Multi-Label Few-Shot ICD Coding as Autoregressive Generation with Prompt

Zhichao Yang[1], **Sunjae Kwon**[1], **Zonghai Yao**[1], **Hong Yu**[1,2,3]
[1]College of Information and Computer Sciences, University of Massachusetts Amherst
[2]Department of Computer Science, University of Massachusetts Lowell
[3]Center for Healthcare Organization and Implementation Research, Veterans Affairs Bedford Healthcare System
zhichaoyang@umass.edu

Abstract

Automatic International Classification of Diseases (ICD) coding aims to assign multiple ICD codes to a medical note with an average of 3,000+ tokens. This task is challenging due to the high-dimensional space of multi-label assignment (155,000+ ICD code candidates) and the long-tail challenge - Many ICD codes are infrequently assigned yet infrequent ICD codes are important clinically. This study addresses the long-tail challenge by transforming this multi-label classification task into an autoregressive generation task. Specifically, we first introduce a novel pretraining objective to generate free text diagnoses and procedures using the SOAP structure, the medical logic physicians use for note documentation. Second, instead of directly predicting the high dimensional space of ICD codes, our model generates the lower dimension of text descriptions, which then infers ICD codes. Third, we designed a novel prompt template for multi-label classification. We evaluate our Generation with Prompt (GP_{soap}) model with the benchmark of all code assignment (MIMIC-III-full) and few shot ICD code assignment evaluation benchmark (MIMIC-III-few). Experiments on MIMIC-III-few show that our model performs with a marco F1 30.2, which substantially outperforms the previous MIMIC-III-full SOTA model (marco F1 4.3) and the model specifically designed for few/zero shot setting (marco F1 18.7). Finally, we design a novel ensemble learner, a cross-attention reranker with prompts, to integrate previous SOTA and our best few-shot coding predictions. Experiments on MIMIC-III-full show that our ensemble learner substantially improves both macro and micro F1, from 10.4 to 14.6 and from 58.2 to 59.1, respectively.

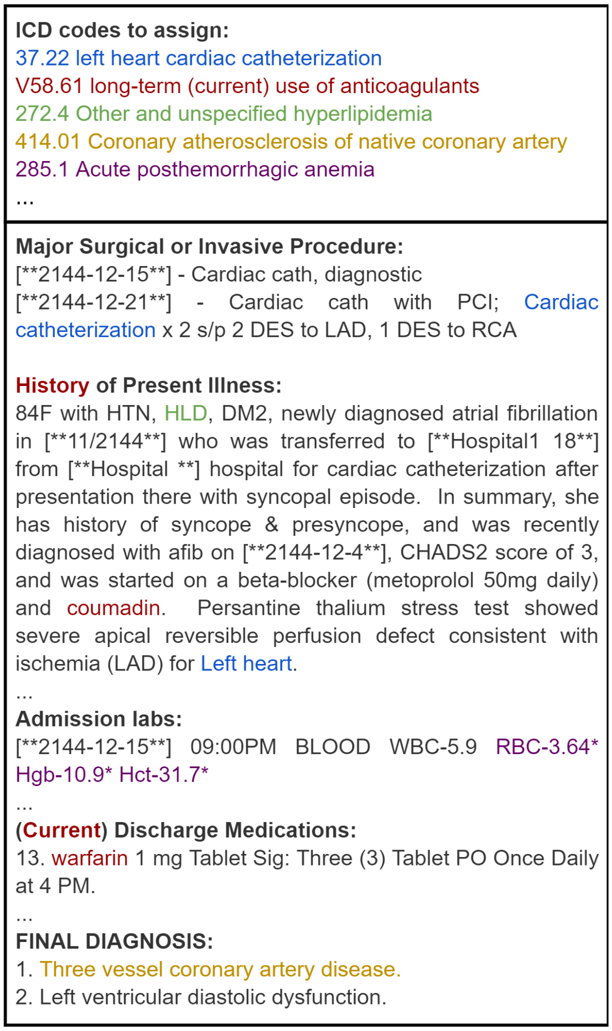
1 Introduction

In real-world tasks, there are often insufficient training data for rare class labels (Atutxa et al. 2019; Koh et al. 2021). Taking automatic international classification of diseases (ICD) coding (Larkey and Croft 1996) as an example, given a discharge note as input, the task is to assign multiple ICD disease label codes associated with each note. The ICD coding task in MIMIC-III dataset (Johnson et al. 2016) contains 4,075 unique ICD-9-CM codes in 3,372 testing data, among which 1,285 (31.5%) codes occur less than 11 times in the training split. In the clinical domain, rare codes may be as clinically important as common codes for a patient. Therefore, a multi-label classifier is required to perform with high precision and recall for every ICD code, including infrequent ones.

Many recent studies tried self-attention BERT-based models on ICD coding (Zhang, Liu, and Razavian 2020; Ji, Holtta, and Marttinen 2021; Pascual, Luck, and Wattenhofer 2021; Dai et al. 2022). Most existing BERT-based models follow a common design choice: each ICD code is associated with a unique entity representation and the coding can be transformed into a multi-label classification across ICD codes. Specifically, previous models use bi-encoder to calculate a low-dimensional dense vector representation for an ICD code and for a note text separately. Later, they take dot product of the two to build a note-aware code representation for each ICD code. This design enables a multi-label classification with binary cross-entropy loss on all code representations. However, such a design has several limitations.

First, bi-encoder may miss fine-grained interactions between target ICD codes and candidate mentions in discharge note texts (Mullenbach et al. 2018). A discharge note contains hundreds of different candidate mentions. Take an ICD code *left heart cardiac catheterization* in Figure 1 as an example, their note encoder aims to capture two candidate mention phrases cardiac catheterization and left heart. The encoder also needs to repeat this process for every code and condenses the note text into a single low-dimensional dense vector. When pooling thousands of tokens from a note text into a single representation, it is possible that some detailed level of information would be lost. This detail-lost representation of note text would then miss fine-grained interactions when cross attend to target code descriptions.

Second, note-aware code representation from cross-attention is unable to generate from missing mention, because language model contains limited amount of medical knowledge (Meng et al. 2022; Yao et al. 2022). Take ICD code *Acute posthemorrhagic anemia* in Figure 1 as an example, there is no mention of *anemia* in this discharge note texts. However, the model is required to infer from lab results such as low level of RBC, Hgb, Hct, which leads to *anemia*. In order to fill in this knowledge gap, previous researches use additional information such as hierarchical code ontology (Tsai, Chang, and Chen 2019; Cai et al. 2022), code co-occurrence (Cao et al. 2020), code frequency



ICD codes to assign:
37.22 left heart cardiac catheterization
V58.61 long-term (current) use of anticoagulants
272.4 Other and unspecified hyperlipidemia
414.01 Coronary atherosclerosis of native coronary artery
285.1 Acute posthemorrhagic anemia
...

