A Residual CNN Model for ICD Assignment

Darryl Lin-Wei Cheng[1], Choo-Yee Ting[1(✉)], and Chiung Ching Ho[2]

[1] Faculty of Computing & Informatics, Multimedia University, 63100 Cyberjaya, Selangor, Malaysia
cyting@mmu.edu.my

[2] Department of Computing and Information Systems, Sunway University, 47500 Petaling Jaya, Selangor, Malaysia

Abstract. International Classification of Diseases (ICD) has been used as a standardized way of classifying a diagnosis or a medical procedure. ICD has also been employed to keep track of illness progression and treatment purposes. However, the assignment methods often require manual input of medical professionals and therefore time consuming and prone to human errors. By automating the assignment of ICD-9 codes to clinical notes we can effectively save time and human resources. In this light, this study proposed a residual convolution neural network leveraging label co-occurrence to measure label correlation and a label attention mechanism to capture label-dependent features. The model was fine-tuned by changing its hyper-parameters which have included dropout probabilities, CNN kernel size and its output size. The empirical findings suggested that the model has outperformed conventional approaches with 93.6% for Micro-AUC, 91.8% for Macro-AUC, 70.0% Micro-F1, and 64.6% for Macro-F1.

Keywords: ICD codes · Residual CNN · MIMIC-III

1 Introduction

International Classification of Diseases (ICD) is a globally used standard maintained by the World Health Organization. It is a standard that assigns unique codes to represent every medical diagnosis and procedure. ICD are used for administrative work such as billing, reimbursements, and standardizing patient medical records. ICD coding is the task of extracting relevant ICD codes from clinical text. The process ICD coding requires manual input from medical professionals which is time-consuming and prone to human errors. Inaccurate ICD coding can lead to inaccurate billing which causes financial losses to both patients and medical providers [1]. Therefore, a model that can assist medical professionals in ICD coding is needed. With machine learning and deep learning, automated ICD coding models that takes clinical text as input and automatically extract the relevant ICD code became possible, and thanks to publicly available Electronic Health Records (EHR) data which enabled broader research in the field of ICD coding.

The challenges faced in ICD coding, in the context of natural language processing is the high dimensionality of ICD codes. There are over 13,000 ICD codes which pose a significant challenge. Furthermore, the imbalance of the distribution of ICD codes adds to

S.-C. Haw and K. Sonai Muthu (Eds.): CITIC 2022, 10, pp. 331–341, 2022.
https://doi.org/10.2991/978-94-6463-094-7_26

the difficulty as there is a portion of ICD codes that are more frequently recorded whereas some are less common. Lastly, dealing with the processing of unstructured clinical notes which are typically noisy, filled with misspellings, uncommon abbreviations, and vocabulary filled with medical terms.

Medical Information Mart for Intensive Care (MIMIC), developed by the Massachusetts Institute of Technology (MIT) is an electronic health records database which is openly available to all [2, 3]. It contains de-identified data of over forty thousand patients with over 58,000 hospital admissions. There are multiple versions of MIMIC available, we will be using MIMIC-III v1.4. The data of interest is the discharge summaries which contain notes on all the information about a patient's stay.

In this study, we automatically assign ICD codes on MIMIC-III clinical text using deep learning models. Our models will be based on a convolution neural network which has been a reliable method for text classification. We propose a residual convolution network with label attention to allow the network to capture the interdependence of ICD code in relation to the clinical notes of MIMIC-III. We also employ label co-occurrence to capture the inter-relationship between pairs of ICD codes.

2 Related Works

Automated ICD coding is a topic of research dating back to the 1900s by a paper proposed by Larkey and Croft [4]. The authors proposed an ensemble of classifiers consisting of K-nearest neighbour, relevance feedback and Bayesian independence to classify ICD codes on patient discharge summaries. de Lima et al. [5] used cosine similarity between ICD code and discharge summary to model their classifier.

In recent years, researchers have explored deep learning models for the task of automated ICD coding. Mullenbach et al. [6] proposed convolution attention for multi-label classification (CAML) which utilizes convolution neural networks with label attention mechanism for the task of ICD coding while simultaneously modelling baseline methods using Logistic Regression and Bi-GRU on the MIMIC-III discharge summary dataset. Mayya et al. [7] proposed an enhanced version of CAML which extends it by using multi-channel CNN. Briefly, it leverages multiple CNN with differing filter sizes instead of a single CNN to capture more information.

