Automatic Mapping of Clinical Classification Systems Using Large Language Models

Anonymous ACL submission

Abstract

Mapping clinical classification systems, such as the International Classification of Diseases (ICD) is crucial for data analysis but is manually intensive and not scalable. We identified two key issues with the standard automatic methods using transformer-based pre-trained encoders: (1) *linguistic variation* and (2) *varying granular details across ICD versions*. To address these issues, we propose a novel method by leveraging the representational capacity of pre-trained encoders and the reasoning abilities of the large language models (LLMs). For each ICD code, we generate: (1) *hierarchy-augmented* and (2) *LLM-generated* descriptions to capture rich semantic nuances, addressing linguistic variation. Furthermore, we leverage the reasoning ability of the LLM to generate the final maps where the source code has been mapped to a parent code, using a *multiple-choice* style prompts. Empirically, we demonstrate the effectiveness of the proposed method by performing *chapter-wise* mapping between ICD-9-CM (Clinical Modification) and ICD-10-CM (Clinical Modification) and ICD-10-AM (Australian Modification) and ICD-11. Our source code is publicly available at:[github link on camera-ready version].

1 Introduction

Disease classification systems such as International Classification of Diseases (ICD) provide standardised codes for diseases and health conditions, facilitating accurate communication, reporting and analysis of healthcare data globally. Clinical classification systems evolve over time into new versions, such as ICD-9, ICD-10, and the most recent ICD-11. In addition, countries often adapt these base classifications for local use, creating national extensions such as Germany's ICD-10-CM (Clinical Modification) and Australia's ICD-10-AM (Australian Modification). These continuous updates and modifications require the development of mapping tables between classification systems to ensure that previously coded data remain consistent and suitable for longitudinal analysis.

These mapping tables are typically constructed manually by domain experts, which is time consuming and not easily scalable. Although some automatic mapping approaches have been proposed, progress remains very limited. Most existing methods rely on *name-based* techniques (e.g. string matching) or *lexical-based* strategies (e.g. exploiting lexical variations and synonym generation) (Allones et al., 2014; Huang et al., 2009; Wang et al., 2008). However, since these approaches are developed primarily for *text-to-concept*[1] mapping, their effectiveness in *concept-to-concept* mapping, such as mapping between ICD versions, remains unclear. Moreover, the lack of implementation details further complicates the evaluation of their suitability for ICD version mapping.

Transformer-based encoder models (e.g., BERT (Devlin et al., 2019)) have emerged as powerful tools for generating discriminative dense representations for texts. A straightforward approach leverages these pre-trained models to project ICD code descriptions (source and target) into a shared embedding space, generating potential mappings based on similarity metrics such as *cosine similarity*. While this method yields promising results (see Appendix C), we identify two key limitations when mapping across ICD versions: (1) *linguistic variation* (e.g., synonyms) and (2) *varying granular detail across ICD versions*.

Given the strong reasoning capabilities of decoder-only large language models (LLMs) numerous methods have been proposed for generating text embeddings using these pre-trained models. These approaches generally fall into two categories: (1) *tuning-free* methods (Jiang et al., 2024; Lei et al., 2024; Thirukovalluru and Dhingra, 2024;

[1]It involves mapping any clinical term to a terminology system, e.g Systemized Nomenclature of Medicine – Clinical Terms (SNOMED-CT)

Zhang et al., 2024) and (2) *tuning-based* methods (Li et al., 2024; Muennighoff et al., 2024; Ji et al., 2025). Both these methods rely on manually crafted prompts and typically use the final hidden state of the last token (e.g., the [EOS] or end-of-sequence token) as the text embedding. Tuning-based methods refine these embeddings further using the *InfoNCE* (Oord et al., 2018) loss to enhance alignment in the embedding space. On the one hand, tuning-free methods are easy to use, but often produce poor embeddings for ICD code descriptions (see Appendix D). On the other hand, tuning-based methods may yield better results but require a complex and resource-intensive training procedure.

To this end, we propose an automatic mapping approach that combines the representation capabilities of pre-trained encoders with the reasoning abilities of large language models (LLMs). For each ICD code description, we (1) *generate a hierarchy-augmented description* and (2) *prompt a pre-trained LLM to produce a concise clinical description*. We encode these two descriptions separately, using a pre-trained encoder model, and take their mean as the final embedding.

To address variation in the level of detail across ICD code descriptions and ensure accurate mapping, we further leverage the reasoning capabilities of LLMs through a prompting framework. In particular, we create a prompt in *multiple-choice* question format, asking the LLM to find the best match for a given source code description from a list of target code descriptions. The prompt also includes a set of manually defined rules, which the model must follow when making decisions. It is important to note that the proposed method does not require any task-specific training or fine-tuning. It is model-agnostic and can be applied using any suitable pre-trained models.

Empirically, we show the effectiveness of the proposed method by mapping different ICD versions, namely ICD-9-CM and ICD-10-CM, and ICD-10-AM and ICD-11. In this work, we opted for *chapter-wise* mapping. We used the equivalent chapters of source and target ICD versions and mapped the codes. Likewise, we restrict our approach to *one-to-one* mapping, i.e. one source code is mapped to one target code. However, if the source concept is broader in meaning than the target concepts, the union of more than one target concept approximates the source concept more closely than the individual target codes (*one-to-many*). In such cases, any partial match is considered complete.

Our main contributions are:

1. We propose an automatic mapping technique to map different ICD version. The proposed method doesn't require any training (or fine-tuning), and doesn't rely on a specific family of pre-trained models.

2. Empirically, we demonstrate the effectiveness of the proposed method by *chapter-wise* mapping between ICD-9-CM and ICD-10-CM, as well as ICD-10-AM and ICD-11, in both directions.

2 Background

The International Classification of Diseases (ICD) is a hierarchical system that organises clinical conditions into chapters, blocks, and groups based on various characteristics, such as affected body systems or causative agents. Figure 1 illustrates an example of the code structure in the eleventh revision of the ICD (ICD-11). Each condition is assigned a unique code with a brief description summarising the clinical condition. Maintained by the World Health Organisation (WHO), the ICD is periodically updated to reflect the advances in medical science and clinical practice. As a result, health data gets encoded using different ICD versions over time, necessitating mapping tables to align historical data and support longitudinal analysis.

```
Certain Infectious or Parasitic Diseases
    Gastorenteritis or Colitis of Infecious Origin
        Bacterial Intestinal Infections
            1A00 Cholera
            1A01 Intestinal Infection due to Other Vibrio
            ...

        Bacterial Foodborne Intoxications
            1A10 Foodborne Staphylococcal Intoxication
            1A11 Botulism
                1A11.0 Foodborne intoxication by ...
                1A11.1 Other forms of Botulism
                1A11.Z Botulism, unspecified
            ...
```

Figure 1: Each clinical condition in ICD-11 is assigned a unique alphanumeric code along with a corresponding description. For example, *1A00* is the ICD-11 code for *Cholera*. The ICD-11 hierarchy is organized into multiple levels of specificity. In this case, *1A00* falls under the broader categories: *Certain infectious or parasitic diseases* → *Gastroenteritis or colitis of infectious origin* → *Bacterial intestinal infections*.

The different ICD versions are not directly comparable. For example, ICD-9 codes are mostly

numeric, whereas ICD-10 codes are alphanumeric. ICD-11 codes are also alphanumeric, but they use completely different structures compared to ICD-10 codes. Thus, it is not possible to directly compare the codes to find equivalent code in the target system.