Major Surgical or Invasive Procedure:
[**2144-12-15**] - Cardiac cath, diagnostic
[**2144-12-21**] - Cardiac cath with PCI; Cardiac catheterization x 2 s/p 2 DES to LAD, 1 DES to RCA

History of Present Illness:
84F with HTN, HLD, DM2, newly diagnosed atrial fibrillation in [**11/2144**] who was transferred to [**Hospital1 18**] from [**Hospital **] hospital for cardiac catheterization after presentation there with syncopal episode. In summary, she has history of syncope & presyncope, and was recently diagnosed with afib on [**2144-12-4**], CHADS2 score of 3, and was started on a beta-blocker (metoprolol 50mg daily) and coumadin. Persantine thalium stress test showed severe apical reversible perfusion defect consistent with ischemia (LAD) for Left heart.
...
Admission labs:
[**2144-12-15**] 09:00PM BLOOD WBC-5.9 RBC-3.64* Hgb-10.9* Hct-31.7*
...
(Current) Discharge Medications:
13. warfarin 1 mg Tablet Sig: Three (3) Tablet PO Once Daily at 4 PM.
...
FINAL DIAGNOSIS:
1. Three vessel coronary artery disease.
2. Left ventricular diastolic dysfunction.

Figure 1: An example of EHR note from the MIMIC-III dataset, which includes ICD code gold label (top) and the discharge note text (bottom). We colored each code and its corresponding candidate mentions/evidence in the note text.

(Zhou et al. 2021), signs and symptoms section in Wikipedia (Wang et al. 2020a, 2022) as an additional input source, and use the same binary cross-entropy objective on codes. However, few research design a specific training objective to distill knowledge into their model.

Besides the accuracy challenge from the long tail distribution, such a high number of ICD codes also increases their model's memory demand. As stated previously, for the ICD-9 testing data from MIMIC-III, the model needs to assign from 4,075 unique diagnosis and procedure codes. In reality, ICD-9 has a total of 14,000 and 3,900 unique diagnosis and procedure codes, respectively. The number of unique ICD codes is also increasing because more and more diseases are being refined or added. The up-to-date ICD-10, which replaces ICD-9, contains 68,000 diagnosis codes and 87,000 procedure codes (Eisfeld 2014). Most previous methods require building a note-aware code representation for each code as shown in Figure 2. The number of code representations stored in memory increases linearly as the number of candidate codes to assign increases. The significant increase in memory makes it hard to deploy in the real-world auto ICD classification setting (Ziletti et al. 2022; Yan et al. 2022).

In this paper, we present a simple and yet effective autoregressive Generation with Prompt (GP) framework for ICD coding. We have made major contributions to ICD coding. Specifically, we designed a novel mask objective for the clinical language model. Our pretraining objective utilizes the unique clinical knowledge inherited by the structure of EHR notes. Specifically, physicians write notes following the subjective, objective, assessment, and plan (SOAP) structure, where the assessment and plan sections can be inferred from the subjective and objective sections (Bodenreider 2004; Yang and Yu 2020). Our GP$_{\text{soap}}$ is a longformer encoder-decoder (LED) pretrained with assessment & plan generation loss, which infers diagnoses and procedures from symptoms and lab results by generating the assessment and plan sections from the subjective and objective sections. This pretraining can mitigate the missing mention challenge.

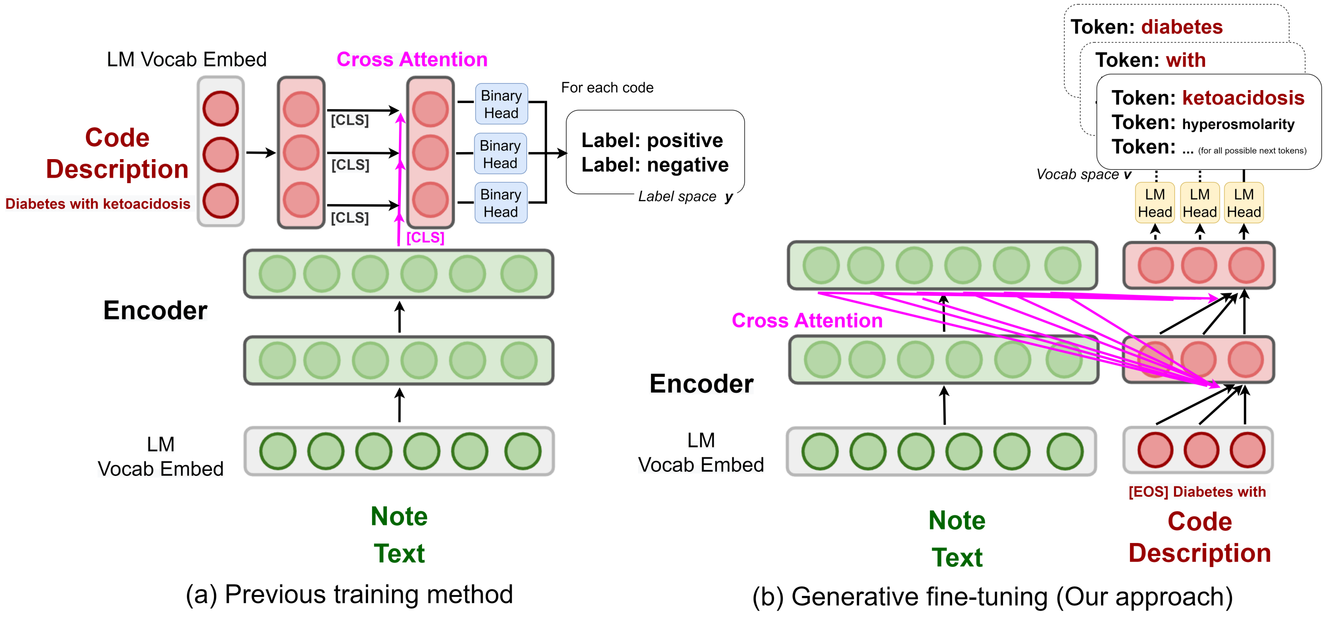
In addition, unlike all previous approaches which directly predict ICD codes, we fine-tune GP$_{\text{soap}}$ to first generate code descriptions with a prefix prompt and then assign the descriptions to the corresponding ICD codes. Code description generation only requires a fixed vocab of 4,501 candidate word tokens, a substantial reduction in dimensional space. Unlike all previous models that build cross-attention on a single pooled representation of input note texts, we perform an autoregressive cross-attention on every single token of EHR notes to avoid the problem of missing fine-grained interactions.

