From Basic to Extra Features: Hypergraph Transformer Pretrain-then-Finetuning for Balanced Clinical Predictions on EHR

Ran Xu RAN.XU@EMORY.EDU
Emory University, United States

Yiwen Lu YIWENLU@SAS.UPENN.EDU
University of Pennsylvania, United States

Chang Liu CHANG.LIU2@EMORY.EDU
Emory University, United States

Yong Chen YCHEN123@PENNMEDICINE.UPENN.EDU
University of Pennsylvania, United States

Yan Sun YAN.V.SUN@EMORY.EDU
Emory University, United States

Xiao Hu XIAO.HU@EMORY.EDU
Emory University, United States

Joyce C Ho JOYCE.C.HO@EMORY.EDU
Emory University, United States

Carl Yang J.CARLYANG@EMORY.EDU
Emory University, United States

Abstract

Electronic Health Records (EHRs) contain rich patient information and are crucial for clinical research and practice. In recent years, deep learning models have been applied to EHRs, but they often rely on massive features, which may not be readily available for all patients. We propose `HTP-Star`[1], which leverages hypergraph structures with a pretrain-then-finetune framework for modeling EHR data, enabling seamless integration of additional features. Additionally, we design two techniques, namely (1) *Smoothness-inducing Regularization* and (2) *Group-balanced Reweighting*, to enhance the model's robustness during finetuning. Through experiments conducted on two real EHR datasets, we demonstrate that `HTP-Star` consistently outperforms various baselines while striking a balance between patients with basic and extra features.

Data and Code Availability We evaluate our framework on two publicly available datasets UK Biobank (Sudlow et al., 2015) and MIMIC-III (Johnson et al., 2016). The research was conducted using data from the UK Biobank Resource under an application number (omitted for anonymization). The UK Biobank makes the data available to all bona fide researchers for all types of health-related research that is in the public interest, without preferential or exclusive access for any persons. All researchers are subject to the same application process and approval criteria as specified by UK Biobank. MIMIC-III is publicly available from the PhysioNet repository.

Institutional Review Board (IRB) UK Biobank has approval from the North West Multicentre Research Ethics Committee (MREC) as a Research Tissue Bank (RTB) approval. MIMIC-III does not need IRB approval.

1. Introduction

Electronic Health Record (EHR) is a digital representation of a patient's medical history that contains a wealth of patient information, including diagnoses, medications, lab results, and so on (Cowie et al., 2017). In clinical research and practice, healthcare

1. Short for **H**ypergraph **T**ransformer **P**retrain-then-Finetuning with **S**moo**th**ness-induced regularization **a**nd **R**eweighting.

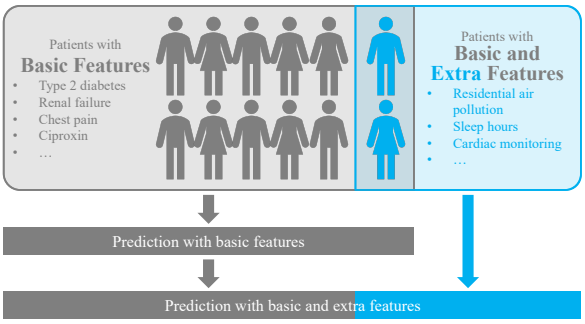


Figure 1: An illustration of basic and extra features.

professionals actively use EHRs for patient health monitoring (Gandrup et al., 2020; Shi et al., 2024), risk predictions (Luo et al., 2020; La Cava et al., 2023) and clinical trial matching (Rogers et al., 2021), thereby harnessing the capabilities of digital repositories to augment patient care and inform clinical decision-making (Tang et al., 2022).

In recent years, various deep learning architectures (Choi et al., 2016; Pang et al., 2021) have gained extensive popularity in predictive tasks on EHR. Typically, these models are trained on a collection of basic features shared across various medical institutions, such as diseases and medications. However, in real-world scenarios, additional features are often collected, which are limited to specific hospitals due to privacy or budget constraints (Taksler et al., 2021; Hong et al., 2021). Figure 1 shows an illustration of extra features collected by some local medical institutes. Due to the often small sample sizes for patients with these extra features, they are not effectively utilized to enhance the modeling of patients, regardless of whether they possess these extra features or not. Hence, our primary objective in this research is to address the challenge: *How can we harness the extra data gathered from local medical institutes on specific patients to enhance clinical prediction tasks within the population?*

This naturally resembles a transfer learning setting (Weiss et al., 2016) with a pretrain-then-finetune pipeline, which requires the model to effectively transfer the knowledge from a more extensive population to individuals with extra sets of features. However, developing clinical predictive models that concurrently incorporate basic and extra features is non-trivial. Directly adapting traditional machine learning (ML) methods such as linear regression or decision trees is problematic since the model trained from patients with basic features cannot directly handle patients with extra features due to different feature dimensions. Although there exist several deep learning architectures (e.g. Transformers (Lee et al., 2019; Choi et al., 2020)) that can model flexible numbers of patient features, they do not explicitly model the interactions between features, so they cannot fully leverage extra features to improve the modeling of basic features, and vice versa.

Given the challenges outlined above, we emphasize the importance of designing a suitable data structure capable of accommodating additional features from local institutes for patients. Inspired by the recent progress in hypergraph learning for clinical predictions on EHRs with strong representative power (Xu et al., 2022; Cai et al., 2022; Wu et al., 2023a), we propose `HTP-Star` to utilize *hypergraph* structure to characterize the EHR data. Building upon this structure, patient visits are conceptualized as hyperedges, with each visit-related feature represented as a node, allowing each hyperedge to be connected to a flexible number of nodes. This approach not only effectively characterizes the relationships between hospital visits and medical codes from a higher-order view, but also enables the seamless integration of new features into the current dataset by simply adding nodes to the existing hyperedges without extensive modifications to the overall graph structure. Additionally, to facilitate information propagation and mutual enhancement between newly incorporated and existing basic features, we employ *hypergraph transformers* (Xu et al., 2022; Cai et al., 2022), which incorporates multi-head self-attention and jointly learns the embeddings for hospital visits and all patient features.

After capturing the relationship between visits and features via hypergraph transformer, it is also crucial to design *effective and balanced training techniques* to enable models to generalize well on both basic and extra features. This is essential as samples with basic and extra features might sometimes have conflicts in their optimization directions. Existing transfer learning models leverage self-supervised learning (Shang et al., 2019; Bo et al., 2022; Park et al., 2022; McDermott et al., 2021) to improve the model's generalization ability with a pretrain-then-finetune pipeline, but they often directly fine-tune on target tasks without effective regularization, which is easy to cause catastrophic forgetting (Ramasesh et al., 2021). There are also generic transfer learning methods (Han et al., 2021; Liu et al., 2021b; Jiang et al., 2023a), but they often have strong assump-

tions on data distributions, and thus may not adapt to the clinical setting well.

Motivated by this, we develop two strategies to enhance the model's generalization ability across patients with varying data volumes: (1) To *mitigate the risk of aggressive model updates*, we maintain a slowly updated predictive model, which takes the form of a momentum-based moving average of the originally fine-tuned model. We add a regularization term to encourage consistent predictions between the original and the slowly updated predictive model to prevent the predictive model from forgetting previous information learned from pretraining. (2) To *resolve conflicts in optimization directions between basic and extra data features*, we introduce a gradient balancing method that adjusts the combination of gradients from patients with different data types. With these two dedicated techniques, `HTP-Star` learns a robust hypergraph model for EHR predictive tasks to accommodate both basic and extra features simultaneously.

We conduct experiments on two datasets, UK-Biobank (Sudlow et al., 2015) and MIMIC-III (Johnson et al., 2016), to evaluate `HTP-Star` and potential baselines. The results demonstrate that `HTP-Star` outperforms various standard ML methods as well as existing finetuning techniques, achieving a balance between patients with basic and extra features. Our contribution can be summarized as follows:

- We study the problem of clinical predictions with basic and extra features and identify the challenges (Sec. 3.2), which have not been widely explored in prior works.
- We design `HTP-Star`, a hypergraph pretrain-then-finetuning framework to enhance the model's robustness over two patient subgroups. We further propose two additional techniques to improve the model's generalization ability during fine-tuning steps.
- We conduct comprehensive experiments on two publicly available datasets (UK Biobank and MIMIC-III) to verify the efficacy of `HTP-Star`.