While mapping between ICD versions is still predominantly a manual task performed by trained professionals with limited progress in automation, recent advancements, particularly in transformer-based encoders for representation learning, offer promising avenues. These models can generate high-quality, discriminative embeddings that capture semantic relationships, placing similar words in close proximity within the embedding space due to their distributional properties. One straightforward automatic mapping approach leverages this by projecting source and target ICD codes into a shared embedding space and then identifying mappings based on a similarity metric such as cosine similarity. We identified the following two key challenges for implementing such automatic mapping approaches:

2.1 Linguistic variation.

The different ICD versions may use varying clinical terms (code descriptions) to describe the same condition. Table 1 illustrates some examples of equivalent ICD-9-CM and ICD-10-CM codes that differ in linguistic structure. These terms are typically short and contain specialised vocabulary. As a result, due to limited contextual information, pre-trained encoders often struggle to generate embeddings that accurately capture their semantic meanings.

ICD-9-CM	ICD-10-CM
Madura foot [0394]	Mycetoma unspecified [B479]
Ornithosis with pneumonia [0730]	Chlamydia psittaci infection [A70]
Herpangina [0740]	Enteroviral vesicular pharyngitis [B085]
Condyloma acuminatum [07811]	Anogenital (venereal) warts [A630]
Toxocariasis [1280]	Visceral larva migrans [B830]
Pneumoconiosis due to other inorganic dust [503]	Stannosis [J635]

Table 1: ICD-9-CM and ICD-10-CM equivalent codes but with different linguistic structures.

Source	Target
0068 Amebic infection of other sites [ICD-9-CM]	A068 Amebic infection of other sites [ICD-10-CM] A0681 Amebic cystitis A0682 Other amebic genitourinary infections A0689 Other amebic infections*
0330 Whooping cough due to bordetella pertussis [ICD-9-CM]	A370 Whooping cough due to Bordetella pertussis [ICD-10-CM] A3700 Whooping cough due to Bordetella pertussis without pneumonia * A3700 Whooping cough due to Bordetella pertussis with pneumonia
11289 Other candidiasis of other specified sites [ICD-9-CM]	B378 Candidiasis of other sites [ICD-10-CM] B3781 Candidal esophagitis B3782 Candidal enteritis B3783 Candidal cheilitis B3784 Candidal otitis externa B3789 Other sites of candidiasis *
A483 Toxic shock syndrome [ICD-10-AM]	1C45 Toxic shock syndrome [ICD-11] 1C450 Streptococcal toxic shock syndrome 1C451 Staphylococcal toxic shock syndrome 1C45Y Toxic shock syndrome due to other specified infectious agent 1C45Z Toxic shock syndrome without specified infectious agent *
B560 Gambiense trypanosomiasis [ICD-10-AM]	1F510 Gambiense trypanosomiasis, [ICD-11] 1F5100 Meningitis in gambiense trypanosomiasis 1F510Y Other specified gambiense trypanosomiasis 1F510Z Gambiense trypanosomiasis, unspecified *

Table 2: Examples of cases where source and target code descriptions are similar, but the target system defines the clinical condition in more granular sub-codes—with $*$ indicating the actual mapped target code.

2.2 Varying granular detail in clinical conditions.

Newer ICD versions are often more specialised than the previous versions and hence may define certain clinical conditions at a more granular level, incorporating distinctions based on specific causative agents or the presence or absence of complications. When mapping to a more specialised ICD version, parent codes in the target system sometimes share similar descriptions with codes in the source system (see Table 2 for examples). Consequently, when relying exclusively on code descriptions, the resulting embeddings for these terms exhibit a high degree of similarity. As a result, these source codes are more likely to get mapped to the parent target code, which is much broader in meaning than the source code.

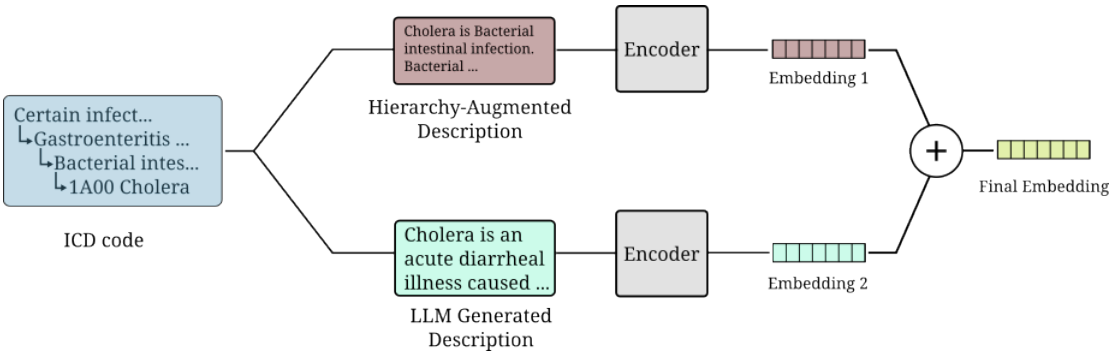

Figure 2: The overall process of generating dense representations for the ICD code descriptions. For each ICD codes, we generate: (1) a hierarchy-augmented description and (2) a concise description generated using a pre-trained LLM. Each descriptions are encoded by an encoder model and we take their mean as the final embedding.

3 Method

3.1 Task Definition

We define an ICD system as a set of codes [2] and it's parent labels, i.e. $\mathcal{C}_* \in \mathcal{C} = \{(c_i, \{p_{i,j}\}_j)\}_i$, where c_i is the ith code in $\mathcal{C}_*$ and p_{ij} is the jth-level parent of c_i. Suppose $\mathcal{C}_{\text{src}}, \mathcal{C}_{\text{tgt}} \in \mathcal{C}$ be the source and target ICD versions respectively. Now, the objective of mapping ICD versions is to generate a mapping set, $\mathcal{M}_{\mathcal{C}_{\text{src}},\mathcal{C}_{\text{tgt}}} = \{(c \in \mathcal{C}_{src}, c' \in \mathcal{C}_{tgt}, s_{c,c'})\}$, where $s : \mathcal{C}_{src} \times \mathcal{C}_{tgt} \rightarrow \mathbb{R}$ is the score function that reflects the semantic similarity between two codes.

3.2 Obtaining Term Embeddings

In this work, we aim to generate the mapping set by projecting both the source and target codes into a shared embedding space, using a pre-trained transformer-based encoders and use *cosine similarity* as the score function, i.e. $\mathcal{M}_{\mathcal{C}_{\text{src}},\mathcal{C}_{\text{tgt}}} = \{(c \in \mathcal{C}_{src}, c' \in \mathcal{C}_{tgt}, s_{\boldsymbol{\mu}_c,\boldsymbol{\mu}_{c'}})\}$, where $\boldsymbol{\mu}_c, \boldsymbol{\mu}_{c'} \in \mathbb{R}^{\mathcal{D}}$ are the dense representation for c and c' respectively, and $s : \mathcal{C}_{src} \times \mathcal{C}_{tgt} \rightarrow [-1, 1]$ is the *cosine similarity* between c and c'. While the pre-trained encoders yield promising results, as discussed in 2, they often fail to capture the semantic meanings of the ICD code descriptions due to the inherent linguistic variation. We aim to address this by generating short descriptions of each term. Specifically, for each code description, we generate: (1) a hierarchy-augmented description, and (2) a concise description generated using an LLM. Figure 2 illustrates the overall process of generating the embeddings.

