Automatic ICD Coding via Interactive Shared Representation Networks with Self-distillation Mechanism

Tong Zhou[1,3,*], **Pengfei Cao**[1,2], **Yubo Chen**[1,2], **Kang Liu**[1,2], **Jun Zhao**[1,2], **Kun Niu**[3]
Weifeng Chong[4], **Shengping Liu**[4]
[1] National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences
[2] School of Artificial Intelligence University of Chinese Academy of Sciences
[3] Beijing University of Posts and Telecommunications
[4] Beijing Unisound Information Technology Co., Ltd
{tongzhou21, niukun}@bupt.edu.cn
{pengfei.cao,yubo.chen,kliu,jzhao}@nlpr.ia.ac.cn
{chongweifeng,liushengping}@unisound.com

Abstract

The ICD coding task aims at assigning codes of the International Classification of Diseases in clinical notes. Since manual coding is very laborious and prone to errors, many methods have been proposed for the automatic ICD coding task. However, existing works either ignore the long-tail of code frequency or the noisy clinical notes. To address the above issues, we propose an **I**nteractive **S**hared Representation Network with Self-**D**istillation mechanism. Specifically, an interactive shared representation network targets building connections among codes while modeling the co-occurrence, consequently alleviating the long-tail problem. Moreover, to cope with the noisy text issue, we encourage the model to focus on the clinical note's noteworthy part and extract valuable information through a self-distillation learning mechanism. Experimental results on two MIMIC datasets demonstrate the effectiveness of our method.

1 Introduction

The International Classification of Diseases (ICD) is a healthcare classification system launched by the World Health Organization. It contains a unique code for each disease, symptom, sign and so on. Analyzing clinical data and monitoring health issues would become more convenient with the promotion of ICD codes (Shull, 2019) (Choi et al., 2016) (Avati et al., 2018). The ICD coding task aims at assigns proper ICD codes to a clinical note. It has drawn much attention due to the importance of ICD codes. This task is usually undertaken by experienced coders manually. However, the manually process is inclined to be labor-intensive and error-prone (Adams et al., 2002). A knowledgeable coder with medical experience has to read the whole clinical note with thousands of words in medical terms and assigning multiple codes from a large number of candidate codes, such as 15,000 and 60,000 codes in the ninth version (ICD-9) and the tenth version (ICD-10) of ICD taxonomies. On the one hand, medical expert with specialized ICD coding skills is hard to train. On the other hand, it is a challenge task even for professional coders, due to the large candidate code set and tedious clinical notes. As statistics, the cost incurred by coding errors and the financial investment spent on improving coding quality are estimated to be $25 billion per year in the US (Lang, 2007).

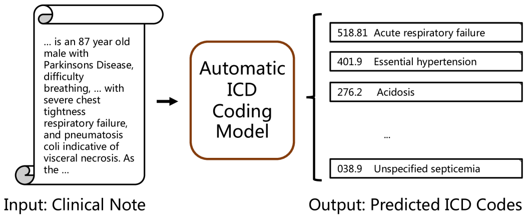
... is an 87 year old male with Parkinsons Disease, difficulty breathing, ... with severe chest tightness respiratory failure, and pneumatosis coli indicative of visceral necrosis. As the ...
Automatic ICD Coding Model
518.81 Acute respiratory failure
401.9 Essential hypertension
276.2 Acidosis
...
038.9 Unspecified septicemia
Input: Clinical Note
Output: Predicted ICD Codes

Figure 1: An example of automatic ICD coding task.

Automatic ICD coding methods (Stanfill et al., 2010) have been proposed to resolve the deficiency of manual annotation, regarding it as a multi-label text classification task. As shown in Figure 1, given a plain clinical text, the model tries to predict all the standardized codes from ICD-9. Recently, neural networks were introduced (Mullenbach et al., 2018) (Falis et al., 2019) (Cao et al., 2020) to alleviate the deficiency of manual feature engineering process of traditional machine learning method (Larkey and Croft, 1996) (Perotte et al., 2014) in ICD coding task, and great progresses have been made. Although effective, those methods either ignore the

[*] Work was done during an internship at National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences.

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics
and the 11th International Joint Conference on Natural Language Processing, pages 5948–5957
August 1–6, 2021. ©2021 Association for Computational Linguistics

long-tail distribution of the code frequency or not target the **noisy text** in clinical note. In the following, we will introduce the two characteristics and the reasons why they are critical for the automatic ICD coding. **Long-tail:** The long-tail problem is unbalanced data distribution phenomenon. And this problem is particularly noticeable in accompanied by a large target label set.

