Memorize and Rank: Elevating Large Language Models for Clinical Diagnosis Prediction

Mingyu Derek Ma[1] **Xiaoxuan Wang**[1] **Yijia Xiao**[1] **Anthony Cuturrufo**[1]
Vijay S Nori[2] **Eran Halperin**[2] **Wei Wang**[1]
[1]UCLA [2]Optum AI
{ma, xw27, yijia.xiao, acc, weiwang}@cs.ucla.edu
vijay.nori@optum.com eran.halperin@uhg.com
mera-diagnosis.github.io

Abstract

Clinical diagnosis prediction models, when provided with a patient's medical history, aim to detect potential diseases early, facilitating timely intervention and improving prognostic outcomes. However, the inherent scarcity of patient data and large disease candidate space often pose challenges in developing satisfactory models for this intricate task. The exploration of leveraging Large Language Models (LLMs) for encapsulating clinical decision processes has been limited. We introduce MERA, a clinical diagnosis prediction model that bridges pertaining natural language knowledge with medical practice. We apply hierarchical contrastive learning on a disease candidate ranking list to alleviate the large decision space issue. With concept memorization through fine-tuning, we bridge the natural language clinical knowledge with medical codes. Experimental results on MIMIC-III and IV datasets show that MERA achieves the state-of-the-art diagnosis prediction performance and dramatically elevates the diagnosis prediction capabilities of generative LMs.

1 Introduction

Electronic Health Records (EHR), containing patient status and diagnoses, embody valuable domain expertise and clinical operation patterns [7]. Clinicians make diagnosis judgments based on their extensive medical knowledge, acquired through years of education from textbooks and literature, as well as their accumulated experience derived from clinical practice. Clinical diagnosis prediction aims to predict patients' diseases that are highly likely to be diagnosed in the upcoming hospital admission by analyzing the patients' past diagnoses. The input and output are both presented in sequences of medical codes, which do not directly convey semantic information nor disease property. The resulting AI-enhanced diagnosis system [34] may enable early warning of diseases [45], optimized clinical resource allocation [58], and better risk estimation for sustainable insurance [14].

Two primary challenges in diagnosis prediction have driven various research efforts [54] but remain unsolved. First, what would be the best practice to incorporate clinical knowledge into the model? Existing works initialize concept embeddings from natural language descriptions [56, 4], or enrich patient representation with external disease ontologies [1, 8]. However, a significant gap persists between the primary knowledge modality, *i.e.* natural language, and the model's hidden representation. Second, how can we handle the large candidate space when making predictions and exploit the supervisory signals induced from this candidate space? The commonly used International Classification of Diseases (ICD) coding system encodes 13k+ diseases [6]. Existing works typically treat the task as k-way classification where k is the number of possible diseases, and then apply cross

GenAI for Health Workshop @ NeurIPS 2024, Vancouver.

entropy loss for each disease individually. These approaches often overlook the dependencies among diseases and the structural nuances within the diagnosis coding system.

Generative Language Models (LM), especially the Large Language Models (LLM), are trained to predict the next token, adhere to task instructions [5, 31], and align with human preferences [39]. These models exhibit superior capabilities in language understanding and reasoning, as shown by their performance on science-based benchmarks [32, 55, 61]. During the pretraining stage, LLMs assimilate a large amount of knowledge extracted from literature and online corpora. However, there remains an underexplored domain in using LLM for clinical diagnosis prediction, due to the aforementioned gap between natural language and medical code, as well as the disparity between the token-level optimization process and the large candidate outcome space. These challenges impede the effective application of generative LMs to diagnosis prediction tasks, even as the state-of-the-art models predominantly rely on graph neural networks without fully harnessing natural language knowledge [60, 56, 1]. Fine-tuning generative LM LLaMA2 [50] directly on diagnosis prediction yields almost 20-point lower recall@20 than GNN-based existing best model [60] as shown in Table 1. There are some studies that use transformer-based LM for clinical outcome prediction, but they either do not support structured data as input [37, 51], not compatible with mainstream LLMs [46, 13], or only work for narrow output space with few classes [51, 48].

To tackle these challenges, we propose MERA, an LLM designed for clinical diagnosis prediction that incorporates a comprehensive understanding of clinical knowledge by leveraging relationships among medical codes and offers extensive supervision over the output space. The patient's historical diagnosis results are formulated as linear sequences and the LLM is tasked with generating a probability distribution for the diagnosis results in the subsequent visit. Compared with the ordinary paradigm that optimizes the probability of generating the correct token, we optimize the outcome directly. To enhance the inter-visit causal reasoning, we employ contrastive learning to compel the model to distinguish true diagnoses from false ones. The contrastive learning process is extended to multiple levels in the hierarchical organization of the medical codes within the ICD coding ontology. The model is learned to distinguish the true diagnoses from a pool of potential diagnoses while the pool is increasingly relevant to the true ones. To regularize the diagnosis predictions to follow intra-visit diagnosis patterns, we develop a teaching-forcing strategy to optimize the medical code ranking, assuming partial diagnoses of the visit are known. To allow the model to grasp the comprehensive clinical semantics and diagnosis property of each medical code, we fine-tune the LM to "memorize" the mapping between medical codes and their natural language definitions. Consequently, this process bridges the gap between raw codes and their contextual medical meanings and equips the LM to capture the intricate code dependencies that are crucial for precise diagnosis assessments.

We validate the effectiveness of MERA in general diagnosis and heart failure prediction tasks on the patient records in MIMIC-III [17] and MIMIC-IV [16] datasets. MERA yields significant improvements over the existing state-of-the-art models across tasks on all datasets while having almost perfect memorization of bidirectional medical code-definition mapping. An extensive analysis of leading LLM's medical code understanding and diagnosis prediction capabilities is conducted, and we observe that GPT-4 is still far behind fine-tuned models on both tasks. We further conduct ablation studies to validate the effectiveness of the proposed novel design choices.

2 Preliminaries

2.1 Task Formulations

MERA can be applied for any task whose output is a collection of candidates belonging to a pre-defined decision space. We introduce widely used diagnosis prediction settings as typical testbeds for MERA [60].

Tasks. The first task is a general **diagnosis prediction** task, in which we aim to predict the diagnoses for the patient's potential next visit V_{T+1} given patient's history diagnoses by selecting a set of medical codes from the medical code ontology O, which can be formally described as f_{DP} : $\{V_1, V_2, \ldots, V_T\} \rightarrow V_{T+1}$. The second task is a disease-specific **heart failure prediction** task, which can be described as a binary classification function $f_{HF} : \{V_1, V_2, \ldots, V_T\} \rightarrow 0, 1$. We are more focused and aim to predict whether a patient would encounter heart failure (ICD-9 codes with head `428`) in any of the future visits.

Input patient record. Given an EHR collection of n patients $\{P_1, P_2, \ldots, P_n\}$, patient historical diagnosis can be represented as a sequence of admissions in chronological order P =

$\{V_1^P, V_2^P, \ldots, V_T^P\}$ where T is the number of existing visits. For a particular visit V, the medical judgment made by clinicians as a result of the visit is an unordered set of diagnoses $V = \{d_1^V, d_2^V, \ldots, d_{|V|}^V\}$ in the format of $|V|$ unique medical code ($d \in O$). The task input has two variants, including 1) history diagnosis *codes* only, and 2) additionally providing patient profile (gender, race, medication and family history) as a *natural language sentence*.

