A General Knowledge Injection Framework for ICD Coding

Xu Zhang[1,2] , Kun Zhang[1,2]*, Wenxin Ma[1,2] ,
Rongsheng Wang[1,2] , Chenxu Wu[1,2] , Yingtai Li[1,2] , S. Kevin Zhou [1,2,3,4]*
[1] School of Biomedical Engineering, Division of Life Sciences and Medicine, USTC
[2] MIRACLE Center, Suzhou Institute for Advance Research, USTC
[3]Jiangsu Provincial Key Laboratory of Multimodal Digital Twin Technology
[4]State Key Laboratory of Precision and Intelligent Chemistry, USTC
xu_zhang@mail.ustc.edu.cn kkzhang@ustc.edu.cn skevinzhou@ustc.edu.cn

Abstract

ICD Coding aims to assign a wide range of medical codes to a medical text document, which is a popular and challenging task in the healthcare domain. To alleviate the problems of long-tail distribution and the lack of annotations of code-specific evidence, many previous works have proposed incorporating code knowledge to improve coding performance. However, existing methods often focus on a single type of knowledge and design specialized modules that are complex and incompatible with each other, thereby limiting their scalability and effectiveness. To address this issue, we propose **GKI-ICD**, a novel, general knowledge injection framework that integrates three key types of knowledge, namely ICD Description, ICD Synonym, and ICD Hierarchy, without specialized design of additional modules. The comprehensive utilization of the above knowledge, which exhibits both differences and complementarity, can effectively enhance the ICD coding performance. Extensive experiments on existing popular ICD coding benchmarks demonstrate the effectiveness of GKI-ICD, which achieves the state-of-the-art performance on most evaluation metrics. Code is available at https://github.com/xuzhang0112/GKI-ICD.

1 Introduction

International Classification of Diseases (ICD) [1] is a globally used medical classification system, developed by the World Health Organization to classify diseases, symptoms, procedures, and external causes. The ICD coding task aims to assign the most accurate ICD codes to clinical texts, typically discharge summaries, for further medical billing and clinical research. Two main challenges arise in the ICD coding process (Edin et al., 2023). First, there is a tremendous number of ICD codes to assign in clinical practice, whose distribution is extremely long-tailed, and most of which are lacking in enough training samples. Second, as shown in Figure 1, the occurrence of multiple ICD codes within a long medical document makes it hard for models to accurately link each ICD code with its corresponding evidence fragments. Human coders do not annotate the evidence of the ICD codes assigned by them, due to the complexity of this operation, only leaving document-level annotations to each medical document.

Medical Text (~1500 words)
[...] Patient then became septic and oliguric as the course of the day went on. He was transferred for evaluation as to whether there was an operation that could salvage him. At current time, he is intubated and sedated on 2 pressors. [...]

Assigned ICD Codes (Ground-Truth)
038.9 unspecified septicemia
995.92 severe sepsis
96.71 continuous invasive mechanical ventilation for less than 96 consecutive hours

Figure 1: An example of ICD coding: Occurrence of multiple codes and noisy content in a long medical text document makes it hard to link each ICD code to its corresponding evidence (marked in same color), explaining the necessity of incorporating code-specific knowledge.

In recent years, numerous studies (Ji et al., 2024) have explored the incorporation of ICD code-related knowledge to assist models in precisely locating evidence fragments related to specific ICD codes, thereby effectively and efficiently improving coding performance. Generally, three types of knowledge are involved in ICD coding: *ICD Description, ICD Synonym, and ICD Hierarchy*. Specifically, 1) ICD Description refers to the meaning of each ICD code, which is directly related to the coding process. Language models can leverage semantic mapping to identify the most relevant evidence fragments within a long medical text document, facilitating accurate classification. 2) ICD

*Corresponding authors
[1]https://www.who.int/standards/classifications/classification-of-diseases

Findings of the Association for Computational Linguistics: ACL 2025, pages 7180–7189

Synonym addresses the diversity of medical terminology, as a single ICD code may have multiple linguistic expressions. Incorporating synonyms helps the model recognize different variants of the same code, enhancing its robustness. 3) ICD Hierarchy organizes the relationships between codes. With tens of thousands of codes in ICD-9, these codes are not entirely independent. ICD Hierarchy provides a structured relationship between codes, particularly grouping rare codes with more common ones. *Inherently, these three types of knowledge exhibit both differences and complementarity.*

However, existing methods typically focus on only one of these different types of knowledge and design **specialized network architectures** accordingly, making it hard to integrate other complementary knowledge. To utilize synonym knowledge, current approaches often employ a multi-synonym-attention mechanism, where each query corresponds to a synonym (Yuan et al., 2022; Gomes et al., 2024). To incorporate hierarchical knowledge, methods primarily rely on graph neural networks, treating the hierarchical structure as an adjacency matrix to aggregate code representations (Xie et al., 2019; Ge et al., 2024). Since these methods design specialized modules for individual knowledge types, the complexity of these modules makes it difficult to scale to advanced models. More importantly, the incompatibility between these specialized modules hinders their integration into a unified model, preventing the comprehensive utilization of all knowledge types.