According to our statistics, the proportion of the top 10% high-frequency codes in MIMIC-III (Johnson et al., 2016) occupied 85% of total occurrence. And 22% of the codes have less than two annotated samples. This is intuitive because people usually catch a cold but seldom have cancer. Trained with these long-tail data, neural automatic ICD coding method would inclined to make wrong predictions with high-frequency codes. Fortunately, intrinsic relationships among different diseases could be utilized to mitigate the deficiency caused by long-tail. For example, *Polyneuropathy in diabetes* is a complication of *diabetes*, with a lower probability than other complications since the long term effect of *vessel lesion* reflect at *nerve* would come out in the late-stage. If a model could learn shared information between *polyneuropathy in diabetes* and more common diseases *diabetes*, the prediction space would range to a set of complication of *diabetes*. Further, utilizing the dynamic code co-occurrence, (the cascade relationship among complications of *diabetes*) the confidence of predicting *polyneuropathy in diabetes* is gradually increased with the occurrence of *vessel blockages*, *angina pectoris*, *hypertorphy of kidney*, respectively. Therefore, how to learn shared information with considering dynamic code co-occurrence characteristics, is a crucial and challenging issue.

Noisy text: The noisy text problem means that plentiful of information showing in clinical notes are redundant or misleading for ICD coding task. Clinical notes are usually written by doctors and nurses with different writing styles, accompanied by polysemous abbreviations, abundant medication records and repetitive records of physical indicators. According to our statistics[1], about 10% of words in a clinical note contribute to the code assign task, on average. Other words are abundant medication records and repetitive records of physical indicators. These words are not just redundant but also misleading to the ICD coding task. For example, two critical patients with entirely different diseases could take similar medicines and have similar physical indicators in the rescue course. We argue that the noisy clinical notes are hard to read for both humans and machines. Training with such noisy text would confuse the model about where to focus on, and make wrong decisions due to the semantic deviation. Therefore, another challenging problem is how to deal with the noisy text in ICD coding task.

In this paper, we propose an **I**nteractive **S**hared Representation Network with Self-**D**istillation Mechanism (ISD) to address the above issues.

To mitigate the disadvantage caused by the long-tail issue, we extract shared representations among high-frequency and low-frequency codes from clinical notes. Codes with different occurrence frequencies all make binary decisions based on shared information rather than individually learning attention distributions. Additional experiments indicate that those shared representations could extract common information relevant to ICD codes. Further, we process the shared representations to an interaction decoder for polishing. The decoder additional supervised by two code completion tasks to ensure the dynamic code co-occurrence patterns were learned.

To alleviate the noisy text issue, we further propose a self-distillation learning mechanism to ensure the extracted shared representations focus on the long clinical note's noteworthy part. The teacher part makes predictions through constructed purified text with all crucial information; meanwhile, the student part takes the origin clinical note as a reference. The student is forced to learn the teacher's shared representations with identical target codes.

The contributions of this paper are as follows:

1) We propose a framework capable of dealing with the long-tail and noisy text issues in the ICD coding task simultaneously.

2) To relieve the long-tail issue, we propose an interactive shared representation network, which can capture the internal connections among codes with different frequencies. To handle the noisy text, we devise a self-distillation learning mechanism, guiding the model focus on important parts of clinical notes.

3) Experiments on two widely used ICD coding datasets, MIMIC-II and MIMIC-III, show our

[1] We randomly select 20 clinical notes in MIMIC-III and manually highlight the essential words.

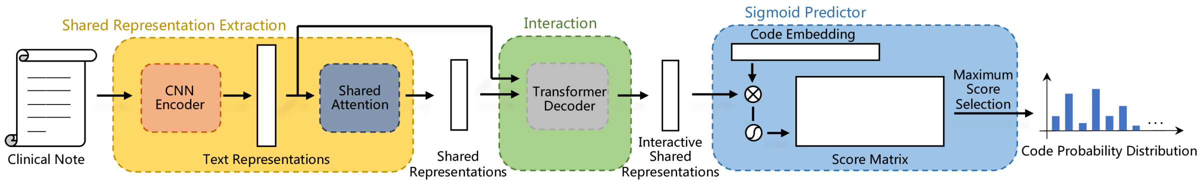

Figure 2: The architecture of Interactive Shared Representation Networks.

method outperforms state-of-the-art methods in macro F1 with 4% and 2%, respectively. The source code is available at `www.github.com/tongzhou21/ISD`.

2 Related Work

ICD coding is an important task in the limelight for decades. Feature based methods firstly brought to solve this task. (Larkey and Croft, 1996) explored traditional machine learning algorithms, including KNN, relevance feedback, and Bayesian applying to ICD coding. (Perotte et al., 2014) utilized SVM for classification in consideration of the hierarchy of ICD codes. With the popularity of neural networks, researchers have proven the effectiveness of CNN and LSTM in ICD coding task. (Mullenbach et al., 2018) propose a convolutional neural network with an attention mechanism to capture each code's desire information in source text also exhibit interpretability. (Xie and Xing, 2018) develop tree LSTM to utilize code descriptions. To further improve the performance, customized structures were introduced to utilize the code co-occurrence and code hierarchy of ICD taxonomies. (Cao et al., 2020) embedded the ICD codes into hyperbolic space to explore their hierarchical nature and constructed a co-graph to import code co-occurrence prior. We argue that they capture code co-occurrence in a static manner rather than dynamic multi-hop relations. (Vu et al., 2020) consider learning attention distribution for each code and introduce hierarchical joint learning architecture to handle the tail codes. Taking advantage of a set of middle representations to deal with the long-tail issue is similar to our shared representation setting, while our method enables every label to choose its desire representation from shared attention rather than its upper-level node, with more flexibility.