There are also many other variations of neural networks such as long-short term memory (LSTM) [8–10], and gated recurrent units [11] for automated ICD coding. Hierarchical methods leveraging the hierarchical structure of ICD codes are also present [12]. Transformer-based methods such as bi-directional encoder representation from Transformer (BERT) which carried state-of-the-art results in the area of natural language processing have also been proposed in the field of automated ICD coding [13, 14].

3 Methodology

3.1 Model Architecture

In this section, we describe the approach of initializing our neural network for multi-label text classification. Our entire neural network is based on a convolution neural network,

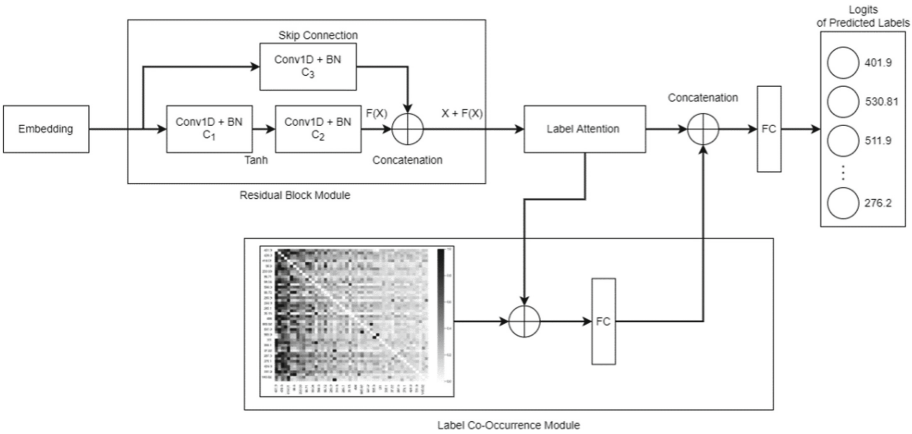

Fig. 1. Network architecture

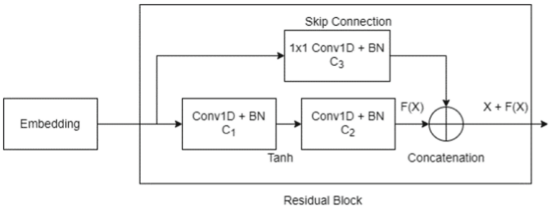

Fig. 2. Architecture of a Residual Block, BN stands for batch normalization

which is widely used in the task of text classification. The network architecture is shown in Fig. 1. The implementation is done using PyTorch. Briefly, the network consists of a convolutional residual block, connected with label attention and label co-occurrence module to produce the logits for each predicted ICD.

3.1.1 Embedding

The embedding layer is initialized with the pre-trained embedding of BioWordVec [15]. Given a clinical note as input, the document is parsed into the embedding layer to generate the word embeddings matrix $\mathbf{E} = [w_1, w_2, \ldots, w_n]$ where n is the length of the document and each word vector $\mathbf{w_i} \in \mathbb{R}^{d_e}$ is represented by a d_e—dimensional word embedding vector.

3.1.2 Residual Convolution Layer

Briefly, the word embedding is fed into a residual block consisting of three 1-dimensional convolution layers as shown in Fig. 2. The convolution operation works to concatenate the vector representation of each word by applying a filter to a window of words to capture local features at different positions. We pass a sliding window over the text represented by the word embeddings $\mathbf{E} \in \mathbb{R}^{n \times d_e}$. For each k-words ngram, we have a

window vector $\mathbf{u}_i = [w_i, \ldots, w_{i+k-1}] \in \mathbb{R}^{d_e \times k}$; $0 \leq i \leq n-k$, where k is the number of filters, and a filter $\mathbf{m} \in \mathbb{R}^{d_e \times k}$. Filter $\mathbf{m}$ convolves with the window vector $\mathbf{u}_i$ to generate a feature map $\boldsymbol{c_i} = f(u_i * m + b)$ where $*$ denotes a convolution operation, f is an element-wise nonlinear transformation and $b \in \mathbb{R}^{d_c}$ is the bias term. The convolution results in a matrix $\mathbf{c}_i \in \mathbb{R}^{nxp}$ where p is the size of the filter output.