Medical code ontology as the decision space. The International Classification of Diseases (ICD) [12] provides a comprehensive ontology O diseases, symptoms and diagnoses. Each leaf node represents a unique disease/diagnosis and is assigned a unique medical code $c \in \{c_1, c_2, \ldots, c_{|O|}\}$ where $|O|$ is the total number of codes. Diseases are organized into disease groups at multiple levels, represented by non-leaf nodes forming a tree hierarchy $G = \{G_{\text{level=0}}, G_{\text{level=1}}, \ldots, G_{\text{level=}depth(O)}\}$. Assuming the root of O is at level 0, at level $j > 0$, there are $|G_{\text{level=}j}|$ disjoint disease groups, *i.e.* $G_{\text{level=}j} = \{g_1, \ldots, g_{|G_{\text{level=}j}|}\}$. There is also a one-to-one mapping between a code c and its natural language definition def_c. For example, in version 9 of ICD, the medical code `250.23` stands for "Diabetes with hyperosmolarity, type I [juvenile type], uncontrolled". It belongs to the first-level group for all "Endocrine, Nutritional, and Metabolic Diseases and Immunity Disorders", and further belongs to the fine-grained disease group "type I uncontrolled diabetes". We use both ICD-9 and ICD-10 coding systems with 13k+ and 68k+ unique codes in this work.

2.2 Existing Paradigm of Generative LMs

The ordinary formulation of generative LMs takes the input sequence $seq_{in} = t_1^{in}, \ldots, t_{|seq_{in}|}^{in}$ and is expected to generate the ground-truth output $seq_{out} = t_1^{out}, \ldots, t_{|seq_{out}|}^{out}$. It produces a probability distribution $P\left(c \mid t_{1:|seq_{in}|}^{in}, \hat{t}_{1:k}^{out}\right)$ over the possible next token ($c \in V$) conditioned on both the input sequence and k generated tokens. Discrete tokens at each autoregressive decoding step are produced by Equation 1. The LM is optimized to minimize the cross-entropy loss shown in Equation 2 applied on the probability of the *gold* next token conditioned on the *gold* output tokens in the previous segment in a teacher-forcing manner, assuming the $|seq_{out}|$-th token marks the end of the decoding.

$$\hat{t}_{k+1}^{out} = \text{argmax}_{c \in V} P\left(c \mid t_{1:|seq_{in}|}^{in}, \hat{t}_{1:k}^{out}\right) \tag{1}$$

$$\mathcal{L}_{CE} = \sum_{k=0}^{|seq_{out}|} -\log P\left(t_{k+1}^{out} \mid t_{1:|seq_{in}|}^{in}, t_{1:k}^{out}\right) \tag{2}$$

3 MERA: Learning to Memorize and Rank

MERA builds upon a large language model LM after pre-training on a natural language corpus, instruction tuning, and potential alignment process. MERA is designed to be compatible with numerous generative LM architectures and inherit knowledge obtained through pre-training, including encoder-decoder LM and decoder-only LM. There are three steps involved as a pipeline: 1) Fine-tuning the model to memorize medical codes used to represent the diagnoses; 2) Further optimizing the model to learn inter-visit causal and temporal relations between patient visits as well as intra-visit patterns from patient history records; 3) During inference, performing autoregressive generation to produce diagnosis predictions given an unseen patient history input.

3.1 Medical Code Memorization

State-of-the-art LLMs struggle to associate medical codes with their correct definitions accurately. GPT-4 can only recall 45% of ICD-9 codes given corresponding definitions (row 3 of Table 2), which may be attributable to the absence of medical codes in the pre-training dataset. MERA explicitly teaches LM the semantic information associated with the medical codes and the relationships within the coding system. We consider all codes in O as special tokens, each unique medical code has a dedicated token embedding and can be represented by a single token. This design reduces the noise of the learning objectives as the diagnosis probability is equivalent to the token probability. The memorization process parameterizes embeddings of the special tokens and further equips the LM with the necessary external knowledge to facilitate downstream diagnosis prediction. To integrate information about medical codes in O and the natural language knowledge contained in LM, we fine-tune LM on synthetic question-answering pairs.

Bidirectional code and definition memorization. For each code c and the natural language definition def_c, we create two input-output pairs. The first pair includes "What is the definition of ICD-9 code c" as seq_{in} and the target answer "def_c" as seq_{out} to train the model to recall its definition given a

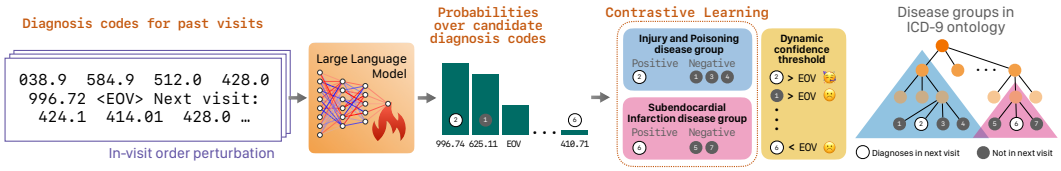

Figure 1: The model design of MERA. The diagnosis probability distribution is induced from token probabilities. It is optimized with hierarchical contrastive learning and dynamic cross-entropy losses.

code. The second pair helps the model memorize the inverse mapping. The question-answer pairs are created according to the O ontology being for the downstream task.

Decision space structure memorization. We further embed code dependencies collectively in LM by training with separate code-category instances. The curated pairs connect a code to its disease groups at various levels $1, \ldots, depth(O)$ in the code ontology O. For example, seq_{in} is "What is the chapter level disease group of the ICD-9 code 998.51?", and seq_{out} is "Injury and Poisoning".

3.2 Seq2seq Data Construction

The second phase aims to equip LM with a temporal and causal understanding of the diagnoses across multiple visits. We train the LM with a collection of sequence-to-sequence training instances $\mathbb{X} = \{\mathbf{X}_1, \ldots, \mathbf{X}_{n_{\text{patient}}}\}$ based on n_{patient} patient records, where $\mathbf{X}_i$ is a set of (diagnosis history, future diagnosis) pairs created based on patient record P_i.

Given the history of a patient containing T visits $P_i = \{V_1^{P_i}, \ldots, V_T^{P_i}\}$, we extract $T-1$ pairs of patient history and the expected diagnoses in the next visit to have maximum utilization of the patient records. For each pair, an input sequence is verbalized from 1-to-k visits following $seq_{in} = instruction, vb(V_1^{P_i}), \ldots, vb(V_k^{P_i})$ ($k \in [1, T-1]$). Additional patient profile sentences can be inserted following the instructions. A ground-truth output sequence is converted from expected diagnoses in the $(k+1)$-th visit following $seq_{out} = vb(V_{k+1}^{P_i})$. The verbalizer function vb concatenates the diagnosis codes within each visit to form a token segment for a specific visit and further prepend the starting prompt phrase ("The diagnosis codes for this visit are: ") and append a special token `EOV` representing "the end of the visit". We show the full prompt design in Appendix C.

Diagnoses order perturbation. The order of patient visits is crucial to convey the dependent relations as a diagnosis in a later visit is conditioned on the previous diagnoses. However, the order of diagnosis codes *within* a particular visit does not carry cognitive rationale as indicated by EHR dataset documentation and papers [16] (more details are in Appendix B.1). An ideal model should preserve the inter-visit orders while ignoring the intra-visit orders. To achieve this goal with a sequential LM, we propose to create n_{perturb} variants of the input patient history sequences and output diagnosis sequences respectively, leading to n_{perturb}^2 diverse combinations. Each variant keeps the same visit order but randomly shuffles the diagnosis codes within each visit. By observing the data instances with shuffled orders and the same target distribution, we teach the LM to ignore the order of diagnosis codes with a model-agnostic design. To summarize, the training sequence-to-sequence data $\mathbb{X}$ contains data instances $\mathbf{X}$ generated according to n_{patient} patient history records. $\mathbf{X}$ contains $T-1$ groups of data instances with different patient history lengths, each group contains combinations among n_{perturb} perturbed input sequences and n_{perturb} perturbed output sequences.