The direct solution to deal with an imbalance label set is re-sampling the training data (Japkowicz and Stephen, 2002) (Shen et al., 2016) or reweighting the labels in the loss function (Wang et al., 2017) (Huang et al., 2016). Some studies treat the classification of tail labels as few-shot learning task. (Song et al., 2019) use GAN to generate label-wise features according to ICD code descriptions. (Huynh and Elhamifar, 2020) proposed shared multi-attention for multi-label image labeling. Our work further constructs a label interaction module for label relevant shared representation to utilize dynamic label co-occurrence.

Lots of effects tried to normalize noisy texts before inputting to downstream tasks. (Vateekul and Koomsubha, 2016) (Joshi and Deshpande, 2018) apply pre-processing techniques on twitter data for sentiment classification. (Lourentzou et al., 2019) utilized seq2seq model for text normalization. Others targeted at noisy input in an end2end manner by designing customized architecture. (Sergio and Lee, 2020) (Sergio et al., 2020). Different from previous works on noisy text, our method neither need extra text processing nor bring in specific parameters.

3 Method

This section describes our interactive shared representation learning mechanism and self-distillation learning paradigm for ICD coding. Figure 2 shows the architecture of interactive shared representation networks and manifest the inference workflow of our method. We first encode the source clinical note to the hidden state with a multi-scale convolution neural network. Then a shared attention module further extracts code relevant information shared among all codes. A multi-layer bidirectional Transformer decoder insert between the shared attention representation extraction module and code prediction, establishes connections among shared code relevant representations.

3.1 Multi-Scale Convolutional Encoder

We employ convolutional neural networks (CNN) for source text representation because the computation complexity affected by the length of clinical notes is non-negligible, although other sequen-

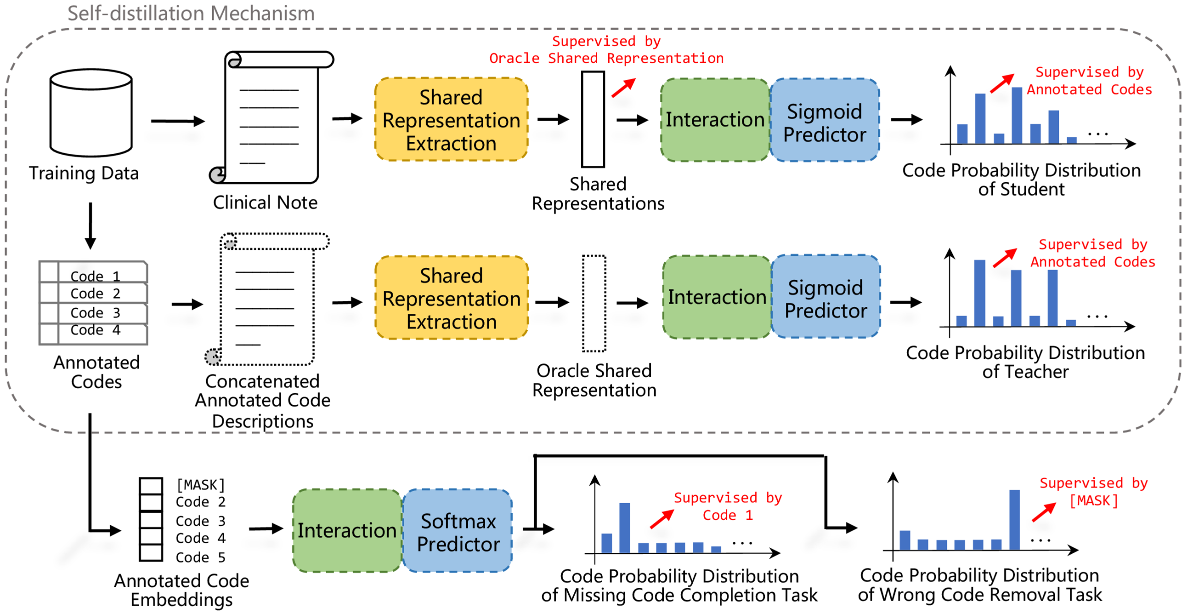
Self-distillation Mechanism
Training Data
Clinical Note
Shared Representation Extraction
Supervised by Oracle Shared Representation
Shared Representations
Interaction
Sigmoid Predictor
Supervised by Annotated Codes
Code Probability Distribution of Student
Code 1
Code 2
Code 3
Code 4
Annotated Codes
Concatenated Annotated Code Descriptions
Shared Representation Extraction
Oracle Shared Representation
Interaction
Sigmoid Predictor
Supervised by Annotated Codes
Code Probability Distribution of Teacher
[MASK]
Code 2
Code 3
Code 4
Code 5
Annotated Code Embeddings
Interaction
Softmax Predictor
Supervised by Code 1
Code Probability Distribution of Missing Code Completion Task
Supervised by [MASK]
Code Probability Distribution of Wrong Code Removal Task

