Transformers in Semantic Indexing of Clinical Codes

Rishivardhan K, Kayalvizhi S, Thenmozhi D, Sachin Krishan T, Aravindan Chandrabose

SSN College Of Engineering, Chennai.
rishivardhan18126@cse.ssn.edu.in, {kayalvizhis,theni_d}@ssn.edu.in,
sachinkrishnan18128@cse.ssn.edu.in, aravindanc@ssn.edu.in

Abstract. International Classification of Diseases (ICD) codes is used to represent the diseases and health statuses which is useful for better communication and to facilitate research. Automatic assignment of these codes to the clinical records is essential nowadays whereas multilingual extraction task of eHealth@CLEF-2020 concentrates on predicting ICD10-CM and ICD10-PCS codes in Spanish language. Automatic prediction of these codes have been approached by using transformers such as BERT, RoBERTa, Electra and XLNet. Among the models, Electra predicts ICD10-CM codes better and RoBERTa predicts ICD10-PCS codes better than the other transformer models.

Keywords: BERT · Electra · RoBERTa · Transformers · XLNet

1 Introduction

The International Classification of Diseases (ICD) is a healthcare classification system maintained by the World Health Organization. Diseases and health statuses are classified according to certain rules and uniquely identified by character codes. The efficiency of ICD coding has begun to receive more attention because of how crucial it is for making clinical and financial decisions. Hospitals with better coding quality see many benefits, including more accurate classification and retrieval of medical records, and better communication with other hospitals to jointly promote healthcare quality and facilitate research.

Human classification of diagnoses is a labor intensive process that consumes significant resources. Most medical practitioners use specially trained medical coders to categorize diagnoses for billing and research purposes. The coding process requires a comprehensive consideration of each patient's health condition. However, very few medical practitioners are capable of taking over the process since they lack training in professional coding.

2 Task description

By the Multilingual Extraction Task [7] from eHealth Lab [3], the goal is to assign ICD10 codes to clinical records. The task comprises of three subtasks wherein we have participated in two of them, namely, ICD10-CM (Clinical Modification) and ICD10-PCS (Procedure Classification System) codes assignment. The ICD10-CM sub task deals with automation of code assignment for classifying and reporting diseases in all healthcare settings while the ICD10-PCS sub task deals with code assignment for hospital reporting of inpatient services. The task is pursued as a multi-label text classification from our viewpoint while Named Entity Recognition and Normalization paths are also plausible.

3 Dataset Description

The CodiEsp corpus of 1,000 clinical case studies, selected manually by a practicing physician was used for training the models. The CodiEsp corpus is distributed in plain text in UTF8 encoding, where each clinical case is stored as a single file whose name is the clinical case identifier. Each clinical case or its identifier is associated with one or multiple medical codes depending on the sub task's golden truth file.

The corpus consists of two variants of the clinical files. The first one being the original set filled with Spanish text and the other one being the English machine-translated version of the former. For our system we had used only the English version of the clinical case files to train the models.

The 1000 clinical case files were distributed as to constitute 500 train files, 250 development files and 250 test files. Additionally 2000 clinical files were released along with test set release to promote that these systems would potentially be able to scale to larger data collections.

4 Methodology

The task was approached as a multi-label text classification task with the use of transformers. The data is preprocessed and then data is made suitable for training by creating a data frame for all the instances of the data. Model training is done by fine tuning the parameters of four transformers that include BERT, RoBERTa, XLNet and Electra. After training, a file with the probability scores of all possible codes is generated which is post-processed to filter the suitable codes for the instances.

4.1 Data preparation

Initially, the data is preprocessed and then data is made suitable for training. Preprocessing steps include removal of punctuation and stop words followed by tokenization and lemmatization. Then, the text files are prepared by mapping the textual data codes to a data frame that has ones and zeros.

For example : If there are 5 unique codes in a scenario whereas text data has code1, code3 as its code,
Then the text file is trained with a data frame that has rows with ones only at the columns with that codes with remaining zeros.
Data frame for that text - 1 0 1 0 0

Here, there are 2194 unique labels in which each text file will have labels in the range 1 to 56 for each text file.Thus, the data frames are created for each text file and then given to the model for training.

Sample Input data:

	text	**labels**
0	describe case 37yearold man previous active.—	[0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, ...]
1	present head neck auscultation cardiac	[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, ...]

4.2 Training the models

The transformer models namely BERT, RoBERTa, Electra and XLNet were used whose implementation is explained below.

BERT

BERT [2] makes use of transformers, an attention mechanism that learns contextual relations between words in a text with two strategies namely masked LM and next sentence prediction. BERT was fine tuned over "Bio-Clinical-BERT" [5], a model trained on all notes from MIMIC III, a database containing electronic health records from ICU patients at the Beth Israel Hospital in Boston for predicting the clinical codes.

In our implementation, the Bio-Clinical-BERT model was made use with ***.

RoBERTa

RoBERTa [6] makes use of a robustly optimized method that improves on BERT by modifying key hyperparameters in BERT.

