MPLite: Multi-Aspect Pretraining for Mining Clinical Health Records

Eric Yang†
Department of Computer Science
Stevens Institute of Technology
Hoboken, NJ, USA
eyang6@stevens.edu

Pengfei Hu†
Department of Computer Science
Stevens Institute of Technology
Hoboken, NJ, USA
phu9@stevens.edu

Xiaoxue Han
Department of Computer Science
Stevens Institute of Technology
Hoboken, NJ, USA
xhan26@stevens.edu

Yue Ning*
Department of Computer Science
Stevens Institute of Technology
Hoboken, NJ, USA
yue.ning@stevens.edu

***Abstract*—The adoption of digital systems in healthcare has resulted in the accumulation of vast electronic health records (EHRs), offering valuable data for machine learning methods to predict patient health outcomes. However, single-visit records of patients are often neglected in the training process due to the lack of annotations of next-visit information, thereby limiting the predictive and expressive power of machine learning models. In this paper, we present a novel framework MPLite that utilizes Multi-aspect Pretraining with Lab results through a light-weight neural network to enhance medical concept representation and predict future health outcomes of individuals. By incorporating both structured medical data and additional information from lab results, our approach fully leverages patient admission records. We design a pretraining module that predicts medical codes based on lab results, ensuring robust prediction by fusing multiple aspects of features. Our experimental evaluation using both MIMIC-III and MIMIC-IV datasets demonstrates improvements over existing models in diagnosis prediction and heart failure prediction tasks, achieving a higher weighted-F_1 and recall with MPLite. This work reveals the potential of integrating diverse aspects of data to advance predictive modeling in healthcare.**

***Index Terms*—EHR, Lab Result, Diagnosis Prediction, Pretraining, Heart Failure Prediction**

I. Introduction

EHR datasets, such as MIMIC-III [1], provide comprehensive medical information, including vital signs, diagnoses, medications, and lab results. These multi-aspect features are valuable resources for predicting personalized health events, such as diagnosis predictions. Meanwhile, deep learning technique have become a common approach for analyzing sequential data within healthcare [2]–[4]. However, many studies often exclude patient examples with only single-visit records, since these records lack labels for prediction tasks involving future admissions. For instance, when training a supervised machine learning model to predict diagnoses in the next visit given previous visits in the MIMIC-III dataset, we need the annotations/labels for the next visit. Therefore, temporal prediction models rely on patient data with at least two visits to complete the training process. Single-visit records are not fully utilized in training predictive models as shown in Figure 1. However, multi-visit patients contribute to only a small portion of the dataset. Among a total of 46,520 patients, only 16.20% have multiple visits. The remaining 83.80% are single-visit patients, which could also provide rich information for models to learn useful patterns and make better predictions.

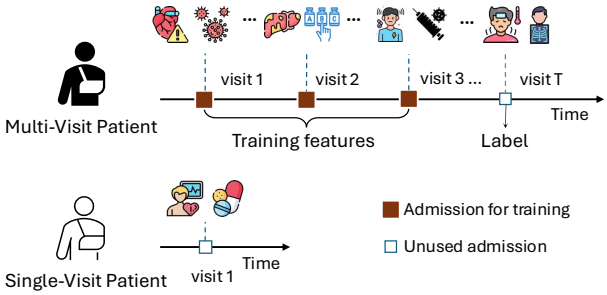

Fig. 1. Example of supervised training on patient-level admission records in most predictive models

To fully utilize these single admission records in an EHR dataset, there are two popular solutions to address this issue: (1) Transformer models like G-BERT [5] leverage single-admission data to design customized self-supervised tasks, which typically treat medical concepts or admissions as masked tokens and further enhance intermediate representation learning within the encoder framework. (2) Multi-aspect learning [6], [7] incorporates diverse features, such as lab test results or clinical notes, to enrich the representation learning of medical concepts, which helps models better capture the complexity and interrelationships inherent in medical data. The former approach, although widely adopted by early studies [8], [9], is susceptible to the order of medical codes and may not be lightweight enough to function as a plug-and-play module. In contrast, while the latter approach demands high-quality

† Equal contribution as co-first authors.
* Corresponding author

and relevant additional medical concepts, it enables models to learn collaborative representations, leading to more accurate predictions with the addition of a lightweight module.

In this study, we leverage single admissions as auxiliary training data to predict diagnoses and health risks, such as heart failure. Recognizing the pivotal role that lab test results play prior to training, we propose a novel framework, **MPLite**, which is an additional plug-in-and-play module to learn relationships between lab results and diagnoses through a **M**ulti-aspect **P**retraining and "**Lite**" module. This framework captures the underlying patterns and associations that are present in both multiple-visit and single-visit data. We then illustrate how incorporating this pre-trained knowledge can significantly enhance the predictive capabilities of temporal neural networks, particularly for forecasting health risks in patients with multiple visits. By fine-tuning the pre-trained subnetwork on two specific health risk prediction tasks, we demonstrate the effective extraction of valuable insights from abundant single-visit patient data. The pretraining module underscore the advantages of pretraining on diverse medical features beyond diagnoses concepts and highlights the broader applicability of lab test data in predictive healthcare.

II. Related Work

Deep learning models have been extensively applied to electronic health records (EHR) to extract representations of medical patterns, addressing various real-world healthcare prediction tasks like diagnosis prediction.