Figure 3: The workflow of our method (ISD) during the training stage. We take the example of training data with a clinical note and annotated four target codes.

tial encoders such as recurrent neural networks or Transformer(Vaswani et al., 2017) could capture longer dependency of text, theoretically. CNN could encode local n-gram pattern, critical in text classification, and with high computational efficiency. The words in source text are first mapped into low-dimensional word embedding space, constitute a matrix $E = \{e_1, e_2, ..., e_{N_x}\}$. Note that N_x is the clinical note's length, e is the word vector with dimension d_e. As shown in Eq. 1 and 2, we concatenate the convolutional representation from kernel set $C = \{c_1, c_2, ..., c_S\}$ with different size k_c to hidden representation matrix $H = \{h_1, h_2, ..., h_{N_x}\}$ with size $N_x \times d_l$:

$$h_i^{c_j} = \tanh(W_c * x_{i:i+k_{c_j}-1} + b_{c_j}) \quad (1)$$

$$h_i = \{h_i^{c_0}; h_i^{c_1}; ...; h_i^{c_S}\} \quad (2)$$

3.2 Shared Attention

The label attention method tends to learn relevant document representations for each code. We argue that the attention of rare code could not be well learned due to lacking training data. Motivated by (Huynh and Elhamifar, 2020) we propose shared attention to bridge the gap between high-frequency and low-frequency codes by learning shared representations H^S through attention. Code set with total number of N_l codes represents in code embedding $E^l = \{e_1^l, e_2^l, ..., e_{N_l}^l\}$ according to their text descriptions. A set of trainable shared queries for attention with size $N_q \times d_l$ is introduced, noted as $E^q = \{e_1^q, e_2^q, ..., e_{N_q}^q\}$, where N_q is the total number of shared queries as a hyperparameter. Then E^q calculates shared attention representation $H^S = \{h_1^S, h_2^S, ..., h_{N_q}^S\}$ with hidden representation H in Eq. 3 to 5:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}}V) \quad (3)$$

$$\alpha_i = \text{Attention}(e_i^q, H, H) \quad (4)$$

$$h_i^S = H \cdot \alpha_i \quad (5)$$

In ideal conditions, those shared representations reflect the code relevant information corresponding to the source text. We can predict codes through H^S. Each code i has its right to choose a shared representation in H^S for code-specific vector through the highest dot product score s_i.

$$s_i = \max(H^S \cdot e_i^l) \quad (6)$$

The product score was further applying to calculate the final score $\hat{y}_l$ through the sigmoid function.

$$\hat{y}_i = \sigma(s_i) \quad (7)$$

With the supervision of binary cross-entropy loss function, the shared representation should have learned to represent code relevant information.

$$\mathcal{L}_{pred} = \sum_{i=1}^{N_l}[-y_i log(\hat{y}_i) - (1 - y_i) log(1 - \hat{y}_i)] \quad (8)$$

3.3 Interactive Shared Attention

Above shared attention mechanism lacks interaction among code relevant information, which is of great importance in the ICD coding task. We implement this interaction through a bidirectional multi-layer Transformer decoder D with an additional code completion task. The shared representation H^S is considered the orderless sequential input of the decoder D. Each layer of the Transformer contains interaction among shared representation H^S through self-attention and interaction between shared representation and source text through source sequential attention.

To make sure the decoder could model the dynamic code co-occurrence pattern, we propose two code set completion tasks, shown at the bottom of Figure 3.

(1) Missing code completion: We construct a code sequence L_{tgt} of a real clinical note X in the training set, randomly masking one code l_{mis}. The decoder takes this code sequence as input to predict the masked code.

$$\mathcal{L}_{mis} = -logP(l_{mis}|L_{tgt} \setminus l_{mis} \cup l_{mask}, X) \quad (9)$$

(2) Wrong code removal: Similar to the above task, we construct a code sequence L_{tgt}, but by randomly adding a wrong code l_{wro}. The decoder is aiming to fade the wrong code's representation with a special mask representation l_{mask}.

$$\mathcal{L}_{rem} = -logP(l_{mask}|L_{tgt} \cup l_{wro}, X) \quad (10)$$

The decoder could generate purificatory code relevant information with higher rationality with the above two tasks' learning. The decoder is plugged to refine the shared representation H^S to $H^{S\prime}$, so the subsequent dot product score is calculated by $H^{S\prime}$.

$$s_i = \max(H^{S\prime} \cdot e_i^l) \quad (11)$$

3.4 Self-distillation Learning Mechanism

We argue that learning the desired shared attention distribution over such a long clinical text is difficult, and the α_i tends to be smooth, brings lots of unnecessary noise information. Therefore we propose a self-distillation learning mechanism showing in the gray dotted lines of Figure 3. With this mechanism, the model could learn superior intermediate representations from itself without introducing another trained model.