3.3 Learning Inter-visit Reasoning

Up to this point, the created seq2seq data instances can be used to conduct supervised fine-tuning of LM following token-level optimization used in conventional generative LM reiterated in §2.2. However, as we analyze theoretically (in §1) and demonstrate empirically (line 14/15 of Table 1), vanilla generative LM does not handle the diagnosis prediction task well. We propose multiple specialized learning objectives to learn the *inter-visit reasoning* to infer upcoming diagnoses and capture *intra-visit diagnosis patterns*. We bridge the sequential modeling capabilities and LM's internal knowledge with the task property and decision space structure (*e.g.* , ICD hierarchy) for diagnosis prediction.

After encoding seq_{in} containing information on existing hospital visits, the LM starts to generate its prediction of the upcoming visit $\hat{seq}_{out}$. As an immediate step, it produces a probability distribution over the possible next token t_1^{out} conditioned on seq_{in}, reflecting the possibility of different tokens in

the vocabulary as one of the diagnoses for visit V_{T+1}. Legit candidate tokens for t_1^{out} are the special code tokens, including $\{c_1, c_2, ..., c_{|O|}\}$. We select the probabilities of all code tokens and then apply softmax, resulting in the probability distribution over the candidate codes

$$P\left(c \mid t^{in}_{1:|seq_{in}|}\right) = \{p_{c_1}, p_{c_2}, ..., p_{c_{|O|}}\}, c \in O. \tag{3}$$

Hierarchical contrastive learning. We design training objectives to identify the real diagnoses among a group of similar candidate diagnoses. With such a design, the model is forced to understand the subtle differences among neighbor diseases in O and learn to infer upcoming diagnoses from a candidate pool under the same disease group.

For a training instance $\mathbf{X}_i$, we first identify all disease groups that the diagnoses of the next visit belong to $G_{\mathbf{X}_i} = \{G_{\text{level=0}}, G_{\text{level=1}}, \ldots, G_{\text{level=}depth(O)}\}$. Then, for each group g_k at level j ($g_k \in G_{\text{level=}j}$), we identify positive diagnosis codes $g_k^{pos} = \{c_1^{pos}, \ldots, c_{|g_k^{pos}|}^{pos}\}$, which are the diseases in g_k that are diagnosed in the next visit. We then use all remaining diseases in g_k as negative codes $g_k^{neg} = g_k - g_k^{pos} = \{c_1^{neg}, \ldots, c_{|g_k^{neg}|}^{neg}\}$. Then, we calculate an InfoNCE loss [38, 29, 33] term for each group in $G_{\mathbf{X}_i}$ and aggregate all the terms to be the aggregated objective $\mathcal{L}_{CL}$.

$$\mathcal{L}_{CL}^{g_k} = -\log \frac{\sum_{c_m^{pos} \in g_k^{pos}} P\left(c_m^{pos} \mid t^{in}_{1:|seq_{in}|}\right)}{\sum_{c_m \in g_k} P\left(c_m \mid t^{in}_{1:|seq_{in}|}\right)} \tag{4}$$

$$\mathcal{L}_{CL} = \frac{1}{|\mathbb{X}|} \sum_{\mathbf{X}_i \in \mathbb{X}} \sum_{G_{\text{level=}j} \in G_{\mathbf{X}_i}} \sum_{g_k \in G_{\text{level=}j}} \mathcal{L}_{CL}^{g_k} \tag{5}$$

The loss term for higher-level groups (where j is smaller) is used to enable the model to recognize disease scopes across a broad spectrum. Optimizing the high-level loss mimics the clinician's training process of making differential diagnoses, the "rough guesses" of possible diseases. Loss terms for lower-level groups focus on nuanced comparisons among diseases within the same family, increasing the model's ability to distinguish rare diseases. The proposed contrastive learning approach is efficient and capable in comparison to in-batch contrastive learning for two reasons: 1) The loss is calculated on the token probability distribution, essential for the typical decoding of generative LM, with no need for additional architecture or forward/backward passes. This ensures efficiency and maximum compatibility with the pre-trained LM. 2) The contradiction for loss calculation pertains to token probabilities, allowing the integration of prediction confidence for each disease into the optimization. This design differs significantly from in-batch contrastive learning, where forward and backward passes must be run for multiple data instances, and batch size significantly limits the size of positive and/or negative samples.

Dynamic confidence threshold. To produce a short list of *confident* diagnoses among the full ranking of *all* diagnosis codes, we learn a dynamic confidence threshold to select the most likely predictions. Existing works apply a fixed threshold to the probability distribution, which is often determined as a hyperparameter observed through the performance of the validation set [35, 44]. This widely used strategy makes shortlisting less flexible, and the model tends to play it safe and produces more diagnoses than it should. To model the confidence threshold dynamically, we use a special token `EOV` to mark the confidence threshold within the token probability ranking list. `EOV` was appended at the end of the diagnosis sequence of each visit as introduced in §3.2.

The model LM learns the placement of the `EOV` in two ways. Implicitly, the visit segments in the input sequence demonstrate that the special token `EOV` represents the end of a visit segment, implying the model should stop generating more diagnosis codes. Training with `EOV`-ended visit sequence segment, LM naturally learns to assign `EOV` a higher probability than other code tokens when the model is not confident to make more diagnoses and chooses to generate `EOV` to end the diagnosis sequence of a particular visit. Explicitly, we design a learning objective to train the LM to place the `EOV` token at the proper rank of the token probability distribution $P\left(c \mid t^{in}_{1:|seq_{in}|}\right)$. We identify the positive medical codes that do appear in the target visit as O^{pos} and the ones not included as O^{neg} ($O^{pos} + O^{neg} = O$). The $\mathcal{L}_{DCE}$ is essentially a dynamic cross-entropy loss that regularizes the probability of each positive code to be not smaller than the probability of `EOV` and further make sure the probability of each negative code is not larger than $P\left(\texttt{EOV} \mid t^{in}_{1:|seq_{in}|}\right)$. The optimization

of the dynamic confidence threshold applies fine-grained supervision to the probability distribution, enabling effective and efficient diagnosis capability learning with sparse patient data.

$$\begin{aligned}\mathcal{L}_{DCE} = &\sum_{c \in O^{pos}} \log\left(ReLU(P\left(\texttt{EOV} \mid t^{in}_{1:|seq_{in}|}\right) - P\left(c \mid t^{in}_{1:|seq_{in}|}\right))\right) \\ &+ \sum_{c \in O^{neg}} \log\left(ReLU(P\left(c \mid t^{in}_{1:|seq_{in}|}\right) - P\left(\texttt{EOV} \mid t^{in}_{1:|seq_{in}|}\right))\right)\end{aligned} \tag{6}$$

3.4 Learning Intra-visit Diagnosis Patterns

Besides training the model to reason between visits, there are many implicit rules and latent dependencies buried in the large pool of diagnoses within each visit. For example, within a group of similar diseases, the clinicians normally only choose the most representative code for the patient's status; some diseases might suppress or correlate with other diagnoses. Modeling the intra-visit dependencies enables us to incorporate real-life clinic operation patterns into realistic diagnosis predictions. The prediction made for a specific visit should consider other diagnoses of the same visit.