2. Related Works

Deep Predictive Model for EHRs In recent years, there have been numerous studies focusing on developing deep healthcare predictive models with various medical concepts. Earlier works attempt to leverage recurrent neural networks (RNN) (Choi et al., 2016; Lipton et al., 2016) as well as Transformers (Li et al., 2020; Pang et al., 2021) to model the chronological relationships among different medical units. Graph-based models have also been proposed for EHR modeling, including graph convolution networks (Zhu and Razavian, 2021; Lu et al., 2022), graph transformers (Choi et al., 2020; Zhu and Razavian, 2021; Jiang et al., 2023b), and hypergraph neural networks (Cai et al., 2022; Xu et al., 2022). These approaches involve constructing a co-occurrence graph based on EHR data and then using graph neural networks (GNNs) to learn the relations among medical codes within each for clinical outcome prediction (Johnson et al., 2023). Despite the impressive performance exhibited by deep learning-based models, these models typically demand massive labeled data and substantial feature richness, making them challenging to deploy in real-life resource-constrained healthcare environments (Erion et al., 2022). In this study, we harness graph-based deep learning models coupled with enhanced training methodologies to address the challenge of data scarcity in EHRs. It is also worth noting that, unlike existing graph-based approaches which concentrate on enhancing performance for patients with only basic or extra features, our approach targets enhancing the generalization ability for patients with both *basic* and *extra* feature profiles.

Training Techniques for Better Generalization Our work is also related to several studies for improving the model's generalization with basic data. *Self-supervised learning* techniques has been widely adopted for CV and NLP tasks (Devlin et al., 2019; Chen et al., 2020), and has also been adopted for EHRs with improved generalization (Shang et al., 2019; McDermott et al., 2021; Bo et al., 2022; Park et al., 2022). *Transfer learning* techniques (Weiss et al., 2016) aims to transfer knowledge across the target and source model, and recent works have proposed to harness attention networks (Xiao et al., 2020), generative models (Desautels et al., 2017; Estiri et al., 2021), reweighting techniques (Li et al., 2023; Han et al., 2021; Yu et al., 2021), or adversarial networks (Dai et al., 2022; Liu et al., 2021b) to facilitate knowledge adaptation to the target domains. Our method is related as we first leverage self-supervised learning techniques to warm up the model training, and then design strategies to better transfer the knowledge to patients with both basic and extra features.

3. Preliminary Studies

Before describing the details of our proposed model, we first give a brief overview of the problem setup, as well as the potential challenges under this scenario.

3.1. Problem Setup

In this study, we focus on predictive tasks on EHR which comprises patient visits with different medical codes. Formally, EHR visits are defined as:

Definition 1 (EHR Visit) *The EHR system generally includes a large amount of hospital visits $\mathcal{H}$ for corresponding patient group $\mathcal{P}$. Each visit $h \in \mathcal{H}$ involves a distinct set of medical codes $c \subset \mathcal{C}$ as features, where $\mathcal{C}$ is the total set of medical codes appearing in $\mathcal{H}$. In this study, $\mathcal{C}$ contains multiple types of medical codes such as diseases, medications, procedures.*

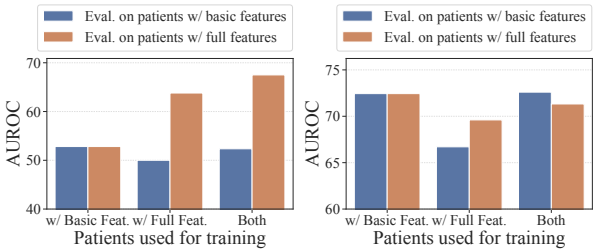
Due to the diversity among various groups of patients, there is usually a large variation in the volume of the medical codes $|\mathcal{C}|$ across different patient groups. In this study, we consider the setting where patients are separated into two subgroups, one with basic features only and the other with both basic and extra features.

Definition 2 (Patients with Basic/Extra Feat.) *Typically, the available data consists of a large set of EHRs $\mathcal{H}$ from a wider population, which have basic patient features $\mathcal{C}_b$. However, as local a medical institute collects extra features $\mathcal{C}_e$, a small subset of EHRs $\mathcal{H}_e \subset \mathcal{H}$ further includes extra features $\mathcal{C}_e$.*

In this work, given the clinical record $\mathcal{H}$ and both basic and extra features $\mathcal{C}_b \cup \mathcal{C}_e$, we aim to develop a model g_θ that predicts the patients' clinical outcomes y so that g_θ can perform well on both patient groups, including those with basic features only and those with full (both basic and extra) features.

3.2. Limitations of Traditional ML Methods

While traditional ML methods usually demonstrate strong performance in predictive EHR tasks, they encounter limitations in our specific scenario due to their inability to handle varying feature dimensions. To illustrate this challenge, we conduct a preliminary study as shown in Figure 2, employing XGBoost (Chen and Guestrin, 2016), one of the most powerful ML models on two datasets: the UK Biobank (Sudlow et al., 2015) and MIMIC-III (Johnson et al., 2016). Both datasets have a large number of patients with basic features and a smaller subset of patients with additional extra features (please refer to Section 5.1 for details of statistics and task descriptions). We conduct three distinct sets of experiments with XGBoost– on patients with basic features only, patients with full features only, and all patients regardless of the features they have (by filling in zeros for patients with basic features only). The experimental results are depicted in Figure 2. From the results, we have the following findings:



(a) UK Biobank (b) MIMIC-III

Figure 2: A preliminary study with XGBoost on the two datasets.

Using patients only with basic or full features hinders the model performance: Figure 2 reveals that when using patients' basic or full features only, the model generally exhibits lower performance. This is mainly due to insufficient information (when using basic features) or the limited amount of training instances (when using full features). It is necessary to design effective approaches for learning with basic and full features simultaneously.

Simply combining patients with basic and full features yields limited performance gains: Training with all patients, on the other hand, involves padding the input vectors of patients with only basic features by zeros to accommodate different feature dimensions. This can potentially lead to biased information, as the model might falsely interpret these zero values as informative features. Therefore, incorporating all patients with all features *does not necessarily enhance* the original model performance in both evaluation scenarios.

In summary, the inherent limitations of traditional ML models lead to unsatisfactory performance in the clinical setting studied in this work. These models either struggle to effectively leverage additional information or excel only in cases where patients have extra features, leaving a substantial performance gap

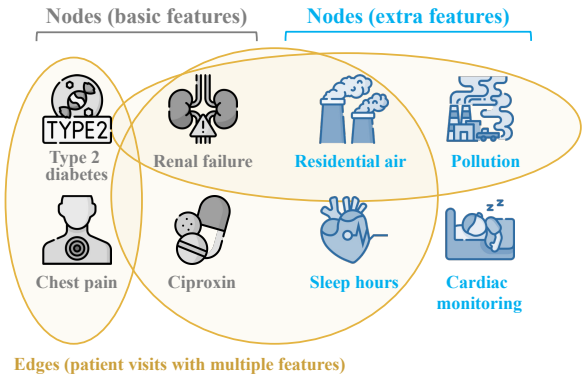


Figure 3: An illustration of the hypergraph structures.

when dealing with patients having only basic features. This observation highlights the importance of developing effective strategies that can simultaneously incorporate patients with both basic and extra features to achieve better generalization.

4. Method

From the above analysis, it is crucial to go beyond the traditional ML modeling techniques to address the intrinsic challenges of learning with patients using basic and extra features. Towards this end, we introduce our framework `HTP-Star` in Figure 4, which leverages *hypergraphs* to model the EHR patient information and adopts the *pretrain-then-finetune* pipeline to incorporate information from both basic and extra features. Additionally, we apply *smoothness-inducing regularization* and *group-balanced reweighting* techniques to mitigate issues related to catastrophic forgetting and excessive updates.