Considering a single clinical note X with target code set L_{tgt} for training, we derive two paths inputted to the model. The teacher's training data consists of the text descriptions $X^{L_{tgt}} = \{X_1^l, X_2^l, ..., X_{N_{l_{tgt}}}^l\}$. We handle those code descriptions separately through the encoder and concatenate them into a flat sequence of hidden state $H^{L_{tgt}} = \{H^{l_1}; H^{l_2}; ...; H^{l_{N_{l_{tgt}}}}\}$, where $N_{l_{tgt}}$ is the number of code in L_{tgt}, so the subsequent process in our model is not affected.

We optimize the teacher's prediction result $\hat{y}_i^{tgt}$ through binary cross-entropy loss.

$$\mathcal{L}_{tgt} = \sum_{i=1}^{N_l}[-y_i log(\hat{y}_i^{tgt}) - (1 - y_i) log(1 - \hat{y}_i^{tgt})] \quad (12)$$

Student takes origin clinical note Xas input and also have BCE loss to optimize. We assume that an origin clinical note with thousands of words contains all desired codes' information, as well as less essential words. The teacher's input contains all desired information that indicates codes to be predicted without any noise. Ideal shared representations obtained from attention are supposed to collect code relevant information only. Hence we treat the teacher's share representation $H^{L_{tgt}}$ as a perfect example to the student. A distillation loss encourages those two representation sequences to be similar.

$$\text{cosine}(H^A, H^B) = \sum_i^N \frac{h_i^A \cdot h_i^B}{\parallel h_i^A \parallel \ \parallel h_i^B \parallel} \quad (13)$$

$$\mathcal{L}_{dist} = \min\{1 - \text{cosine}(H^{S\prime}, H^{L_{tgt}\prime})\} \quad (14)$$

Since we treat the shared representations without order restrict, every teacher have its rights to choose a suitable student, meanwhile, considering other teachers' appropriateness. It implements with Hungarian algorithm (Kuhn, 1955) to calculates the cosine distance globally minimum. Where $\prime$ denotes any shuffle version of the origin representation sequence.

3.5 Training

The complete training pipeline of our method is shown in Figure 3. The final loss function is the

Model	MIMIC-III-full					MIMIC-III 50				
	AUC		F1		P@8	AUC		F1		P@5
	Macro	Micro	Macro	Micro		Macro	Micro	Macro	Micro	
CAML	0.895	0.986	0.088	0.539	0.709	0.875	0.909	0.532	0.614	0.609
DR-CAML	0.897	0.985	0.086	0.529	0.690	0.884	0.916	0.576	0.633	0.618
MSATT-KG	0.910	0.992	0.090	0.553	0.728	0.914	0.936	0.638	0.684	0.644
MultiResCNN	0.910	0.986	0.085	0.552	0.734	0.899	0.928	0.606	0.670	0.641
HyperCore	0.930	0.989	0.090	0.551	0.722	0.895	0.929	0.609	0.663	0.632
LAAT	0.919	0.988	0.099	0.575	0.738	0.925	0.946	0.666	0.715	0.675
JointLAAT	0.921	0.988	0.107	**0.575**	0.735	0.925	0.946	0.661	0.716	0.671
ISD (Ours)	**0.938**	**0.990**	**0.119**	0.559	**0.745**	**0.935**	**0.949**	**0.679**	**0.717**	**0.682**
	±0.003	±0.003	±0.002	±0.002	±0.001	±0.004	±0.001	±0.009	±0.003	±0.005

Table 1: Comparison of our model and other baselines on the MIMIC-III dataset. We run our model 10 times and each time we use different random seeds for initialization. We report the *mean* ± *standard deviation* of each result.

Model	AUC		F1		P@8
	Macro	Micro	Macro	Micro	
HA-GRU	-	-	-	0.366	-
CAML	0.820	0.966	0.048	0.442	0.523
DR-CAML	0.826	0.966	0.049	0.457	0.515
MultiResCNN	0.850	0.968	0.052	0.464	0.544
HyperCore	0.885	0.971	0.070	0.477	0.537
LAAT	0.868	0.973	0.059	0.486	0.550
JointLAAT	0.871	0.972	0.068	0.491	0.551
ISD (Ours)	**0.901**	**0.977**	**0.101**	**0.498**	**0.564**
	±0.004	±0.002	±0.004	±0.002	±0.002

Table 2: Experimental results are shown in *means* ± *standard deviations* on the MIMIC-II dataset.

weighting sum of the above losses.

$$\mathcal{L} = \lambda_{pred}\mathcal{L}_{pred} + \lambda_{mis}\mathcal{L}_{mis} + \lambda_{rem}\mathcal{L}_{rem} + \lambda_{tgt}\mathcal{L}_{tgt} + \lambda_{dist}\mathcal{L}_{dist} \tag{15}$$

4 Experiments

4.1 Datasets

For fair comparison, we follow the datasets used by previous work on ICD coding (Mullenbach et al., 2018) (Cao et al., 2020), including MIMIC-II (Jouhet et al., 2012) and MIMIC-III (Johnson et al., 2016). The third edition is the extension of II. Both datasets contain discharge summaries that are tagged manually with a set of ICD-9 codes. The dataset preprocessing process is consistent with (Mullenbach et al., 2018). For MIMIC-III full dataset, there are 47719, 1631, 3372 different patients' discharge summaries for training, development, and testing, respectively. Totally 8921 unique codes occur in those three parts. MIMIC-III 50 dataset only retains the most frequent codes appear in full setting, leave 8067, 1574, 1730 discharge summaries for training, development, and testing, respectively. MIMIC-II dataset contains 5031 unique codes divided into 20533 and 2282 clinical notes for training and testing, respectively.