Hierarchy-Augmented (HA) Description. The hierarchy-augmented variants utilise the structural context provided by a code's position within the ICD hierarchy—specifically its parent or ancestor codes—to clarify and enrich the meaning of a code. To construct this, we concatenate the original code description with its hierarchical labels using the "is a" relation to form a short, context-aware description as follows:

$$d^h = \text{“}[c_i] \texttt{ is } [p_{i,1}] \ldots [p_{i,j-1}] \texttt{ is } [p_{i,j}]\text{.”}, \quad (1)$$

where c_i is the ith code description and $p_{i,1}$, ..., $p_{i,j}$ are its parent labels with $p_{i,1}$ being the immediate parent. For example, using the template as shown in 1, the hierarchy-augmented description for *Cholera* in ICD-11 is: *Cholera is a Bacterial Intestinal Infection. Bacterial Intestinal Infection is a Gastroenteritis or Colitis of Infectious Origin. Gastroenteritis or Colitis of Infectious Origin is a Certain Infectious or Parasitic Disease.*

This context-aware description is then encoded using a pre-trained encoder model. In this work, we use the *Sentence-Transformer* (SBERT) (Reimers and Gurevych, 2019) model, specifically *all-mpnet-base-v2*, as the preferred encoding model.

$$\mathbf{e}^h \in \mathbb{R}^{\mathcal{D}} = \text{SBERT}(d^h), \quad (2)$$

where $\mathcal{D}$ is the dimension of the embedding space.

LLM-Generated (LG) Description. Recent studies have demonstrated that fine-tuning models on synthetic data generated by large language models (LLMs) can enhance performance across various downstream tasks, such as representation learning (Peng et al., 2024; Wang et al., 2024), fake news detection (Ma et al., 2024), and instance detection (Wagner et al., 2025). Inspired by this, we

[2] We do not use the codes themselves to generate the embeddings, but the corresponding code descriptions.

generate a concise description for each code description using an LLM via a prompting method. This is particularly effective for reducing lexical variation, as LLMs tend to produce consistent outputs for similar prompts.

We construct prompts using a template (see Figure 3) and instruct a pre-trained LLM to generate the concise description. The output is then encoded using SBERT. Finally, we compute the mean of the two embedding vectors (hierarchy-augmented and LLM-generated) to obtain the final representation. To evaluate the effectiveness of this approach, we conducted experiments using several open-source LLMs, including *LLaMA-3.1-8B-Instruct* (Lei et al., 2024), *Qwen3-8B* (Yang et al., 2025), *Mistral-7B-Instruct-v0.3*[3], and *Microsoft-Phi-4-mini-Instruct* (Abdin et al., 2024).

$$d^l = \text{LLM}(\text{Prompt}(X)) \quad (3)$$

$$\mathbf{e}^l \in \mathbb{R}^{\mathcal{D}} = \text{SBERT}(d^l), \quad (4)$$

$$\text{And finally, } \mathbf{e} = \frac{1}{2}(\mathbf{e}^h + \mathbf{e}^l) \quad (5)$$

```
Prompt(X) = Provide a concise clinical
description (max 100 words) of the condition '[X
]'. Include (if possible) common synonyms, known
 causative agents, and typically affected body
parts. Avoid bullet points.
```

Figure 3: Prompt template to generate a concise description of an ICD code description. Here X is the placeholder for the code description.

3.3 Generating Maps with Rule-Based Prompts (RP)

Given the source and the target code embeddings, the proposed method used *cosine similarity* score as the metric to find the potential maps.

$$\mathbf{t}_i^* = \arg \max_{\mathbf{s}_i \in \mathcal{S}; \mathbf{t}_j \in \mathcal{T}} \cos(\mathbf{s}_i, \mathbf{t}_j), \quad (6)$$

where $\mathcal{S} = \{\mathbf{s}_i\}_i$ and $\mathcal{T} = \{\mathbf{t}_j\}_j$ are the set of source and target embeddings respectively, and $\cos(\mathbf{a}, \mathbf{b})$ is the cosine similarity score between $\mathbf{a}$ and $\mathbf{b}$.

However, as discussed earlier (see section 2.2), some source codes get mapped to parent target codes, i.e. *source-to-parent* mapping. Often, these cases arise when mapping a less specialised version to a more specialised version. One potential alternative would be to remove all the *parent-level* codes and map only to the *leaf-nodes*. However, we identified some cases where source codes are mapped to the parent-level codes.

To mitigate these cases, the proposed method leverages the reasoning ability of an LLM. In particular, we construct a *multiple-choice-style* prompt asking the LLM to select the best option for the given source codes, from a list of target codes. Figure 4 shows an example of the prompt for ICD-9-CM code 0020 ('Typhoid fever'). The prompts also include a set of manually defined rules (see Appendix F for details on the rules) and instruct the LLM to follow these rules while selecting the best option. We use *Qwen3-8B* as it allows a hard switch to enable the model's thinking behaviour.

```
Please apply the rules below to answer the following question.
Rules:
1. Select the most specific target option that represents the closest
 clinical equivalent to the level of detail provided in the given
clinical term.
2. In cases where the given clinical term lacks specific details,
select the options that include terms like 'unspecified' or 'other
specified'.
3. Maintain consistency by selecting 'other' for 'other specified'
and 'unspecified' for 'unspecified'.
4. Take into account the clinical context of the given clinical term
and select the option that reflect common clinical manifestations or
broader categories relevant to its clinical implications.
Which of the following is the best match for 'Typhoid fever'?
A010 Typhoid fever
A0100 Typhoid fever unspecified
A0101 Typhoid meningitis
A0102 Typhoid fever with heart involvement
A0103 Typhoid pneumonia
A0104 Typhoid arthritis
A0105 Typhoid osteomyelitis
A0109 Typhoid fever with other complications
Please do not include explanations or code descriptions, just return
the code.
```

Figure 4: An example of a prompt template to select the best ICD-10-CM match for ICD-9-CM code 0020 (*Typhoid fever*) based on the provided rules.

4 Experiment Details

4.1 Dataset

We evaluated the effectiveness of the proposed methods by mapping ICD-9-CM and ICD-10-CM, as well as ICD-10-AM and ICD-11, for three different chapters: the Disease of the Digestive System, Intestinal Infectious Diseases and the Diseases of the Respiratory System (see Appendix B for details, including the particular versions and chapter details). For mappings between ICD-9-CM and ICD-10-CM (in both directions), we relied on the General Equivalence Mappings (GEMs) provided by the Centres for Medicare and Medicaid Ser-

[3] https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3

vices (CMS)[4]. Since no official mapping tables are available for ICD-10-AM to ICD-11 and vice versa, similar to Xu et al. (2022), we used a sequential approach—first, we map ICD-10-AM to ICD-10 using the mapping tables provided by the Independent Health and Aged Care Pricing Authority (IHACPA)[5], and then ICD-10 to ICD-11 using the conversion tables made available by the World Health Organisation (WHO). For all ICD versions, we included all available codes, including three- and four-digit codes[6]. Consequently, some source codes lacked a valid mapping in the ground truth, and we excluded those instances when calculating the final accuracies (see Appendix B.2 for more details).