To model the intra-visit dependencies, we apply the objectives over the token probability distribution introduced in §3.3 to multiple training instance variants with partial output sequences as conditions. This enables teacher-forcing training. For each (seq_{in}, seq_{out}) pair in $\mathbf{X}_i$ for patient record P_i where the seq_{out} expresses all diagnoses in the visit $V^{P_i}_{k+1}, k \in [1, T-1]$, we create $|V^{P_i}_{k+1}|$ variants to move partial diagnosis results in seq_{out} to be part of the input of LM together with seq_{in}. Given the new input including the patient history and m known diagnoses in the upcoming visit, LM produces probability over the candidate medical code $P\left(c \mid t^{in}_{1:|seq_{in}|}, t^{out}_{1:m}\right)$. Since the m known diagnoses have been part of the input sequence, we remove the corresponding medical codes from the positive code set for the calculation of $\mathcal{L}_{DCE}$ and $\mathcal{L}_{CL}$ to prevent the model from generating duplicated codes. Formally, the conditions for probability P in Equation 3, 4, and 6 are $t^{in}_{1:|seq_{in}|}, t^{out}_{1:m}$ instead of $t_{1:|seq_{in}|}$. The m known diagnoses in $V^{P_i}_{k+1}$ are removed from g^{pos}_k, O^{pos} and added to g^{neg}_k and O^{neg}.

3.5 Training and Inference Pipeline

Training objectives. For code memorization, LM is trained with the ordinary cross-entropy loss in Equation 2. The hierarchical contrastive learning loss (Equation 5) is additionally applied to the instances whose output is a medical code. For the diagnosis prediction task, the LM fine-tuned from the memorization task is further optimized with the hierarchical contrastive learning loss (Equation 5) and the dynamic cross-entropy loss (Equation 6) on $|V^{P_i}_{k+1}|$ teaching force variants. Unlike language modeling, no loss has been applied to the reconstruction of the input segment for both fine-tuning stages. We perform full-parameter fine-tuning.

Autoregressive decoding. The produced LM can be used for inference on unseen patient history. Given seq_{in}, LM performs autoregressive decoding to output discrete diagnosis code with the highest probability in the ranking list for each output step until the EOV token is generated.

4 Experiments

4.1 Experimental Setup

Datasets. We use MIMIC-III [17] and MIMIC-IV [16] EHR datasets containing patient records to train and evaluate. The MIMIC-III dataset focuses on patients eventually admitted to the ICU, while the MIMIC-IV dataset includes both ICU patients and other patients. We conduct data preprocessing following previous works [25] and split the train/dev/test sets by patients to avoid information leak. Please refer to Appendix B.3 for data processing details and data statistics.

Metrics. We report the weighted F1 and recall@k, where k is the number of top-ranked predictions, and AUC and F1 for diagnosis prediction and heart failure, respectively.

Baselines. *RNN/CNN and attention-based models:* **RETAIN** [10], **Dipole** [28], **Timeline** [2], **HiTANet** [27], and **Deepr** [36]. *Graph-based models:* **GRAM** [9], **G-BERT** [47], **CGL** [26], **Chet** [25], and **MCDP** [20]. **KGxDP** [60] formulates each patient as a personalized medical KG, combining medical KGs with patient admission history. Note that additional medical notes are used by CGL, and additional Unified Medical Language System resource [3] is used as external knowledge by KGxDP. Please refer to Appendix B.4 for details about these baselines. *Transformer-based models:* We adapt two encoder-only LM. **RoBERTa** [24] with 125M and **MedBERT** [44] with 109M parameters and append a $|O|$-way classification head. We choose MedBERT among other similar

encoder-only architectures for medical sequence [40, 21, 46] because other models require additional input information such as lab test results which is not available under our setting. **Seq2seq** uses ordinary generative LM's formulation introduced in §2.2 to fine-tune a LM to generate diagnosis codes as output. We include definition sentences in the prompt following each code, so these baselines are exposed to the same external knowledge used by MERA.

Base LMs. We evaluate MERA using diverse LMs, including BioMistral [18] trained on PubMed Central, LLaMA2 [50], GPT-2 [42], T5 [43] and Flan-T5 [11].

Table 1: Diagnosis prediction comparison with baselines using ICD-9 as the decision space with code-only input (%).

#	Model	Diagnosis Prediction						Heart Failure			
		MIMIC-III			MIMIC-IV			MIMIC-III		MIMIC-IV	
		w-F1	R@10	R@20	w-F1	R@10	R@20	AUC	F1	AUC	F1
	RNN/CNN and attention-based models										
1	Deepr	18.87	24.74	33.47	24.08	26.29	33.93	81.36	69.54	88.43	61.36
2	Dipole	19.35	24.98	34.02	23.69	27.38	35.48	82.08	70.35	88.69	66.22
3	Timeline	20.46	25.75	34.83	25.26	29.00	37.13	82.34	71.03	87.53	66.07
4	RETAIN	20.69	26.13	35.08	24.71	28.02	34.46	83.21	71.32	89.02	67.38
5	HiTANet	21.15	26.02	35.97	24.92	27.45	36.37	82.77	71.93	88.10	68.21
	Graph-based models										
6	G-BERT	19.88	25.86	35.31	24.49	27.16	35.86	81.50	71.18	87.26	68.04
7	GRAM	21.52	26.51	35.80	23.50	27.29	36.36	83.55	71.78	89.61	68.94
8	CGL	21.92	26.64	36.72	25.41	28.52	37.15	84.19	71.77	89.05	69.36
9	MCDP	-	28.30	39.60	-	25.80	36.10	-	-	-	-
10	Chet	22.63	28.64	37.87	26.35	30.28	38.69	86.14	73.08	90.83	71.14
11	KGxDP	27.35	30.98	41.29	30.38	34.19	43.47	86.57	74.74	95.66	79.87
	Transformer-based models										
12	RoBERTa	17.39	22.84	32.07	22.54	24.89	32.38	79.74	68.28	87.03	60.21
13	MedBERT	19.01	23.68	34.39	24.13	25.88	33.81	81.06	69.96	88.73	61.81
14	Seq2seq (LLaMA2-7B)	18.05	18.38	23.56	20.47	20.77	24.19	77.62	66.06	85.98	59.14
15	Seq2seq (BioMistral-7B)	19.14	19.83	24.97	22.11	22.03	26.24	78.57	67.87	87.04	61.07
16	MERA (LLaMA2-7B)	32.77	35.94	47.48	34.64	38.16	46.94	89.49	77.21	97.26	82.31
17	MERA (BioMistral-7B)	**33.24**	**36.73**	**49.01**	**36.16**	**39.57**	**49.09**	**90.78**	**79.13**	**98.74**	**84.03**

4.2 Performance of Diagnosis Prediction

We show the performance comparison on the diagnosis prediction and heart failure prediction tasks (described in §2.1) using ICD-9 as decision space with history diagnosis code as input in Table 1 and the influence of base pre-trained LM selection in Table 2. We further show that MERA can be generalized to richer input with natural language patient profile, and the larger ICD-10 decision space in Table 3.

Encoder-only & vanilla generative LM perform poorly. The encoder-only LMs exhibit limited performance (rows 12-13 of Table 1), possibly because they do not account for the specialized modeling of intra-visit order and the extensive output space. When employing a vanilla generative LM (rows 14-15), the performance is further diminished. This is attributed to sparse supervision distributed in token-level loss. For each pass, only the probability of the single ground-truth token is optimized following Equation 2, while MERA optimizes the probabilities of all candidate diagnoses.

Gap between zero-shot and fine-tuned LMs. There remains a 20-point deficit in recall@20 comparing the best zero-shot LLM (row 3 of Table 2) to the fine-tuned model. This underscores the importance of leveraging patient data.

MERA is the state-of-the-art diagnosis prediction model. Finally, MERA achieves significantly better performance in both diagnosis and heart failure prediction tasks on both MIMIC datasets. MERA exhibits a 5.89 point higher weighted F1 score and almost 8 points higher recall@20 for MIMIC-III compared to the existing best model (row 17 vs 11 of Table 1). In Table 2, we showcase the diagnosis prediction performance using different pre-trained LMs, noting that even MERA with GPT-2 large (row 10) achieves comparable performance with the existing best KGxDP.