To address the above issue, we propose GKI-ICD, a novel synthesis-based multi-task learning fr to inject knowledge. In contrast to existing methods that often struggle with complex architectural designs and integration challenges, our method jointly leverages all types of knowledge without relying on specialized modules. Specifically, GKI-ICD consists of two key components: guideline synthesis and multi-task learning. The guideline synthesis incorporates ICD code knowledge to synthesize a guideline, ensuring that all the knowledge relevant to the raw sample is embedded within the guideline. Meanwhile, the multi-task learning mechanism requires the model to not only correctly classify the original samples but also make accurate predictions based on the synthesized guidelines. Additionally, it encourages the model to align the information extracted from the raw samples with that from the provided guidelines as closely as possible, thereby facilitating effective knowledge integration.

Our main contributions are summarized as:

- To our knowledge, we are the first to inject ICD code knowledge **without requiring any additional specially-designed networks or prompts**, thus being able to integrate the three kinds of ICD code knowledge separately utilized before.
- We propose a novel synthesis-based multi-task learning mechanism, including guideline synthesis and multi-task learning, to inject ICD code knowldge into the coding model.
- We achieve state-of-the-art performance on most evaluation metrics on the ICD coding benchmarks MIMIC-III and MIMIC-III-50, proving not only the effectiveness of our knowledge injection framework, but also the necessity of multiple knowledge integration.

2 Related Work

2.1 ICD Coding Network

The automatic ICD coding task is well established in the healthcare domain, and most of the approaches fisrt encode the discharge summary with a text encoder, and then use a label attention mechanism to attend, aggregate, and make predictions.

Text encoder. Early ICD coding methods (Mullenbach et al., 2018; Vu et al., 2021; Li and Yu, 2020; Liu et al., 2021) primarily utilized convolutional neural networks (CNNs), recurrent neural networks (RNNs), and their variants as backbones, while recent approaches (Huang et al., 2022; Edin et al., 2024) have been based on pretrained language models (LMs). Besides, large language models (LLMs) have been proved to perform worse on this task (Boyle et al., 2023), compared to fine-tuned small models.

Label attention. Instead of making predictions based on a pooled vector, label attention use a linear layer to compute relationships between each ICD code and each token in the clinical text, aggregate different information for different codes and then make predictions (Mullenbach et al., 2018). Subsequently, this linear layer was replaced by a multilayer perceptron (Vu et al., 2021), and was finally replaced by the standard cross attention (Edin et al., 2024), both improving the training stability and slightly enhancing its performance.

2.2 Knowledge Injection

Considering the rich prior knowledge in biomedical domain, many efforts have been made to incorporate medical knowledge to enhance model performance on ICD coding tasks. Knowledge injection methods can generally be divided into two categories: task-agnostic and task-specific.

Task-agnostic knowledge. Extensive biomedical corpora, such as electronic health records and biomedical academic papers, can be utilized for pretraining language models. These pretrained models, including BioBERT (Lee et al., 2020), ClinicalBERT (Huang et al., 2019), PubMedBERT (Gu et al., 2021) and RoBERTa-PM (Lewis et al., 2020), serve as powerful biomedical text encoders, significantly enhancing the performance of downstream tasks, including ICD coding.

Task-specific knowledge. Task-specific knowledge refers to information related to ICD codes, such as the meaning of each code and the hierarchical structure among codes. By injecting this kind of knowledge during the fine-tuning stage, the model's performance on the ICD coding task can be improved. MSATT-KG (Xie et al., 2019) leverages graph convolutional neural network to capture the hierarchical relationships among medical codes and the semantics of each code. ISD (Zhou et al., 2021) proposes a self-distillation learning mechanism, utilizing code descriptions help the model ignore the noisy text in clinical notes. MSMN (Yuan et al., 2022) uses multiple synonyms of code descriptions to initialize the code query embeddings. KEPTLongformer (Yang et al., 2022) incorporates a medical knowledge graph for self-alignment contrastive learning, and then adds a sequence of ICD code descriptions as prompts in addition to each clinical note as model input. DKEC (Ge et al., 2024) propose a heterogeneous graph network to encode knowledge from multiple sources, and generate knowledge-based queries for each ICD code. MRR (Wang et al., 2024a) and AKIL (Wang et al., 2024b) incorporates diagnosis-related group (DRG) codes, current procedural terminology (CPT) codes, and medications prescribed to patients to generate a dynamic label mask, which can help down-sample the negative labels and focus the classifier on candidate labels. Unlike previous methods that design specialized networks for knowledge injection, we propose a general knowledge injection framework, making it applicable to various models and diverse types of knowledge.

3 Methodology

We first provide an overview in Section 3.1, highlighting the key differences between our proposed GKI-ICD and previous works. Next, we elaborate on its details in Section 3.2. In addition, we briefly describe the ICD coding network adopted in our work in Section 3.3.

3.1 Overview

Typically, the ICD coding task involves optimizing an ICD coding network to assign specific ICD codes to the given medical text, defined as:

$$\theta^* = \arg\min_{\theta} \mathcal{L}(f(\mathbf{x};\theta), y), \tag{1}$$

where $\mathbf{x}$ represents the input medical text and $\mathbf{y}$ denotes the corresponding ground-truth ICD codes, θ denotes the parameters of the ICD coding network.