4.2 Metrics and Parameter Settings

As in previous works (Mullenbach et al., 2018), we evaluate our method using both the micro and macro, F1 and AUC metrics. As well as P@8 indicates the proportion of the correctly-predicted codes in the top-8 predicted codes. PyTorch (Paszke et al., 2019) is chosen for our method's implementation. We perform a grid search over all hyperparameters for each dataset. The parameter selections are based on the tradeoff between validation performance and training efficiency. We set the word embedding size to 100. We build the vocabulary set using the CBOW Word2Vec method (Mikolov et al., 2013) to pre-train word embeddings based on words in all MIMIC data, resulting in the most frequent 52254 words included. The multi-scale convolution filter size is 5, 7, 9, 11, respectively. The size of each filter output is one-quarter of the code embedding size. We set code embedding size to 128 and 256 for the MIMIC-II and MIMIC-III, respectively. The size of shared representation is 64. We utilize a two-layer Transformer for the interactive decoder. For the loss function, we set $\lambda_{mis} = 0.5$, $\lambda_{mis} = 5e-4$, $\lambda_{rem} = 5e-4$, $\lambda_{tgt} = 0.5$, and $\lambda_{dist} = 1e-3$ to adjust the scale of different supervisory signals. We use Adam for optimization with an initial learning rate of 3e-4, and other settings keep the default.

4.3 Baselines

We compare our method with the following baselines:

HA-GRU: A Hierarchical Attention Gated Recurrent Unit model is proposed by (Baumel et al., 2017) to predict ICD codes on the MIMIC-II dataset.

CAML & DR-CAML: (Mullenbach et al., 2018) proposed the Convolutional Attention Net-

Model	AUC		F1		P@8
	Macro	Micro	Macro	Micro	
ISD (Ours)	**0.938**	**0.990**	**0.119**	**0.559**	**0.745**
w/o distillation loss	0.935	0.986	0.103	0.551	0.743
w/o self-distillation	0.934	0.981	0.099	0.547	0.724
w/o code completion task	0.931	0.988	0.061	0.522	0.728
w/o co-occurrence decoder	0.936	0.989	0.084	0.547	0.743

Table 3: Ablation results on the MIMIC-III-full test set.

work for Multi-Label Classification (CAML), which learning attention distribution for each label. DR-CAML indicates Description Regularized CAML, an extension incorporating the text description of codes.

MSATT-KG: The Multi-Scale Feature Attention and Structured Knowledge Graph Propagation was proposed by (Xie et al., 2019) They capture variable n-gram features and select multi-scale features through densely connected CNN and a multi-scale feature attention mechanism. GCN is also employed to capture the hierarchical relationships among medical codes.

MultiResCNN: The Multi-Filter Residual Convolutional Neural Network was proposed by (Li and Yu, 2020). They utilize the multi-filter convolutional layer capture variable n-gram patterns and residual mechanism to enlarge the receptive field.

HyperCore: Hyperbolic and Co-graph Representation was proposed by (Cao et al., 2020). They explicitly model code hierarchy through hyperbolic embedding and learning code co-occurrence thought GCN.

LAAT & JointLAAT: (Vu et al., 2020) Label Attention model (LAAT) for ICD coding was proposed by (Vu et al., 2020), learning attention distributions over LSTM encoding hidden states for each code. JointLAAT is an extension of LAAT with hierarchical joint learning.

4.4 Compared with State-of-the-art Methods

The left part of Table 1 and Table 2 show the results of our method on the MIMIC-III and MIMIC-II dataset with the whole ICD code set. Compared with previous methods generating attention distribution for each code, our method achieves better results on most metrics, indicating the shared attention mechanism's effectiveness. It is noteworthy that the macro results have more significant improvement compare to micro than previous methods. Since the macro indicators are mainly affected by tail codes' performance, our approach benefits from the interactive shared representations among codes with different frequencies.

Compared with the static code interaction of co-occurrence implemented in (Cao et al., 2020), our method achieves higher scores, indicating that the dynamic code interaction module could capture more complex code interactive information other than limit steps of message passing in GCN.

The right part of Table 1 shows the results of our method on the MIMIC-III dataset with the most frequent 50 codes. It proved that our approach's performance would not fall behind with a more balanced label set.

4.5 Ablation Experiments

To investigate the effectiveness of our proposed components of the method, we also perform the ablation experiments on the MIMIC-III-full dataset. The ablation results are shown in Table 3, indicating that none of these models can achieve a comparable result with our full version. Demonstrate that all those factors contribute a certain improvement to our model.