4.1. Hypergraph Learning

Graph Construction To better model the patient visit information as well as medical codes, it is crucial to learn the hypergraph structural information. In this work, we model the patient visits $\mathcal{H}$ as hyperedges $\mathcal{E}$ and the full collection of medical codes (*i.e.* features) $\mathcal{C}_b \cup \mathcal{C}_e$ as nodes $\mathcal{V}$. Each hyperedge $e \in \mathcal{E}$ represents a patient visit and can connect to various nodes, where each node $v \in \mathcal{V}$ stands for a medical code. We construct $\mathcal{G}_b = (\mathcal{V}_b, \mathcal{E})$ as the hypergraph that includes all the patients and their basic features, and $\mathcal{G}_e = (\mathcal{V}, \mathcal{E}_e)$ as the hypergraph that contains patients with extra features and all their features.

Figure 3 illustrates the hypergraph structures used in our approach. In this figure, the yellow circles represent hyperedges, which correspond to patient visits. Each hyperedge encompasses all the nodes (i.e., features) that are present in that particular hospital visit. This modeling approach captures the higher-order interactions among patient visits and features. Additionally, extra features can be easily incorporated into the hyperedges without the need to create new edges, providing flexibility in feature integration.

Hypergraph Transformer Architecture Denote the representation of nodes and hyperedges on l-th layer as $\boldsymbol{X}^{(l)} \in \mathbb{R}^{|\mathcal{V}| \times d}$, $\boldsymbol{E}^{(l)} \in \mathbb{R}^{|\mathcal{E}| \times d'}$ where d and d' are two hyperparameters. Let $\mathcal{V}_{e,\boldsymbol{X}} = \{\boldsymbol{X}_{v,:} : v \in e\}$ denote the set of hidden representations for *nodes* in the hyperedge e and $\mathcal{E}_{v,\boldsymbol{E}} = \{\boldsymbol{E}_{e,:} : v \in e\}$ denote the set of hidden representations of *hyperedges* that contain the node v, respectively. In this work, we leverage the hypergraph transformer architecture $g(\cdot; \mathcal{G}, \theta)$ (Xu et al., 2022), which comprises several sequential layers. In the l-th layer, the message passing follows two steps:

$$\boldsymbol{E}_e^{(l)} = f_{\mathcal{V}\to\mathcal{E}}\left(\mathcal{V}_{e,\boldsymbol{X}^{(l-1)}}\right), \tag{1}$$

$$\boldsymbol{X}_v^{(l)} = f_{\mathcal{E}\to\mathcal{V}}\left(\mathcal{E}_{v,\boldsymbol{E}^{(l)}}\right). \tag{2}$$

To realize the propagation function $f(\cdot)$ for each layer, we use two sub-layers: a multi-head self-attention (MHA) and a fully connected feed-forward neural network (FFNN). The details of these two components are deferred to Appendix A. Formally, the propagation rule $f(\cdot)$ can be expressed as

$$\boldsymbol{Y} = \mathrm{MHA}(\boldsymbol{X}), \tag{3}$$

$$f(\boldsymbol{X}) = \mathrm{LN}(\boldsymbol{Y} + \mathrm{FFNN}(\boldsymbol{Y})). \tag{4}$$

By harnessing the strong representative power of self-attention, we can identify the most relevant elements within the set for message passing, which is crucial for encoding the relationships among rich and sparse features and facilitating knowledge transfer.

Predictions on Target Tasks To support downstream clinical prediction tasks with the learned patient representations, we stack a classification layer on top of the visit embeddings from *all layers* $\widetilde{\boldsymbol{E}}_i^{(l)} (1 \le l \le L)$ from Eq. 1 to obtain final predictions. Specifically, for patient i, the prediction can be expressed as

$$\widehat{y}_i = g(e_i; \mathcal{G}, \theta) = \sigma\left(\boldsymbol{W}_{\mathrm{cls}}\left(\|_{l=1}^{L} \widetilde{\boldsymbol{E}}_i^{(l)}\right)\right); \tag{5}$$

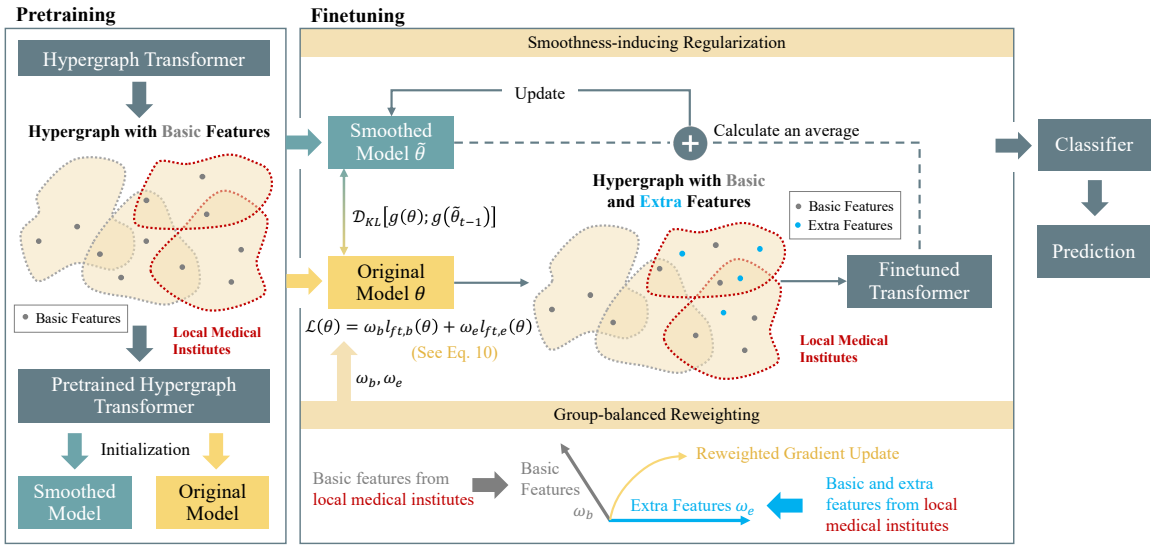


Figure 4: The framework of `HTP-Star`.

where e_i is the corresponding hyperedge for patient i, $\boldsymbol{W}_{\text{cls}}$ is a linear classification head that converts the vector to a value for binary classification and $\sigma(x) = 1/(1+\exp(-x))$ is the sigmoid function. The target prediction task involves a binary classification task, we use the binary cross-entropy as the learning objective defined as

$$\ell_{\text{cls}}(e_i, y_i) = -y \log(\widehat{y}_i) - (1-y)\log(1-\widehat{y}_i). \quad (6)$$

4.2. Pretrain-then-Finetune Pipeline

The previous section mainly discusses the hypergraph construction and learning models. Now, the key challenge becomes how to design an effective training scheme to better model the basic and extra clinical features of different groups of patients. As shown in Section 3.2, using only part of the features, as well as simply combining patients with basic and extra clinical features results in unsatisfactory performances.

To tackle this issue, we introduce a two-stage training approach, beginning with a pre-training phase followed by fine-tuning. Note that such a pretrain-then-finetune pipeline has been widely adopted for various domains including computer vision (Chen et al., 2020), text (Devlin et al., 2019), and time series (McDermott et al., 2021). Initially, the predictor is trained on the hypergraph $\mathcal{G}_b$. Then, to tailor the learned model $g(\cdot; \mathcal{G}_b, \theta)$ for the specific task involving patients with additional features, we fine-tune it on the hypergraph $\mathcal{G}_e$. The details of this two-stage training pipeline is described in the following sections.