4.2 Baseline Method

We construct the baseline method, by generating embeddings using only the ICD code descriptions. We evaluated various transformer-based encoders (see Appendix C for details) with some specifically trained on the clinical data, and general text data. We use *mean pooling* to generate a single fixed-length sentence-level representations from a variable-length *token-level* embeddings. Compared to all other models, the *Sentence-Transformer* (Reimers and Gurevych, 2019) (SBERT) (*all-mpnet-base-v2*[7]) performed significantly better, and hence we chose it as the preferred baseline encoder.

4.3 Models

To validate the effectiveness of the proposed method, we experimented with various open-source large language models (LLMs), including *Llama-3.1-8B-Instruct*, *Mistral-7B-Instruct-v0.3*, *Phi-4-mini-instruct* and *Qwen3-8B*, to generate the clinical descriptions. For the reasoning task, i.e. for rule-based prompt **RP**, we used *Qwen3-8B*. See Appendix A for the implementation details.

[4] https://www.cms.gov/medicare/coding-billing/icd-10-codes/2018-icd-10-cm-gem

[5] https://www.ihacpa.gov.au/resources/icd-10-am-and-achi-mapping-tables

[6] The three-digit codes are a general group of related conditions (or a single specific condition in some cases), and the four-digit codes represent more specific conditions that are further subdivided (in some cases) based on various features, for example, the causative agent, and with or without some complications.

[7] https://sbert.net/docs/sentence_transformer/pretrained_models.html

4.4 Evaluation Metric

For evaluation metric, we report the *Top-1* accuracy:

$$\text{Top-1 Accuracy} = \frac{C}{N - N_{nm}}, \tag{7}$$

where C is the number of correct maps, N is the total number of source codes and N_{nm} is the number of source codes that do not have any maps.

Given the inherent stochasticity of LLM-generated text, which introduces slight variations in output across multiple runs with the same prompt, and that the proposed method uses LLM-generated descriptions to generate the final representation, we adopted a strategy of multiple runs to ensure a robust evaluation of performance. Specifically, we report the mean and standard deviation of the accuracy calculated from five independent runs for each prompt.

5 Results

5.1 Main Results

Comparison with the Baseline. We present our main results, a detailed comparison of the proposed method against the baseline, in table 3 for *chapter-wise* mapping across different ICD versions. We used the Top-1 accuracy to evaluate the mapping performance. The baseline method, which generates the embeddings using only the code descriptions, exhibits consistent but lower performance across all tasks (accuracies ranging from 0.59 to 0.80). For instance, incorporating hierarchy-augmented (**HA**) and *Qwen3-8B* generated descriptions (**LG**) resulted in an average gain of approximately 5% (0.0483). This performance was further enhanced by roughly 6% (0.0575) when employing the rule-based prompting (**RP**) technique (as discussed in 3.3) for generating final maps.

Similarly, in all the cases, except for the ICD-9-CM to ICD-10-CM for the Disease of the Digestive System, the proposed method outperformed the baseline, even without the rule-based map generation step. In the case where the *Top-1* accuracy was below the baseline, the maximum performance difference was only 3% (0.03), i.e. for the *Llama-3.1-8B-Instruct*.

Consistency of Results. In table 3, we also reported the standard deviation to assess the consistency of the mapping performance across five runs. The proposed method, across all evaluated LLMs and both configurations (**HA+LG** and

		ICD-9-CM to ICD-10-CM			ICD-10-CM to ICD-9-CM			ICD-10-AM to ICD-11			ICD-11 to ICD-10-AM		
		Dig	**Inf**	**Resp**	**Dig**	**Inf**	**Resp**	**Dig**	**Inf**	**Resp**	**Dig**	**Inf**	**Resp**
Baseline		0.80 ±0.0	0.69 ±0.0	0.75 ±0.0	0.62 ±0.0	0.70 ±0.0	0.59 ±0.0	0.66 ±0.0	0.66 ±0.0	0.71 ±0.0	0.60 ±0.0	0.67 ±0.0	0.61 ±0.0
HA+LG	**Qwen3-8B**	0.79 ±0.007	0.74 ±0.004	0.76 ±0.010	0.70 ±0.005	0.77 ±0.005	0.67 ±0.005	0.67 ±0.004	0.69 ±0.005	0.76 ±0.010	0.64 ±0.005	0.72 ±0.005	0.71 ±0.011
	Llama-3.1-8B-Instruct	0.77 ±0.005	0.74 ±0.005	0.73 ±0.016	0.68 ±0.013	0.78 ±0.008	0.66 ±0.011	0.66 ±0.012	0.69 ±0.004	0.73 ±0.004	0.65 ±0.008	0.71 ±0.011	0.71 ±0.012
	Phi-4-mini-instruct	0.78 ±0.0	0.72 ±0.0	0.75 ±0.0	0.68 ±0.0	0.77 ±0.0	0.68 ±0.0	0.66 ±0.0	0.69 ±0.0	0.71 ±0.0	0.62 ±0.0	0.72 ±0.0	0.71 ±0.0
	Mistral-7B-Instruct-v0.3	0.79 ±0.0	0.72 ±0.0	0.74 ±0.0	0.68 ±0.0	0.79 ±0.0	0.66 ±0.0	0.66 ±0.0	0.69 ±0.0	0.74 ±0.0	0.62 ±0.0	0.71 ±0.0	0.66 ±0.0
HA+LG+RP	**Qwen3-8B**	**0.87** ±0.005	**0.80** ±0.004	**0.82** ±0.010	**0.80** ±0.008	0.77 ±0.005	0.73 ±0.005	**0.75** ±0.009	0.73 ±0.004	**0.80** ±0.012	0.71 ±0.004	0.79 ±0.006	0.74 ±0.017
	Llama-3.1-8B-Instruct	0.86 ±0.008	0.79 ±0.005	0.79 ±0.010	0.78 ±0.014	0.78 ±0.008	0.73 ±0.015	0.73 ±0.008	**0.74** ±0.001	0.78 ±0.006	**0.72** ±0.005	**0.80** ±0.014	0.73 ±0.016
	Phi-4-mini-instruct	**0.87** ±0.0	0.79 ±0.0	0.80 ±0.0	0.78 ±0.0	0.77 ±0.0	**0.75** ±0.0	0.74 ±0.0	0.73 ±0.0	0.78 ±0.0	**0.72** ±0.0	**0.80** ±0.0	**0.76** ±0.0
	Mistral-7B-Instruct-v0.3	0.86 ±0.0	0.78 ±0.0	0.79 ±0.0	0.79 ±0.0	**0.79** ±0.0	0.74 ±0.0	0.73 ±0.0	**0.74** ±0.0	0.78 ±0.0	0.71 ±0.0	0.79 ±0.0	0.69 ±0.0

Table 3: Comparison of the proposed method against the baseline on *chapter-wise* mapping of different ICD versions. **HA** and **LG** denote hierarchical-augmented description and LLM-generated description, respectively. **RP** denotes rule-based map generation if the target code is a parent code. **Dig**, **Inf** and **Resp** are respectively the diseases of the Digestive System, the Intestinal Infectious Diseases and the Diseases of the Respiratory System chapters. The numbers are the Mean *Top-1* Accuracies and the Standard Deviation after five runs.