4.3 Performance on Medical Code Memorization

Table 2 shows the evaluation of the memorization results for the ICD-9 medical code system while using various base LMs. We report code and definition accuracy, indicating the proportion of correct

Table 2: Memorization and diagnosis prediction (after fine-tuning on the memorization task) results on MIMIC-III data using different pre-trained LMs.

#	Model	Med. Code Mem. Code Acc	Med. Code Mem. Def Acc	Diagnosis Pred. w-F1	Diagnosis Pred. R@20
	Zero-shot LM				
1	LLaMA2	4.69	0.61	5.62	15.64
2	GPT-3.5	33.50	9.31	6.11	17.07
3	GPT-4	45.16	48.48	6.46	21.56
	Fine-tuned encoder-decoder LM				
4	T5 base	81.71	1.26	20.53	30.13
5	T5 large	85.28	2.32	23.19	33.85
6	Flan-T5 base	88.58	0.19	21.01	32.24
7	Flan-T5 large	89.97	0.29	25.32	35.25
	Fine-tuned decoder-only LM				
8	GPT-2 base	0.00	95.68	23.29	32.06
9	GPT-2 medium	0.00	98.30	25.50	34.59
10	GPT-2 large	80.05	98.56	29.59	40.96
11	LLaMA2 7B	**99.87**	99.12	32.77	47.48
12	BioMistral 7B	99.61	**99.58**	**33.24**	**49.01**

Table 3: Diagnosis prediction results (recall@20, %) on the MIMIC-IV dataset using ICD-10 as the decision space with or without additional natural language patient profile.

Model	w NL info	w/o NL info
Chet	17.51	17.51
Seq2seq (BioMistral 7B)	16.31	13.47
MERA (BioMistral 7B)	43.66	40.39

output full ICD codes/definitions given their definitions/ICD codes as input by exact match. We observed that **1) Almost perfect medical code recall using large-enough 7B LM. 2) Pre-trained LLMs alone do not know medical codes well.** GPT models exhibit better memorization of medical codes compared to LLaMA2 (rows 1-3 of Table 2), but they still lag far behind the fine-tuned models (line 3 vs 12). **3) Model scaling-up boosts memorization.** Increasing models' parameters significantly enhances their memorization capabilities, as evidenced by an 80-point improvement in code accuracy from GPT-2 medium to large. However, this does not fully translate into improvement of the same magnitude in diagnosis prediction (row 9 vs 10 in Table 2). **4) Encoder-decoder vs decoder-only.** Comparing rows 4-7 with rows 8-12 in Table 2, we observe that encoder-decoder LMs tend to perform well on definition-to-code mapping while performing significantly worse on producing the accurate definition given the code. However, the observation is different for decoder-only LMs who can handle code-to-definition mapping at the early stage. Derived from these observations, it is optimal to use a large-size decoder-only LM as the backbone for diagnosis prediction.

4.4 Ablation Studies on Method Design

Knowledge injection approach. In rows 1-2 of Table 4, we observed that simply training the medical code sequence without providing meanings of the codes (row 1) leads to a 3.5-point lower recall@20. Providing the natural language definition of medical code in the input prompt along with the history diagnosis code (row 2 vs 1) is also helpful. However, the NL prompt method suffers from incomplete patient history due to the LM's input length limit, resulting in a 2.5-point lower recall@20 compared to memorization. Fine-tuning for concept memorization is the most effective knowledge injection approach.

Training objectives. Results in row 3-7 of Table 4 show that removing hierarchical contrastive learning leads to more than a 10-point drop in F1. Among the contrastive terms for disease groups categorized by different granularities, the 0-th level loss (row 4) is the most beneficial, which provides comparisons among the most involved diseases. The finest level loss (row 6) is the second most

Table 4: Ablation study on model design choices compared with full MERA (row 16 of Table 1) on MIMIC-III dataset.

#	Method Variant	w-F1	R@20
	Knowledge injection approach		
1	No external knowledge	-2.33	-3.54
2	Code definition in the prompt	-1.69	-2.46
	Training objectives		
3	w/o hierarchical contrastive learning	-10.34	-10.27
4	- w/o 0-th level CL loss only	-9.24	-8.4
5	- w/o chapter level CL loss only	-5.86	-4.08
6	- w/o finest level CL loss only	-7.74	-6.81
7	w/o dynamic confidence threshold	-4.10	-2.57
	Outputting strategies MERA = decode (our losses)		
8	Decode (cross-entropy loss)	-10.31	-17.33
9	Rank (cross-entropy loss)	-6.72	-13.32
10	Rank (our losses)	-2.63	-3.16

important, as the chapter-level disease is relatively easier to mine from data, while the fine-grained diagnosis decision involves distinguishing diseases that are similar in manifestation or etiology. Dynamic confidence threshold (row 7) also contributes more than 4-point F1 score improvement.

Outputting strategies. In rows 8-10 of Table 4, we explore optimal approaches to produce the diagnosis prediction set. LM can conduct autoregressive *decoding* to generate diagnosis codes as an output sequence. Alternatively, we can obtain the *ranking* list based on the token probability over the vocabulary of the first output token. Using decoding trained with sparse correct token cross-entropy loss (§2.2, row 8) compromises performance by 17 points in recall@20. The confusing in-visit diagnosis code order makes producing the result from the first token ranking list (row 9) a better choice than decoding along. When applying rich supervision with contrastive learning and dynamic confidence threshold, we observe a 10-point higher recall@20 with ranking output (row 10 vs 9). The comparison between row 10 and full MERA validates the effectiveness of intra-visit modeling, yielding a 3-point higher recall@20, where we decode token-by-token conditioned on other diagnoses but with specialized trained token probability for *each* decoding step.

5 Related Works

Diagnosis prediction. Existing works leverage structured diagnosis data [35]. They use sequential models like RNN and LSTM [10, 2] to model the longitudinal patient history and GNNs to encapsulate spatial features [41, 25]. To inject external knowledge, they conduct multi-task or transfer learning to borrow supervision from other tasks or domains [59, 62], use pre-trained embedding to incorporate natural language into initial features [56, 4], or utilizing external knowledge graphs or ontologies [1, 8, 22]. We propose to use the capable LLM architecture to learn patterns from patient history sequences and inject external knowledge with a unified and shared architecture across the pipeline. Existing works apply contrastive learning on intermediate latent for KG relations [1] or patient embedding [15], while we apply contrastive learning on diagnosis output space directly.

Transformer models for medical event prediction. Existing works either handle NL medical notes and other modalities [37, 62, 52, 23], or they use a non-unified architecture that cannot inherit the pretrained knowledge [46, 21, 40, 13] or needs adaptation for downstream tasks [49, 19, 30, 57]. [51, 48, 53] fine-tune the generative LM for classification tasks. We develop a model that is compatible with mainstream LLMs to use the pretrained knowledge and specializes in producing predictions from large diagnosis decision space.

6 Conclusion

MERA stands out by seamlessly integrating clinical knowledge and addressing the challenges associated with a large candidate space. Contrasting learning, tailored to the coding system's hierarchical structure, enables effective distinguishing between accurate and inaccurate diagnosis codes. Through validation on MIMIC datasets, MERA emerges as a leading approach to diagnosis prediction.

References

[1] Y. An, H. Tang, B. Jin, Y. Xu, and X. Wei. KAMPNet: Multi-source medical knowledge augmented medication prediction network with multi-level graph contrastive learning. *BMC Medical Informatics and Decision Making*, 23(1):243, Oct. 2023.

[2] T. Bai, S. Zhang, B. L. Egleston, and S. Vucetic. Interpretable Representation Learning for Healthcare via Capturing Disease Progression through Time. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 43–51, London United Kingdom, July 2018. ACM.