A. CNN/RNN-based Models

Most early studies in this area can be categorized into two main subcategories: (i) RNN-based models, where predictive methods like GRU [10], RETAIN [11], and Timeline [12] combine attention mechanisms and RNN for prediction. Other models [2], [13], [14] leverage RNNs to handle time-series data effectively; (ii) CNN-based models, such as Deepr [15] and AdaCare [16], use convolution and pooling layers to process features in EHR. However, these methods often overlook relations among encoded medical concepts and other critical aspects such as lab test results.

B. Graph/Transformer-based Models

Recently, there has been a trend towards using ontology graphs to incorporate additional information related to medical concepts in predictions such as GRAM [17], G-Bert [5], GCT [7], Variationally Regularized GNN [18], GraphCare [19], ME2Vec [20], RGNN [21]. However, most existing works primarily rely on admission medical concepts as features for various deep learning models. Meanwhile, following the success of the transformer architecture, researchers have quickly adopted it for EHR data. Encoder-decoder structures offer the advantage of fully utilizing single-visit data in the pretraining process by customizing proxy tasks for different prediction tasks. Early studies, like G-Bert [5] treat medical codes as tokens and incorporate hierarchical domain knowledge along with diagnosis codes. Recent models like HiTANet [22], Med-BERT [8], and Sherbet [23] have also been trained to precisely identify patient information based on various medical concepts. However, the pretraining phase in most works cannot be easily separated into a plug-and-play module, limiting its generalizability when transferring pretraining information to new tasks or different structures.

TABLE I
Notations Used in This Paper

Notation	Definition
$\mathcal{S}$	EHR dataset
P_i	i-th patient
$\mathcal{C}, \mathcal{D}, \mathcal{L}$	Sets of medical concepts, diagnosis codes, and lab test codes
$\|\mathcal{C}\|, \|\mathcal{D}\|, \|\mathcal{L}\|$	Cardinality of medical concepts, diagnoses, and lab test codes
T_i	The number of visits for patient $\mathbf{p_i}$
$\mathbf{x}_t, \mathbf{x}_t^D, \mathbf{x}_t^L$	Multi-hot vector for the t-th visit of a patient

C. Models with Mutli-Aspect Features

Beyond traditional medical concepts, such as condition, medication, and treatment codes, researchers [9] also involve additional information (e.g. demographic features and timestamps) in each admission record. To augment representation from different modalities, both CGL [6] and MedGTX [24] integrates disease-patient graphs and unstructured text from clinical notes through encoder structures to demostrates the importance of involving additional information other than sequence of medical concepts. MiME [25] and GCT [7] are preliminary tries to involve lab results as input features to further optimize medical hidden representation. However, these integrated models cannot work well in the absence of corresponding records, and they always have complicated preprocessing or fusion steps which cannot be generalized as lightweight modules.

In this paper, we propose **MPLite** that allows different models to jointly learn representations of medical diagnosis codes and lab results. Our framework provides a novel perspective for integrating different features to achieve more accurate predictions. The experimental results demonstrate a significant improvement with the extensive pretraining module in predicting health outcomes over several baselines, as confirmed by confidence intervals obtained from repeated experiments.

III. Proposed Method

We begin by describing the notations and then introduce our proposed framework, which includes a pretraining module with lab results, along with instructions on how to seamlessly integrate and utilize the module for downstream tasks.

A. General Notations

An EHR dataset $\mathcal{S}$ is a collection of patient admission records of N patients $\{P_1, P_2, ..., P_N\} \in \mathcal{S}$ in total. For admission records of each patient, the i-th patient can be represented as a sequence of T_i admission records $\{\mathbf{x}_1, \mathbf{x}_2, .., \mathbf{x}_{T_i}\} \in \mathbf{p}_i$ in chronological order, where T_i is the number of admissions for the patient. The goal of our predefined prediction tasks is