The residual connection allows training much deeper networks without suffering performance degradation issues such as vanishing gradient. Residual blocks are simply an identity function which maps a hidden state forward in the network. The skip connection brings forward information learnt in the previous layers thus allowing the network to retain information learned in previous layers activation early in the network. Overall, the residual block as shown in Fig. 2 composed of two convolution layers with one convolution layer at the skip connection and can be formulated as follows:

$$\begin{aligned} \mathrm{x}_1 &= \mathrm{c}_1(x) = tanh(f(x_{i:i+k-1} * m_1)) \\ x_2 &= c_2(x_1) = f(x_{1_{i:i+k-1}} * m_2) \\ \mathrm{x}_3 &= \mathrm{c}_3(x) = f(x_{i:i} * m_3) \\ R &= tanh(x_2 + x_3) \end{aligned} \tag{1}$$

where $\mathbf{x}$ is the input matrix, $\mathbf{R}$ is the output of the residual block, and $\mathbf{m}$ is the filter. $x_{i:i+k-1}$ represents the window vector which convolves with filter $\mathbf{m}$ to compute the results of the convolution.

The input matrix $\mathbf{x}$ is the word embeddings $\mathbf{E} \in \mathbb{R}^{n \times d_e}$, where n is the length of the document, and each word is represented by a d_e-dimensional vector. The first convolution layer $\mathbf{c}_1$ will produce a matrix $\mathbf{x}_1 \in \mathbb{R}^{nxp}$, with p as the filter output size and k as the number of filter as shown in Eq. 1. The matrix $\mathbf{x}_1$ is then passed into the next convolution layer $\boldsymbol{c}_2$ as an input where the number of filter k and the output size p is set to be the same as $\mathbf{c}_1$. The resulting matrix is $\mathbf{x}_2 \in \mathbb{R}^{nxp}$. $\mathbf{c}_3$ is a convolution layer at the skip connection where kernel size is set to 1 to transform the shape of input $\mathbf{x}$ to be the same as the output of $\mathbf{c}_2$ for the addition operation. In the end, the addition operation between $\mathbf{x}_2$ and $\mathbf{x}_3$ resulted in a matrix of $\mathbf{R} \in \mathbb{R}^{nxp}$.

3.1.3 Attention Layer

The output of the residual block $\mathbf{R} \in \mathbb{R}^{nxp}$ is then fed into a per-label attention layer which computes the attention weights on a per label basis. It allows the ICD codes to attend to different parts of the document to produce its own dense representations. Simply, we compute a vector $\mathbf{Q}_l \in \mathbb{R}^p$ for each label l where the resulting matrix will be $\mathbf{Q} \in \mathbb{R}^{p \times l}$ and compute a matrix-vector product of $\mathbf{Q}$ and $\mathbf{R}$. The attention layer can be formulated as:

$$\begin{aligned} a &= softmax\left(R^T Q\right), \\ V &= aR \end{aligned} \tag{2}$$

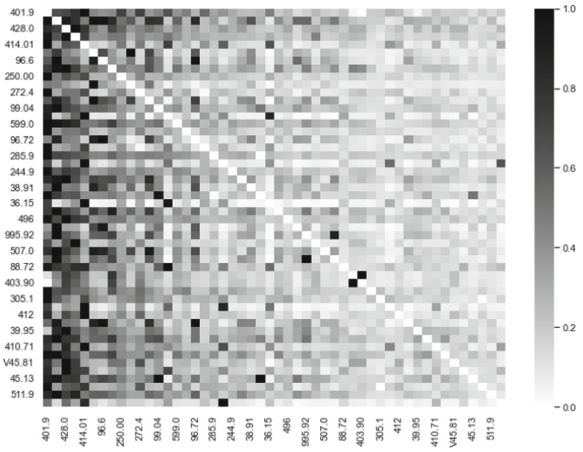

Fig. 3. Label Co-occurrence Matrix of MIMIC-III Top 50 ICD

where $\mathbf{V} \in \mathbb{R}^{p \times l}$ represent the output of the attention layer and $\mathbf{a} \in \mathbb{R}^{nxl}$ represent the attention weights for each ICD code.

3.1.4 Label Co-occurrence

Lastly, to exploit the relationship and correlation between ICD labels, we employed label co-occurrence matrix to model label correlation. Figure 3 shows the label co-occurrence matrix where darker colour indicates the higher frequency that both labels appear together.