To further boost performance, existing methods (Xie et al., 2019; Yang et al., 2022; Yuan et al., 2022; Ge et al., 2024; Gomes et al., 2024; Luo et al., 2024) generally devise additional neural networks to inject knowledge, i.e.,

$$\theta^*; \theta_i^* = \arg\min_{\theta;\theta_i} \mathcal{L}(g_i(\mathbf{x};\theta;\theta_i), y), \tag{2}$$

where g_i is a neural network specially designed to incorporate the i-th type of knowledge, and θ_i denotes the corresponding additional module parameters. To be specific, g_i can be graph neural networks for hierarchy knowledge (Xie et al., 2019) or multi-synonym attention networks for synonym knowledge (Yuan et al., 2022), etc.

However, considering these extra modules are complex and hard to integrate simultaneously, our approach aims to propose a new training framework that can inject knowledge without extra parameters. By leveraging knowledge to synthesize guidelines $\hat{x}$ and modifying the training pipeline, we enable the injection of all necessary knowledge to be free of extra parameters or interactions. The proposed knowledge injection framework can be defined as:

$$\theta^* = \arg\min_{\theta} \mathcal{L}'(f(\mathbf{x};\hat{\mathbf{x}};\theta), y), \tag{3}$$

where f is the simplest ICD coding network, having the merit to be adapted to any state-of-the-art network. In the following, as illustrated in Fig. 2, we give the details including guideline synthesis based on knowledge in 3.2.1 and multi-task learning based on synthetic guidelines in 3.2.2.

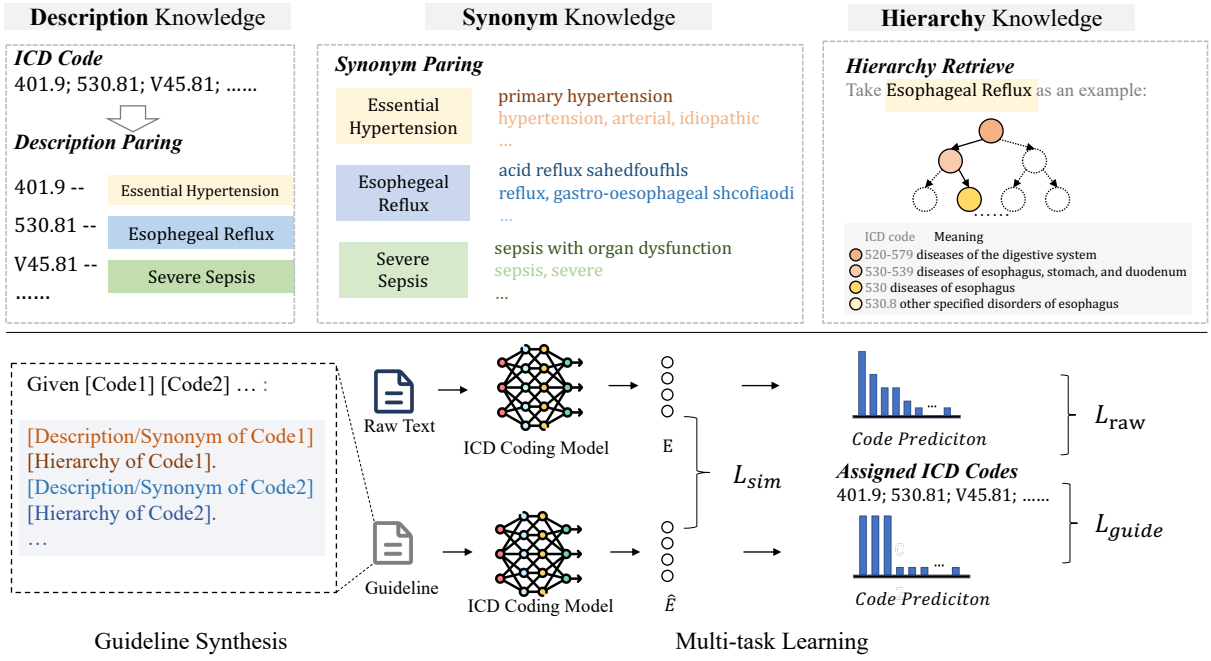

Figure 2: Our proposed general knowledge injection training framework for ICD coding, GKI-ICD. For each training sample, we first retrieve code-specific knowledge to synthesize a guideline, and then use this guideline and multi-task learning to inject knowledge into the model. Note that our method only incorporates knowledge in the training stage, which has no effect on the computation cost of the model during the inference stage.

3.2 Proposed Method

3.2.1 Guideline Synthesis

Given a medical text document with a set of ICD codes, we synthesize a guideline by retrieving relevant knowledge associated with each ICD code assigned to this document. This guideline can assist the model in learning to localize evidence fragments and make accurate code predictions.

Description parsing. Given document-level annotations $\{y_i\}, y_i \in \{0, 1\}$, we can extract the set of ICD codes present in the document, referred to as the positive code set. Let the full code set be denoted as $\{c_1, ..., c_n\}$, and the positive code set be represented as:

$$C_p = \{c_i | y_i = 1\}, \tag{4}$$

Since each code c_i has an offical description in ICD-9, it can be denoted as Description(c_i). We can easily retrieve the descriptions of these assigned ICD codes in the positive code set:

$$D_p = \{\text{Description}(c_i) | c_i \in C_p\}, \tag{5}$$

which can be used to build the synthetic guideline. We remove the term "NOS" (Not Otherwise Specified) to standardize expressions.