4.2.1. Pretraining on Basic Features From Broader Population

In order to equip the information from the basic features, we pretrain the model on the hypergraph $\mathcal{G}_b$ where the basic features from all patients are considered. During this pretraining stage, the learning objective is denoted as

$$\ell_{\text{pt}}(\theta) = \mathbb{E}_{e_i \sim \mathcal{E}} \ell_{\text{cls}}(e_i, y_i), \quad (7)$$

where ℓ_{cls} is defined in Eq. 6. This pretrained hypergraph transformer serves as the starting point in the fine-tuning stage.

4.2.2. Finetuning with Customized Techniques

After pretraining on a broader population, we then finetune our model on local data with a small number of patient visits with basic and extra features. To better harness the knowledge from pretraining and balance the performance between patients with basic and extra features, we design two additional techniques for our scenarios, namely *Smoothness-inducing Regularization* and *Group-balanced Reweighting*. The details of these two models are described as follows.

Algorithm 1: Training Process of `HTP-Star`.

Input: Patient Visit $\mathcal{H} = ((\mathcal{C}_b, \mathcal{C}_e), \mathcal{P})$, Numbers of iterations for pretraining and finetuning Iter_{pt}, Iter_{ft}.
Output: Finetuned hypergraph transformer g_θ.
// Step 1: Hypergraph Construction
$\mathcal{E} \leftarrow \mathcal{H}$, $\mathcal{V}_b \leftarrow \mathcal{C}_b$, $\mathcal{G}_b \leftarrow (\mathcal{V}_b, \mathcal{E})$
// Step 2: Hypergraph Transformer Pretraining
for $i \leftarrow 1$ **to** Iter_{pt} **do**
  Update hypergraph transformer $g(\cdot; \mathcal{G}_b, \theta)$ with $\ell_{\text{pt}}(\theta)$ in Eq. 7.
end
// Step 3: Hypergraph Transformer Finetuning
$\mathcal{E}_b \leftarrow \mathcal{H}_b$, $\mathcal{V} \leftarrow (\mathcal{C}_b, \mathcal{C}_e)$, $\mathcal{G}_e \leftarrow (\mathcal{V}, \mathcal{E}_e)$, $g(\cdot; \mathcal{G}_e, \theta) \leftarrow g(\cdot; \mathcal{G}_b, \theta)$
for $i \leftarrow 1$ **to** Iter_{ft} **do**
  Calculate loss for patients with basic and extra features $\ell_{\text{ft,b}}$, $\ell_{\text{ft,e}}$ using Eq. 9.
  Calculate weights (ω_b, ω_e) for two groups with Eq. 15.
  Update the hypergraph transformer $g(\cdot; \mathcal{G}_e, \theta)$ with Eq. 10.
end

Smoothness-inducing Regularization Due to the limited data from the target task, the standard fine-tuning of the hypergraph transformer model can lead to overfitting on the training instances, resulting in poor generalization to test data (Ramasesh et al., 2021).

To alleviate this issue, we maintain an additional smoothed model $g(\widetilde{\theta})$, initialized by the pretrained model $g(\cdot; \mathcal{G}_b, \theta)$. In the t-th step, the parameter for the smoothed model $\widetilde{\theta}_t$ is updated as

$$\widetilde{\theta}_t = (1-\beta)\theta_t + \beta\widetilde{\theta}_{t-1}, \tag{8}$$

where β represents the smoothing factor, creating an exponential moving average between the parameters of the original predictor θ and the smoothed model from the previous timestep. To encourage consistency between predictions made by the original model $g(\theta)$ and the smoothed model $g(\widetilde{\theta})$ during fine-tuning, we add the additional consistency regularization between the original and the smoothed model to the learning objective as

$$\ell_{\text{ft}}(\theta) = \ell_{\text{cls}}(\theta) + \mu\mathbb{E}_{e_i\sim\mathcal{E}_e}\mathcal{D}_{\text{KL}}\left(g(e_i; \mathcal{G}_e, \theta); g(e_i; \mathcal{G}_e, \widetilde{\theta}_{t-1})\right), \tag{9}$$

where $\mathcal{D}_{\text{KL}}$ is the Kullback–Leibler (KL) divergence and μ is the weight for the consistent loss. This regularization strategy effectively prevents aggressive parameter updates and enhances the model's generalization capabilities for the target prediction (Tarvainen and Valpola, 2017; Nichol et al., 2018).

Group-balanced Reweighting Apart from the issue of aggressive updates, an equally, if not more, important challenge for finetuning the predictor to target patient visits is the balance between patients with basic only and extra features. To address this, we propose a reweighting scheme for dynamically adjusting the weight of patients with basic and extra features (denoted as ω_{b} and ω_{e}, respectively) during the finetuning process. Note that in the finetuning stage, only a small subset of EHRs from patients with basic features is available, aiming to better replicate the perspective of a local medical institute. Thus, the learning objective after the reweighting stage can be written as

$$\mathcal{L}(\theta) = \omega_{\text{b}}\ell_{\text{ft,b}}(\theta) + \omega_{\text{e}}\ell_{\text{ft,e}}(\theta). \tag{10}$$

Here $\ell_{\text{ft,b}}(\theta)$ and $\ell_{\text{ft,e}}(\theta)$ stand for the finetuning loss, which is defined in Eq. 9. When learning with two groups of patients simultaneously, we hypothesize that an ideal choice of ω_{b} and ω_{e} would provide the biggest reduction on the training loss of the two groups. We approximate the loss reduction using first-order Taylor expansion:

$$\Delta\ell^{(t)} = \sum_{i\in\{\text{b,e}\}} \left(\ell_i(\theta - \alpha\nabla_\theta\mathcal{L}(\theta)) - \ell_i(\theta)\right) \tag{11}$$

$$\approx -\alpha \sum_{i\in\{\text{b,e}\}} \sum_{j\in\{\text{b,e}\}} \omega_i^{(t)} \left(\nabla_\theta\ell_i(\theta)\right)^{\intercal} \nabla_\theta\ell_j(\theta), \tag{12}$$

where α is the learning rate. In addition, we avoid the potential rapid change of weights for finetuning stability, by adding a KL divergence regularization between $\boldsymbol{\omega} = (\omega_{\text{b}}, \omega_{\text{e}})$ at different steps. This leads to the following optimization target:

$$\min_{\omega^{(t)}} \ \Delta\ell^{(t)} + \mathcal{D}_{\text{KL}}(\boldsymbol{\omega}^{(t)}, \boldsymbol{\omega}^{(t-1)}), \tag{13}$$

$$\text{s.t.} \quad \omega_{\text{b}} + \omega_{\text{e}} = 1. \tag{14}$$

By using the Lagrangian multiplier with KKT conditions, we obtain the closed-form solution for the weight on patients with basic and extra features in step t as:

$$\omega_i^{(t)} = \frac{\omega_i^{(t-1)} \cdot \exp\left(\langle s_i, \sum_j s_j\rangle\right)}{\sum_{k\in\{\text{b,e}\}} \omega_k^{(t-1)} \cdot \exp\left(\langle s_k, \sum_j s_j\rangle\right)}, \tag{15}$$

Table 1: Dataset Statistics.

Stats	UK Biobank	MIMIC-III
# basic features	642	846
# extra features	1371	6577
# health records	1629	12353
└─ # train samples w/ basic only	1140	8647
└─ # train samples w/ extra	164	1235
└─ # validate samples	162	1236
└─ # test samples	488	3706

where $i \in \{\mathrm{b}, \mathrm{e}\}$, $s_i = \nabla_\theta \ell_i(\theta)$ is the gradient for loss l_i. The ideal solution naturally takes into account the similarity of gradients between patients who have basic features and those with extra features. It prioritizes the allocation of weights that share more common needs with others, to enhance the robustness of the model across different patients.

4.3. Overall Algorithm

To better illustrate the learning procedure, the overall procedure is listed in Algorithm 1. It is worth noting that `HTP-Star` can be trained in an end-to-end manner, without heavy parameter tuning.

5. Experiments

5.1. Datasets and Tasks

We conduct experiments on two datasets: UK Biobank (Sudlow et al., 2015) and MIMIC-III (Johnson et al., 2016), with the statistics shown in Table 1.