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

[4] A. Bornet, D. Proios, A. Yazdani, F. Jaume-Santero, G. Haller, E. Choi, and D. Teodoro. Comparing neural language models for medical concept representation and patient trajectory prediction, June 2023.

[5] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language Models are Few-Shot Learners, July 2020.

[6] D. J. Cartwright. ICD-9-CM to ICD-10-CM Codes: What? Why? How? *Advances in Wound Care*, 2(10):588–592, Dec. 2013.

[7] J. H. Caufield, Y. Zhou, Y. Bai, D. A. Liem, A. O. Garlid, K.-W. Chang, Y. Sun, P. Ping, and W. Wang. A comprehensive typing system for information extraction from clinical narratives. *medRxiv*, page 19009118, 2019.

[8] C. W. Cheong, K. Yin, W. K. Cheung, B. C. M. Fung, and J. Poon. Adaptive Integration of Categorical and Multi-relational Ontologies with EHR Data for Medical Concept Embedding. *ACM Transactions on Intelligent Systems and Technology*, Sept. 2023.

[9] E. Choi, M. T. Bahadori, L. Song, W. F. Stewart, and J. Sun. GRAM: Graph-based Attention Model for Healthcare Representation Learning. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 787–795, Halifax NS Canada, Aug. 2017. ACM.

[10] E. Choi, M. T. Bahadori, J. Sun, J. Kulas, A. Schuetz, and W. Stewart. RETAIN: An Interpretable Predictive Model for Healthcare using Reverse Time Attention Mechanism. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[11] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, A. Castro-Ros, M. Pellat, K. Robinson, D. Valter, S. Narang, G. Mishra, A. Yu, V. Zhao, Y. Huang, A. Dai, H. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, and J. Wei. Scaling Instruction-Finetuned Language Models, Dec. 2022.

[12] M. T. Cuadrado. Icd-9-cm: International classification of diseases, ninth revision, clinical modification, 2019.

[13] L. L. Guo, E. Steinberg, S. L. Fleming, J. Posada, J. Lemmon, S. R. Pfohl, N. Shah, J. Fries, and L. Sung. EHR foundation models improve robustness in the presence of temporal distribution shift. *Scientific Reports*, 13(1):3767, Mar. 2023.

[14] W. Hsu, S. X. Han, C. W. Arnold, A. A. Bui, and D. R. Enzmann. A data-driven approach for quality assessment of radiologic interpretations. *Journal of the American Medical Informatics Association*, 23(e1):e152–e156, 2016.

[15] H. Jeong, N. Oufattole, A. Balagopalan, M. Mcdermott, P. Chandak, M. Ghassemi, and C. Stultz. Event-Based Contrastive Learning for Medical Time Series, Dec. 2023.

[16] A. E. W. Johnson, L. Bulgarelli, L. Shen, A. Gayles, A. Shammout, S. Horng, T. J. Pollard, S. Hao, B. Moody, B. Gow, L.-w. H. Lehman, L. A. Celi, and R. G. Mark. MIMIC-IV, a freely accessible electronic health record dataset. *Scientific Data*, 10(1):1, Jan. 2023.

[17] A. E. W. Johnson, T. J. Pollard, L. Shen, L.-w. H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Anthony Celi, and R. G. Mark. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3(1):160035, May 2016.

[18] Y. Labrak, A. Bazoge, E. Morin, P.-A. Gourraud, M. Rouvier, and R. Dufour. BioMistral: A Collection of Open-Source Pretrained Large Language Models for Medical Domains, Feb. 2024.

[19] T. M. Lai, C. Zhai, and H. Ji. KEBLM: Knowledge-Enhanced Biomedical Language Models. *Journal of Biomedical Informatics*, 143:104392, July 2023.

[20] R. Li and J. Gao. Multi-modal Contrastive Learning for Healthcare Data Analytics. In *2022 IEEE 10th International Conference on Healthcare Informatics (ICHI)*, pages 120–127, Rochester, MN, USA, June 2022. IEEE.

[21] Y. Li, M. Mamouei, G. Salimi-Khorshidi, S. Rao, A. Hassaine, D. Canoy, T. Lukasiewicz, and K. Rahimi. Hi-BEHRT: Hierarchical Transformer-Based Model for Accurate Prediction of Clinical Events Using Multimodal Longitudinal Electronic Health Records. *IEEE Journal of Biomedical and Health Informatics*, 27(2):1106–1117, Feb. 2023.

[22] Y. Li, B. Qian, X. Zhang, and H. Liu. Knowledge guided diagnosis prediction via graph spatial-temporal network. In *Proceedings of the 2020 SIAM International Conference on Data Mining (SDM)*, Proceedings, pages 19–27. Society for Industrial and Applied Mathematics, Jan. 2020.

[23] S. Liu, X. Wang, Y. Hou, G. Li, H. Wang, H. Xu, Y. Xiang, and B. Tang. Multimodal Data Matters: Language Model Pre-Training Over Structured and Unstructured Electronic Health Records. *IEEE Journal of Biomedical and Health Informatics*, 27(1):504–514, Jan. 2023.

[24] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach, July 2019.

[25] C. Lu, T. Han, and Y. Ning. Context-Aware Health Event Prediction via Transition Functions on Dynamic Disease Graphs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(4):4567–4574, June 2022.

[26] C. Lu, C. K. Reddy, P. Chakraborty, S. Kleinberg, and Y. Ning. Collaborative Graph Learning with Auxiliary Text for Temporal Event Prediction in Healthcare. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pages 3529–3535, Montreal, Canada, Aug. 2021. International Joint Conferences on Artificial Intelligence Organization.

[27] J. Luo, M. Ye, C. Xiao, and F. Ma. HiTANet: Hierarchical Time-Aware Attention Networks for Risk Prediction on Electronic Health Records. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 647–656, Virtual Event CA USA, Aug. 2020. ACM.

[28] F. Ma, R. Chitta, J. Zhou, Q. You, T. Sun, and J. Gao. Dipole: Diagnosis Prediction in Healthcare via Attention-based Bidirectional Recurrent Neural Networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1903–1911, Halifax NS Canada, Aug. 2017. ACM.

[29] M. D. Ma, M. Chen, T.-L. Wu, and N. Peng. HyperExpan: Taxonomy expansion with hyperbolic representation learning. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4182–4194, Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics.

[30] M. D. Ma, A. Taylor, W. Wang, and N. Peng. DICE: Data-Efficient Clinical Event Extraction with Generative Models. In A. Rogers, J. Boyd-Graber, and N. Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15898–15917, Toronto, Canada, July 2023. Association for Computational Linguistics.

[31] M. D. Ma, X. Wang, P.-N. Kung, P. J. Brantingham, N. Peng, and W. Wang. Star: Boosting low-resource information extraction by structure-to-text data generation with large language models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17):18751–18759, Mar. 2024.

[32] M. D. Ma, C. Ye, Y. Yan, X. Wang, P. Ping, T. Chang, and W. Wang. Clibench: A multifaceted and multigranular evaluation of large language models for clinical decision making. June 2024.

[33] Y. Meng, C. Xiong, P. Bajaj, P. Bennett, J. Han, X. Song, et al. Coco-lm: Correcting and contrasting text sequences for language model pretraining. *Advances in Neural Information Processing Systems*, 34:23102–23114, 2021.

[34] M. A. Morid, O. R. L. Sheng, and J. Dunbar. Time Series Prediction Using Deep Learning Methods in Healthcare. *ACM Transactions on Management Information Systems*, 14(1):2:1–2:29, Jan. 2023.

[35] M. A. Morid, O. R. L. Sheng, and J. Dunbar. Time Series Prediction Using Deep Learning Methods in Healthcare. *ACM Transactions on Management Information Systems*, 14(1):1–29, Mar. 2023.