(1) Effectiveness of Self-distillation. Specifically, when we discard the whole self-distillation part (w/o self-distillation), the performance drops, demonstrate the effectiveness of the self-distillation. To further investigate the contribution of the self-distillation module, whether the more training data we constructed, we retain the teacher path and remove the loss between shared representations (w/o distillation loss), the performance still slightly drops. It can be concluded that although the positive effects of the constructed training data in the teacher path, the distillation still plays a role.

(2) Effectiveness of Shared Representation. When we remove the self-distillation mechanism (w/o self-distillation), the contribution of shared representation part can be deduced compared to the performance of CAML. Result showing our version still have 1.1% advantage in macro F1, indicating the effectiveness of shared representation.

Size	AUC		F1		P@8
	Macro	Micro	Macro	Micro	
1	0.899	0.980	0.081	0.532	0.723
32	0.937	0.990	0.104	0.557	0.737
64	**0.938**	**0.990**	0.119	**0.559**	**0.745**
128	0.938	0.988	**0.124**	0.558	0.743
1159	0.935	0.990	0.116	0.543	0.731

Table 4: Experimental results of our method with different size of shared representations on MIMIC-III-full dataset.

(3) Effectiveness of Code Completion Task. When we neglect the missing code completion task and wrong code removal task (w/o code completion tasks), the code interactive decoder optimizes with final prediction loss only. The performance is even worse than the model without the whole code interaction module (w/o co-occurrence decoder). It indicates that the additional code completion task is the guarantee of modeling dynamic code co-occurrence characteristics. Further compared with the model with label attention rather than our proposed shared representations (w/o shared representation), the performance even worse, showing the code completion task is also the guarantee of the effectiveness of shared representations. Without this self-supervised task, the shared information is obscure and the performance drops due to the join of dubiously oriented model parameters.

4.6 Discussion

To further explore our proposed interactive shared attention mechanism, we conduct comparisons among various numbers of shared representations in our method. And visualization the attention distribution over source text of different shared representations, as well as the information they extracted.

(1) The Analysis of Shared Representations Size. As shown in Table 4, both large or small size would harm the final performance. When the shared size is set to 1, the shared representation degrades into a global representation. A single vector compelled to predict multiple codes causes the performance drops, as Table 4 shows. We also initialize the shared embeddings with ICD's hierarchical parent node. Specifically, there are 1159 unique first three characters in the raw ICD code set of MIMIC-III-full. We initialize those shared embeddings with the mean vector of their corresponding child codes. Although the hierarchical priori knowledge is introduced, the computation complexity and uneven node selection could cause the model to be hard to optimize and overfit high frequent parent nodes.

Clinical Note: chief complaint elective admit major surgical or invasive procedure recoiling acomm aneurysm history of present illness on she had a crushing headache but stayed at home the next day ... angiogram with embolization and or stent placement medication take aspirin 325mg ...
Codes: 437.3 (cerebral aneurysm, nonruptured); 39.75 (endovascular repair of vessel); 88.41 (arteriography of cerebral arteries)

Table 5: The attention distribution visualization over a clinical note of different shared representations. We determine the shared representations according to the target codes' choice. Since we calculate the attention score over hidden states encoded by multi-scale CNN, we take the most salient word as the center word of 5-gram and highlight.

Model	Standard Deviation
ISD (Ours)	0.013992
w/o self-distillation	0.004605

Table 6: The average standard deviation calculated from the attention weights of clinical text in MIMIC-III-full dataset.

(2) Visualization of Shared Attention Distribution. The attention distribution of different shared representations shown in Table 5 indicates that they have learned to focus on different source text patterns in the noisy clinical note to represent code relevant information.

(3) The Analysis of Self-distillation. As shown in Table 6, the attention weights over clinical text learned by model with the training of self-distillation mechanism are more sharp than origin learning process. In combination with Table 5, it can be concluded that the self-distillation mechanism could help the model more focus on the desire words of clinical text.

5 Conclusion

This paper proposes an interactive shared representation network and a self-distillation mechanism for the automatic ICD coding task, to address the long-tail and noisy text issues. The shared representations can bridge the gap between the learning process of frequent and rare codes. And the code interaction module models the dynamic code co-occurrence characteristic, further improving the performance of tail codes. Moreover, to address the noisy text issue, the self-distillation learning mechanism helps the shared representations focus on code-related information in noisy clinical notes. Experimental results on two MIMIC datasets indicate that our proposed model significantly outperforms previous state-of-the-art methods.

Acknowledgments

This work is supported by the National Key Research and Development Program of China (No.2017YFB1002101), the National Natural Science Foundation of China (No. 61806201, 61976211). This work is also supported by Beijing Academy of Artificial Intelligence (BAAI2019QN0301), the Key Research Program of the Chinese Academy of Sciences (Grant NO. ZDBS-SSW-JSC006), independent research project of National Laboratory of Pattern Recognition and the CCF-Tencent Open Research Fund.

References

Diane L Adams, Helen Norman, and Valentine J Burroughs. 2002. Addressing medical coding and billing part ii: a strategy for achieving compliance. a risk management approach for reducing coding and billing errors. *Journal of the National Medical Association*, 94(6):430.