Synonym replacement. To enhance the diversity of synthetic samples and enrich the representation of each code, we incorporate synonyms (Yuan et al., 2022) derived from biomedical knowledge bases. For instance, code 401.9 in ICD-9 is defined as "unspecified essential hypertension", but may be referred to in alternative terminologies such as "primary hypertension" or "hypertension nos." These variations can be systematically identified within the Unified Medical Language System (UMLS) (Bodenreider, 2004), a structured repository of biomedical terminologies that provides multiple synonymous expressions for all ICD codes.

We first map each ICD code to its corresponding Concept Unique Identifier (CUI) and extract the English synonyms associated with the same CUI. For a specific code c_i with multiple synonyms, we randomly sample one of these synonyms, i.e.,

$$s_i = \text{Synonym}(c_i) \sim \{s_i^1, s_i^2, ..., s_i^k\}, \tag{6}$$

where s_i^k is the k-th synonym. Then we replace the code descriptions with these sampled synonyms to obtain:

$$S_p = \{s_i | c_i \in C_p\}. \tag{7}$$

This synonym substitution strategy facilitates diverse and robust code representation and enhances the adaptability to real-world medical texts.

Hierarchy retrieve. Another important source of prior knowledge is the hierarchical relationships between ICD codes. For example, code 038.9 ("unspecified septicemia") belongs to code groups 030-041 ("other bacterial diseases") and 001-139 ("infectious and parasitic diseases"), which include many similar but distinct codes. The hierarchical information of a code can be defined as Hierarchy(c_i), which contains the descriptions of all the groups to which this ICD code belongs.

While code hierarchy knowledge is commonly incorporated by designing graph neural networks with predefined adjacency matrices, we assume that the language model can adaptively retrieve semantic information from the complete descriptions. This is achieved by simply adding all hierarchical knowledge to the guideline as:

$$H_p = \{\text{Hierarchy}(c_i) | c_i \in C_p\}. \tag{8}$$

Shuffle and concatenate. Next, we shuffle the order of the assigned codes C_p, replace them with their descriptions and hierarchical descriptions, and concatenate them to form a long string sequence $\hat{x}$.

Thus, for each training sample (x, y), we generate a synthetic guideline $\hat{x}$ that encapsulates the relevant knowledge of the ICD codes assigned to the raw training sample.

3.2.2 Multi-task Learning

Retrieve and prediction from raw text. In an ordinary setting, the ICD coding model makes a binary prediction based on the raw clinical document as:

$$L_{raw} = L_{BCE}(f(x), y), \tag{9}$$

where x is the medical document, and y is the binary vector whose dimension equals the total number of ICD codes. The predictions are supervised by the binary labels using cross-entropy loss:

$$L_{BCE} = -\frac{1}{C}\sum_{i=1}^{C}(y_i \log p_i + (1 - y_i)\log(1 - p_i)), \tag{10}$$

where C is the total number of ICD-9 codes and i refers to the dimension of the predicted vector and ground truth vector.

Retrieve and prediction from guideline. Given that the guideline encapsulates all the semantic information of the assigned ICD codes, the model is guided to retrieve code-specific details and predict the corresponding ICD codes. We employ

$$L_{guide} = L_{BCE}(f(\hat{x}), y), \tag{11}$$

to achieve this goal. This guideline, free from noisy content such as social and family history, simplifies the assignment of ICD codes and facilitates smoother learning for the ICD coding model.

Semantic similarity constraint: We apply a similarity loss function to enforce consistency between the code-specific representations aggregated from the raw sample and its corresponding guideline. Only the assigned ICD codes are considered, using the binary ground truth vector to select the aggregated vector of these positive codes by:

$$E = y \odot E \tag{12}$$

$$\hat{E} = y \odot \hat{E}, \tag{13}$$

where $E \in R^{C \times D}$ and $\hat{E} \in R^{C \times D}$ are code-specific representations obtained by the ICD coding model elaborated in Section 3.3. Then, we compute the similarity between each of the two retrieved features: one from a normal clinical document, and the other from the guideline, as the loss function:

$$L_{sim} = 1 - cosine(E, \hat{E}), \tag{14}$$

to make them consistent in the semantic space.

The total optimization function can be formulated as:

$$L = L_{raw}(x, y) + L_{guide}(\hat{x}, y) + \lambda L_{sim}(E, \hat{E}), \tag{15}$$

where λ is a coefficient to control the similarity, considering the gap between theoretical code knowledge and clinical code expressions.