[36] P. Nguyen, T. Tran, N. Wickramasinghe, and S. Venkatesh. \mathtt Deepr: A Convolutional Net for Medical Records. *IEEE Journal of Biomedical and Health Informatics*, 21(1):22–30, Jan. 2017.

[37] S. Niu, J. Ma, L. Bai, Z. Wang, L. Guo, and X. Yang. EHR-KnowGen: Knowledge-enhanced multimodal learning for disease diagnosis generation. *Information Fusion*, 102:102069, Feb. 2024.

[38] A. v. d. Oord, Y. Li, and O. Vinyals. Representation learning with contrastive predictive coding. *ArXiv preprint*, abs/1807.03748, 2018.

[39] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, and R. Lowe. Training language models to follow instructions with human feedback, Mar. 2022.

[40] C. Pang, X. Jiang, K. S. Kalluri, M. Spotnitz, R. Chen, A. Perotte, and K. Natarajan. CEHR-BERT: Incorporating temporal information from structured EHR data to improve prediction tasks. In *Proceedings of Machine Learning for Health*, pages 239–260. PMLR, Nov. 2021.

[41] D. Proios, A. Yazdani, A. Bornet, J. Ehrsam, I. Rekik, and D. Teodoro. Leveraging patient similarities via graph neural networks to predict phenotypes from temporal data. In *2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 1–10, Oct. 2023.

[42] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language Models are Unsupervised Multitask Learners.

[43] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, Sept. 2023.

[44] L. Rasmy, Y. Xiang, Z. Xie, C. Tao, and D. Zhi. Med-BERT: Pretrained contextualized embeddings on large-scale structured electronic health records for disease prediction. *npj Digital Medicine*, 4(1):1–13, May 2021.

[45] C. M. Rochefort, D. L. Buckeridge, and A. J. Forster. Accuracy of using automated methods for detecting adverse events from electronic health record data: a research protocol. *Implementation Science*, 10(1):1–9, 2015.

[46] M. Rupp, O. Peter, and T. Pattipaka. ExBEHRT: Extended Transformer for Electronic Health Records. In H. Chen and L. Luo, editors, *Trustworthy Machine Learning for Healthcare*, Lecture Notes in Computer Science, pages 73–84, Cham, 2023. Springer Nature Switzerland.

[47] J. Shang, T. Ma, C. Xiao, and J. Sun. Pre-training of Graph Augmented Transformers for Medication Recommendation. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pages 5953–5959, Macao, China, Aug. 2019. International Joint Conferences on Artificial Intelligence Organization.

[48] O. B. Shoham and N. Rappoport. CPLLM: Clinical Prediction with Large Language Models. 2023.

[49] E. Steinberg, Y. Xu, J. Fries, and N. Shah. MOTOR: A Time-To-Event Foundation Model For Structured Medical Records, Sept. 2023.

[50] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M.-A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, and T. Scialom. Llama 2: Open Foundation and Fine-Tuned Chat Models, July 2023.

[51] H. Wang, C. Gao, C. Dantona, B. Hull, and J. Sun. DRG-LLaMA : Tuning LLaMA Model to Predict Diagnosis-related Group for Hospitalized Patients, Sept. 2023.

[52] X. Wang, J. Luo, J. Wang, Z. Yin, S. Cui, Y. Zhong, Y. Wang, and F. Ma. Hierarchical Pretraining on Multimodal Electronic Health Records, Oct. 2023.

[53] M. Wornow, R. Thapa, E. Steinberg, J. A. Fries, and N. H. Shah. EHRSHOT: An EHR Benchmark for Few-Shot Evaluation of Foundation Models, Nov. 2023.

[54] M. Wornow, Y. Xu, R. Thapa, B. Patel, E. Steinberg, S. Fleming, M. A. Pfeffer, J. Fries, and N. H. Shah. The shaky foundations of large language models and foundation models for electronic health records. *npj Digital Medicine*, 6(1):135, July 2023.

[55] C. Wu, W. Lin, X. Zhang, Y. Zhang, Y. Wang, and W. Xie. PMC-LLaMA: Towards Building Open-source Language Models for Medicine, Aug. 2023.

[56] J. Wu, K. He, R. Mao, C. Li, and E. Cambria. MEGACare: Knowledge-guided multi-view hypergraph predictive framework for healthcare. *Information Fusion*, 100:101939, Dec. 2023.

[57] J. Xu, M. D. Ma, and M. Chen. Can NLI Provide Proper Indirect Supervision for Low-resource Biomedical Relation Extraction? In A. Rogers, J. Boyd-Graber, and N. Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2450–2467, Toronto, Canada, July 2023. Association for Computational Linguistics.

[58] K. Yadav, E. Sarioglu, M. Smith, and H.-A. Choi. Automated outcome classification of emergency department computed tomography imaging reports. *Academic Emergency Medicine*, 20(8):848–854, 2013.

[59] K. Yang, Y. Xu, P. Zou, H. Ding, J. Zhao, Y. Wang, and B. Xie. KerPrint: Local-Global Knowledge Graph Enhanced Diagnosis Prediction for Retrospective and Prospective Interpretations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(4):5357–5365, June 2023.

[60] Z. Yang, Y. Lin, Y. Xu, J. Hu, and S. Dong. Interpretable Disease Prediction via Path Reasoning over medical knowledge graphs and admission history. *Knowledge-Based Systems*, 281:111082, Dec. 2023.

[61] Y. Zhang, S. Hou, M. D. Ma, W. Wang, M. Chen, and J. Zhao. Climb: A benchmark of clinical bias in large language models, 2024.

[62] H.-Y. Zhou, Y. Yu, C. Wang, S. Zhang, Y. Gao, J. Pan, J. Shao, G. Lu, K. Zhang, and W. Li. A transformer-based representation-learning model with unified processing of multimodal input for clinical diagnostics. *Nature Biomedical Engineering*, 7(6):743–755, June 2023.

A Potential Questions

Why not include more information, such as patient clinical notes, in the input? We experimented with two settings of input prompts, one with diagnosis code only and one with additional patient information as a natural language sentence. The text-code mixing input result shows that MERA is capable of handling natural language input along with diagnosis codes. Table 3 shows the performance with the additional patient profile as input.

Two main reasons motivate us to use the setting of using only diagnosis codes as input for diagnosis prediction, without additional records. 1) The setting is commonly used in existing literature (introduced in the baselines paragraph in §4.1 and §5), enabling fair head-to-head comparison with recent and state-of-the-art works. 2) The diagnosis code-only setting is realistic for many situations where additional patient health records, such as lab test results and medical notes, are not available. Billing-related stakeholders like insurance companies only have access to diagnosis codes included in insurance filing without access to other medical records. Predicting potential future diseases is an urgently needed capability.

How does MERA handle missing or incomplete patient histories and data sparsity? For missing or incomplete data, we filter out patient records that do not have a consistent diagnosis history from our training and evaluation data. We handle the missing data in the same way as a wide range of works, many listed under the "Baselines" paragraph in §4.1. To mitigate the influence of data sparsity, we create multiple training records from the record of a single patient described at the beginning of §3.2. For a patient with T visits, we create $T - 1$ pairs of patient history with various lengths and next-visit diagnoses. This enables us to exploit the patient records to the maximum utilization.

Code memorization seems not to be a challenging task given almost perfect performance? Code memorization is not challenging only after the optimal strategy of memorizing medical code definition is found. Simply performing memorization does not work (as shown in lines 4-9 in Table 2). Code memorization is part of our approach to improve the diagnosis prediction capabilities, instead of the ultimate task at hand. The high performance of code memorization indicates our approach is effective. We explore different recipes to conduct code memorization in Table 2. We observe that 1) encoder-decoder architecture does not work for definition recall, and 2) small size model cannot memorize well. These motivate us to use decoder-only large generative language models for the diagnosis prediction task.