Let $\mathbf{B} \in \mathbb{R}^{l \times l}$ where l is the number of labels. Each entry in the matrix $\mathbf{B}(i, j)$ is computed by taking the number of observations where label i and label j both appear together in the same document. The co-occurrence matrix is also normalized in a range of [0,1] and the matrix can be formulated as:

$$B(i,j) = \begin{cases} 0, i = j \\ \frac{B(i,j) - \min(B(\cdot,j))}{\max(B(\cdot,j)) - \min(B(\cdot,j))}, i \neq j \end{cases} \tag{3}$$

The label co-occurrence matrix $\mathbf{B} \in \mathbb{R}^{l \times l}$ will be concatenated with the output of the attention layer $\mathbf{V} \in \mathbb{R}^{p \times l}$ to produce $\mathbf{F} \in \mathbb{R}^{(p+l) \times l}$. Finally, the matrix $\mathbf{F}$ will be fed into the classification layer to compute the logits for each label l.

4 Empirical Study

We define our problem as a multi-label text classification problem. Each discharge summary for each patient can be associated with more than one ICD code. Given discharge summary as our input, our objective is to determine the correct set of ICD codes associated with each discharge summary.

Based on the literature, there are two settings the dataset is experimented on, either using the full labels of MIMIC-III or only the top 50 most frequent ICD codes. The full

Table 1. Descriptive statistics of MIMIC-III 50 training data

Styles	MIMIC-III 50
Training documents	8,067
Mean ICD per document	5.7
Mean tokens per document	934
Vocabulary size	54578

label of MIMIC-III contains over 8,929 unique ICD codes. For this experiment, we will work on MIMIC-III with the top 50 most frequent labels.

4.1 Dataset

From MIMIC-III database, we have extracted discharge summaries from NOTEEVENTS.csv which contain all the clinical notes, and diagnosis ICD code and procedures ICD code tables are extracted from their respective files. Discharge summaries along with their respective diagnosis and procedure ICD code are grouped together based on columns like HADM_ID and SUBJECT_ID that acted as key for the grouping operation. Alongside the discharge summaries, there are also addendums which are additional text added in for the report, we merged the addendums to their respective discharge summary report based on HADM_ID. The final table consists of 52,722 rows of discharge summary with a total of 8,929 ICD codes. Next, we pre-processed each discharge summary before it is tokenized for model training. Stop words, punctuations, numeric characters, and all non-alphanumeric characters are removed (with the exception of terms like "25mg"). To further prune the total amount of words, we have also removed uncommon words that appear in less than three documents. All our documents are also truncated to a max length of 2,500 words. Every word is lowercased and tokenized. Lastly, out of the 8,929 ICD codes, we only extracted the top 50 most frequently appearing ICD codes. Table 1 is the descriptive statistic of the MIMIC-III 50 training dataset after a train-test-split containing 8,067 discharge summaries for training and 1,574 and 1,730 documents for validation and testing, respectively.

4.2 Pre-trained Word Embedding

Besides that, we have opted to use a pre-trained word embedding called BioWordVec [15], a word embedding that are pre-trained on clinical documents and clinical notes from both PubMed and MIMIC-III. BioWordVec utilizes skip-gram approach that learns character n-grams where each word is represented by the sum of its character n-grams. Each word is represented with a *200*-dimensional word embedding. Using a pre-trained word embedding that are domain specific to our work would mean that medical vocabularies are more likely to be represented.

4.3 Hyperparameters

For the hyperparameters, our CNN have adopted a kernel size of 5 and a feature map of 350 after a series of hyperparameter tuning. The word embedding dimension is 200 as per BioWordVec [15]. Dropout rate used was 0.5. AdamW optimizer with a learning rate of 0.001 and weight decay of 0.01. The batch size was 32.

4.4 Evaluation Metrics

The performance of our model will be compared against a few of the works mentioned in the related works. Two standard metrics will be used to assess the performance of our models, micro-F1 and macro-F1. Macro-F1 is calculated by the average precision and recall of individual classes while micro-F1 is the harmonic mean of the global precision and recall scores with total true positives, false negatives, and false positives. Macro-averaged values put more emphasis on infrequent labels as every metric is calculated per label before averaging. Our training objective was to maximize the macro-F1 score, as we believe in the setting of an imbalanced multi-label classification that macro-F1 makes sense as it is an average of the performance of each individual classes.