3.3 Model Architecture

Following PLM-CA (Edin et al., 2024), our model comprises an encoder and a decoder. The encoder transforms a sequence of N tokens into a sequence of contextualized token representations $H \in \mathbb{R}^{N \times D}$. We use RoBERTa-PM (Lewis et al., 2020), a transformer pre-trained on PubMed articles and clinical notes, as the encoder. However, the length of clinical documents is larger than the max input length of RoBERTa-PM, so we chunk the raw document text into pieces, feed them into the PLM separately, and concatenate them along

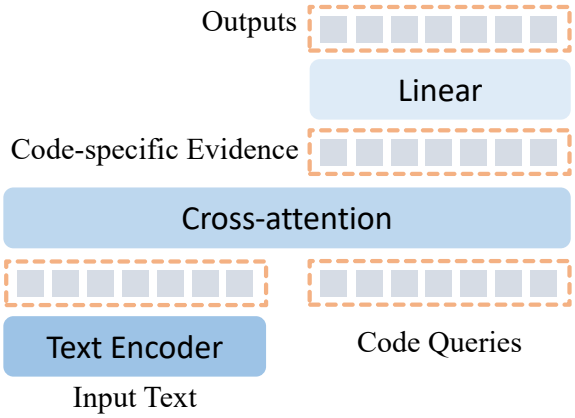

Figure 3: The model architecture adopted in our work.

with the axis of length in feature space. For simplicity, we describe this process as:

$$\mathbf{H} = \text{PLM}(x). \tag{16}$$

After obtaining the contextual representations of the input text, we use a standard cross attention to aggregate information for different ICD codes. The code-specific evidence E_i can be obtained by:

$$A_i = \text{softmax}(Q_i(HW_k))^T), \tag{17}$$
$$E_i = \text{layernorm}(A_i(HW_v)), \tag{18}$$

where $Q_i \in R^D$ is the learnable code query of the i-th ICD code, $A_i \in \mathbb{R}^{C\times N}$ is the attention matrix from the i-th code to the input text, and W_k, $W_v \in \mathbb{R}^{D\times D}$ are the linear transform matrices.

Based on the aggregated evidence of the i-th ICD code, a linear classifier is applied to compute the predicted probability for i-th ICD code:

$$\hat{y}_i = \text{sigmoid}(E_i W_i), \tag{19}$$

where $W_i \in \mathbb{R}^D$ is an independent linear classifier applied to the i-th ICD code.

4 Experiments

4.1 Experiment Setting

Dataset	N_{Train}	N_{Dev}	N_{Test}	N_{Codes}
Full	47,723	1,631	3,372	8,929
Top-50	8,066	1,573	1,729	50

Table 1: Statistics of MIMIC-III Dataset Splits. N_{Train}, N_{Dev} and N_{Test} refer to the number of samples in the train, development and test split. N_{Codes} refers to the number of unique ICD codes in the whole dataset.

Dataset. We use the MIMIC-III dataset (Johnson et al., 2016), which is the largest publicly available clinical dataset. We follow the experimental setting of Mullenbach et al. (2018) to form MIMIC-III-Full and MIMIC-III-Top-50. The statistical data for the two datasets are presented in Table 1. Following the setting of Edin et al. (2024), we train and test the models on raw text, only truncating all documents to a maximum of 8,192 tokens without any other pre-processing.

Evaluation metrics. Following previous work (Mullenbach et al., 2018), we evaluate our method using both macro and micro F1 and AUC metrics, mean average precision (MAP), and precision at K (P@K) that indicates the proportion of the correctly predicted labels in the top-K predictions. For MIMIC-III-Full Dataset, we set K as 8, 15, while for the MIMIC-III-Top-50 Dataset, we set K as 5.

Implementation details. We implement our model in PyTorch (Paszke et al., 2019) on a single NVIDIA H20 96G GPU. We use the Adam optimizer and the learning rate is initialized to 5e-5. We train the model for 12 epochs, the learning rate increases in the first 2000 steps, and then decays linearly in the further steps. The batch size is 8, which indicates that there are 8 raw samples and 8 synthetic guidelines in a batch in our proposed framework. We initialize each code query with its ICD description by encoding the text and employing a maximum pooling, inspired by Wang et al. (2018). We use R-Drop (Wu et al., 2021) regularization techniques to alleviate overfitting, and set α as 5 for MIMIC-III-Full Dataset and 10 for MIMIC-III-Top-50 Dataset.

4.2 Comparison with SOTA models

To demonstrate the superiority of proposed GKI-ICD framework, we compare it with the state-of-the-art methods for ICD coding.

Methods without knowledge. CAML (Mullenbach et al., 2018) is a CNN-based model, which is the first work to propose explainable ICD coding; PLM-ICD (Huang et al., 2022) and PLM-CA (Edin et al., 2024) are transformer-based models, which are popular these years in ICD coding.