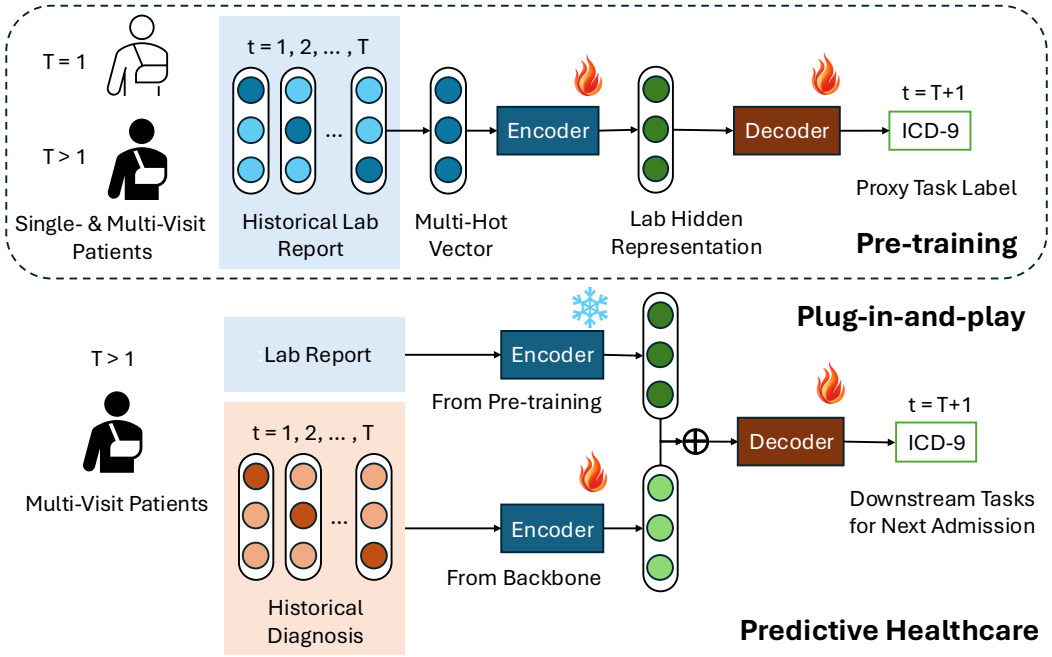

Fig. 2. Overview of the proposed MPLite Framework

to predict the label at the end of the sequence, $\mathbf{y} \in \{0,1\}^s$, which can be either a one-hot or multi-hot vector. We then omit i in the rest of the sections and explain our framework using single patient to avoid misunderstanding.

Specifically, a single admission $\mathbf{x}_t$ ($t \in \{1\}$) can be also represented as a multi-hot vector with dimensions corresponding to the medical concepts $\mathcal{C} = \{c_1, c_2, ..., c_{|\mathcal{C}|}\}$ where $|\mathcal{C}|$ is the total number of medical concepts. Each element in the vector is a boolean value indicating the presence (1) or absence (0) of the corresponding medical concept. Note that, we consider both *ITEM_ID* from lab results and *ICD-9* codes from diagnoses as medical concepts in our experiment, thus medical concepts might be either diagnosis codes $\mathbf{x}_t^D$ within vocabulary $\mathcal{D} \in \mathcal{C}$ or lab items codes $\mathbf{x}_t^L$ within vocabulary $\mathcal{L} \in \mathcal{C}$ in single admission. As medical concepts depend on problem formulation and real-world EHR data, procedures, drugs, and some other medical concepts can also be considered medical codes from a broader perspective. In the following sections, we also use abstract symbols like *MLP* to denote specific frameworks with mutable settings.

B. MPLite Framework

1) Multi-Aspect Pretraining: To fully leverage the EHR data, it is essential to utilize records from single-admission patients, who constitute the majority of the dataset. Since these records lack labels for future admissions, our focus is on learning the relationship between lab test results and diagnoses from the current visit. A single-visit patient has only one admission record, so we consider a single-visit patient equivalent to a single visit in this part. We hypothesize that additional aspects of features (e.g, lab tests) reflect important information about a patient's existing diagnoses. Thus, we identify lab results as additional medical concepts for each patient, considering that lab results are one of the most crucial components in describing diagnoses results.

In terms of the pretraining step of Figure 2, we define a novel proxy task that predicts the diagnoses shown in the sequence of visits by historical lab results in the pretraining step. As mentioned in the section III-A, diagnoses and lab results sets are denoted by $\mathcal{D}$ and $\mathcal{L}$ respectively, and we aim to decode the item code set from lab results $\mathbf{x}^L$ into the probability distribution $\hat{\mathbf{y}} = P(\mathbf{x}^D|\mathbf{x}^L)$ for each patient. Here we use a multi-layer perceptron (MLP) for parametrization to transform lab results to diagnoses of patients:

$$\hat{\mathbf{y}} = \sigma(\text{MLP}(\mathbf{x}_1^L \mid P_{\text{single}})) \tag{1}$$

$$\hat{\mathbf{y}} = \sigma(\text{MLP}(\text{Integrate}(\{\mathbf{x}_t^L\}_{t=1}^T) \mid P_{\text{multi}})) \tag{2}$$

Here $P_{\text{single}}, P_{\text{multi}}$ denotes single-visit and multi-visit patients, and σ means the activation function. There are two main reasons we chose MLP as the backbone model for the pretraining module: (1) It directly predicts the probability distribution of diagnosis codes efficiently, requiring minimal computational resources. (2) It achieves competitive predictive performance for the defined proxy task, even when compared to models incorporating embedding or convolution modules. Moreover, `Integrate` means we integrate the sequence of multiple visit into a single vector, which can be aligned with the input of single-visit patients as shown in equation 3.

$$\text{Integrate}(\{\mathbf{x}_t^L\}_{t=1}^T) = \bigvee_{t=1}^{T} \mathbf{x}_t^L \tag{3}$$