The UK Biobank dataset (Sudlow et al., 2015) is a comprehensive biomedical national biobank and research initiative based in the United Kingdom. It involves participants aged 40 to 69 who were enrolled between 2006 and 2010. It recruits a small subset of patients to take part in an assessment, where extra features such as sleep hours and cardiac monitoring are recorded[2]. We conduct an outcome prediction task which predicts whether the patients with type 2 diabetes would experience cardiovascular disease (CVD) endpoints within 10 years after their initial diagnosis. Specifically, CVD endpoints represent the presence of coronary heart disease (CHD), congestive heart failure (CHF), dilated cardiomyopathy (DCM), myocardial infarction (MI), or Stroke. Please refer to Appendix B for preprocessing details.

The MIMIC-III dataset (Johnson et al., 2016) contains over 40,000 de-identified patients in critical care units of the Beth Israel Deaconess Medical Center from 2001 to 2012. We conduct phenotyping prediction on MIMIC-III, which is formulated as a multi-label classification on the 25 pre-defined phenotypes by Harutyunyan et al. (2019). Specifically, given the patients' health records, we aim to predict whether the 25 acute care conditions are present in their next visits. See Appendix B for the detailed list of the phenotypes. In the preprocessing stage, we extract patients with multiple hospital visits and create pairs of consecutive visits for each patient. For each pair, we extract the diseases, medications, procedures, and services in the former visit as input features. Among them, diseases are considered as the basic features, and the others are considered as extra features. MIMIC-III is in a simulated setting and thus the extra features are intentionally masked out for most patients, even though some of them might have that information available in the dataset. This is because diseases are typically readily available through claims data, and often serve as the primary focus in various analytical tasks (Wu et al., 2023b). The phenotypes present in the latter visit serve as the corresponding labels.

For both datasets, we construct two subgroups for evaluation. To ensure a fair comparison, we maintain the same set of patients in both subgroups, with one group having only basic features, and the other having additional extra features. Moreover, we evaluate `HTP-Star` and all baselines on both subgroups to show their capabilities in two different scenarios.

5.2. Baselines

We mainly compare `HTP-Star` with two groups of baselines. The detailed description of baselines is deferred to Appendix C.We employ Accuracy, Macro AUROC and Macro AUPR as evaluation metrics.

Traditional ML Baselines These methods do not leverage graph structure to model the relationships between patients and features. For these methods, we consider three variants — *basic* (patients with basic features only), *extra* (patients with extra features only), and *combined* (considering all patients, and zero-padding is used to ensure the alignment of basic and extra features dimensions). In this group of baselines, we consider three techniques: (1) **Logistic Regression** (LR, Keyhani et al. (2008)), (2) **XGBoost** (Chen and Guestrin, 2016) and (3) **Transformer** (Li et al., 2020).

2. Some other features could be found at https://biobank.ctsu.ox.ac.uk/crystal/browse.cgi

Table 2: Performance on UK Biobank and MIMIC-III compared with baselines. "P/F" stands for methods with pretrain-then-finetuning. "HyG" represents hypergraph. **Bold** indicates the best result across all models. The result is averaged over 5 runs. * denotes statistical significant results ($p < 0.05$).

Model	UK Biobank						MIMIC-III					
	Basic			Full			Basic			Full		
	ACC	AUROC	AUPR	ACC	AUROC	AUPR	ACC	AUROC	AUPR	ACC	AUROC	AUPR
LR w/ Basic Feat.	67.90	51.76	46.69	67.90	51.76	46.69	75.85	72.31	54.25	75.85	72.31	54.25
LR w/ Full Feat.	64.20	57.36	48.05	69.14	68.50	64.32	74.98	66.86	50.30	74.59	67.41	49.11
LR w/ Both	62.96	52.35	43.58	67.90	64.79	60.82	75.01	72.28	54.25	75.10	68.09	49.50
XGBoost w/ Basic Feat.	61.73	52.82	44.54	61.73	52.82	44.54	75.97	72.44	54.61	75.97	72.44	54.61
XGBoost w/ Full Feat.	64.20	50.00	35.80	64.20	63.79	57.27	76.83	66.71	50.19	76.06	69.61	51.85
XGBoost w/ Both	64.20	52.35	44.84	64.20	67.51	56.63	76.88	72.60	54.54	75.31	71.33	53.25
Transformer w/ Basic Feat.	62.96±0.32	59.88±0.21	46.42±0.32	62.96±0.32	59.88±0.21	46.42±0.32	72.45±0.92	74.64±0.61	59.47±0.76	72.45±0.92	74.64±0.61	59.47±0.76
Transformer w/ Full Feat.	64.20±0.00	41.35±0.28	31.43±0.45	64.20±0.00	54.21±0.35	37.24±0.42	71.72±0.13	72.76±0.24	57.46±0.36	71.72±0.13	72.92±0.25	57.45±0.50
Transformer w/ Both	64.20±0.00	40.22±0.36	30.71±0.40	64.20±0.00	54.31±0.39	37.63±0.40	72.21±0.27	74.55±0.42	59.49±0.69	71.47±0.26	72.35±0.47	56.07±0.59
HyG + vanilla P/F	62.96±0.92	56.79±0.38	48.74±0.26	**69.14±0.99**	71.56±0.37	61.26±1.84	75.04±0.69	75.77±0.59	62.66±0.38	74.67±0.60	76.54±0.92	63.36±0.84
HyG + Reweight P/F	65.43±0.00	57.33±0.49	49.40±0.09	65.43±0.00	70.23±0.58	64.44±0.20	75.10±1.14	76.04±0.53	63.14±0.96	74.36±1.29	76.65±0.87	62.72±0.80
HyG + AUX-TS P/F	62.96±0.83	56.13±0.44	49.63±0.15	64.20±0.86	58.69±0.43	38.09±1.36	75.89±1.01	77.06±0.40	66.27±0.62	64.69±1.27	63.84±1.34	47.02±0.97
HyG + G-Adv P/F	65.43±0.00	59.98±0.27	48.50±0.11	64.20±1.12	57.29±0.78	39.78±2.01	73.79±1.20	76.71±0.58	62.56±0.94	73.15±1.35	75.23±0.97	60.85±0.73
HyG + ForkMerge P/F	64.20±1.10	58.39±0.35	50.28±0.21	66.67±0.94	74.20±0.23	64.28±0.19	75.56±1.70	76.11±0.54	64.14±0.88	70.85±1.21	70.12±1.63	54.75±1.18
`HTP-Star` (HyG + proposed P/F)	**72.84±0.71***	**60.11±0.43**	**50.78±0.16***	67.90±0.88	**74.40±0.28**	**65.54±0.12***	**78.17±1.09***	**81.00±0.42***	**69.54±0.70***	**77.06±0.69***	**79.56±0.92***	**67.13±0.55***

Pretrain-then-Finetune Baselines These baselines propose additional training techniques to facilitate knowledge transfer and improve the model's generalization ability. Specifically, we consider the following baselines: (4) **PT-FT** (Xu et al., 2022), (5) **Reweight** (Li et al., 2023), (6) **AUX-TS** (Han et al., 2021), (7) **G-Adv** (Dai et al., 2022; Liu et al., 2021b), (8) **ForkMerge** (Jiang et al., 2023a). We are aware that there are additional techniques for EHR-based clinical predictions, however, they either focus on designing neural architectures (Zhu and Razavian, 2021) or leverage additional knowledge (Park et al., 2022; Cui et al., 2023; Jiang et al., 2023b; Xu et al., 2024), thus are orthogonal to the focus of this work.

5.3. Implementation Details

We implement our model in PyTorch (Paszke et al., 2019). We tune the learning rate (α) in the range of {1e-4, 2e-4, 1e-3} and set it as 1e-3 in the pretraining stage and 2e-4 in the finetuning stage. We use Adam (Kingma and Ba, 2014) as the optimizer with a weight decay of 1e-3. We set μ in Eq. 9 as 0.5, β in Eq. 8 as 0.5, and number of layers l in the hypergraph transformer as 3. For the local The experiment is run on a single NVIDIA Titan RTX GPU. We study the effect of μ and β in Section 5.6.