Does the model compatible with encoder-only models like BERT? We did not adapt MERA to encoder-only models such as BERT since it would require us to remove some proposed techniques, such as intra-visit dependencies modeling.

Is it possible to apply the method to other tasks with different output space ontology? The coding system O to be used is an experimental setup choice. Our method design supports various coding systems. In our experiments, we show the results when using both ICD-9 (Table 1, Table 2 and Table 4) and ICD-10 (Table 3) as the coding system.

B Method, Implementation and Experiments Details

B.1 In-visit Code Order

Though the priority among the diagnosis code list for a visit is provided in the dataset, both the MIMIC-IV dataset paper [16] and dataset documentation[1] mentioned that there are few incentives for the operator to ensure the rank reflects the diagnosis's importance. This motivates us to train the model to ignore the order for most of the codes with perturbed code sequences.

B.2 Implementation Details

Loss over completion only. Instead of language modeling where the entire sequence is used to optimize the model, we only calculate loss over the completion part assuming the input is given. For example, for definition-to-code memorization, we only apply loss to let the model output the correct

[1] https://mimic.mit.edu/docs/iv/modules/hosp/diagnoses_icd/#seq_num

code. We do not apply the next token prediction for the question part and do not require the model to learn to reconstruct the question "What is the ICD-9 code with the definition ...".

Getting top-k predictions from free-text responses. The LM will general responses as an output sequence; we then parse the output sequence to a set of diagnosis codes separately by white space. The LM decides the number of predictions made by the LM. When calculating metrics like recall@20 and the number of prediction codes is less than 20 codes, we do not force the model to generate more codes.

B.3 Experimental Setup Details

We use the records of patients having multiple visits and use the complete medical code to represent the diagnosis. This poses a more challenging task compared to some existing studies that use simplified and higher-level codes. To prevent data duplication within the overlapping time frame of the two datasets, only the patient information from MIMIC-IV with multiple visits between 2013 and 2019 was used. In the validation and test sets, we designate the patients' last visit as the label for prediction, with the preceding visit(s) used to construct input.

For the settings using ICD-10 as decision space, we use a subset of patients in the MIMIC-IV dataset whose diagnosis records are all in ICD-10 to avoid potential errors during code version conversion. Records of 4277 patients are used for training, and the remaining 500 are used for evaluation. The input of this setting would be a sequence of ICD-10 codes, and the expected diagnoses to be predicted are also selected from a list of all ICD-10 codes.

We show the data statistics of the training and evaluation data used in our experiments in Table 5.

Dataset	MIMIC-III	MIMIC-IV
# unique patients	7493	10000
Train/valid/test splits	6000/493/1000	8000/1000/1000
Max. # visit	42	55
Avg. # visit	2.66	3.66
# unique diagnosis codes	4880	6102
Max. # codes per visit	39	50
Avg. # codes per visit	13.06	13.38

Table 5: Data statistics.

B.4 Baseline Details

RNN/CNN and attention-based models. **RETAIN** [10] employs two attention mechanisms to model two-way visit-disease mapping. **Dipole** [28] proposes a bidirectional RNN to address the issue of lengthy medical visit records. **Timeline** [2] designs an attention mechanism that combines time intervals and attention weights of each entity. **HiTANet** [27] employs a hierarchical temporal attention mechanism. **Deepr** [36] predicts future risks from medical records by converting records into discrete element sequences and using a CNN to detect predictive local clinical patterns.

Graph-based models. **GRAM** [9] employs the structure of medical ontologies. **G-BERT** [47] integrates pretrained language models and considers the hierarchical information of ICD codes. **CGL** [26] introduces a collaborative graph learning model. **Chet** [25] computes the diagnosis neighbor and global neighbor for each disease. **MCDP** [20] uses hyperbolic space to preserve the hierarchical structure of diagnostic codes. **KGxDP** [60] formulates each patient as a personalized medical KG, combining medical KGs with patient admission history. Note that additional medical notes are used by CGL, and additional Unified Medical Language System resource [3] is used as external knowledge by KGxDP.

We cannot reproduce MCDP [20], so we only report results in the paper.

B.5 Computational Complexity

We perform full parameter supervised fine-tuning without using any parameter-efficient training techniques as we observe that full parameter tuning leads to better performance. We use 4 A6000

GPUs for an average of 53 hours to train our model (line 15 of Table 1) until the validation F1 score does not improve.

C Complete Prompt Example

We show an exemplar input and output for the diagnosis prediction task. The green background segment is the output sequence. All the token that starts with "ICD9_" are special tokens that would map to a unique token representation. EOV is also a special token representing the end of a hospital visit.

The task is to predict the diagnosis codes for the next patient visit given the patient diagnosis history.
Patient history: Diagnosis codes for the visit: ICD9_443.9 ICD9_785.4 ICD9_585.9 ICD9_584.9 ICD9_250.70 ICD9_250.60 ICD9_357.2 ICD9_369.4 ICD9_403.90 ICD9_V58.67 ICD9_V12.59 EOV Diagnosis codes for the visit: ICD9_584.9 ICD9_427.5 ICD9_348.30 ICD9_276.2 ICD9_403.11 ICD9_428.22 ICD9_250.80 ICD9_428.0 ICD9_585.6 ICD9_790.4 ICD9_787.01 ICD9_285.21 ICD9_272.4 ICD9_782.3 ICD9_786.6 ICD9_794.31 ICD9_607.9 ICD9_608.9 ICD9_564.00 ICD9_V58.67 ICD9_E932.3 ICD9_V02.54 EOV
Diagnosis for the next visit:
ICD9_038.9 ICD9_585.6 ICD9_403.91 ICD9_428.20 ICD9_276.2 ICD9_995.91 ICD9_608.83 ICD9_428.0 EOV

D Limitations

We would like to raise awareness that there might be miscoded diagnosis codes in the patient records. The billing ICD diagnosis codes are used as "ground-truth" diagnosis decisions to train our model and evaluate the performance for diagnosis prediction. We acknowledge that the diagnosis code extracted from the EHR dataset should not be considered the best/perfect diagnosis decision. We also raise the potential data distribution issue as the training and evaluation data used in this work is largely collected for patients with ICU stay history. Thus, the evaluation result does not represent the generalized diagnosis prediction capability, and the trained model may yield compromised performance when different kinds of patient records are queried.

E Ethical Statement

While MERA demonstrates improved performance in diagnostic prediction tasks, the trained model may incorporate biases from multiple sources, including the pre-training corpus or the medical records distribution utilized for fine-tuning, and more. Therefore, the model necessitates comprehensive evaluation prior to its consideration for real-world clinical application. Additionally, the outcomes of the diagnostic prediction model may not be utilized to attribute discriminatory labels to specific diseases. Healthcare institutions and insurance entities may not use the predictive diagnoses of MERA as a basis for changing patient services.

F Broader Impact Statement

MERA contributes to the potential improvement of healthcare delivery and patient outcomes. By aiming to accurately predict diseases based on patient medical histories, these models offer a possibility for earlier detection and intervention, which might lead to better patient outcomes over time. Given the challenges associated with limited patient data and the complexity of diagnosing a wide range of diseases, MERA's approach, which leverages Large Language Models (LLMs) and hierarchical contrastive learning, represents a step towards addressing these issues. While its performance on the MIMIC datasets indicates promising results in diagnosis prediction, the real-world application of such models underscores the cautious optimism for AI's role in enhancing clinical decisions and healthcare efficiency. These developments suggest a direction where AI could support more informed clinical decisions, potentially improving patient care and management, albeit with ongoing evaluation and validation needed to fully realize these benefits.