For the lab result data, we assume that lab results are all up-to-date, and we considered single lab-test code $c_k^{(\mathcal{L})}$ normal if it

has not been taken or was tested normal in the most recent test. For each patient, given lab results prior to the $(t+1)$-th visit is $\mathbf{x}_t^L \in \mathbb{R}^{|\mathcal{L}|}$ ($|\mathcal{L}| = 697$ in MIMIC-III dataset). The defined proxy task is a multi-label classification task. Given multi-hot vector of lab results $\mathbf{x}_t^L \in \{0,1\}^{|\mathcal{L}|}$, we use the first dense layer as encoder to get H-dimensional hidden representation $\mathbf{h}^L$ for each patient, and then we leverage the second dense layer as decoder to convert such hidden representation as diagnose classifier with output $\hat{y}$. The pretraining dense layers and corresponding loss function are defined as follows:

$$\mathbf{h}^L = \text{Encoder}(\{\mathbf{x}_t^L\}_{t=1}^T \mid P) \in \mathbb{R}^h \tag{4}$$

$$\hat{\mathbf{y}} = \text{Decoder}(\mathbf{h}^L) = \sigma(\mathbf{w}_k \mathbf{h}^L) \in \mathbb{R}^{|\mathcal{D}|} \tag{5}$$

$$\mathcal{L}_{\text{patient}} = -\left[\mathbf{y}\log(\hat{\mathbf{y}}) + (1-\mathbf{y})\log(1-\hat{\mathbf{y}})\right] \tag{6}$$

At the pretraining step, t depends on the number of available admission records for each patient, and we use binary cross entropy as loss function through N single-visit patients. Note that, such module is learnable within both single-visit and multi-visit patients, and involved parameters are fixed after the pretraining process. As a self-supervised learning problem, this proxy task is not simply input reconstruction and will not be affected by the order of medical concepts in a single visit. This is also the main advantage compared to traditional transformer-base models. The pretraining process does not have access to the validation and test sets of the prediction model. Thus there are no data leakage issues.

2) Integration and Inference: Now let us focus on how to fuse both representations from a backbone prediction model and the proposed pretraining module. Note that, subscript t might be also involved in model in terms of the training setting across different baseline. For example, some works feed model by admission-level data, which means patient with multiple visits can be fed consecutively into the model. For the adaptation ability of our framework, we also transfer this setting into the description of our framework.

Assuming we already have the final output $\mathbf{o}_t \in \mathbb{R}^{|\mathcal{C}|}$ for prediction of the t-th admission before feeding into the classifier of existing baselines, we can also retrieve lab results vector $\mathbf{x}_t^L$ as the input of the pre-trained module. $|\mathcal{C}|$ is the output dimension, which is also considered as the vocabulary size. We keep the same format of input for $\mathbf{x}_t^L$ and get hidden representation $\mathbf{h}_t^L \in \mathbb{R}^h$ in terms of patient's lab results through the pretrained encoder dense layer. We then use a classifier with single dense layer to get prediction $\hat{y}_t$ for multiple prediction tasks after concatenating both patient-level representations as shown in Figure 2. Finally, the integration step and classifier are defined as follows, the output dimension of classifier can be modified for various prediction tasks:

$$\mathbf{o}_t = \text{Encoder}(\{\mathbf{x}_t^L\}_{t=1}^T \mid P_{\text{multi}}) \tag{7}$$

$$\mathbf{o}_t' = \mathbf{o}_t \parallel \mathbf{h}_t^L \in \mathbb{R}^{|\mathcal{C}|+h} \tag{8}$$

$$\hat{\mathbf{y}}_t = \text{Classifier}(\mathbf{o}_t') \in \mathbb{R}^{|\mathcal{C}|} \tag{9}$$

We can still remain the same loss function $\mathcal{L}$ as the one already defined in the backbone module. Through the definition of inference part, we can easily plug in pretraining module and optimize current model's output by integrating lab results for more precise prediction.

TABLE II
STATISTICS OF THE MIMIC-III DATASET

# patients in total	46,520
# patients with multiple visits	7,537
# patients with multiple visits utilized in experiments	7,493
# patients with single visit	38,983
# patients with single visit utilized in experiments	26,085
Avg. visits per patient in MIMIC-III	1.27
# Medical codes (disease)	4,880
# Items (lab results)	697

C. Downstream Tasks

The proposed framework can be adapted for various prediction tasks. Consider a patient with $T+1$ admission records, we can build one sample with admission history $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T\}$ for each patient. We perform two prediction tasks in our experiments by the following definition:

(1) **Diagnosis (DG) Prediction** predicts the diagnosis result of the next admission given previous admission records. Formally, we learn a function $f : (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_t) \rightarrow \mathbf{y}[\mathbf{x}_{t+1}]$ where $t \leq T$ and $\mathbf{y}[\mathbf{x}_{t+1}] \in \mathbb{R}^{|\mathcal{D}|}$ is a multi-hot vector where $|\mathcal{D}|$ denotes the number of all diagnosis codes.

(2) **Heart Failure (HF) Prediction** predicts if heart failure (i.e., ICD-9 prefixed code of 428) is diagnosed in the next admission. Formally, we learn a function $f : (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_t) \rightarrow \mathbf{y}[\mathbf{x}_{t+1}]$ where $t \leq T$ and $\mathbf{y}[\mathbf{x}_{t+1}] \in \{0,1\}$ is a binary label indicating whether heart failure is diagnosed in the admission.

The binary cross-entropy (BCE) loss is used with a sigmoid function to train the learning framework for both binary and multi-label classifications tasks.

IV. EVALUATION

A. Dataset Description

To evaluate our proposed model, we focus on two pubic and widely-used EHR datasets: MIMIC-III [26] and MIMIC-IV [27]. Both datasets are derived from extensive de-identified clinical data collected from patients admitted to Intensive Care Units (ICUs). We employed a randomized approach to divide both datasets into training, validation, and testing segments. Specifically, MIMIC-III and MIMIC-IV datasets were divided into 6000/493/1000 and 8000/1000/1000 for the training, validation, and test sets, respectively.