Experiments on the public ICD coding benchmark (MIMIC-III-full) show that our model GP$_{\text{soap}}$ outperforms the state-of-the-art (SOTA) MSMN on MIMIC-III-full using the macro evaluation metrics. In order to verify its few-shot capability, we also show that our model substantially outperforms both MSMN (Yuan, Tan, and Huang 2022) and AGMHT (Song et al. 2020). Last but not least, we show that an ensemble learner which simply reranks the predictions of our model and MSMN achieves the new SOTA on MIMIC-III-full. Our codes are attached in supplementary material and will be publicly available upon publication.

2 Methods

2.1 Task Definition

ICD coding is typically formulated as a multi-label classification task (McCallum 1999). Specifically, considering thousands of words from an input discharge summary note text t (green as shown in Figure 2), the task is to assign a binary label $y_i \in \{0, 1\}$ for the ith ICD code in the label space Y, where 1 means that note is positive for an ICD diagnosis or procedure. Each candidate code has a short code description phrase c_i in free text. For instance, code 250.1 has description *diabetes with ketoacidosis*. Code descriptions c is the set of N_c number of c_i where $N_c = 4,075$ is the number of total ICD codes unless otherwise specified.



(a) Previous training method

(b) Generative fine-tuning (Our approach)

Figure 2: An illustration of (a) standard training method and (b) our proposed generative fine-tuning. The number of candidate code is 3 for illustration in (a). Solid lines represent attention in step 3 and dash lines represent previous steps in (b).

2.2 Pretraining to Learn Clinical Inference

Besides token-level masked language modeling (MLM) objective, which has shown to be limited in biomedical tasks (Moradi et al. 2021; Gutierrez et al. 2022), our GP$_{\text{soap}}$ model is also pretrained with a paragraph-level autoregressive encoder-decoder objective. Specifically, given subject and object (SO) sections including symptoms and lab results, the task is to generate free-text assessment&plan (AP) including diagnoses and procedures (refer to figure 1 of Yang and Yu (2020) for an example[1]). The SOAP is identified using the document section classification tool (Kwon, Yang, and Yu 2022). Similar to PEGASUS (Zhang et al. 2020), we masked selected sentences in assessment&plan section with [MASK1] for target generation text (GSG). The selected sentences include diagnoses and procedures entities found using biomedical named entity recognition and linking (NER+L) tool MedCAT (Kavuluru, Rios, and Lu 2015). The other sentences in subject and object section remain almost the same as input, but with some phrases randomly masked by [MASK2] for MLM. This pretraining objective is similar compared to ICD coding task, which aims to assign ICD diagnoses and procedures codes, but differs from ICD coding in that it generates sentences other than the specific code. ICD coding also has the whole medical note as input, but our GSG only includes part of clinical notes without diagnoses and procedures. Hence, our GP$_{\text{soap}}$ could learn such inferences from symptoms and lab results to diagnoses and procedures.

2.3 Generative Fine-tuning

Our generative fine-tuning is different from standard fine-tuning. We use autoregressive encoder-decoder model to assign ICD codes to a note by first generating unique code descriptions. An example to assign the ICD code 250.1 with the corresponding text description *diabetes with ketoacidosis* is illustrated in Figure 2 (b). This step is repeated N times for each code description to generate, and PLM implicitly decides N by generating the end of text token. In this way, we transfer a downstream multi-label classification task into a sequence to sequence task like pretraining. Specifically, we design a free text prompt template as input:

$x_i =$ t. diagnoses and procedures : c_1 ; c_2 ; ... ; c_i .

Given ith prompt, the next $i+1$th code label probability would be calculated as:

$$P(\hat{c}_{i+1}|x_i) = \prod_{j=1}^{L_c} P(u_j|u_{<j}, x_i) \tag{1}$$

where u is the sequence of L_c tokens in $i+1$th code description $\hat{c}_{i+1}$ to generate. Finally, we grouped the generated tokens as a code description phrase $\hat{c}_{i+1}$, which is then mapped to the corresponding ICD code. During prediction, the ICD code is assigned with the binary label of $y = 1$, and use the same code to evaluate and compare to standard fine-tuning. Notice that this decoding step would cross-attend each token of code descriptions and note text in every layer of PLM decoder. Generative fine-tuning reuses all parameters during pretraining, and does not introduce new randomly initialized parameters, making the whole model easy to fine-tune in a few-shot setting.

[1] https://aclanthology.org/2020.findings-emnlp.336.pdf

2.4 Inference with Clinical Vocab Constraint

Ideally, the generated tokens would match to a valid code. However in reality, PLM would hallucinate and generate random tokens that do not match to any code descriptions, especially given that code description is usually a long phrase of average 9.81 tokens. Hence, we exploit beam search (Sutskever, Vinyals, and Le 2014) with vocab constraint, which forces to only generate valid code descriptions. Similar to previous research on entity retrieval (De Cao et al. 2021), we applied a trie structure to efficiently limit the vocab. Since beam search only examines one step forward during decoding, we only limit the generation of one future token given the preceding tokens. Hence, we construct the trie by the following rule: for each node in the trie, its children are allowed next tokens to formulate into a valid code description. While querying the allowed next tokens for a sub phrase, we simply traverse from root node using tokens in sub phrase to find the children. An illustrative example is shown in Figure 3.

By querying this trie, we are able to generate a valid description for a single code. In order to generate multiple codes, we design 2 different ending tokens for each code description: [EOS] (end of text token) or ; (semicolon), where [EOS] means this is the last code to assign to this note text, and semicolon means continue generating other codes. We let decoder decide how many codes to assign.

2.5 Code Reranking

Instead of generative prompt intended for few-shot learning, we propose a novel cloze style prompt to rerank predicted code candidates generated from different models. The concept of 2nd stage reranker originates from information retrieval (Dang, Bendersky, and Croft 2013; Nogueira and Cho 2019). This reranker serves as a fine-grained coding model. Specifically, we reformulate multi-label classification tasks with free text prompt template as input:

x_p =t . c_1 : [MASK] , c_2 : [MASK] , ... , c_{N_b} : [MASK] . where c_i is the descriptions of code candidates generated from previous step models. We use KEPTLongformer (Yang et al. 2022b) to decide if code is positive (or negative) for a note by filling [MASK] with vocab token yes (or no). This step is very similar to generative fine-tuning but differs in the design of the prompt. Instead of using the prompt to autoregressively generate the next code description, our reranker formulates the prompt with the code description given as the candidate, and verifies if the code should be assigned to the discharge note texts or not.

3 Experiments

3.1 Dataset

To reduce potential bias with small cohort(Weber et al. 2017), we pretrain on a large cohort containing 3.6 million clinical notes following SOAP structure. The pretraining cohort is from US Veterans Health Administration Corporate Data Warehouse. We collect every patient progress notes from calendar year 2016 to 2020. The average number of tokens for SO section and AP section is 1456 and 175 respectively.