For our implementation, Roberta was fine tuned over "BioMed-RoBERTa-base" [4], a language model based on the RoBERTa-base architecture is trained over 100 epochs with train batch size of 20 and learning rate of 4e-5.

Electra

In general, electra [1], a modification of BERT corrupts the input by replacing some tokens and trains a discriminative model that predicts whether each token in the corrupted input was replaced by a generator sample or not so that the contextual representations are learned in a better way.

For our work, electra was fined tuned over "Electra-base" model of 12 Transformer blocks, 768 hidden dimensions and 110 M parameters, trained over 100 epochs with train batch size of 20 and learning rate of 4e-5 to predict the probability of codes.

XLNet

XLNet[8] is a generalized AR pretraining method that uses a permutation language modeling objective to combine the advantages of AR (autoregression) and AE (autoencoding) methods that integrate Transformer-XL and the careful design of the two-stream attention mechanism.

For predicting the codes, XLNet was fine tuned over "xlnet-base-cased" model of parameters, 12 Transformer blocks, 12 self-attention heads and 768 hidden dimensions which is trained over 100 epochs with train batch size of 20 and learning rate of 4e-5.

4.3 Post-Processing

After training, a prediction file with a probability score of all 2194 codes is generated which has to be post processed. Post processing includes filtering the suitable codes. Here, we filtered with a threshold value of '0.002'. which seems to be effective. The probability of the codes greater than 0.002 are filtered and considered as codes.

5 Results

File	**Model**	**MAP**	**Precision**	**Recall**	**F1**
Run-1	BERT	0.001	0.009	0.014	0.011
Run-2	**Electra**	**0.007**	**0.025**	**0.049**	**0.033**
Run-3	RoBERTa	0.004	0.014	0.019	0.016
Run-4	XLNet	0	0	0.001	0.001

Table 1. subtask 1 results

Table 1 and Table 2 shows the results of our submission in subtask1 and subtask2 respectively. From Table 1, Electra model has attained a better Precision score 0f 0.007, MAP score of 0.025, F1 score of 0.033 and recall of 0.049 than the other models for subtask 1 whereas Roberta attains the better scores in subtask2. The Roberta model of subtask 2 has the best recall 0.075 among all our submissions.

File	**Model**	**MAP**	**Precision**	**Recall**	**F1**
Run-1	Electra	**0**	0.001	0.002	0.001
Run-2	**RoBERTa**	**0.028**	**0.016**	**0.075**	**0.027**
Run-3	XLNet	0.002	0.002	0.007	0.003

Table 2. subtask 2 results

6 Conclusion

Semantic indexing of clinical codes is very desirable since term mapping of clinical codes are crucial now. We approached the semantic indexing by utilizing the transformers to predict the diagonstic (ICD10-CM) and procedural (ICD10-PCS) codes for the data. BERT, RoBERTa, Electra and XLNet are the transformers that have been fine tuned to predict the codes. All the four transformer models have been used in ICD10-CM codes which Electra performs better and RoBERTa performs better in predicting ICD10-PCS codes among RoBERTa, Electra and XLNet.

Acknowledgement

We would like to thank DST-SERB and HPC laboratory for providing resources needed for this work.

References

1. Clark, K., Luong, M.T., Le, Q.V., Manning, C.D.: ELECTRA: Pre-training text encoders as discriminators rather than generators. In: ICLR (2020), https://openreview.net/pdf?id=r1xMH1BtvB
2. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding (2018)
3. Goeuriot, L., Suominen, H., Kelly, L., Miranda-Escalada, A., Krallinger, M., Liu, Z., Pasi, G., Saez Gonzales, G., Viviani, M., Xu, C.: "overview of the CLEF eHealth evaluation lab 2020. In: Arampatzis, A., Kanoulas, E., Tsikrika, T., Vrochidis, S., Joho, H., Lioma, C., Eickhoff, C., Névéol, A., andNicola Ferro, L.C. (eds.) Experimental IR Meets Multilinguality, Multimodality, and Interaction: Proceedings of the Eleventh International Conference of the CLEF Association (CLEF 2020). LNCS Volume number: 12260 (2020)

4. Gururangan, S., Marasović, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D., Smith, N.A.: Don't stop pretraining: Adapt language models to domains and tasks. In: Proceedings of ACL (2020)
5. Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C.H., Kang, J.: Biobert: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics (Sep 2019). https://doi.org/10.1093/bioinformatics/btz682, http://dx.doi.org/10.1093/bioinformatics/btz682
6. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach (2019)
7. Miranda-Escalada, A., Gonzalez-Agirre, A., Armengol-Estapé, J., Krallinger, M.: Overview of automatic clinical coding: annotations, guidelines, and solutions for non-english clinical cases at codiesp track of CLEF eHealth 2020. In: Working Notes of Conference and Labs of the Evaluation (CLEF) Forum. CEUR Workshop Proceedings (2020)
8. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., Le, Q.V.: Xlnet: Generalized autoregressive pretraining for language understanding (2019)