Table II shows the basic statistics in MIMIC-III. Note that, while there are 85,155 patients in MIMIC-IV with multiple visits, we remove the patients with the overlapped time range and then randomly sample 10,000 patients from MIMIC-IV from 2013 to 2019 for training, which retains the same setting as Chet [28]. Hence, the basic statistics of MIMIC-IV which is omitted in paper might change for every runtime, since the random sampling method is adopted to get the comparable sample size of patients with MIMIC-III. We select patients with multiple admission records (# of visits $\geq$ 2) for the

TABLE III

PREDICTION RESULTS ON MIMIC-III AND MIMIC-IV FOR DIAGNOSIS AND HEART FAILURE PREDICTION. WE REPORT THE AVERAGE PERFORMANCE (%) AND STANDARD DEVIATION (IN BRACKETS) OF EACH MODEL OVER 10 RUNS. "NO" IN THE PRETRAIN COLUMN MEANS THE ORIGINAL BASELINES, AND "+MPLITE" MEANS THAT WE PLUG IN THE PRETRAINING MODULE INTO THE CORRESPONDING BASELINES

		MIMIC-III					**MIMIC-IV**				
		DG Prediction			**HF Prediction**		**DG Prediction**			**HF Prediction**	
Models	**Pretrain**	**w-F$_1$**	**R@10**	**R@20**	**AUC**	**F$_1$**	**w-F$_1$**	**R@10**	**R@20**	**AUC**	**F$_1$**
GRU	No	$17.82_{(0.43)}$	$31.56_{(0.40)}$	$33.64_{(0.38)}$	$80.54_{(0.60)}$	$68.93_{(0.53)}$	$19.55_{(0.48)}$	$35.12_{(0.57)}$	$37.91_{(0.54)}$	$81.33_{(0.71)}$	$69.31_{(0.56)}$
GRU	+MPLite	$19.58_{(0.34)}$	$33.82_{(0.39)}$	$35.97_{(0.35)}$	$82.01_{(0.55)}$	$70.56_{(0.47)}$	$21.87_{(0.37)}$	$37.84_{(0.43)}$	$40.63_{(0.48)}$	$83.12_{(0.62)}$	$71.02_{(0.42)}$
Dipole	No	$14.66_{(0.21)}$	$28.73_{(0.28)}$	$29.44_{(0.20)}$	$82.08_{(0.45)}$	$70.35_{(0.51)}$	$17.16_{(0.36)}$	$32.21_{(0.30)}$	$38.74_{(0.32)}$	$84.80_{(0.47)}$	$69.52_{(0.44)}$
Dipole	+MPLite	$18.27_{(0.30)}$	$30.91_{(0.37)}$	$32.97_{(0.29)}$	$83.56_{(0.53)}$	$71.53_{(0.46)}$	$20.63_{(0.33)}$	$38.12_{(0.36)}$	$40.75_{(0.41)}$	$85.67_{(0.56)}$	$71.02_{(0.50)}$
Deepr	No	$11.68_{(0.17)}$	$26.47_{(0.15)}$	$27.53_{(0.12)}$	$81.36_{(0.39)}$	$69.54_{(0.49)}$	$18.58_{(0.31)}$	$36.79_{(0.29)}$	$39.45_{(0.21)}$	$83.61_{(0.50)}$	$70.46_{(0.53)}$
Deepr	+MPLite	$18.43_{(0.28)}$	$31.08_{(0.25)}$	$33.22_{(0.30)}$	$82.91_{(0.58)}$	$71.12_{(0.42)}$	$19.75_{(0.32)}$	$38.97_{(0.34)}$	$41.11_{(0.38)}$	$85.08_{(0.60)}$	$71.55_{(0.47)}$
RETAIN	No	$18.37_{(0.28)}$	$32.12_{(0.38)}$	$32.54_{(0.27)}$	$83.21_{(0.43)}$	$71.32_{(0.32)}$	$23.11_{(0.47)}$	$37.32_{(0.36)}$	$40.15_{(0.41)}$	$84.14_{(0.34)}$	$71.23_{(0.38)}$
RETAIN	+MPLite	$20.42_{(0.35)}$	$34.56_{(0.42)}$	$36.87_{(0.39)}$	$84.73_{(0.52)}$	$72.94_{(0.39)}$	$24.85_{(0.41)}$	$39.68_{(0.35)}$	$42.67_{(0.44)}$	$85.82_{(0.51)}$	$72.83_{(0.47)}$
Timeline	No	$20.46_{(0.39)}$	$30.73_{(0.31)}$	$34.83_{(0.28)}$	$82.34_{(0.38)}$	$71.03_{(0.44)}$	$23.76_{(0.35)}$	$37.89_{(0.40)}$	$40.87_{(0.34)}$	$83.45_{(0.37)}$	$72.30_{(0.39)}$
Timeline	+MPLite	$22.64_{(0.30)}$	$32.89_{(0.29)}$	$36.94_{(0.38)}$	$83.92_{(0.49)}$	$72.98_{(0.36)}$	$24.38_{(0.33)}$	$39.72_{(0.36)}$	$42.84_{(0.40)}$	$84.98_{(0.50)}$	$73.54_{(0.33)}$
GRAM	No	$20.78_{(0.19)}$	$34.17_{(0.21)}$	$35.46_{(0.20)}$	$81.55_{(0.44)}$	$68.78_{(0.46)}$	$24.39_{(0.34)}$	$38.42_{(0.33)}$	$41.62_{(0.31)}$	$85.55_{(0.40)}$	$69.82_{(0.48)}$
GRAM	+MPLite	$22.78_{(0.32)}$	$35.96_{(0.35)}$	$38.61_{(0.32)}$	$83.22_{(0.54)}$	$70.94_{(0.38)}$	$25.93_{(0.31)}$	$40.42_{(0.34)}$	$43.68_{(0.37)}$	$86.98_{(0.55)}$	$71.06_{(0.52)}$
KAME	No	$21.10_{(0.20)}$	$29.97_{(0.23)}$	$33.99_{(0.25)}$	$82.88_{(0.46)}$	$72.03_{(0.42)}$	$25.01_{(0.29)}$	$38.86_{(0.28)}$	$42.12_{(0.30)}$	$84.80_{(0.35)}$	$72.34_{(0.43)}$
KAME	+MPLite	$23.64_{(0.37)}$	$31.98_{(0.32)}$	$35.92_{(0.40)}$	$83.67_{(0.47)}$	$73.48_{(0.31)}$	$26.22_{(0.33)}$	$40.74_{(0.36)}$	$43.95_{(0.39)}$	$85.92_{(0.52)}$	$73.95_{(0.40)}$
CGL	No	$22.63_{(0.29)}$	$33.64_{(0.33)}$	$37.87_{(0.27)}$	$84.19_{(0.34)}$	$71.77_{(0.41)}$	$25.74_{(0.32)}$	$39.23_{(0.37)}$	$42.67_{(0.36)}$	$87.91_{(0.44)}$	$70.71_{(0.35)}$
CGL	+MPLite	$24.82_{(0.40)}$	$35.68_{(0.28)}$	$39.97_{(0.35)}$	$85.82_{(0.43)}$	$72.81_{(0.36)}$	$26.97_{(0.38)}$	$41.92_{(0.39)}$	$44.61_{(0.41)}$	$88.89_{(0.45)}$	$72.52_{(0.39)}$
G-BERT	No	$22.28_{(0.25)}$	$35.62_{(0.29)}$	$36.46_{(0.26)}$	$81.50_{(0.38)}$	$71.18_{(0.43)}$	$25.12_{(0.30)}$	$39.91_{(0.31)}$	$43.25_{(0.28)}$	$85.76_{(0.50)}$	$72.88_{(0.45)}$
G-BERT	+MPLite	$24.31_{(0.36)}$	$37.14_{(0.30)}$	$38.98_{(0.33)}$	$82.99_{(0.56)}$	$72.72_{(0.42)}$	$27.58_{(0.35)}$	$42.56_{(0.34)}$	$44.62_{(0.39)}$	$86.88_{(0.58)}$	$74.12_{(0.47)}$
HiTANet	No	$23.15_{(0.28)}$	$34.68_{(0.35)}$	$35.97_{(0.31)}$	$85.13_{(0.31)}$	$73.15_{(0.39)}$	$24.53_{(0.33)}$	$38.42_{(0.37)}$	$41.89_{(0.29)}$	$86.34_{(0.36)}$	$71.35_{(0.44)}$
HiTANet	+MPLite	$25.87_{(0.33)}$	$36.91_{(0.36)}$	$39.02_{(0.34)}$	$86.74_{(0.47)}$	$74.45_{(0.40)}$	$26.91_{(0.30)}$	$42.12_{(0.38)}$	$43.94_{(0.33)}$	$87.99_{(0.49)}$	$72.85_{(0.42)}$