Methods with extra knowledge. MSATT-KG (Xie et al., 2019) captures code hierarchical relationships with graph neural networks; MSMN (Yuan et al., 2022) proposes multi-synonym-attention to learn diverse code representations; KEPTLongformer (Yang et al., 2022) adds the description of each ICD code to a long prompt;

Models	MIMIC-III-Full						MIMIC-III-Top-50				
	AUC		F1		P@K		AUC		F1		P@5
	Macro	Micro	Macro	Micro	P@8	P@15	Macro	Micro	Macro	Micro	
CAML (Mullenbach et al., 2018)	0.895	0.986	0.088	0.539	0.709	0.561	0.875	0.909	0.532	0.614	0.609
MSATT-KG (Xie et al., 2019)	0.910	0.992	0.090	0.553	0.728	0.581	0.914	0.936	0.638	0.684	0.644
MSMN (Yuan et al., 2022)	0.950	0.992	0.103	0.584	0.752	0.599	0.928	0.947	0.683	0.725	0.680
KEPTLongformer (Yang et al., 2022)	-	-	0.118	0.599	0.771	0.615	0.926	0.947	0.689	0.728	0.672
PLM-ICD (Huang et al., 2022)	0.926	0.989	0.104	0.598	0.771	0.613	0.910	0.934	0.663	0.719	0.660
PLM-CA (Edin et al., 2024)	0.916	0.989	0.103	0.599	0.772	0.616	0.916	0.936	0.671	0.710	0.664
CoRelation (Luo et al., 2024)	0.952	0.992	0.102	0.591	0.762	0.607	0.933	0.951	**0.693**	0.731	**0.683**
GKI-ICD (Ours)	**0.962**	**0.993**	**0.123**	**0.612**	**0.777**	**0.624**	**0.933**	**0.952**	0.692	**0.735**	0.681
MRR (Wang et al., 2024a)	0.949	0.995	0.114	0.603	0.775	0.623	0.927	0.947	0.687	0.732	0.685
AKIL (Wang et al., 2024b)	0.948	0.994	0.112	0.605	0.784	0.637	0.928	0.950	0.692	0.734	0.683

Table 2: Comparison with previous SOTA methods. Note that MRR and AKIL rely on DRG codes, CPT codes and medications, which are additionally annotated to each sample by human coders. We list these methods for reference although directly comparing our method with them is unfair.

CoRelation (Luo et al., 2024) integrates context, synonyms and code relationships to enhance the learning of ICD code representations.

Methods with additional human annotated data. AKIL (Wang et al., 2024b) and MRR (Wang et al., 2024a) improve ICD coding using additional human annotations, e.g., DRG codes, and CPT codes. Although directly comparing our methods with them is unfair, we list them for reference.

Methods using LLMs. Currently, LLMs under zero-shot prompting perform worse than fine-tuned PLMs on ICD coding tasks, according to Boyle et al. (2023). To our knowledge, no published work has applied fine-tuned LLMs to ICD coding to achieve comparable performance to PLMs.

Table 2 shows the quantitative results of these approaches on MIMIC-III-Full and MIMIC-III-Top-50. Our method outperforms state-of-the-arts significantly on all evaluation metrics. Specifically, compared with PLM-CA, on whose basis our model builds, our method obtains 4.6% improvement on MacroAUC and 2.0% improvement on MicroAUC, respectively, on MIMIC-III-Full. It also obtains 1.7% gains on Macro AUC and 2.6% gains on Micro AUC on MIMIC-III-Top-50, which only considers the most common ICD codes in MIMIC-III-Full. Moreover, even compared with methods that rely on extra annotated inputs, e.g., AKIL and MRR, our method shows comparable performance and is even better on many metrics. The improvement shows the effectiveness of GKI-ICD for using knowledge-based synthetic data to guide the learning process, and further verifies that jointly using the real samples and synthetic samples can obtain more accuracy.

Models	AUC		F1		MAP
	Macro	Micro	Macro	Micro	
w/o knowledge	0.917	0.989	0.109	0.606	0.653
w/ desc	0.960	0.993	0.118	0.609	0.658
w/ desc + syn	0.962	0.993	0.123	0.611	0.660
w/ desc + hie	0.962	0.993	0.123	0.611	0.661
w/ desc + syn + hie	**0.962**	**0.993**	**0.123**	**0.612**	**0.661**

Table 3: Ablation of multiple knowledge injection on MIMIC-III-Full Dataset. The abbreviations "desc", "syn", "hie" stand for description knowledge, synonym knowledge and hierarchy knowledge, respectively. We apply our proposed knowledge injection training framework to the baseline model, and add different types of ICD code knowledge. Different from PLM-CA, all these models use R-drop regularization techniques and truncate input text into 8192 tokens, not 6144 tokens.

4.3 Ablation Study

We conduct extensive ablation studies on MIMIC-III-Full dataset to verify the effectiveness of each component of our method.

Effectiveness of proposed knowledge injection training framework. To address the challenges of long-tailed distribution and missing annotations, GKI-ICD injects knowledge through synthetic sample generation and multi-task learning. As shown in Table 3, after incorporating any type of ICD code knowledge, the model demonstrates improvements across various evaluation metrics, highlighting the effectiveness of GKI-ICD and the importance of knowledge infusion. Furthermore, the model's performance is further enhanced by integrating all kinds of knowledge, demonstrating the compatibility of our approach with diverse types of knowledge and its potential for broader applications.

Effectiveness of integrating multiple types of ICD code knowledge. The impact of integrating multiple types of ICD code knowledge is explored.