4.5 Baselines

4.5.1 CNN

A single one-dimensional convolution neural network has been used by [6] for ICD coding on the MIMIC-III dataset.

4.5.2 CAML & DR-CAML

Convolutional Attention network for Multi-Label classification (CAML) is proposed by [6] which used a single CNN layer with per-label attention to generate label-aware features. Description Regularized CAML (DR-CAML) are simply extensions of CAML which adds regularization to the model by utilizing text descriptions for each ICD code. The model has achieved impressive results on MIMIC-II and MIMIC-III datasets.

4.5.3 LEAM

Label Embedding Attentive Model (LEAM) proposed by [16] is implemented by jointly embedding the word and label in the same latent space, and the text representations are constructed directly using the text-label compatibility.

4.5.4 MultiResCNN

Multi-Filter Residual Convolution Neural Network (MultiResCNN) [17] employed multi-channel CNN using different kernel sizes and a residual convolutional layer to enlarge the receptive field on the MIMIC-III dataset.

Table 2. Experimental results

Model	Micro-AUC	Macro-AUC	Micro-F1	Macro-F1
CNN [6]	90.7	87.6	62.5	57.6
CAML [6]	90.9	87.5	61.4	53.2
DR-CAML [6]	91.6	88.4	63.3	57.6
LEAM [16]	91.2	88.1	61.9	54.0
MultiResCNN [17]	92.8	89.9	67.0	60.6
JointLAAT [12]	94.6	92.5	71.5	66.1
CNN with label attention	93.1	90.6	67.1	60.5
Residual CNN with label attention	93.8	91.6	68.9	62.8
Residual CNN with label co-occurrence and label attention	93.6	91.8	70.0	64.6

4.5.5 JointLAAT

JointLAAT is a hierarchical joint learning model proposed by [12] which utilizes bidirectional-LSTM and label attention. It leverages the hierarchical relationship between ICD codes by building a hierarchical architecture which first predict the higher-level ICD code comprising of the first three characters in an ICD code and utilizing that information to predict the raw ICD code.

4.6 Experimental Results

The experimental results are all done on the 50 most frequent ICD from MIMIC-III dataset. We use the same experimental settings as previous works. Our MIMIC-III 50 dataset contains 8,067 discharge summaries for training and 1,574 and 1,730 documents for validation and testing, respectively. The train-test-split would be similar to previous works [6, 9]. The max length of the document is also truncated to 2500, the same as previous literature for a fair comparison. None of the clinical notes that belong to the same patient end up in both the training and testing dataset.

Our experimental results are shown in Table 2. We compare our model against our own baseline model, CNN with label attention which employs a single CNN layer with an attention layer. The baseline is compared with the proposed architecture using residual blocks and label co-occurrence. This is done so that we can assess the performance impact of each component. Comparing the Residual CNN model with label attention against our baseline model, the model utilizing residual blocks had an improvement of 2.3% in Macro-F1. By further adding label co-occurrence, the model using both residual CNN, label attention and label co-occurrence had an improvement of 4.1% in Macro-F1 score.

Comparing our results against other baselines, we can get significantly better results when compared against CAML [6], DR-CAML [6], LEAM [16] and MultiResCNN [17]. Our own baseline model, CNN with label attention is similar to CAML [6], and our

baseline model performs 7.3% better in Macro-F1. One of the reasons for the difference can be due to our use of a pre-trained word embedding, BioWordVec which are trained on multiple sources of clinical text. However, our result fall short when compared against JointLAAT. The difference between our work is that JointLAAT uses Bi-directional LSTM and utilizes the hierarchical structure of ICD that predicts a higher-level ICD code comprising of the first three characters in an ICD and uses that information to predict the raw ICD code. With that, JointLAAT have a higher macro-F1 score of 66.1% which are 1.5% higher than our work.

5 Conclusion

In this paper, we have presented a residual convolution neural network with label attention and label co-occurrence. We have conducted the experiment on MIMIC-III dataset, while our result does not achieve state-of-the-art results, we can conclude that the proposed method of utilizing label attention, label co-occurrence and residual connections can bring a positive impact on model performance.