diagnosis prediction task and only consider patients without missing diagnosis codes in all visits. For instance, among the 38,983 single-visit patients in MIMIC-III, we only consider patients with previous lab results before their admission. We conduct multiple experiments with uniform baseline hyperparameter sets, measuring average and standard deviation values.

B. Baselines

To check the improvement of MPLite for predictive models, we select the following state-of-art methods as baselines:

- RNN/CNN-based models: GRU [10], Timeline [12], RETAIN [11], Deepr [15], and Dipole [13].
- Graph-based models: GRAM [17], KAME [29], and CGL [6].
- Transformer-based models: G-BERT [5], HiTANet [22].

Note that GRU uses multi-hot vectors of medical codes as inputs, while other baselines use medical code embeddings. For G-BERT, both pretraining and medication inputs is discarded which requires extra information other than diagnose features. Moreover, we remove the clinical notes parsing module in CGL and the timestamp feature in HiTANet to ensure the consistency of training data for all models. We also do not consider MiME [25] and GCT [7] because of additional requirement on input data and supported tasks.

C. Parameters Setting

The parameter settings used for pretraining module, we find the optimal output dimension 200 of the first dense layer from a search space of [100, 200] and set Drop-out rate as 0.4 in the final classifier for fine-tuning and final prediction. For the baseline GRU, the units of RNN module are all set as 128. For other baselines, we do our best to follow the parameter setting described in original papers. Different learning rate decay schedulers with Adam optimizer are experimented, resulting a decay learning rate from $1e-2$ to $1e-5$ between epochs. Moreover, we set batch size as 64 and use 100 epochs for training process. We conduct 10 repeated experiments for each baseline model and the corresponding model with pretrained lab results. All evaluation metrics are recorded and calculated for each experiment, and we can then assess whether *MPLite* can help model get more accurate prediction.