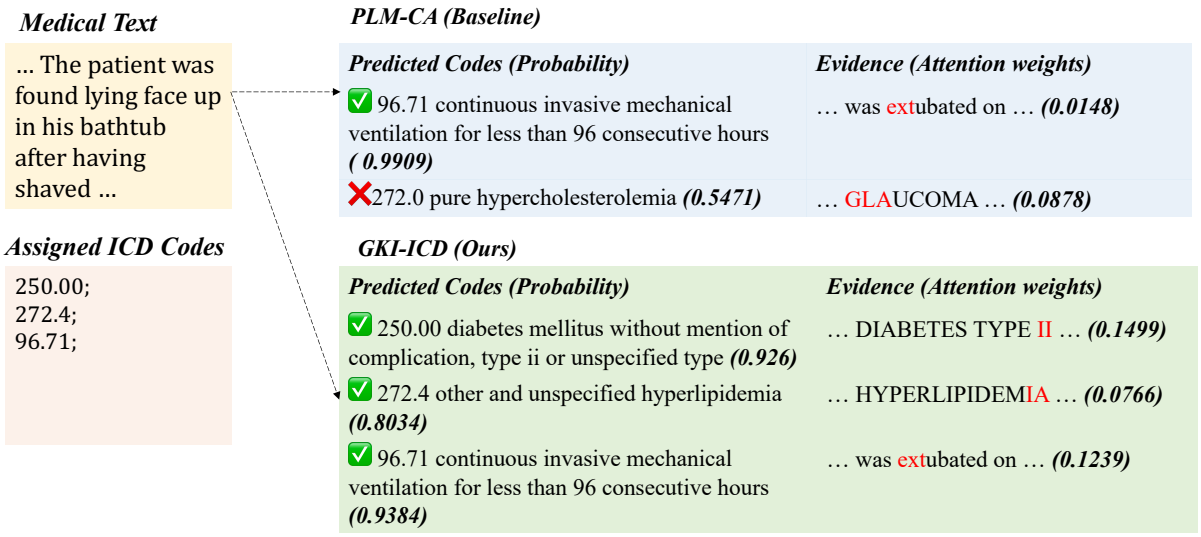

Figure 4: Case Study on MIMIC-III-Top-50 Dataset. We visualize the predicted ICD codes and the retrieved evidence of PLM-CA and our method. The red means the token which gains the greatest attention weight.

Code Frequency	PLM-CA	GKI-ICD
>500	0.684	0.687
101-500	0.508	0.509
51-100	0.413	0.420
11-50	0.293	0.322
1-10	0.029	0.132

Table 4: Comparison of F1-scores of PLM-CA and GKI-ICD on different code groups on MIMIC-III-Full Dataset.

In addition to ICD code definitions, we incorporate synonym knowledge from a medical knowledge graph and hierarchy knowledge defined in ICD-9 system. These additional knowledge sources can be seamlessly integrated into GKI-ICD framework as supplementary information. As shown in Table 3, incorporating richer knowledge enhances the ICD coding performance. This improvement highlights the importance of leveraging diverse and structured medical knowledge to better capture the semantic and relational nuances of ICD codes, leading to more accurate and robust predictions.

4.4 Effectiveness on Rare Codes

We classify the ICD codes into groups based on their frequencies in the training set, and test the F1 scores on different groups separately. As shown in Table 4, GKI-ICD leads to improved accuracy across all code groups. Specifically, for rare codes (occurrence <= 10), GKI-ICD demonstrates an improvement of 0.103 micro-F1 score over PLM-CA, highlighting its superior capability in handling rare codes, as well as its potential to address other long-tailed distribution problem.

4.5 Case Study

We visualize an example from the test set, as shown in Figure 4, comparing the attention weights and predictions before and after knowledge injection. Before knowledge injection, only half of the codes are correctly predicted by the model, and the evidence of the false positive code "272.0" is totally irrelevant to this code. However, after knowledge injection, the predicted codes are the same as the ground truth. Notably, the model pays attention to "Diabetes Type II", which is specially mentioned in the description of code "250.00". Moreover, the model pays more attention to the word "extubation", which is related to code "96.71", compared to the baseline. These changes substantiate the efficacy of knowledge injection.

5 Conclusion

In this paper, we propose GKI-ICD, a novel, general knowledge injection framework, which integrates multiple kinds of ICD code knowledge for guideline synthesis and inject code knowledge to the ICD coding model via multi-task learning. Experimental results demonstrate that our proposed method outperforms the baseline models and is even comparable to models relying on extra human annotations. In addition, our framework does not make any changes to model architecture, thus being easy to be applied to other multi-label classification problems, using label-specific knowledge to improve the performance on rare labels.

Limitations

Our proposed general knowledge injection framework, while offering an effective approach for the injection of knowledge to improve ICD coding performance, has notable limitations. First, it focuses on the ICD-9 code system, which, though widely used in prior research, is outdated compared to the more comprehensive ICD-10 system (e.g., over 70,000 diagnosis codes in ICD-10-CM vs. 14,000 in ICD-9). Future work should adapt our approach to ICD-10. Second, our framework does not incorporate the Alphabetic Index, a key tool in ICD coding. Coders use the Alphabetic Index to map clinical terms to a set of candidates before assigning the final ICD codes, ensuring accurate ICD coding. Future work should also integrate the Alphabetic Index.

Ethics Statement

We use the publicly available clinical dataset MIMIC-III, which contains de-identified patient information. We do not see any ethics issues here in this paper.