HA+LG+RG), demonstrated a high degree of consistency with very low standard deviation, the maximum being only 0.017. Notably, with the default parameter values (see Appendix A) *Phi-4-mini-instruct* and *Mistral-7B-Instruct-v0.3* consistently produced identical results across all five runs, and hence resulting in a standard deviation of 0. *Qwen3-8B* and *Llama-3.1-8B-Instruct* showed slight variations across runs, and yielded a small non-zero standard deviations (ranging from 0.001 to 0.017). These low standard deviation values indicate that the performance of the proposed method is highly stable.

5.2 Discussion

As shown in table 4, using different terms (i.e. only the code descriptions, hierarchy-augmented descriptions and LLM generated descriptions) yielded comparable performance across different chapters and ICD version mapping directions. While these metrics provide an overview of Top-1 accuracies and their consistency, we conducted a qualitative analysis of the generated maps to evaluate the effectiveness of LLM-generated descriptions in capturing the linguistic variation across ICD versions. We identified cases where, despite significant vocabulary differences between source and target code descriptions, using LLM-generated descriptions enabled the successful identification of correct maps. In these cases, the correct mappings did not even rank within the top 100 predicted codes using the baseline method. For example, the target ICD-10-CM code for the ICD-9-CM code **0730** [*Ornithosis with pneumonia*] is **A70** [*Chlamydia psittaci infection*]. Using the concise clinical description generated using *Qwen3-8B*, these codes were correctly mapped. Table 5 shows the LLM-generated descriptions for these codes. See Appendix E for more examples.

		Dig	**Inf**	**Resp**
Terms-Only		0.67 ±0.0	**0.68** ±0.0	0.67 ±0.0
HA		0.66 ±0.0	**0.68** ±0.0	**0.68** ±0.0
LG	**Qwen3-8B**	**0.68** ±0.058	**0.68** ±0.02	**0.68** ±0.038
	Llama-3.1-8B-Instruct	0.64 ±0.041	0.66 ±0.023	0.63 ±0.04
	Phi-4-mini-instruct	0.65 ±0.0	0.66 ±0.0	0.65 ±0.0
	Mistral-7B-Instruct-v0.3	0.64 ±0.0	0.66 ±0.0	0.66 ±0.0

Table 4: Comparison of the ICD version mapping performance using different description types. **Terms-Only** uses only the ICD code descriptions (*Baseline)*. **HA** and **LG** use hierarchical-augmented description and LLM-generated description, respectively. Results are presented for specific chapters: Diseases of the Digestive System (**Dig**), Intestinal Infectious Diseases (**Inf**), and Diseases of the Respiratory System (**Resp**), across various ICD version mapping pairs: ICD-9-CM to ICD-10-CM, ICD-10-CM to ICD-9-CM, ICD-10-AM to ICD-11, and ICD-11 to ICD-10-AM. The numbers are Mean *Top-1* Accuracy ± Standard Deviation over five runs.

Furthermore, since the performance gain with the rule-based prompting (**RP**) depends on the number of correct *source-to-parent* (i.e. the source codes are mapped to parent target codes) mapping cases, we evaluated the effectiveness of using the

Ornithosis with pneumonia: Ornithosis with pneumonia, also known as psittacosis or parrot fever, is a zoonotic infection caused by Chlamydia psittaci. It primarily affects the respiratory system, leading to pneumonia characterized by fever, cough, and respiratory distress. The disease is transmitted through inhalation of aerosolized particles from infected birds. Commonly affected body parts include the lungs and occasionally the liver and spleen. Symptoms may range from mild flu-like illness to severe pneumonia. Diagnosis is often confirmed through serological testing or PCR. Treatment typically involves antibiotics such as doxycycline or tetracycline.
Chlamydia psittaci infection: Chlamydia psittaci infection is a zoonotic respiratory illness caused by the bacterium Chlamydia psittaci, commonly found in birds. It is also known as psittacosis or parrot fever. The infection typically affects the lungs, causing pneumonia, and may spread to other organs. Symptoms include fever, cough, headache, and muscle pain. Transmission occurs through inhalation of contaminated aerosols from infected birds. It can also cause systemic illness, particularly in immunocompromised individuals. Diagnosis is confirmed by serology or PCR, and treatment involves antibiotics such as tetracyclines.

Table 5: Examples of clinical descriptions generated by *Qwen3-8B* for ICD-9-CM code **0730** [*Ornithosis with pneumonia*] and ICD-10-CM code **A70** [*Chlamydia psittaci infection*]. These are equivalent codes as per the mapping file.

hierarchy-augmented and LLM-generated descriptions to generate the embeddings, by focusing on these cases. Table 6 shows the chapter-wise average percentage of correct source-to-parent mappings, across mapping ICD-9-CM and ICD-10-CM, and ICD-10-AM and ICD-11 versions in both directions. The result suggests that using both the descriptions, i.e. **HA + LG**, generally achieved a higher percentage of correct parent mappings than the baseline.

		Dig (%)	Inf (%)	Resp (%)
Baseline		74.4(61.8)	81.9(44.3)	74.6(26.8)
HA + LG	**Qwen3-8B**	78.4(68.9)	86.3(54.2)	77.1(23.1)
	Llama-3.1-8B-Instruct	80.3(71.0)	**89.8**(58.8)	77.8(23.4)
	Phi-4-mini	75.8(71.5)	85.5(55.0)	78.6(24.8)
	Mistral-7B-Instruct-v0.3	**81.4**(69.8)	86.3(54.2)	**82.5**(22.8)

Table 6: comparison between the baseline and the proposed method (using hierachy-augmented (**HA**) and LLM-generated (**LG**) descriptions) on *Source-to-Parent* code mappings, across various ICD version mapping pairs: ICD-9-CM to ICD-10-CM, ICD-10-CM to ICD-9-CM, ICD-10-AM to ICD-11, and ICD-11 to ICD-10-AM. The numbers are the chapter-wise average correct percentage of source-to-parent mappings, with the total number of cases in parentheses.

Likewise, table 7 shows the chapter-wise average percentage of cases where the rule-based prompting method generated correct final mappings across different LLMs. The numbers are comparable across the different LLMs, with *Phi-4-mini* having slightly better results. We analysed the "thinking" steps of the model for some cases and identified a key property in the ground-truth mappings from ICD-9-CM to ICD-10-CM: *some ICD-9-CM codes map to all of their ICD-10-CM siblings, but not to their parent codes*. For example, ICD-9-CM code

	Dig (%)	Inf (%)	Resp (%)
Qwen3-8B	77.2(54.0)	69.6(48.25)	68.9(17.85)
Llama-3.1-8B-Instruct	76.6(56.95)	70.5(53.6)	69.8(18.2)
Phi-4-mini-instruct	**77.9**(54.25)	**74.5**(50.0)	**78.2**(19.5)
Mistral-7B-Instruct-v0.3	76.7(56.75)	68.1(51.75)	73.3(18.75)

Table 7: Evaluation of rule-based prompt (**RP**) method to generate the final maps in case of *Source-to-Parent* mappings, across different ICD version mapping pairs—ICD-9-CM and ICD-10-CM, and ICD-10-AM and ICD-11 in both directions. The numbers are the chapter-wise average percentage of correct cases generated by **RP**, with the total number of correct cases in parentheses. **Dig**, **Inf** and **Resp** are respectively the diseases of the Digestive System, Intestinal Infectious Diseases and the diseases of the Respiratory System.