All programs are implemented using Python 3.10, Tensorflow 2.10, and Pytorch 2.3.1 with CUDA 12.3 on a machine with two AMD EPYC 9254 24-Core Processors, 528GB RAM, and four Nvidia L40S GPUs. [1]

D. Evaluation Metrics

Since among the 4880 and 6102 diseases we are predicting in MIMIC-III and MIMIC-IV, the distribution of disease codes

[1]The source code of the MPLite model can be found at https://github.com/EricY090/MPLite.

is very sparse and the occurrence for each disease is highly imbalance, we adopt the weighted F1 score (w-F_1 [12]) and top k recall (R@k [11]) for diagnosis predictions. In the context of the weighted F_1 score, the contributions of individual classes (diseases) are weighted based on their prevalence in our dataset. Unlike traditional recall, Recall@k focuses on the ratio of true positive samples among positive samples in the top k predictions, and we set k values as 10 and 20 for evaluation which is the same as other works. For heart failure predictions in case study, we add the area under the ROC curve (AUC) as binary classification metrics besides F_1 score, since label distribution is imbalanced in MIMIC datasets.

E. Experimental results

1) Diagnosis Prediction: As demonstrated in Table III, both the mean and standard deviation are reported across different baselines within two datasets. The results indicate that the integration of the proposed framework consistently enhances the predictive performance of various baselines.

From the results on the MIMIC-III dataset, we observe significant improvements in w-F_1 scores when the proposed framework is applied. For example, GRU with *MPLite* improves over the vanilla GRU by approximately +1.76 in w-F_1, +2.26 in R@10, and +2.33 in R@20. This trend is consistent across other models such as Dipole, Deepr, and RETAIN, demonstrating similar enhancements in w-F_1 and recall metrics. Specifically, Dipole with *MPLite* achieves a w-F_1 score improvement of +3.61, and an increase of +2.18 in R@10 and +3.53 in R@20, highlighting the efficacy of pretraining with *MPLite*.

On the MIMIC-IV dataset, the improvement trends are similar. GRU shows an increase of +1.32 in w-F_1 and notable gains of +2.72 in R@10 and +2.72 in R@20 with the pretraining module. These results suggest that *MPLite* not only boosts w-F_1 scores but also enhances recall rates, indicating better model sensitivity in capturing relevant diagnostic information.

The consistent performance boost across both datasets underscores the generalization capability of the proposed module. The most likely reason for the improved performance is that lab results typically include detailed physiological and biochemical indicators, which directly reflect patients' health status, providing crucial information about disease conditions and bodily functions for doctors to diagnose diseases.

2) Case Study - Heart Failure Prediction: Following the results of diagnosis predcition, a research question arises: *Assuming different training tasks, where the final prediction and the proxy task in the pretraining module are different, can MPLite still improve the predictive performance of the baseline models?* Table III also shows the heart failure prediction results in both MIMIC-III and MIMIC-IV. We observe that *MPLite* still allows all involved baselines to achieve higher AUC and F_1 scores. Therefore, we conclude that lab results can complement other clinical information, contributing collectively to precise prediction, which also demonstrates the generalization ability of the proposed framework.

V. Discussion and Conclusion

In this paper, we present a flexible plug-in-and-play framework called **MPLite**, which integrates lab results to enable backbone models to collaboratively learn more precise representations for patients. We conduct experiments on 2 widely-used EHR datasets across 10 predictive baselines with different architectures, demonstrating the effectiveness of the plug-in pretraining module through significant improvements over the original backbone models. Additionally, we performed a case study on heart failure prediction to verify the generalization ability of *MPLite* across various prediction tasks. All pretraining experiments are based on lab results which can be limited when such features are not available. While the framework can be adapted to other input types, we believe further extensive testing is necessary.

In the future, we plan to evaluate the effectiveness of the pretraining model on more complex network architectures than MLP and diverse health risk prediction tasks. For instance, we can further refine proxy tasks to help model get rid of limitation on laboratory input, which is also the common problem as other baselines upon lab tests. Furthermore, an initial screening process could be applied to single-visit patients to enhance training quality by ensuring adequate diversity of single-visit and multi-visit patients. Another potential direction for future research is to incorporate more feature modalities such as clinical notes into the pre-training process.

VI. Acknowledgement

This work is supported in part by the US National Science Foundation under grants 2047843 and 2437621. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] A. Johnson, P. Tom, and R. Mark, "Mimic-iii clinical database," 2016, data retrieved from PhysioNet, https://doi.org/10.13026/C2XW26.

[2] E. Choi, M. T. Bahadori, A. Schuetz, W. F. Stewart, and J. Sun, "Doctor AI: predicting clinical events via recurrent neural networks," in *Proceedings of the 1st Machine Learning in Health Care*, vol. 56, 2016, pp. 301–318.

[3] W. Zhang, B. Ingale, H. Shabir, T. Li, T. Shi, and P. Wang, "Event detection explorer: An interactive tool for event detection exploration," in *Companion Proceedings of the 28th International Conference on Intelligent User Interfaces*, 2023, pp. 171–174.