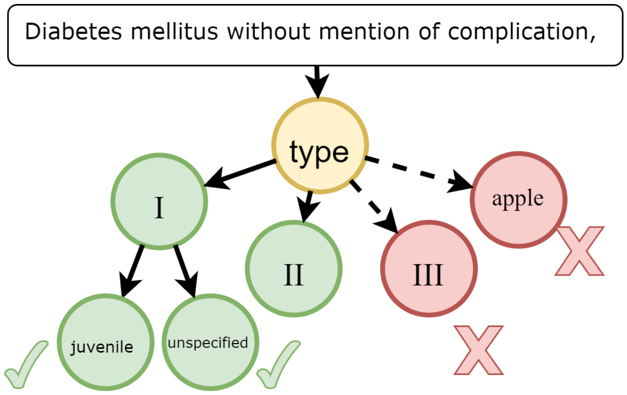


Figure 3: An illustration of trie constraint beam search which forces to decode only valid code descriptions. Full valid description is *Diabetes mellitus without mention of complication, type 1 juvenile type, not stated as uncontrolled.*

The fine-tuning dataset (Johnson et al. 2016) contains clinical data from real patients. It contains data instances of de-identified discharge summary note texts with expert-labeled ICD-9 codes. We applied the following text preprocessing steps before tokenization: (1) removing all de-identification tokens; (2) replacing characters other than punctuation marks and alphanumerical into white space (e.g. /n); (3) stripping extra white spaces. The average number of documents per label is 12.6, and the average labels per document is 15.8.

For all codes prediction tasks (MIMIC-III-full), we used the same splits as the previous work (Mullenbach et al. 2018; Yuan, Tan, and Huang 2022). Previous work truncated discharge summaries at 4,000 words. Since longformer used tokens instead of words, we truncated discharge summaries at 8,192 tokens unless otherwise specified.

To benchmark ICD coding task on few shot learning, we also created a few-shot codes prediction dataset (MIMIC-III-few), which is a subset of the MIMIC-III-full dataset. Following previous works on few-shot classification (Lu et al. 2020; Song et al. 2020), among 4,075 different types of ICD-9 codes in test set, we collect those which occur within 5 times (but occur at least 1 time) in the training set. We found a total of 685 (16.8%) unique ICD-9 codes under the few-shot setting.

3.2 Baselines

MSMN (Yuan, Tan, and Huang 2022) use synonyms with adapted multi-head attention, which achieved SOTA performance on the MIMIC-III-full dataset.

AGMHT (Song et al. 2020) use adversarial generative model to generate feature for zero/few shot labels, which achieved SOTA performance on zero-shot MIMIC-III task.

GP$_{\textbf{wiki}}$ use our GP framework (without GSG pretraining objective in section 2.2). During fine-tuning, it is initialized from LED pretrained on Wikipedia and Book corpus in Beltagy, Peters, and Cohan (2020).

GP$_{\textbf{pubmed}}$ use our GP framework but initialized from LED pretrained on the Pubmed dataset.

Model	F1		Prec		Recall					
	Mac	Mic	Mac	Mic	Mac	Mic	P@15	R@15	P@50	R@50
Standard	7.7	47.2	7.6	42.4	7.7	53.4	47.2	43.2	24.9	76.1
MSMN	10.4	**58.1**	10.4	**61.2**	10.5	**55.4**	**59.6**	**55.8**	**27.0**	**78.4**
AGMHT	1.3	0.6	1.2	36.6	1.5	0.3	-	-	-	-
$\text{GP}_{\text{pubmed}}$	8.3	44.2	10.1	54.7	7.1	37.0	-	-	-	-
GP_{soap}	**13.4**	49.8	**13.2**	53.7	**13.7**	46.4	-	-	-	-
Reranker Oracle	28.8	67.9	28.2	63.7	29.4	72.7	63.6	59.0	35.9	99.8
Reranker (MSMN+AGMHT)	11.9	**59.7**	12.5	**65.3**	12.2	57.5	**61.4**	**57.4**	27.0	78.6
Concater (MSMN+AGMHT)	11.6	58.2	11.4	61.0	11.9	55.6	-	-	-	-
Reranker (MSMN+$\text{GP}_{\text{pubmed}}$)	11.7	57.2	12.2	57.9	11.2	56.5	58.7	55.5	27.0	78.5
Concater (MSMN+$\text{GP}_{\text{pubmed}}$)	11.4	54.8	11.0	52.6	12.6	59.4	-	-	-	-
Reranker (MSMN+GP_{soap})	**14.6**	59.1	**14.3**	58.3	14.9	59.9	60.5	56.8	**27.8**	**80.2**
Concater (MSMN+GP_{soap})	14.0	55.0	12.0	50.2	**17.6**	**60.8**	-	-	-	-

Table 1: Results on the MIMIC-III-full set, compared between GP and baselines (top), reranking from combination of different baselines (down). Standard is standard fine-tuning with longformer as encoder. AGMHT is only built for zero/few-shot.

3.3 Metrics

We report both macro and micro scores of precision, recall, f1, and accuracy. Given the predicted text code pair, macro-averaged values are calculated by first computing metric for each ICD code and then taking the average among all classes. Micro-averaged values are calculated by treating each pair as a separate prediction and average among all predictions. Take precision as an example, the metrics are distinguished as follows:

$$MacroP = 1/N_c \sum_{i=1}^{N_c} \frac{TP_i}{TP_i + FP_i} \tag{2}$$

$$MicroP = \frac{\sum_{i=1}^{N_c} TP_i}{\sum_{i=1}^{N_c} TP_i + FP_i} \tag{3}$$

where TP_i is True Positives for code label i, and FP_i is False Positives for code label i. This shows Macro metrics place much more emphasis on rare code prediction.

We also report precision or recall at K. This is inspired by the need of clinical decision support tool when applied to real world, in which a doctor is given a certain number of anticipated codes to examine. We select $K = 15$ which is roughly the average number of codes in MIMIC-III per note texts, and $K = 50$ as this is the maximum candidate our largest GPU memory support as a reranker.

3.4 Implementation Details

We order ICD codes to generate by its SEQ_NUM (priority ranking for diagnoses and perform order for procedures) labeled by medical experts from DIAGNOSES_ICD PROCEDURES_ICD table in MIMIC-III. For punctuation semicolons in ICD code descriptions, we replace them to punctuation commas. Pretraining on SOAP data took about 140 hours with 4 NVIDIA RTX 6000 GPU with 24 GB memory. Fine-tuning took about 40 hours with 4 NVIDIA RTX 6000 GPU with 24 GB memory. Our reranker training took about 12 hours with 2 NVIDIA A100 GPU with 40 GB memory. During pretraining, we used warmup ratio of 0.1, learning rate $5e^{-5}$, dropout rate 0.1, L2 weight decay $1e^{-3}$ and batch size of 64 with fp16. During fine-tuning, we grid searched learning rate [$1e^{-5}$, $\mathbf{2e^{-5}}$, $3e^{-5}$], dropout rate [$\mathbf{0.1}$, 0.3, 0.5], with batch size of 4. Best hyper-parameters set is bolded. The random seed is 42. For each evaluation on generation models, we run generation once with a beam search size of 2. We set the number of candidate code to rerank $N_b = 50$ as this is the maximum candidate our largest GPU memory support. Our evaluation code is publicly available[2].