52107 (*Dental caries of smooth surface*) is mapped to ICD-10-CM codes **K0261** (*Dental caries on smooth surface limited to enamel*), **K0262** (*Dental caries on smooth surface penetrating into dentin*) and **K0263** (*Dental caries on smooth surface penetrating into pulp*), all of which are the child code for **K026** (*Dental caries on smooth surface*). However, our prompt, specifically Rule 1, instructed the LLM to select the single most appropriate target option representing a similar level of detail as the source code description. Consequently, the model chose the parent code when the individual child codes were more specific. For example **K026** in the above case (See Appendix G for the detail thinking steps used by *Qwen3-8B* to generate the final maps).

6 Conclusion

In this work, we proposed an automatic method for mapping different ICD versions leveraging the representational capacities of pre-trained transformer-based encoders and the reasoning capabilities of the large language models (LLMs). Specifically, to address the inherent linguistic variation across ICD versions, we generate the embeddings using: (1) the hierarchy-augmented description, and (2) the LLM-generated description. Furthermore, since these ICD versions may define some clinical conditions at different granular levels, we propose a rule-based prompting method to generate the final maps for cases where the source codes were mapped to the parent target codes. Empirically, we demonstrate the effectiveness of our proposed method by *chapter-wise* mapping of ICD-9-CM and ICD-10-CM, and ICD-10-AM and ICD-11, across three different chapters.

7 Limitations

We identified the following limitation in our work:

First, in this work, we focused only on the *one-to-one* mappings, however, it is also possible to have *one-to-many, many-to-one and many-to-many* maps. For any mapping system, it is crucial to handle all these cases. Likewise, we did not extend our experiment to ICD versions in different languages (e.g. the German Modification, the Korean Modification).

Second, several studies have shown that the LLMs are very sensitive to the input prompt (Sclar et al.; Lu et al., 2022; Pezeshkpour and Hruschka, 2023). However, in this work, we limit ourselves to a single prompting template.

Third, in this work, we attempt to capture the hierarchical information of the ICD system by generating a simple *hierarchy-augmented* description. Even though this approach is simple and produces comparable results, it is interesting to explore other techniques, for example, a hyperbolic representation method (Cao et al., 2020).

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, and 1 others. 2024. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*.

JL Allones, Diego Martinez, and Maria Taboada. 2014. Automated mapping of clinical terms into snomed-ct. an application to codify procedures in pathology. *Journal of medical systems*, 38:1–14.

Emily Alsentzer, John R Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical bert embeddings. *arXiv preprint arXiv:1904.03323*.

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, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.

Kuo-Chuan Huang, James Geller, Michael Halper, Yehoshua Perl, and Junchuan Xu. 2009. Using wordnet synonym substitution to enhance umls source integration. *Artificial intelligence in medicine*, 46(2):97–109.

Yifan Ji, Zhipeng Xu, Zhenghao Liu, Yukun Yan, Shi Yu, Yishan Li, Zhiyuan Liu, Yu Gu, Ge Yu, and Maosong Sun. 2025. Learning more effective representations for dense retrieval through deliberate thinking before search. *arXiv preprint arXiv:2502.12974*.

Ting Jiang, Shaohan Huang, Zhongzhi Luan, Deqing Wang, and Fuzhen Zhuang. 2024. Scaling sentence embeddings with large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 3182–3196, Miami, Florida, USA. Association for Computational Linguistics.

Yibin Lei, Di Wu, Tianyi Zhou, Tao Shen, Yu Cao, Chongyang Tao, and Andrew Yates. 2024. Meta-task prompting elicits embeddings from large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10141–10157, Bangkok, Thailand. Association for Computational Linguistics.

Chaofan Li, Zheng Liu, Shitao Xiao, Yingxia Shao, and Defu Lian. 2024. Llama2vec: Unsupervised adaptation of large language models for dense retrieval. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3490–3500.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Xiaoxiao Ma, Yuchen Zhang, Kaize Ding, Jian Yang, Jia Wu, and Hao Fan. 2024. On fake news detection with llm enhanced semantics mining. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 508–521.

George Michalopoulos, Yuanxin Wang, Hussam Kaka, Helen Chen, and Alexander Wong. 2021. UmlsBERT: Clinical domain knowledge augmentation of contextual embeddings using the Unified Medical Language System Metathesaurus. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1744–1753, Online. Association for Computational Linguistics.

Niklas Muennighoff, SU Hongjin, Liang Wang, Nan Yang, Furu Wei, Tao Yu, Amanpreet Singh, and Douwe Kiela. 2024. Generative representational instruction tuning. In *ICLR 2024 Workshop: How Far Are We From AGI*.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Letian Peng, Yuwei Zhang, Zilong Wang, Jayanth Srinivasa, Gaowen Liu, Zihan Wang, and Jingbo Shang. 2024. Answer is all you need: Instruction-following text embedding via answering the question. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 459–477.

Pouya Pezeshkpour and Estevam Hruschka. 2023. Large language models sensitivity to the order of options in multiple-choice questions. *arXiv preprint arXiv:2308.11483*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. In *The Twelfth International Conference on Learning Representations*.

Raghuveer Thirukovalluru and Bhuwan Dhingra. 2024. Geneol: Harnessing the generative power of llms for training-free sentence embeddings. *arXiv preprint arXiv:2410.14635*.

Stefan Sylvius Wagner, Maike Behrendt, Marc Ziegele, and Stefan Harmeling. 2025. The power of LLM-generated synthetic data for stance detection in online political discussions. In *The Thirteenth International Conference on Learning Representations*.

Guangyu Wang, Xiaohong Liu, Zhen Ying, Guoxing Yang, Zhiwei Chen, Zhiwen Liu, Min Zhang, Hongmei Yan, Yuxing Lu, Yuanxu Gao, and 1 others. 2023. Optimized glycemic control of type 2 diabetes with reinforcement learning: a proof-of-concept trial. *Nature Medicine*, 29(10):2633–2642.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Improving text embeddings with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11897–11916.

Yefeng Wang, Jon Patrick, Graeme Miller, and Julie O'Hallaran. 2008. A computational linguistics motivated mapping of icpc-2 plus to snomed ct. In *BMC medical informatics and decision making*, volume 8, pages 1–8. Springer.

Julia Xu, Kin Wah Fung, and Olivier Bodenreider. 2022. Sequential mapping–a novel approach to map from icd-10-cm to icd-11. *Studies in health technology and informatics*, 290:96.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Bowen Zhang, Kehua Chang, and Chunping Li. 2024. Simple techniques for enhancing sentence embeddings in generative language models. In *International Conference on Intelligent Computing*, pages 52–64. Springer.

A Implementation Details

Our source code is implemented in Python 3.11 and runs all the experiments on an Nvidia A30 GPU with cuda-12.6. We used Huggingface's *transformer-v4.51.3* to load the LLMs. We used *sentence-transformers-v4.1.0* to load the **SBERT** and all other encoder models. For the clinical description generation task, we used the default values for all the hyperparameters, for example, *temperature=1.0* and *do_sample=False*, and set the *max_new_tokens=512*. And for the reasoning task, we set *max_new_tokens=32768* and *enable_thinking=True* when applying the chat template.