[4] W. Zhang, K. Zeng, X. Yang, T. Shi, and P. Wang, "Text-to-esq: A two-stage controllable approach for efficient retrieval of vaccine adverse events from nosql database," in *Proceedings of the 14th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, 2023, pp. 1–10.

[5] 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, IJCAI 2019*, 2019, pp. 5953–5959.

[6] 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, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, 2021, pp. 3529–3535.

[7] E. Choi, Z. Xu, Y. Li, M. Dusenberry, G. Flores, E. Xue, and A. M. Dai, "Learning the graphical structure of electronic health records with graph convolutional transformer," in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, 2020, pp. 606–613.

[8] 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 Digit. Medicine*, vol. 4, 2021.

[9] R. Poulain and R. Beheshti, "Graph transformers on EHRs: Better representation improves downstream performance," in *The Twelfth International Conference on Learning Representations*, 2024.

[10] K. Cho, B. van Merrienboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, 2014, pp. 1724–1734.

[11] E. Choi, M. T. Bahadori, J. Sun, J. Kulas, A. Schuetz, and W. F. Stewart, "RETAIN: an interpretable predictive model for healthcare using reverse time attention mechanism," in *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016*, 2016, pp. 3504–3512.

[12] 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, KDD 2018*, 2018, pp. 43–51.

[13] 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, Halifax, NS, Canada, August 13 - 17, 2017*, 2017, pp. 1903–1911.

[14] L. Ma, C. Zhang, Y. Wang, W. Ruan, J. Wang, W. Tang, X. Ma, X. Gao, and J. Gao, "Concare: Personalized clinical feature embedding via capturing the healthcare context," in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020*, 2020, pp. 833–840.

[15] P. Nguyen, T. Tran, N. Wickramasinghe, and S. Venkatesh, "`Deepr`: A convolutional net for medical records," *IEEE J. Biomed. Health Informatics*, vol. 21, no. 1, pp. 22–30, 2017.

[16] L. Ma, J. Gao, Y. Wang, C. Zhang, J. Wang, W. Ruan, W. Tang, X. Gao, and X. Ma, "Adacare: Explainable clinical health status representation learning via scale-adaptive feature extraction and recalibration," in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020*, 2020, pp. 825–832.

[17] 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, Halifax, NS, Canada, August 13 - 17, 2017*, 2017, pp. 787–795.

[18] W. Zhu and N. Razavian, "Variationally regularized graph-based representation learning for electronic health records," in *ACM CHIL '21: ACM Conference on Health, Inference, and Learning, Virtual Event, USA, April 8-9, 2021*, 2021, pp. 1–13.

[19] P. Jiang, C. Xiao, A. R. Cross, and J. Sun, "Graphcare: Enhancing healthcare predictions with personalized knowledge graphs," in *The Twelfth International Conference on Learning Representations*, 2023.

[20] T. Wu, Y. Wang, Y. Wang, E. Zhao, and Y. Yuan, "Leveraging graph-based hierarchical medical entity embedding for healthcare applications," *Scientific reports*, vol. 11, no. 1, p. 5858, 2021.

[21] S. Liu, T. Liu, H. Ding, B. Tang, X. Wang, Q. Chen, J. Yan, and Y. Zhou, "A hybrid method of recurrent neural network and graph neural network for next-period prescription prediction," *Int. J. Mach. Learn. Cybern.*, vol. 11, no. 12, pp. 2849–2856, 2020.

[22] J. Luo, M. Ye, C. Xiao, and F. Ma, "Hitanet: Hierarchical time-aware attention networks for risk prediction on electronic health records," in *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, 2020, pp. 647–656.

[23] C. Lu, C. K. Reddy, and Y. Ning, "Self-supervised graph learning with hyperbolic embedding for temporal health event prediction," *IEEE Trans. Cybern.*, vol. 53, no. 4, pp. 2124–2136, 2023.

[24] S. Park, S. Bae, J. Kim, T. Kim, and E. Choi, "Graph-text multi-modal pre-training for medical representation learning," in *Conference on Health, Inference, and Learning, CHIL 2022, 7-8 April 2022, Virtual Event*, vol. 174, 2022, pp. 261–281.

[25] E. Choi, C. Xiao, W. F. Stewart, and J. Sun, "Mime: Multilevel medical embedding of electronic health records for predictive healthcare," in *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018*, 2018, pp. 4552–4562.

[26] A. E. 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*, vol. 3, no. 1, pp. 1–9, 2016.

[27] A. Johnson, L. Bulgarelli, T. Pollard, S. Horng, L. A. Celi, and R. Mark, "Mimic-iv," *PhysioNet. Available online at: https://physionet.org/content/mimiciv/1.0/(accessed August 23, 2021)*, pp. 49–55, 2020.

[28] C. Lu, T. Han, and Y. Ning, "Context-aware health event prediction via transition functions on dynamic disease graphs," in *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, 2022, pp. 4567–4574.

[29] F. Ma, Q. You, H. Xiao, R. Chitta, J. Zhou, and J. Gao, "KAME: knowledge-based attention model for diagnosis prediction in healthcare," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018*, 2018, pp. 743–752.