3.5 Results

Results show that our generation with prompt framework pretrained on clinical SOAP data (GP_{soap}) migrates the long-tail issue compared to the previous SOTA model MSMN. For the all disease code assignment (MIMIC-III-full) task, our GP_{soap} achieves macro F1 of 13.4 (+3.0), macro precision of 13.2 (+2.8), and macro recall of 13.7 (+3.2). The number in parentheses shows the improvements compared to MSMN. For the rare disease code assignment (MIMIC-III-few) task, our GP_{soap} achieves macro F1 of 30.2 (+25.9), micro F1 of 35.3 (+26.8), macro recall of 32.9 (+28.7), micro recall of 32.6 (+28.1), macro accuracy of 25.1 (+21.3), micro accuracy of 21.4 (+17.0). This further confirms the strong advantage of our GP_{soap} for long-tail codes. Finally, by combining the prediction of GP_{soap} and MSMN, our reranker with code descriptions as prompt shows SOTA performance for both macro and micro evaluations.

[2] https://github.com/whaleloops/KEPT

Model	F1		Prec		Recall	
	Mac	Mic	Mac	Mic	Mac	Mic
MSMN	4.3	8.5	4.5	**70.9**	4.2	4.5
AGMHT	18.7	29.2	17.6	49.4	19.9	20.7
GP$_{\text{wiki}}$	2.9	4.9	3.0	61.1	2.7	2.6
GP$_{\text{pubmed}}$	7.3	12.3	7.9	47.3	6.8	7.1
GP$_{\text{soap}}$	**30.2**	**35.3**	**27.9**	38.5	**32.9**	**32.6**

Table 2: Results on the MIMIC-III-few set, compared between baselines and different pretraining of GP.

4 Discussions

Our final model is a combination of 1) autoregressive generation; 2) SOAP pretraining objective; 3) reranker with prompt. We will discuss details in the following paragraphs:

Are autoregressive coding models few-shot learners? Our autoregressive model GP$_{\text{soap}}$ outperforms non-autoregressive auto coding model MSMN in few-shot coding task as shown in Section 3.5. Moreover, our GP$_{\text{soap}}$ also significantly outperforms AGMHT which is specifically designed for few/zero-shot ICD coding. Compared to AGMHT, our GP$_{\text{soap}}$ improves macro and micro F1 by +11.5 and +6.1, macro and micro precision by +10.3, -10.9, macro and micro recall by +13.0, +11.9, macro and micro accuracy by +9.2, +4.3. Our findings support that clinical PLMs are few shot learners (Taylor et al. 2022; Lewis et al. 2020; Yang et al. 2022a). From a network architecture perspective, our autoregressive ICD generation model uses cross-attention to capture note aware code representations, whereas previous traditional ICD coding studies use dot product label attention (Zhang, Liu, and Razavian 2020; Ji, Holtta, and Marttinen 2021; Pascual, Luck, and Wattenhofer 2021; Dai et al. 2022). A dot product label attention is required to aggregate many tokens from note into a single note representation before a final label cross-attention layer to build note aware code representations. Such a selection of which tokens best represent note is training intensive and does not generalize well in out-of-domain or zero/few-shot setting (Wang et al. 2020c; Thakur et al. 2021). Traditional methods avoid this by using a label attention when creating note aware code representations (Mullenbach et al. 2018; Bai and Vucetic 2019; Ji, Cambria, and Marttinen 2020; Wang et al. 2020b; Zhou et al. 2021; Liu et al. 2021; Kim and Ganapathi 2021; Luo et al. 2021; Sun et al. 2021; Wang et al. 2022; Yuan, Tan, and Huang 2022; Ren et al. 2022). However, the majority of them used unpretrained LSTM or CNN. This increased the number of unpretrained parameters and has shown to limit its ability in few-shot setting (Brown et al. 2020; Schick and Schütze 2021; Gao, Fisch, and Chen 2021). In contrast, our model combines the merit of both PLM and traditional ICD coding model. Our model is pretrained without introducing new unpretrained parameters during fine-tuning and also cross-attend code with every token in the note like traditional label attention. Such cross-attention has shown to be effective (compared to a typical bi-encoder network) in combining query and document in information retrieval passage ranking (Zhou and Devlin 2021), but has not yet been deployed for multi-label classification tasks.

MIMIC-III-few		F1	Prec	Recall
GP$_{\text{pubmed}}$	Mentioned	17.3	17.9	17.9
	UnMentioned	7.0	8.6	6.7
GP$_{\text{soap}}$	Mentioned	21.3	17.2	35.5
	UnMentioned	17.1	15.0	25.2

Table 3: The impact of whether diagnoses are mentioned (or not) in discharge summaries. GP$_{\text{soap}}$ is pretrained with GSG objective while GP$_{\text{pubmed}}$ is not.

Comparison among different pretraining objectives in adding clinical knowledge into PLM.
In-domain pretraining is important for clinical PLM to perform well in few-shot setting. Hence, we designed a new training objective on the clinical notes. Instead of the traditional masked language model which captures linguistic features, we designed a novel assessment&plan generation objective that focuses on predicting clinical outcomes in tokens. Our GP$_{\text{soap}}$ outperforms GP$_{\text{pubmed}}$ in recall and F1 scores in both MIMIC-III-few and MIMIC-III-full. This is mainly due to the learned inference ability during pretraining where the diagnosis may be missing. We want to know if GP$_{\text{soap}}$ is able to assign ICD codes when a missing mention occurs (e.g. *anemia* in Figure 1.). We annotate the discharge summaries with ICD-9 diagnosis codes with MedCat (Kavuluru, Rios, and Lu 2015). We then evaluate on codes that are explicitly mentioned in the text and those that are not. Table 3 shows that the accuracy on unmentioned diagnoses does not drop significantly from mentioned diagnoses using our GP$_{\text{soap}}$, especially when it is compared to GP$_{\text{pubmed}}$ without assessment&plan generation objective. The accuracy differences between the two models indicate that assessment&plan generation objective helps learn more clinical inference ability.

From the view of pretraining objective, our work is most similar to gap sentences generation objective such as PEGASUS (Zhang et al. 2020), which has shown to be effective in medical paper summarization (Guo et al. 2022) and healthcare question summarization (Yadav, Gupta, and Demner-Fushman 2022). Wan and Bansal (2022) augment the sentence selection strategy when pretraining and shows improvement in the factuality of the generated summary. Similarly, instead of random text format on MIMIC-III, we select paragraphs following the SOAP format from real-world data, because assessment and plan paragraphs in SOAP contain key clinical information of a patient encounter (Stupp et al. 2022).