B Dataset Details

B.1 Source

We used the ICD-9-CM (version 32) from the Centres for Medicare and Medicaid Services (CMS)[8] and the ICD-10-CM (FY22 release) from the Centers for Disease Control and Prevention (CDC)[9]. Likewise, we used the ICD-10-AM (twelfth edition) provided by the Independent Health and Aged Care Pricing Authority (IHACPA)[10]. We accessed the ICD-11 codes via the WHO API (version 2.5)[11]. It is important to note that the WHO API provides only pre-coordinated ICD-11 codes. Therefore, we used the parent codes in those cases where the source codes are mapped to the post-coordinated codes.

B.2 Chapters

We employed a chapter-wise mapping strategy, concentrating on the *Infectious Diseases*, *Diseases of the Respiratory System*, and *Diseases of the Digestive System* chapters. We used this approach to limit the search space for the potential maps. Also, we include all the three- and four-digit codes. Hence, as shown in Table 8, several codes have no maps because they are either the immediate parents

[8] https://www.cms.gov/medicare/coding-billing/icd-10-codes

[9] https://www.cdc.gov/nchs/icd/icd-10-cm/files.html

[10] https://www.ihacpa.gov.au/resources/icd-10-amachiacs-twelfth-edition

[11] https://icd.who.int/icdapi

or a broader category in the hierarchy. Additionally, this also include the number of cases where the source codes are mapped to a different target chapters.

Chapters	ICD9CM-ICD10CM		ICD10CM-ICD9CM		ICD10AM-ICD11		ICD11-ICD10AM	
	N	N_{nm}	N	N_{nm}	N	N_{nm}	N	N_{nm}
Diseases of the Digestive System	757	274	795	213	617	290	969	437
Intestinal Infectious Diseases	889	0	1158	117	921	207	1004	398
Diseases of the Respiratory System	320	93	369	72	281	74	342	136

Table 8: Total number of codes (N) and cases where there were no maps in the ground truth (N_{nm})

C Evaluation of Various BERT Models for Mapping Between ICD Versions

We evaluated multiple pre-trained BERT models to generate dense vector representations of ICD code descriptions. ClinicalBERT (Wang et al., 2023), BioClinicalBERT (Alsentzer et al., 2019), and UMLSBert (Michalopoulos et al., 2021) are trained specifically on clinical texts. *Sentence-Transformer* (SBERT) (Reimers and Gurevych, 2019) provides a set of models trained on general text to generate sentence-level embeddings. In this work, we used *all-mpnet-base-v2* as the SBERT encoder. Table 9 reports the mapping accuracy achieved by each model. Interestingly, SBERT consistently outperformed all other models.

D Evaluation of Large Language Models (LLMs) for Mapping ICD Versions

We used several open-source LLMs, each with parameters in the range of 7 to 8 billion, to generate dense representations of ICD code descriptions and evaluated their performance on mapping different ICD versions. A tuning-free approach was adopted: for each code description, we applied a template function to construct a prompt. Specifically, we used the knowledge-enhanced *promptEOL* and *promptSUM* templates, whose details are presented in Table 10. Following standard practice, all prompts were lowercase and appended with an end-of-sequence (EOS) token. We used the final hidden state corresponding to the EOS token as the final representation.

Table 11 lists the performance of various open-source LLMs, on mapping different ICD versions. Although LLMs have significantly more parameters and are trained on much larger text corpora using substantial computational resources, they achieve considerably lower mapping accuracies compared to much smaller encoders, which are typically trained on less data with fewer compute resources. One possible reason for this performance gap is the *causal language modeling* objective used during pre-training, which optimizes the model to predict the next token rather than to perform structured alignment tasks like code mapping.

E Comparison Between LLM-Generated Summaries and the Code Description for Mapping ICD Versions

We analyzed the generated maps using LLM-generated descriptions and terms-only, to evaluate the effectiveness of the LLM-generated descriptions at capturing the linguistic variation in the code description across different ICD versions. we identified cases where the ground-truth target codes did not appear among the top-100 predicted mappings when using only the code descriptions, but were correctly retrieved when using the summaries generated by *Qwen3-8B*. Table 12 presents some examples of such cases. This suggests that, LLM-generated texts do provide meaningful context to generate better embeddings.

F Rules for Handling Varying Granular Detail in Clinical Conditions

1. **Select the most specific target option that represents the closest clinical equivalent to the level of detail provided in the source code.**
   (a) Example: Source: ‘tuberculosis of hip’ → Target: ‘Tuberculous arthritis of other joints’ (hip is a specific joint).
   (b) Example: Source: ‘typhoid fever’ → Target: ‘Typhoid fever unspecified’ (when no specific complication is mentioned).

2. **In cases where the source code lacks specific details, select the options that include terms like ‘unspecified’ or ‘other specified’.**
   (a) Example: Source: ‘roseola infantum, unspecified’ → Target: ‘Exanthema subitum [sixth disease] unspecified’.
   (b) Example: Source: ‘tuberculosis of limb bones’ → Target: ‘Tuberculosis of other bones’ (limb bones are part of other bones).

	ICD9CM-ICD10CM			ICD10CM-ICD9CM			ICD10AM-ICD11			ICD11-ICD10AM		
	Dig	**Inf**	**Resp**	**Dig**	**Inf**	**Resp**	**Dig**	**Inf**	**Resp**	**Dig**	**Inf**	**Resp**
BioClinicalBERT	0.67	0.57	0.63	0.51	0.53	0.55	0.58	0.53	0.63	0.43	0.56	0.57
ClinicalBERT	0.73	0.60	0.68	0.53	0.56	0.58	0.63	0.57	0.67	0.46	0.57	0.60
UmlsBERT	0.75	0.58	0.67	0.53	0.56	0.55	0.62	0.56	0.65	0.47	0.57	0.54
SBERT	**0.80**	**0.69**	**0.75**	**0.62**	**0.70**	**0.59**	**0.66**	**0.66**	**71**	**0.60**	**0.67**	**61**

Table 9: Evaluation of various pre-trained BERT models for mapping between different ICD versions. These models are used to generate the embeddings for the ICD code descriptions and the potential maps are identified using *cosine-similarity*.

PromptEOL "The term: '[X]' means in one word"
PromptSUM "The term: '[X]' can be summarized as"
Knowledge Enhanced PromptEOL "In clinical terminology, a clinical condition can be described in multiple ways, and many synonyms are used interchangeably. With this in mind, the term: '[X]' means in one word"
Knowledge Enhanced PromptSUM "In clinical terminology, a clinical condition can be described in multiple ways, and many synonyms are used interchangeably. With this in mind, the term: '[X]' can be summarized as"

Table 10: Prompt templates to generate dense representation from LLM

3. **Maintain consistency by selecting 'other' for 'other specified' and 'unspecified' for ' unspecified'.**
   (a) Example: Source: 'other specified tuberculosis of central nervous system' → Target: 'Other tuberculosis of nervous system'.
   (b) Example: Source: 'whooping cough, unspecified organism' → Target: 'Whooping cough unspecified species without pneumonia'.
4. **Take into account the clinical context of the source code and select the option that reflect common clinical manifestations or broader categories relevant to its clinical implications.**
   (a) Example: Source: 'chickenpox with other specified complications' → Target: 'Varicella meningitis' (meningitis is a known severe complication).
   (b) Example: Source: 'other specified diseases due to chlamydiae' → Target: 'Other chlamydial diseases'.

.