5.4. Experimental Results

Table 2 summarizes the experimental results of `HTP-Star` compared with baselines. Note that *AUROC* is the main metric for the model performance. From the results, we have the following findings:

◇ Models following the pretrain-then-finetune pipeline generally exhibit better performance compared to traditional ML methods which face challenges in implementing this pipeline due to the dimension mismatch issue mentioned in section 3.2.

◇ Directly leveraging the vanilla pretrain-then-finetune can be suboptimal, as it performs well on patients with full features but is less satisfactory for patients with basic features. We attribute this phenomenon to the issue of *catastrophic forgetting*, where the model may *forget* the knowledge learned during the pretraining stage (Mehta et al., 2023). This further highlights the need for designing effective finetuning techniques to circumvent this issue.

◇ When compared to other transfer learning techniques, our framework achieves better performance. This is because some baselines (e.g. G-Adv, ForkMerge) are mainly proposed to improve the robustness of finetuning without modeling the relations between patients with basic and extra features, while other baselines (e.g. Reweight, AUX-TS) mainly use loss scales and gradient cosine similarity to reweight different group, which fail to consider overall loss reduction. On the contrary, we propose dynamically reweight different patient groups to avoid sacrificing the average performance.

5.5. Ablation Study

We study the effect of different components of `HTP-Star` on the two datasets, shown in Figure 5. We observe that both Smoothness-inducing Regularization and Group-balanced Reweighting are beneficial to the model performance, as they address the catastrophic forgetting issue and identify an opti-

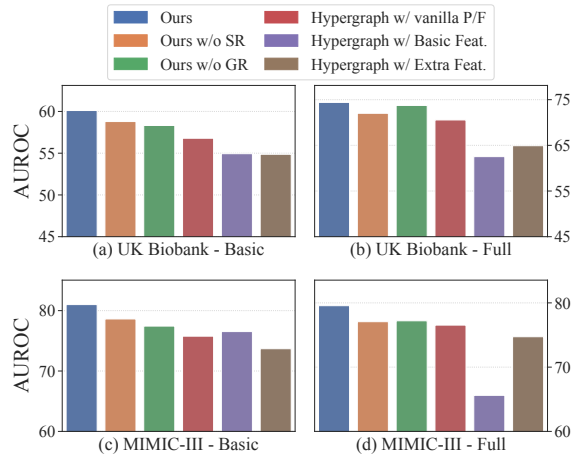


Figure 5: Effect of different components of `HTP-Star` on the two datasets. SR and GR stands for Smoothness-inducing Regularization and Group-balanced Reweighting, respectively.

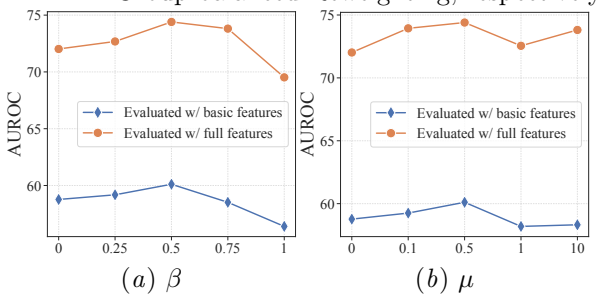


Figure 6: Parameter studies on UK Biobank.

mized gradient direction that balances between basic features and extra features. Additionally, we also demonstrate that the pretrain-then-finetune pipeline generally enhances the model performance in both evaluation scenarios. Simply using only patients with basic features, or patients with extra features do not harness all the information.

5.6. Parameter Study

We study the effect of β and μ in Eq. 8 and Eq. 9, respective, in Figure 6. The results indicate that the model achieves its optimal performance when the smoothing factor β is set to 0.5, which evenly balances the influence of both the smoothed model and the original model. When β equals 1, only the smoothed model is considered, while β at 0 implies the model operates without smoothness-inducing regularization. In both extreme cases, the model neglects information from either the smoothed or original model, leading to reduced performance. Additionally, the parameter μ serves as the weight for the consistency loss in Eq. 9. A higher value of μ signifies a greater alignment with the previous smoothed model, while a lower value of μ places more emphasis on the original model. The optimal model performance is achieved when μ is set to 0.5.

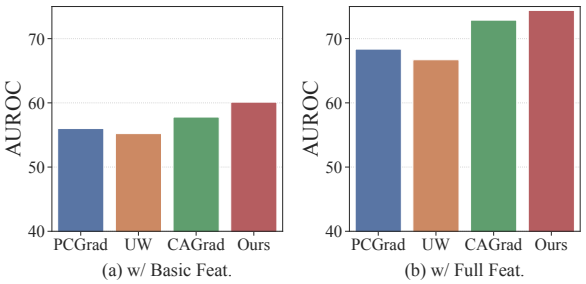


Figure 7: Performance of `HTP-Star` with different balancing method.

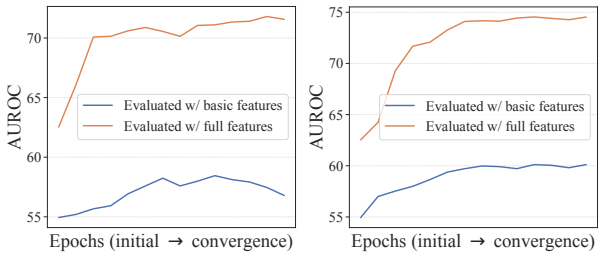
5.7. Study on Different Balancing Methods

As `HTP-Star` includes a reweighting step to balance the weight between two different patient groups, we further compare with other generic reweighing methods originally proposed for multi-task learning to understand the benefit of our design further. Specifically, we compare with three representative methods: *Uncertainty Weighting* (Kendall et al., 2018) that leverages task homoscedastic uncertainty to weight each group; *CAGrad* (Liu et al., 2021a) and *PCGrad* (Yu et al., 2020), which design gradient harmonization approaches to avoid negative transfer.

Figure 7 illustrates the result. We observe that UW does not perform well in our setting, as we observe that the training process can be highly unstable, especially for patients with full features. Incorporating gradient harmonization approaches is beneficial, but the gain is not so significant as they do not take the overall loss reduction into account. These results corroborate the advantage of our proposed group-balanced reweighting.

5.8. A Closer Look at the Finetuning Stage

Figure 8 illustrates the learning curve for the finetuning phase of `HTP-Star`, in comparison with the hypergraph transformer incorporating vanilla pretraining and finetuning. Our model effectively balances performance across patients with basic and extra features, enhancing their performance simultaneously. In contrast, the baseline model experiences a decline



(*a*) HG + vanilla P/F (*b*) `HTP-Star`

Figure 8: Learning curve of the finetuning stage from `HTP-Star`, compared with hypergraph transformer with vanilla pretraining and finetuning on UK Biobank.

in performance on patients with basic features while showing continuous improvement for those with full features. This disparity occurs because the baseline model fails to maintain a balance between the two evaluation scenarios and gradually forgets pretraining information during finetuning.

6. Conclusion

We introduce `HTP-Star`, a framework leveraging hypergraph structures within a pretrain-then-finetune framework for EHR modeling, facilitating seamless integration of additional features. Additionally, we propose two techniques: (1) *Smoothness-inducing Regularization* and (2) *Group-balanced Reweighting*, to enhance model robustness during finetuning. Through experiments on two real EHR datasets, we demonstrate that `HTP-Star` consistently outperforms various baselines, maintaining a balance between patients with basic and extra features.

7. Limitation

In this work, we mainly focus on other medical codes as extra features, but in real clinical applications, there could be other types of features from other modalities, e.g. text (Park et al., 2022), images (Lee et al., 2023), or time series (McDermott et al., 2021; King et al., 2023). It is important to design techniques to incorporate data from these modalities to further broaden the application range of `HTP-Star`.