From the view of adding medical knowledge into language models, besides applying masked language modeling which captures linguistic features in the medical domain (Li et al. 2019; Rongali et al. 2020), previous work has integrated knowledge from unlabelled clinical documents through clinical outcome pretraining (van Aken et al. 2021), patient journey prediction (Peng et al. 2020), amplified vocabulary (Wada et al. 2020), and medical entity retrieving (Meng et al. 2022) in non-autoregressive BERT. Here we provide an alternative way of incorporating clinical inference knowledge into an autoregressive model.

Combining few-shot and non-few-shot using cross-attention reranker with prompt.
Since AGMHT and our pretrained generation model $\mathrm{GP}_{\mathrm{soap}}$ improve the accuracy in rare codes but not for common codes, we further collect the ICD prediction of these models and rerank them with cross-attention prompt based fine-tuning. Compared to MSMN only model, our MSMN+$\mathrm{GP}_{\mathrm{soap}}$ reranker achieves the best macro F1 of 14.6 (+4.2), marco precision of 14.3 (+3.9), macro recall of 14.9 (+4.5), and micro recall of 59.9 (+4.6). MSMN+AGMHT reranker achieves the best micro F1 of 59.7 (+1.5) and micro precision of 62.0 (+0.7). Both models achieve the new SOTA macro and micro F1 on MIMIC-III coding benchmark. We also compare our reranker to a simple concater which directly concatenates predictions of two models. As shown at the bottom of Table 1, all rerankers outperform the corresponding concaters in terms of F1. Finally, our final result with reranker shows better F1 micro and macro score compared to RAC (Kim and Ganapathi 2021), which already reached past the human-level coding baseline.

5 Related Work

5.1 Prompt-based Fine-tuning

Prompt-based fine-tuning has been shown to be effective in few-shot tasks (Le Scao and Rush 2021; Gao, Fisch, and Chen 2021), even when PLM is relatively small (Schick and Schütze 2021) because they introduce no new parameter during few shot fine-tuning. However, most previous works focus on single-label multi-class classification task such as sentiment classification (Gao, Fisch, and Chen 2021), clinical ICD-9 triage (Taylor et al. 2022), disease phenotyping (Sivarajkumar and Wang 2022). To the best of our knowledge, our GP is the first work that applies prompting to multi-label classification task.

5.2 Autoregressive Entity Linking

ICD coding is similar to entity linking, as it is the task of assigning a unique textual entity identifier (i.e. entity name) to an entity mention given context. In contrast, ICD coding assigns multiple codes to whole discharge note document. De Cao et al. (2021) first addresses entity linking by using autoregressive sequence-to-sequence transformer to generate identifier, and experiments show high micro F1 score on both in-domain and out-of-domain entity linking benchmarks. Yuan, Yuan, and Yu (2022) further enhance autoregressive entity linking using biomedical knowledge base guided pretraining, and achieve SOTA results on biomedical entity linking such as BC5CDR. Instead of pretraining using a pair of entities within a sentence, we used a pair of subjective-objective paragraphs and assessment-plan paragraphs within SOAP notes each containing multiple entities.

6 Limitations

Our best MSMN+$\mathrm{GP}_{\mathrm{soap}}$ has limited accuracy in candidate code assignment, and still has room for improvement compared to oracle. An oracle candidate code assigning model is able to include all ground truth labels within its top-50 predictions (R@50 is 100). If such a near oracle model exists, then our reranker is able to achieve macro F1 of 28.8 and micro F1 of 67.9, where our current SOTA only has R@50 of 80.2 and thus achieves lower macro F1 of 14.6 and micro F1 of 59.7. One potential exploration is to change MSMN into more recent ICD coding works like PLM-ICD (Huang, Tsai, and Chen 2022) which shows higher micro.

Our $\mathrm{GP}_{\mathrm{soap}}$ is pretrained on 3.6 million proprietary clinical notes under SOAP format. We cannot pretrain on publicly available dataset mimic, because mimic discharge summary contains less than 1,000 notes under SOAP format, and thus is not suitable for pretraining. GP is an autoregressive generation method, and hence, slow as it generates one token at a time. In ICD coding task, our generative fine-tuning is about 24 times slower than the non-autoregressive standard fine-tuning.

7 Conclusions

In this paper, we investigate a pretrained clinical language model on ICD coding task for both full and rare codes. Adapted from recent advances in autoregressive entity linking, our SOAP pretrained autoregressive model achieves a competitive performance over SOTA systems in rare ICD coding task. This autoregressive model also preserves the order of the code. In contrast, previous auto ICD coding models neglect this potential error source but are actually common in the real world. (O'malley et al. 2005). We further propose a cross-attention reranker to combine best prediction from common ICD coding task and rare ICD coding task. Experiments show that our reranker model significantly outperforms the previous SOTA model on full ICD coding task. Our reranker can be easily plugged-in to rerank predictions generated from any model.

Acknowledgements

Research reported in this study was supported by the National Science Foundation under award 2124126. The work was also in part supported by the National Institutions of Health R01DA045816 and R01MH125027. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Science Foundation and National Institutes of Health.

References

Atutxa, A.; de Ilarraza, A. D.; Gojenola, K.; Oronoz, M.; and de Viñaspre, O. P. 2019. Interpretable deep learning to map diagnostic texts to ICD-10 codes. *International journal of medical informatics*, 129: 49–59.

Bai, T.; and Vucetic, S. 2019. Improving Medical Code Prediction from Clinical Text via Incorporating Online Knowledge Sources. In *WWW*, 72–82.

Beltagy, I.; Peters, M. E.; and Cohan, A. 2020. Longformer: The Long-Document Transformer. *ArXiv*, abs/2004.05150.

Bodenreider, O. 2004. The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, 32(1): 267–270.

Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. In *NeurIPS*, 1877–1901.

Cai, P.; Liu, F.; Bajracharya, A.; Sills, J.; Kapoor, A.; Liu, W.; Berlowitz, D.; Levy, D.; Pradhan, R.; and Yu, H. 2022. Generation of Patient After-Visit Summaries to Support Physicians. In *Proceedings of the 29th International Conference on Computational Linguistics*.

Cao, P.; Chen, Y.; Liu, K.; Zhao, J.; Liu, S.; and Chong, W. 2020. HyperCore: Hyperbolic and Co-graph Representation for Automatic ICD Coding. In *ACL*, 3105–3114.

Dai, X.; Chalkidis, I.; Darkner, S.; and Elliott, D. 2022. Revisiting Transformer-based Models for Long Document Classification. *ArXiv*, abs/2204.06683.

