2018) which proposed the use of code descriptions from (World Health Organization, 2016) and embed these descriptions as vectors to perform regularisation on the model parameters so that this regulariser can assign similar model parameters to codes with similar descriptions.
After tinkering and working on the project for a fair bit especially on the pre-processing pipeline, I realised more needs to be done to reduce the dimensionality of the problem and I have decided to adopt (James Mullenbach, 2018) approach where only samples from the top 50 most used labels are used. This approach has proven to be very good at dealing with dimensionality of the problem.

5. Conclusion

In this work, we built a deep learning model to perform ICD-9 code predictions from clinical notes. We will evaluate the performance of the BERT-CAML model using the top 50 codes (to reduce the dimensionality and class imbalance of the dataset). If time permits, we'll try to validate the assumption of (James Mullenbach, 2018) that "important information is hidden in small snippets of text and for every label they are different". Given the fact that there are 53,000 different clinical notes and 8,000 codes, it is a challenging problem to work with. We'll first try working on just the top 50 most used labels for model training and validation using a simple RNN or CNN before moving on to BERT-CAML which serves as the baseline and also to ensure that the text pre-processing pipelines are indeed without errors before moving on to utilizing code descriptions to assign regularization weights so that rarely observed codes will be assigned similar parameters to similar codes and descriptions. We believe the use of pre-trained models such as BERT and fine-tuning should achieve a better model performance than the state-of-the-art and we also hope to show that the attention mechanism of CAML can identify meaningful explanations for each code prediction.

6. References

1. Anand Avati, Kenneth Jung, Stephanie Harman, Lance Downing, Andrew Ng, and Nigam H. Shah. 2017. Improving palliative care with deep learning. *arXiv preprint arXiv:1711.06402*.
2. Edward Choi, Mohammad Taha Bahadori, Andy Schuetz, Walter F Stewart, and Jimeng Sun. 2016. Doctor AI: Predicting clinical events via recurrent neural networks. In *Machine Learning for Healthcare Conference*. pages 301–318.
3. Rajesh Ranganath, Adler J. Perotte, Noémie Elhadad, and David M. Blei. 2015. The survival filter: Joint survival analysis with a latent time series. In *UAI*. pages 742–751.
4. Joshua C. Denny, Marylyn D. Ritchie, Melissa A. Bas- ford, Jill M. Pulley, Lisa Bastarache, Kristin Brown- Gentry, Deede Wang, Dan R. Masys, Dan M. Ro- den, and Dana C. Crawford. 2010. PheWAS: demon- strating the feasibility of a phenome-wide scan to discover gene–disease associations. *Bioinformatics* 26(9):1205–1210.
5. Luciano R.S. de Lima, Alberto H.F. Laender, and Berthier A. Ribeiro-Neto. 1998. A hierarchical ap- proach to the automatic categorization of medical documents. In *Proceedings of the seventh international conference on Information and knowledge management*. ACM, pages 132–139.
6. World Health Organization. 2016. International statistical classification of diseases and related health problems 10th revision. http://apps.who.int/classifications/icd10/browse/2016/en
7. Elena Birman-Deych, Amy D. Waterman, Yan Yan, David S. Nilasena, Martha J. Radford, and Brian F Gage. 2005. Accuracy of ICD-9-CM codes for identifying cardiovascular and stroke risk factors. *Medical care* 43(5):480–485.
8. Andrew M Dai and Quoc V Le. 2015. Semi-supervised sequence learning. In *Advances in neural information processing systems*, pages 3079–3087.
9. Peter F Brown, Peter V Desouza, Robert L Mercer, Vincent J Della Pietra, and Jenifer C Lai. 1992. Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–479.
10. Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding with unsupervised learning. Technical re- port, OpenAI
11. Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *ACL*. Association for Computational Linguistics.
12. Alistair E.W. Johnson, Tom J. Pollard, Lu Shen, Li- wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. 2016. MIMIC-III, a freely accessible critical care database. *Scientific data* 3.
13. J. Mullenbach, S. Wiegreffe, J. Duke, J. Sun, and J. Eisenstein. Explainable prediction of medical codes from clinical text. ACL Anthology, 2018.
14. Andrew McCallum. 1999. Multi-label text classification with a mixture model trained by EM. In *AAAI workshop on Text Learning*. pages 1–7.
15. Yoon Kim. 2014. Convolutional neural networks for sentence classification. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. pages 1746–1751.
16. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 6000–6010.
17. Diederik Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*.
18. Ankit Vani, Yacine Jernite, and David Sontag. 2017. Grounded recurrent neural networks. *arXiv preprint arXiv:1705.08557* .
19. Aaditya Prakash, Siyuan Zhao, Sadid A Hasan, Vivek V Datla, Kathy Lee, Ashequl Qadir, Joey Liu, and Oladimeji Farri. 2017. Condensed memory networks for clinical diagnostic inferencing. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*. pages 3274–3280.
20. Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. pages 3111–3119.