Besides, the inclusion of a pretrain-then-finetune pipeline leads to longer training time, which can be problematic when there are large amount of patient data. A promising avenue for future research involves designing efficient training techniques to improve the scalability of our proposed framework.

Acknowledgement

We thank the anonymous reviewers and area chairs for valuable feedbacks. This research was partially supported by the Emory Global Diabetes Center of the Woodruff Sciences Center, Emory University. Research reported in this publication was supported by the National Institute Of Diabetes And Digestive And Kidney Diseases of the National Institutes of Health under Award Number K25DK135913. The research also receives partial support by the National Science Foundation under Award Number IIS-2145411. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

References

Jessica Y Bo, Hen-Wei Huang, Alvin Chan, and Giovanni Traverso. Pretraining ecg data with adversarial masking improves model generalizability for data-scarce tasks. *arXiv preprint arXiv:2211.07889*, 2022.

Derun Cai, Chenxi Sun, Moxian Song, Baofeng Zhang, Shenda Hong, and Hongyan Li. Hypergraph contrastive learning for electronic health records. In *Proceedings of the 2022 SIAM International Conference on Data Mining (SDM)*, pages 127–135. SIAM, 2022.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 785–794, 2016.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

Edward Choi, Mohammad Taha Bahadori, Jimeng Sun, Joshua Kulas, Andy Schuetz, and Walter Stewart. Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. *Advances in neural information processing systems*, 29, 2016.

Edward Choi, Zhen Xu, Yujia Li, Michael Dusenberry, Gerardo Flores, Emily Xue, and Andrew Dai. Learning the graphical structure of electronic

health records with graph convolutional transformer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 606–613, 2020.

Martin R Cowie, Juuso I Blomster, Lesley H Curtis, Sylvie Duclaux, Ian Ford, Fleur Fritz, Samantha Goldman, Salim Janmohamed, Jörg Kreuzer, Mark Leenay, et al. Electronic health records to facilitate clinical research. *Clinical Research in Cardiology*, 106:1–9, 2017.

Hejie Cui, Jiaying Lu, Shiyu Wang, Ran Xu, Wenjing Ma, Shaojun Yu, Yue Yu, Xuan Kan, Tianfan Fu, Chen Ling, et al. A survey on knowledge graphs for healthcare: Resources, application progress, and promise. In *ICML 3rd Workshop on Interpretable Machine Learning in Healthcare (IMLH)*, 2023.

Quanyu Dai, Xiao-Ming Wu, Jiaren Xiao, Xiao Shen, and Dan Wang. Graph transfer learning via adversarial domain adaptation with graph convolution. *IEEE Transactions on Knowledge and Data Engineering*, 35(5):4908–4922, 2022.

Thomas Desautels, Jacob Calvert, Jana Hoffman, Qingqing Mao, Melissa Jay, Grant Fletcher, Chris Barton, Uli Chettipally, Yaniv Kerem, and Ritankar Das. Using transfer learning for improved mortality prediction in a data-scarce hospital setting. *Biomedical informatics insights*, 9: 1178222617712994, 2017.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019.

Gabriel Erion, Joseph D Janizek, Carly Hudelson, Richard B Utarnachitt, Andrew M McCoy, Michael R Sayre, Nathan J White, and Su-In Lee. Coai: Cost-aware artificial intelligence for health care. *Nature biomedical engineering*, 6(12):1384, 2022.

Hossein Estiri, Sebastien Vasey, and Shawn N Murphy. Generative transfer learning for measuring plausibility of ehr diagnosis records. *Journal of the American Medical Informatics Association*, 28(3): 559–568, 2021.

Julie Gandrup, Syed Mustafa Ali, John McBeth, Sabine N van der Veer, and William G Dixon. Remote symptom monitoring integrated into electronic health records: a systematic review. *Journal of the American Medical Informatics Association*, 27(11):1752–1763, 2020.

Xueting Han, Zhenhuan Huang, Bang An, and Jing Bai. Adaptive transfer learning on graph neural networks. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 565–574, 2021.

Hrayr Harutyunyan, Hrant Khachatrian, David C Kale, Greg Ver Steeg, and Aram Galstyan. Multitask learning and benchmarking with clinical time series data. *Scientific data*, 6(1):96, 2019.

Chuan Hong, Everett Rush, Molei Liu, Doudou Zhou, Jiehuan Sun, Aaron Sonabend, Victor M Castro, Petra Schubert, Vidul A Panickan, Tianrun Cai, et al. Clinical knowledge extraction via sparse embedding regression (keser) with multi-center large scale electronic health record data. *NPJ digital medicine*, 4(1):151, 2021.

Junguang Jiang, Baixu Chen, Junwei Pan, Ximei Wang, Dapeng Liu, jie jiang, and Mingsheng Long. Forkmerge: Mitigating negative transfer in auxiliary-task learning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023a.

Pengcheng Jiang, Cao Xiao, Adam Cross, and Jimeng Sun. Graphcare: Enhancing healthcare predictions with open-world personalized knowledge graphs. *arXiv preprint arXiv:2305.12788*, 2023b.

Alistair E. W. Johnson, Tom J. Pollard, Lu Shen, Liwei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3(160035), 2016.

Ruth Johnson, Michelle M Li, Ayush Noori, Owen Queen, and Marinka Zitnik. Graph ai in medicine. *arXiv preprint arXiv:2310.13767*, 2023.

Alex Kendall, Yarin Gal, and Roberto Cipolla. Multitask learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491, 2018.

Salomeh Keyhani, Paul L Hebert, Joseph S Ross, Alex Federman, Carolyn W Zhu, and Albert L Siu. Electronic health record components and the quality of care. *Medical care*, pages 1267–1272, 2008.

Ryan King, Tianbao Yang, and Bobak J. Mortazavi. Multimodal pretraining of medical time series and notes. In *Proceedings of the 3rd Machine Learning for Health Symposium*, volume 225 of *Proceedings of Machine Learning Research*, pages 244–255. PMLR, 10 Dec 2023.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

William G La Cava, Elle Lett, and Guangya Wan. Fair admission risk prediction with proportional multicalibration. In *Proceedings of the Conference on Health, Inference, and Learning*, volume 209 of *Proceedings of Machine Learning Research*, pages 350–378. PMLR, 22 Jun–24 Jun 2023.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International conference on machine learning*, pages 3744–3753. PMLR, 2019.

Kwanhyung Lee, Soojeong Lee, Sangchul Hahn, Heejung Hyun, Edward Choi, Byungeun Ahn, and Joohyung Lee. Learning missing modal electronic health records with unified multi-modal data embedding and modality-aware attention. *arXiv preprint arXiv:2305.02504*, 2023.

Can Li, Sirui Ding, Na Zou, Xia Hu, Xiaoqian Jiang, and Kai Zhang. Multi-task learning with dynamic re-weighting to achieve fairness in healthcare predictive modeling. *Journal of Biomedical Informatics*, 143:104399, 2023.

Yikuan Li, Shishir Rao, José Roberto Ayala Solares, Abdelaali Hassaine, Rema Ramakrishnan, Dexter Canoy, Yajie Zhu, Kazem Rahimi, and Gholamreza Salimi-Khorshidi. Behrt: transformer for electronic health records. *Scientific reports*, 10(1):7155, 2020.

Zachary C Lipton, David C Kale, Randall Wetzel, et al. Modeling missing data in clinical time series with rnns. *Machine Learning for Healthcare*, 56 (56):253–270, 2016.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. *Advances in Neural Information Processing Systems*, 34:18878–18890, 2021a.

Xiaofeng Liu, Xiongchang Liu, Bo Hu, Wenxuan Ji, Fangxu Xing, Jun Lu, Jane You, C-C Jay Kuo, Georges El Fakhri, and Jonghye Woo. Subtype-aware unsupervised domain adaptation for medical diagnosis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 2189–2197, 2021b.

Chang Lu, Tian Han, and Yue Ning. Context-aware health event prediction via transition functions on dynamic disease graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 4567–4574, 2022.