Dang, V.; Bendersky, M.; and Croft, W. B. 2013. Two-stage learning to rank for information retrieval. In *ECIR*, 423–434.

De Cao, N.; Izacard, G.; Riedel, S.; and Petroni, F. 2021. Autoregressive Entity Retrieval. In *ICLR*.

Eisfeld, J. 2014. International statistical classification of diseases and related health problems. *Transgender Studies Quarterly*, 1(1-2): 107–110.

Gao, T.; Fisch, A.; and Chen, D. 2021. Making Pretrained Language Models Better Few-shot Learners. In *ACL-IJCNLP*, 3816–3830.

Guo, M.; Ainslie, J.; Uthus, D.; Ontanon, S.; Ni, J.; Sung, Y.-H.; and Yang, Y. 2022. LongT5: Efficient Text-To-Text Transformer for Long Sequences. In *Findings of NAACL*.

Gutierrez, B. J.; McNeal, N.; Washington, C.; Chen, Y.; Li, L.; Sun, H.; and Su, Y. 2022. Thinking about GPT-3 In-Context Learning for Biomedical IE? Think Again. *ArXiv*, abs/2203.08410.

Huang, C.-W.; Tsai, S.-C.; and Chen, Y.-N. 2022. PLM-ICD: Automatic ICD Coding with Pretrained Language Models. In *Clinical NLP*, 10–20.

Ji, S.; Cambria, E.; and Marttinen, P. 2020. Dilated Convolutional Attention Network for Medical Code Assignment from Clinical Text. In *Clinical NLP*, 73–78.

Ji, S.; Holtta, M.; and Marttinen, P. 2021. Does the Magic of BERT Apply to Medical Code Assignment? A Quantitative Study. *Computers in biology and medicine*, 139: 104998.

Johnson, A. E. W.; Pollard, T. J.; Shen, L.; wei H. Lehman, L.; Feng, M.; Ghassemi, M. M.; Moody, B.; Szolovits, P.; Celi, L. A.; and Mark, R. G. 2016. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3.

Kavuluru, R.; Rios, A.; and Lu, Y. 2015. An empirical evaluation of supervised learning approaches in assigning diagnosis codes to electronic medical records. *Artificial intelligence in medicine*, 65 2: 155–66.

Kim, B.-H.; and Ganapathi, V. 2021. Read, Attend, and Code: Pushing the Limits of Medical Codes Prediction from Clinical Notes by Machines. In *MLHC*, 196–208.

Koh, P. W.; Sagawa, S.; Marklund, H.; Xie, S. M.; Zhang, M.; Balsubramani, A.; Hu, W.; Yasunaga, M.; Phillips, R. L.; Gao, I.; et al. 2021. WILDS: A Benchmark of in-the-Wild Distribution Shifts. In *ICML*, 5637–5664.

Kwon, S.; Yang, Z.; and Yu, H. 2022. An Automatic SOAP Classification System Using Weakly Supervision And Transfer Learning. *ArXiv*, abs/2211.14539.

Larkey, L. S.; and Croft, W. B. 1996. Combining Classifiers in Text Categorization. In *SIGIR*, 289–297.

Le Scao, T.; and Rush, A. 2021. How many data points is a prompt worth? In *NAACL*, 2627–2636.

Lewis, P.; Ott, M.; Du, J.; and Stoyanov, V. 2020. Pretrained Language Models for Biomedical and Clinical Tasks: Understanding and Extending the State-of-the-Art. In *Clinical NLP*, 146–157.

Li, F.; Jin, Y.; Liu, W.; Rawat, B. P. S.; Cai, P.; and Yu, H. 2019. Fine-Tuning Bidirectional Encoder Representations From Transformers (BERT)–Based Models on Large-Scale Electronic Health Record Notes: An Empirical Study. *JMIR Medical Informatics*, 7.

Liu, Y.; Cheng, H.; Klopfer, R.; Gormley, M. R.; and Schaaf, T. 2021. Effective Convolutional Attention Network for Multi-label Clinical Document Classification. In *EMNLP*.

Lu, J.; Du, L.; Liu, M.; and Dipnall, J. 2020. Multi-label Few/Zero-shot Learning with Knowledge Aggregated from Multiple Label Graphs. In *EMNLP*, 2935–2943.

Luo, J.; Xiao, C.; Glass, L.; Sun, J.; and Ma, F. 2021. Fusion: Towards Automated ICD Coding via Feature Compression. In *Findings of ACL-IJCNLP*, 2096–2101.

McCallum, A. K. 1999. Multi-label text classification with a mixture model trained by EM. In *Text Learning Workshop*.

Meng, Z.; Liu, F.; Shareghi, E.; Su, Y.; Collins, C.; and Collier, N. 2022. Rewire-then-Probe: A Contrastive Recipe for Probing Biomedical Knowledge of Pre-trained Language Models. In *ACL*, 4798–4810. Dublin, Ireland: Association for Computational Linguistics.

Moradi, M.; Blagec, K.; Haberl, F.; and Samwald, M. 2021. GPT-3 Models are Poor Few-Shot Learners in the Biomedical Domain. *ArXiv*, abs/2109.02555.

Mullenbach, J.; Wiegreffe, S.; Duke, J.; Sun, J.; and Eisenstein, J. 2018. Explainable Prediction of Medical Codes from Clinical Text. In *NAACL*, 1101–1111.

Nogueira, R.; and Cho, K. 2019. Passage Re-ranking with BERT. *ArXiv*, abs/1901.04085.

O'malley, K. J.; Cook, K. F.; Price, M. D.; Wildes, K. R.; Hurdle, J. F.; and Ashton, C. M. 2005. Measuring diagnoses: ICD code accuracy. *Health services research*, 40 5 Pt 2: 1620–39.

Pascual, D.; Luck, S.; and Wattenhofer, R. 2021. Towards BERT-based Automatic ICD Coding: Limitations and Opportunities. In *BioNLP*, 54–63.

Peng, X.; Long, G.; Shen, T.; Wang, S.; and Jiang, J. 2020. Self-attention enhanced patient journey understanding in healthcare system. In *ECML-PKDD*, 719–735.

Ren, W.; Zeng, R.; Wu, T.; Zhu, T.; and Krishnan, R. G. 2022. HiCu: Leveraging Hierarchy for Curriculum Learning in Automated ICD Coding. *arXiv:2208.02301*.

Rongali, S.; Jagannatha, A.; Rawat, B. P. S.; and Yu, H. 2020. Continual Domain-Tuning for Pretrained Language Models. *arXiv:2004.02288*.

Schick, T.; and Schütze, H. 2021. It's Not Just Size That Matters: Small Language Models Are Also Few-Shot Learners. In *NAACL*, 2339–2352.

Sivarajkumar, S.; and Wang, Y. 2022. HealthPrompt: A Zero-shot Learning Paradigm for Clinical Natural Language Processing. *ArXiv*, abs/2203.05061.