For future work, we would like to extend our work by taking advantage of the hierarchical structure of ICD code in our network architecture. We also want to look into extending our work to the full ICD code dataset instead of restricting it to the top-50 ICD dataset. Lastly, we want to explore the explainability of our models by extracting the words/phrases that are important to the model. Explainable models are important as they can increase trust in human coders in using the model as well as to detect potential errors and biases in the model. Explainable models are also essential to conform to regulatory laws such as the European Union's General Data Protection Regulation (GDPR) which require any automated decision-making system to be able to explain why a decision has been made.

Acknowledgments. The author would like to thank Multimedia University and Ministry of Higher Education under the Fundamental Research Grant Scheme (FRGS) number: FRGS/1/2019/ICT02/MMU/02/16 for funding this project.

References

1. Zafirah, S. A., Nur, A. M., Puteh, S. E. W., & Aljunid, S. M. (2018). Potential loss of revenue due to errors in clinical coding during the implementation of the Malaysia diagnosis related group (MY-DRG®) Casemix system in a teaching hospital in Malaysia. *BMC health services research*, *18*(1), 1-11.
2. Johnson, A., Pollard, T., & Mark, R. (2016). MIMIC-III Clinical Database (version 1.4). PhysioNet. https://doi.org/10.13026/C2XW26.
3. Johnson, A. E., Pollard, T. J., Shen, L., Lehman, L. W. H., Feng, M., Ghassemi, M., ... & Mark, R. G. (2016). MIMIC-III, a freely accessible critical care database. Scientific data, 3(1), 1-9.

4. Larkey, L. S., & Croft, W. B. (1996, August). Combining classifiers in text categorization. In Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 289–297).
5. de Lima, L. R., Laender, A. H., & Ribeiro-Neto, B. A. (1998, November). A hierarchical approach to the automatic categorization of medical documents. In Proceedings of the seventh international conference on Information and knowledge management (pp. 132–139).
6. Mullenbach, J., Wiegreffe, S., Duke, J., Sun, J., & Eisenstein, J. (2018). Explainable prediction of medical codes from clinical text. arXiv preprint arXiv:1802.05695.
7. Mayya, V., Kamath, S., Krishnan, G. S., & Gangavarapu, T. (2021). Multi-channel, convolutional attention based neural model for automated diagnostic coding of unstructured patient discharge summaries. Future Generation Computer Systems, 118, 374-391.
8. Ayyar, S., Don, O., & Iv, W. (2016). Tagging patient notes with icd-9 codes. In Proceedings of the 29th Conference on Neural Information Processing Systems (pp. 1–8).
9. Shi, H., Xie, P., Hu, Z., Zhang, M., & Xing, E. P. (2017). Towards automated ICD coding using deep learning. arXiv preprint arXiv:1711.04075.
10. Xie, P., & Xing, E. (2018, July). A neural architecture for automated ICD coding. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 1066–1076).
11. Baumel, T., Nassour-Kassis, J., Cohen, R., Elhadad, M., & Elhadad, N. (2018, June). Multi-label classification of patient notes: case study on ICD code assignment. In Workshops at the thirty-second AAAI conference on artificial intelligence.
12. Vu, T., Nguyen, D. Q., & Nguyen, A. (2020). A label attention model for icd coding from clinical text. arXiv preprint arXiv:2007.06351.
13. Heo, T. S., Yoo, Y., Park, Y., Jo, B., Lee, K., & Kim, K. (2021, December). Medical Code Prediction from Discharge Summary: Document to Sequence BERT using Sequence Attention. In 2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA) (pp. 1239–1244). IEEE.
14. Schäfer, H., & Friedrich, C. M. (2020). Multilingual ICD-10 Code Assignment with Transformer Architectures using MIMIC-III Discharge Summaries. In CLEF (Working Notes).
15. Zhang, Y., Chen, Q., Yang, Z., Lin, H., & Lu, Z. (2019). BioWordVec, improving biomedical word embeddings with subword information and MeSH. Scientific data, 6(1), 1-9.
16. Wang, G., Li, C., Wang, W., Zhang, Y., Shen, D., Zhang, X., ... & Carin, L. (2018). Joint embedding of words and labels for text classification. arXiv preprint arXiv:1805.04174.
17. Li, F., & Yu, H. (2020, April). Icd coding from clinical text using multi-filter residual convolutional neural network. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 34, No. 05, pp. 8180–8187).