G Examples of the Thiniking Steps for the ICD-9-CM Code 52107

		ICD-9-CM to ICD-10-CM			ICD-10-CM to ICD-9-CM			ICD-10-AM to ICD-11			ICD-11 to ICD-10-AM		
		Dig	Inf	Resp	Dig	Inf	Resp	Dig	Inf	Resp	Dig	Inf	Resp
Knowledge-Enhanced PromptSUM	**Qwen3-8B**	0.35	0.24	0.38	0.27	0.22	0.31	0.34	0.31	0.46	0.24	0.33	0.39
	Llama-3.1-8B-Instruct	0.42	0.34	0.51	0.33	0.29	0.40	0.41	0.41	0.52	0.26	0.42	0.46
	Phi-4-mini-instruct	0.13	0.06	0.10	0.06	0.05	0.09	0.04	0.07	0.08	0.07	0.07	0.07
	Mistral-7B-Instruct-v0.3	0.52	0.42	0.46	0.41	0.42	0.39	0.48	0.46	0.57	0.35	0.46	0.43
Knowledge-Enhanced PromptEOL	**Qwen3-8B**	0.30	0.18	0.33	0.20	0.17	0.29	0.29	0.25	0.39	0.22	0.26	0.35
	Llama-3.1-8B-Instruct	0.43	0.33	0.49	0.30	0.29	0.37	0.43	0.40	0.50	0.28	0.41	0.47
	Phi-4-mini-instruct	0.05	0.02	0.04	0.04	0.03	0.06	0.06	0.04	0.08	0.05	0.04	0.06
	Mistral-7B-Instruct-v0.3	0.59	0.46	0.56	0.41	0.49	0.47	0.54	0.52	0.56	0.42	0.55	0.50

Table 11: Evaluation between different pre-trained LLMs on mapping ICD versions. We used knowledge enhanced prompteEOL and promptSUM to generate the dense representations for the ICD code descriptions.

Source	**Target**
ICD9CM-ICD10CM	
Madura Foot [0394]	Mycetoma unspecified [B479]
Geniculate Herpes Zoster [05311]	Postherpetic geniculate ganglionitis [B0221]
Ornithosis with pneumonia [0730]	Chlamydia psittaci infection [A70]
Condyloma acuminatum [07811]	Anogenital (venereal) warts[A630]
Hand, foot, and mouth disease [0743]	Enteroviral vesicular stomatitis with exanthem [B084]
Blood in stool [5781]	Melena[K921]
ICD10CM-ICD9CM	
Enteroviral vesicular pharyngitis [B085]	Herpangina [0740]
Tinea cruris [B356]	Dermatophytosis of groin and perianal area [1103]
Naegleriasis [B602]	Other specific infections by free-living amebae [13629]
Cercarial dermatitis [B653]	Cutaneous schistosomiasis [1203]
Visceral larva migrans [B830]	Toxocariasis [1280]
Stannosis [J635]	Pneumoconiosis due to other inorganic dust [503]
ICD10AM-ICD11	
Hypercementosis [K034]	Cementum dysplasia [DA075]
Glossodynia [K146]	Burning mouth syndrome[DA0F0]
Exanthema subitum (sixth disease) [B082]	Roseola infantum [1F01]
Tinea unguium [B351]	Dermatophytosis of nail [1F281]
Penicilliosis [B484]	Talaromycosis [1F2K]
ICD11-ICD10AM	
Postdiphtheritic paralysis of uvula [1C1700]	Pharyngeal diphtheria [A360]
Dermatophytosis of nail [1F281]	Tinea Unguium [B351]
Dermatophytosis of foot [1F282]	Tinea Pedis [B353]
Talaromycosis [1F2K]	Penicilliosis [B484]
Burning mouth syndrome [DA0F0]	Glossodynia [K146]

Table 12: Examples of cases where ground-truth target codes were not in the Top-100 predictions using the ICD code descriptions but correctly mapped using summaries generated by *Qwen3-8B*.

```
[Prompt]:
Please apply the rules below to answer the following question.
Rules:
1. Select the most specific target option that represents the closest clinical equivalent to the level of detail provided in the given clinical term.
2. In cases where the given clinical term lacks specific details, select the options that include terms like 'unspecified' or 'other specified'.
3. Maintain consistency by selecting 'other' for 'other specified' and 'unspecified' for 'unspecified'.
4. Take into account the clinical context of the given clinical term and select the option that reflect common clinical manifestations or broader
categories relevant to its clinical implications.

Which of the following is the best match for 'Dental caries of smooth surface'?
K026,Dental caries on smooth surface
K0261,Dental caries on smooth surface limited to enamel
K0262,Dental caries on smooth surface penetrating into dentin
K0263,Dental caries on smooth surface penetrating into pulp
Please do not include explanations or code descriptions, just return the code.

[Thinking Steps]:
Okay, let's tackle this question. The user is asking for the best match for 'Dental caries of smooth surface' from the given options. The options are
K026, K0261, K0262, K0263.

First, I need to recall the rules provided. Rule 1 says to select the most specific target option that's the closest clinical equivalent. Rule 2
mentions using 'unspecified' or 'other specified' if the term lacks details. Rule 3 says to use 'other' for 'other specified' and 'unspecified' for '
unspecified'. Rule 4 emphasizes considering clinical context and common manifestations.

The clinical term here is "Dental caries of smooth surface". Looking at the options, K026 is "Dental caries on smooth surface", which seems to match
exactly. But there are more specific codes: K0261, K0262, K0263. These specify different depths of caries\-enamel, dentin, pulp. The original term
doesn't mention the depth, just the location (smooth surface).

So, since the term doesn't specify the depth, but the options have more detailed categories, I need to check if the original term is general enough. If
 the term is only about the location (smooth surface) without depth, then the most specific would be K026. However, sometimes in coding, if the depth
isn't specified, it might be considered 'unspecified'. But the options don't have an 'unspecified' code here. Wait, the options given are K026, K0261,
K0262, K0263. The first one is just 'Dental caries on smooth surface', which is the general term. The others are more specific. Since the original term
 doesn't mention depth, maybe the correct answer is K026. But wait, the rules say to select the most specific if possible. However, if the term doesn't
 have enough detail, maybe we should pick the most general. Wait, the rules say to select the most specific that's the closest equivalent. But if the
term is 'smooth surface' without depth, then the most specific would be K026, as the others require more detail. However, maybe in some coding systems,
 if depth isn't specified, it's considered as unspecified. But the options don't have an unspecified code here. Let me check the options again. The
options are K026, K0261, K0262, K0263. The first one is the general code. So if the term is exactly 'Dental caries on smooth surface', then K026 is the
 direct match. However, sometimes in ICD-10 coding, the more specific codes are preferred when possible. But the original term doesn't specify depth,
so maybe K026 is the correct one. But the user's question is about selecting the most specific. Wait, the original term is 'Dental caries of smooth
surface', which is the same as K026. However, the other codes are more specific. But since the original term doesn't specify depth, perhaps the answer
is K026. Alternatively, maybe the term 'smooth surface' in the original is considered to be the same as K026. Therefore, the best match is K026. But I
need to make sure. Let me think again. If the original term is 'Dental caries of smooth surface', then the code K026 is exactly that. The other codes
add more details (depth). Since the original term doesn't mention depth, the correct code would be K026. So the answer should be K026.

[Final Answer]: K026
```

Figure 5: An example of *Qwen3-8B*'s thinking steps for generating maps for ICD-9-CM code **K026** (*Dental caries of smooth surface*).