Song, C.; Zhang, S.; Sadoughi, N.; Xie, P.; and Xing, E. 2020. Generalized Zero-Shot Text Classification for ICD Coding. In *IJCAI*, 4018–4024.

Stupp, D.; Barequet, R.; Lee, I.-C.; Oren, E.; Feder, A.; Benjamini, A.; Hassidim, A.; Matias, Y.; Ofek, E.; and Rajkomar, A. 2022. Structured Understanding of Assessment and Plans in Clinical Documentation. *medRxiv*.

Sun, W.; Ji, S.; Cambria, E.; and Marttinen, P. 2021. Multitask Balanced and Recalibrated Network for Medical Code Prediction. *ArXiv*, abs/2109.02418.

Sutskever, I.; Vinyals, O.; and Le, Q. V. 2014. Sequence to Sequence Learning with Neural Networks. In Ghahramani, Z.; Welling, M.; Cortes, C.; Lawrence, N.; and Weinberger, K., eds., *NeurIPS*. Curran Associates, Inc.

Taylor, N.; Zhang, Y.; Joyce, D. W.; Nevado-Holgado, A. J.; and Kormilitzin, A. 2022. Clinical Prompt Learning with Frozen Language Models. *ArXiv*, abs/2205.05535.

Thakur, N.; Reimers, N.; Ruckl'e, A.; Srivastava, A.; and Gurevych, I. 2021. BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models. *ArXiv*, abs/2104.08663.

Tsai, S.-C.; Chang, T.-Y.; and Chen, Y.-N. 2019. Leveraging Hierarchical Category Knowledge for Data-Imbalanced Multi-Label Diagnostic Text Understanding. In *LOUHI*.

van Aken, B.; Papaioannou, J.-M.; Mayrdorfer, M.; Budde, K.; Gers, F.; and Loeser, A. 2021. Clinical Outcome Prediction from Admission Notes using Self-Supervised Knowledge Integration. In *EACL*, 881–893.

Wada, S.; Takeda, T.; Manabe, S.; Konishi, S.; Kamohara, J.; and Matsumura, Y. 2020. A Pre-Training Technique to Localize Medical BERT and to Enhance Biomedical BERT. *arXiv: Computation and Language*.

Wan, D.; and Bansal, M. 2022. FactPEGASUS: Factuality-Aware Pre-training and Fine-tuning for Abstractive Summarization. In *NAACL*, 1010–1028.

Wang, K.; Chen, X.; Chen, N.; and Chen, T. 2020a. Automatic Emergency Diagnosis with Knowledge-Based Tree Decoding. In *IJCAI*, 3407–3414.

Wang, S.; Ren, P.; Chen, Z.; Ren, Z.; Nie, J.-Y.; Ma, J.; and de Rijke, M. 2020b. Coding Electronic Health Records with Adversarial Reinforcement Path Generation. In *SIGIR*.

Wang, T.; Zhang, L.; Ye, C.; Liu, J.; and Zhou, D. 2022. A Novel Framework Based on Medical Concept Driven Attention for Explainable Medical Code Prediction via External Knowledge. In *Findings of ACL*, 1407–1416.

Wang, W.; Wei, F.; Dong, L.; Bao, H.; Yang, N.; and Zhou, M. 2020c. MiniLM: Deep Self-Attention Distillation for Task-Agnostic Compression of Pre-Trained Transformers. In *NeurIPS*, 5776–5788.

Weber, G. M.; Adams, W. G.; Bernstam, E. V.; Bickel, J. P.; Fox, K. P.; Marsolo, K. A.; Raghavan, V. A.; Turchin, A.; Zhou, X.; Murphy, S. N.; and Mandl, K. D. 2017. Biases introduced by filtering electronic health records for patients with "complete data". *Journal of the American Medical Informatics Association*, 24: 1134–1141.

Yadav, S.; Gupta, D. K.; and Demner-Fushman, D. 2022. CHQ-Summ: A Dataset for Consumer Healthcare Question Summarization. *ArXiv*, abs/2206.06581.

Yan, C.; Fu, X.; Liu, X.; Zhang, Y.; Gao, Y.; Wu, J.; and Li, Q. 2022. A survey of automated International Classification of Diseases coding: development, challenges, and applications. *Intelligent Medicine*.

Yang, X.; Nejatian, N. P.; Shin, H. C.; Smith, K.; Parisien, C.; Compas, C.; Flores, M.; Zhang, Y.; Magoc, T.; Harle, C.; et al. 2022a. GatorTron: A Large Clinical Language Model to Unlock Patient Information from Unstructured Electronic Health Records. *medRxiv*.

Yang, Z.; Wang, S.; Rawat, B. P. S.; Mitra, A.; and Yu, H. 2022b. Knowledge Injected Prompt Based Fine-tuning for Multi-label Few-shot ICD Coding. *ArXiv*, abs/2210.03304.

Yang, Z.; and Yu, H. 2020. Generating Accurate Electronic Health Assessment from Medical Graph. In *Findings of EMNLP*, 3764–3773.

Yao, Z.; Cao, Y.; Yang, Z.; and Yu, H. 2022. Context Variance Evaluation of Pretrained Language Models for Prompt-based Biomedical Knowledge Probing. *ArXiv*, abs/2211.10265.

Yuan, H.; Yuan, Z.; and Yu, S. 2022. Generative Biomedical Entity Linking via Knowledge Base-Guided Pre-training and Synonyms-Aware Fine-tuning. In *NAACL*, 4038–4048.

Yuan, Z.; Tan, C.; and Huang, S. 2022. Code Synonyms Do Matter: Multiple Synonyms Matching Network for Automatic ICD Coding. In *ACL*, 808–814.

Zhang, J.; Zhao, Y.; Saleh, M.; and Liu, P. J. 2020. PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization. *ArXiv*, abs/1912.08777.

Zhang, Z.; Liu, J.; and Razavian, N. 2020. BERT-XML: Large Scale Automated ICD Coding Using BERT Pretraining. In *ClinicalNLP*, 24–34.

Zhou, G.; and Devlin, J. 2021. Multi-Vector Attention Models for Deep Re-ranking. In *EMNLP*, 5452–5456.

Zhou, T.; Cao, P.; Chen, Y.; Liu, K.; Zhao, J.; Niu, K.; Chong, W.; and Liu, S. 2021. Automatic ICD Coding via Interactive Shared Representation Networks with Self-distillation Mechanism. In *ACL-IJCNLP*, 5948–5957.

Ziletti, A.; Akbik, A.; Berns, C.; Herold, T.; Legler, M.; and Viell, M. 2022. Medical Coding with Biomedical Transformer Ensembles and Zero/Few-shot Learning. *ArXiv*, abs/2206.02662.