Anand Avati, Kenneth Jung, Stephanie Harman, Lance Downing, Andrew Ng, and Nigam H Shah. 2018. Improving palliative care with deep learning. *BMC medical informatics and decision making*, 18(4):122.

Tal Baumel, Jumana Nassour-Kassis, Raphael Cohen, Michael Elhadad, and Noémie Elhadad. 2017. Multi-label classification of patient notes a case study on icd code assignment. *arXiv preprint arXiv:1709.09587*.

Pengfei Cao, Yubo Chen, Kang Liu, Jun Zhao, Shengping Liu, and Weifeng Chong. 2020. Hypercore: Hyperbolic and co-graph representation for automatic icd coding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3105–3114.

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.

Matus Falis, Maciej Pajak, Aneta Lisowska, Patrick Schrempf, Lucas Deckers, Shadia Mikhael, Sotirios Tsaftaris, and Alison O'Neil. 2019. Ontological attention ensembles for capturing semantic concepts in icd code prediction from clinical text. In *Proceedings of the Tenth International Workshop on Health Text Mining and Information Analysis*, pages 168–177.

Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. 2016. Learning deep representation for imbalanced classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5375–5384.

Dat Huynh and Ehsan Elhamifar. 2020. A shared multi-attention framework for multi-label zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8776–8786.

Nathalie Japkowicz and Shaju Stephen. 2002. The class imbalance problem: A systematic study. *Intelligent data analysis*, 6(5):429–449.

Alistair EW Johnson, Tom J Pollard, Lu Shen, H Lehman Li-Wei, 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(1):1–9.

Shaunak Joshi and Deepali Deshpande. 2018. Twitter sentiment analysis system. *arXiv preprint arXiv:1807.07752*.

Vianney Jouhet, Georges Defossez, Anita Burgun, Pierre Le Beux, P Levillain, Pierre Ingrand, Vincent Claveau, et al. 2012. Automated classification of free-text pathology reports for registration of incident cases of cancer. *Methods of information in medicine*, 51(3):242.

Harold W Kuhn. 1955. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97.

Dee Lang. 2007. Consultant report-natural language processing in the health care industry. *Cincinnati Children's Hospital Medical Center, Winter*, 6.

Leah S Larkey and W Bruce Croft. 1996. Combining classifiers in text categorization. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 289–297.

Fei Li and Hong Yu. 2020. Icd coding from clinical text using multi-filter residual convolutional neural network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8180–8187.

Ismini Lourentzou, Kabir Manghnani, and ChengXiang Zhai. 2019. Adapting sequence to sequence models for text normalization in social media. In *Proceedings of the International AAAI Conference* *on Web and Social Media*, volume 13, pages 335–345.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. 2018. Explainable prediction of medical codes from clinical text. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pages 1101–1111.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, pages 8026–8037.

Adler Perotte, Rimma Pivovarov, Karthik Natarajan, Nicole Weiskopf, Frank Wood, and Noémie Elhadad. 2014. Diagnosis code assignment: models and evaluation metrics. *Journal of the American Medical Informatics Association*, 21(2):231–237.

Gwenaelle Cunha Sergio and Minho Lee. 2020. Stacked debert: All attention in incomplete data for text classification. *Neural Networks*.

Gwenaelle Cunha Sergio, Dennis Singh Moirangthem, and Minho Lee. 2020. Attentively embracing noise for robust latent representation in bert. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3479–3491.

Li Shen, Zhouchen Lin, and Qingming Huang. 2016. Relay backpropagation for effective learning of deep convolutional neural networks. In *European conference on computer vision*, pages 467–482. Springer.

Jessica Germaine Shull. 2019. Digital health and the state of interoperable electronic health records. *JMIR medical informatics*, 7(4):e12712.

Congzheng Song, Shanghang Zhang, Najmeh Sadoughi, Pengtao Xie, and Eric Xing. 2019. Generalized zero-shot icd coding. *arXiv preprint arXiv:1909.13154*.

Mary H Stanfill, Margaret Williams, Susan H Fenton, Robert A Jenders, and William R Hersh. 2010. A systematic literature review of automated clinical coding and classification systems. *Journal of the American Medical Informatics Association*, 17(6):646–651.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Peerapon Vateekul and Thanabhat Koomsubha. 2016. A study of sentiment analysis using deep learning techniques on thai twitter data. In *2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, pages 1–6. IEEE.

Thanh Vu, Dat Quoc Nguyen, and Anthony Nguyen. 2020. A label attention model for icd coding from clinical text. *arXiv preprint arXiv:2007.06351*.

Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. 2017. Learning to model the tail. In *Advances in Neural Information Processing Systems*, pages 7029–7039.

Pengtao Xie and Eric Xing. 2018. A neural architecture for automated icd coding. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 1066–1076.

Xiancheng Xie, Yun Xiong, Philip S Yu, and Yangyong Zhu. 2019. Ehr coding with multi-scale feature attention and structured knowledge graph propagation. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 649–658.