Acknowledgement

Supported by the National Natural Science Foundation of China under Grant 62271465, the Suzhou Basic Research Program under Grant SYG202338, and the China Postdoctoral Science Foundation under Grant 2024M763178.

References

Olivier Bodenreider. 2004. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270.

Joseph Boyle, Antanas Kascenas, Pat Lok, Maria Liakata, and Alison O'Neil. 2023. Automated clinical coding using off-the-shelf large language models. In *Deep Generative Models for Health Workshop NeurIPS 2023*.

Joakim Edin, Alexander Junge, Jakob D Havtorn, Lasse Borgholt, Maria Maistro, Tuukka Ruotsalo, and Lars Maaløe. 2023. Automated medical coding on mimic-iii and mimic-iv: a critical review and replicability study. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2572–2582.

Joakim Edin, Maria Maistro, Lars Maaløe, Lasse Borgholt, Jakob Drachmann Havtorn, and Tuukka Ruotsalo. 2024. An unsupervised approach to achieve supervised-level explainability in healthcare records. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4869–4890, Miami, Florida, USA. Association for Computational Linguistics.

Xueren Ge, Abhishek Satpathy, Ronald Dean Williams, John Stankovic, and Homa Alemzadeh. 2024. DKEC: Domain knowledge enhanced multi-label classification for diagnosis prediction. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12798–12813, Miami, Florida, USA. Association for Computational Linguistics.

Goncalo Gomes, Isabel Coutinho, and Bruno Martins. 2024. Accurate and well-calibrated ICD code assignment through attention over diverse label embeddings. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2302–2315, St. Julian's, Malta. Association for Computational Linguistics.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23.

Chao-Wei Huang, Shang-Chi Tsai, and Yun-Nung Chen. 2022. PLM-ICD: Automatic ICD coding with pretrained language models. In *Proceedings of the 4th Clinical Natural Language Processing Workshop*, pages 10–20, Seattle, WA. Association for Computational Linguistics.

Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. 2019. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *arXiv preprint arXiv:1904.05342*.

Shaoxiong Ji, Xiaobo Li, Wei Sun, Hang Dong, Ara Taalas, Yijia Zhang, Honghan Wu, Esa Pitkänen, and Pekka Marttinen. 2024. A unified review of deep learning for automated medical coding. *ACM Computing Surveys*, 56(12):1–41.

Alistair EW 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(1):1–9.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Patrick Lewis, Myle Ott, Jingfei Du, and Veselin Stoyanov. 2020. Pretrained language models for biomedical and clinical tasks: understanding and extending the state-of-the-art. In *Proceedings of the 3rd clinical natural language processing workshop*, pages 146–157.

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.

Yang Liu, Hua Cheng, Russell Klopfer, Matthew R. Gormley, and Thomas Schaaf. 2021. Effective convolutional attention network for multi-label clinical document classification. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5941–5953, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Junyu Luo, Xiaochen Wang, Jiaqi Wang, Aofei Chang, Yaqing Wang, and Fenglong Ma. 2024. CoRelation: Boosting automatic ICD coding through contextualized code relation learning. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 3997–4007, Torino, Italia. ELRA and ICCL.

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 (Long Papers)*, pages 1101–1111, New Orleans, Louisiana. Association for Computational Linguistics.

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. *Advances in neural information processing systems*, 32.

Thanh Vu, Dat Quoc Nguyen, and Anthony Nguyen. 2021. A label attention model for icd coding from clinical text. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, IJCAI'20.

Guoyin Wang, Chunyuan Li, Wenlin Wang, Yizhe Zhang, Dinghan Shen, Xinyuan Zhang, Ricardo Henao, and Lawrence Carin. 2018. Joint embedding of words and labels for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2321–2331, Melbourne, Australia. Association for Computational Linguistics.

Xindi Wang, Robert Mercer, and Frank Rudzicz. 2024a. Multi-stage retrieve and re-rank model for automatic medical coding recommendation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4881–4891, Mexico City, Mexico. Association for Computational Linguistics.

Xindi Wang, Robert E. Mercer, and Frank Rudzicz. 2024b. Auxiliary knowledge-induced learning for automatic multi-label medical document classification. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2006–2016, Torino, Italia. ELRA and ICCL.

Lijun Wu, Juntao Li, Yue Wang, Qi Meng, Tao Qin, Wei Chen, Min Zhang, Tie-Yan Liu, et al. 2021. R-drop: Regularized dropout for neural networks. *Advances in neural information processing systems*, 34:10890–10905.

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.

Zhichao Yang, Shufan Wang, Bhanu Pratap Singh Rawat, Avijit Mitra, and Hong Yu. 2022. Knowledge injected prompt based fine-tuning for multi-label few-shot ICD coding. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1767–1781, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Zheng Yuan, Chuanqi Tan, and Songfang Huang. 2022. Code synonyms do matter: Multiple synonyms matching network for automatic ICD coding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 808–814, Dublin, Ireland. Association for Computational Linguistics.

Tong Zhou, Pengfei Cao, Yubo Chen, Kang Liu, Jun Zhao, Kun Niu, Weifeng Chong, and Shengping Liu. 2021. Automatic ICD coding via interactive shared representation networks with self-distillation mechanism. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5948–5957, Online. Association for Computational Linguistics.