Junyu Luo, Muchao Ye, Cao Xiao, and Fenglong 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, 2020.

Matthew McDermott, Bret Nestor, Evan Kim, Wancong Zhang, Anna Goldenberg, Peter Szolovits, and Marzyeh Ghassemi. A comprehensive ehr timeseries pre-training benchmark. In *Proceedings of the Conference on Health, Inference, and Learning*, pages 257–278, 2021.

Sanket Vaibhav Mehta, Darshan Patil, Sarath Chandar, and Emma Strubell. An empirical investigation of the role of pre-training in lifelong learning. *Journal of Machine Learning Research*, 24(214):1–50, 2023.

Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018.

Chao Pang, Xinzhuo Jiang, Krishna S Kalluri, Matthew Spotnitz, RuiJun Chen, Adler Perotte, and Karthik Natarajan. Cehr-bert: Incorporating temporal information from structured ehr data to improve prediction tasks. In *Machine Learning for Health*, pages 239–260. PMLR, 2021.

Sungjin Park, Seongsu Bae, Jiho Kim, Tackeun Kim, and Edward Choi. Graph-text multi-modal pretraining for medical representation learning. In *Proceedings of the Conference on Health, Inference,*

and Learning, volume 174 of *Proceedings of Machine Learning Research*, pages 261–281. PMLR, 07–08 Apr 2022.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Vinay Venkatesh Ramasesh, Aitor Lewkowycz, and Ethan Dyer. Effect of scale on catastrophic forgetting in neural networks. In *International Conference on Learning Representations*, 2021.

James R Rogers, Cong Liu, George Hripcsak, Ying Kuen Cheung, and Chunhua Weng. Comparison of clinical characteristics between clinical trial participants and nonparticipants using electronic health record data. *JAMA Network Open*, 4 (4):e214732–e214732, 2021.

Junyuan Shang, Tengfei Ma, Cao Xiao, and Jimeng Sun. Pre-training of graph augmented transformers for medication recommendation. In *28th International Joint Conference on Artificial Intelligence, IJCAI 2019*, pages 5953–5959. International Joint Conferences on Artificial Intelligence, 2019.

Wenqi Shi, Ran Xu, Yuchen Zhuang, Yue Yu, Jieyu Zhang, Hang Wu, Yuanda Zhu, Joyce Ho, Carl Yang, and May D Wang. Ehragent: Code empowers large language models for complex tabular reasoning on electronic health records. *arXiv preprint arXiv:2401.07128*, 2024.

Cathie Sudlow, John Gallacher, Naomi Allen, Valerie Beral, Paul Burton, John Danesh, Paul Downey, Paul Elliott, Jane Green, Martin Landray, et al. Uk biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLoS medicine*, 12(3): e1001779, 2015.

Glen B Taksler, Jarrod E Dalton, Adam T Perzynski, Michael B Rothberg, Alex Milinovich, Nikolas I Krieger, Neal V Dawson, Mary J Roach, Michael D Lewis, and Douglas Einstadter. Opportunities, pitfalls, and alternatives in adapting electronic health records for health services research. *Medical Decision Making*, 41(2):133–142, 2021.

Shengpu Tang, Maggie Makar, Michael Sjoding, Finale Doshi-Velez, and Jenna Wiens. Leveraging factored action spaces for efficient offline reinforcement learning in healthcare. *Advances in Neural Information Processing Systems*, 35:34272–34286, 2022.

Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30, 2017.

Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big data*, 3(1):1–40, 2016.

Jialun Wu, Kai He, Rui Mao, Chen Li, and Erik Cambria. Megacare: Knowledge-guided multi-view hypergraph predictive framework for healthcare. *Information Fusion*, 100:101939, 2023a.

Kevin Wu, Dominik Dahlem, Christopher Hane, Eran Halperin, and James Zou. Collecting data when missingness is unknown: a method for improving model performance given under-reporting in patient populations. In *Conference on Health, Inference, and Learning*, pages 229–242. PMLR, 2023b.

Cao Xiao, Trong Nghia Hoang, Shenda Hong, Tengfei Ma, and Jimeng Sun. Cheer: Rich model helps poor model via knowledge infusion. *IEEE transactions on knowledge and data engineering*, 34(2): 531–543, 2020.

Ran Xu, Yue Yu, Chao Zhang, Mohammed K Ali, Joyce C Ho, and Carl Yang. Counterfactual and factual reasoning over hypergraphs for interpretable clinical predictions on ehr. In *Proceedings of the 2nd Machine Learning for Health symposium*, volume 193 of *Proceedings of Machine Learning Research*, pages 259–278. PMLR, 28 Nov 2022.

Ran Xu, Wenqi Shi, Yue Yu, Yuchen Zhuang, Bowen Jin, May D Wang, Joyce C Ho, and Carl Yang. Ram-ehr: Retrieval augmentation meets clinical predictions on electronic health records. *arXiv preprint arXiv:2403.00815*, 2024.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in Neural Information Processing Systems*, 33:5824–5836, 2020.

Yue Yu, Simiao Zuo, Haoming Jiang, Wendi Ren, Tuo Zhao, and Chao Zhang. Fine-tuning pre-trained language model with weak supervision: A contrastive-regularized self-training approach. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1063–1077, 2021.

Weicheng Zhu and Narges Razavian. Variationally regularized graph-based representation learning for electronic health records. In *Proceedings of the Conference on Health, Inference, and Learning*, pages 1–13, 2021.

Appendix A. Details about Hypergraph Transformer

MHA computes the attention in parallel h heads as:

$$\text{MHA}(\boldsymbol{X}) = \text{Concat}\left(\text{head}_1, \ldots, \text{head}_h\right) \boldsymbol{W}_o, \quad (16)$$

$$\text{head}_i = \text{Softmax}\left(\boldsymbol{W}_{q_i}(\boldsymbol{X}\boldsymbol{W}_{k_i})^T/\sqrt{d_h}\right)\boldsymbol{X}\boldsymbol{W}_{v_i}, \quad (17)$$

where $\boldsymbol{X} \in \mathbb{R}^{n\times d}$ is the input, $d_h = d/h$, and $\boldsymbol{W}_{q_i} \in \mathbb{R}^{1\times d_h}$, $\boldsymbol{W}_{k_i}, \boldsymbol{W}_{v_i} \in \mathbb{R}^{d\times d_h}$ are query, key, and value projection matrices for i-th head, respectively[3]. $\boldsymbol{W}_o \in \mathbb{R}^{d\times d}$ is an output projection matrix. The fully connected feed-forward neural network (FFNN) comprises two linear layers with an activation function:

$$\text{FFNN}(\boldsymbol{X}) = \sigma\left(\boldsymbol{X}\boldsymbol{W}_{f_1} + \boldsymbol{b}_1\right)\boldsymbol{W}_{f_2} + \boldsymbol{b}_2 \quad (18)$$

where $\boldsymbol{W}_{f_1} \in \mathbb{R}^{d\times d_m}, \boldsymbol{W}_{f_2} \in \mathbb{R}^{d_m\times d}$, $\sigma(\cdot)$ is the activate function. A residual connection is used followed by a layer normalization.

Appendix B. Datasets and Tasks Details

UK Biobank In the preprocessing stage, we consider We consider patients in both inpatient and outpatient EHR with type 2 diabetes (ICD10 code of 'E11.XX'). The task labels are whether the patients develop CHD (ICD10 code of 'I25.XX'), CHF (ICD10 code of 'I50.XX'), DCM (ICD10 code of 'I42.XX'), MI (ICD10 code of 'I21.XX') and Stroke (ICD10 code of 'I66.XX') within 10 years of the diagnosis of type 2 diabetes. Note that we also consider death with causes of CHD, CHF, MI or Stroke as a positive outcome (label should be 1). Patients with an interval between their initial and last medical record of less than 10 years or with a documented medical history of any of the specified outcomes (CHD, CHF, DCM, MI, or Stroke) before their initial diabetes diagnosis are excluded from our analysis. To formulate a subset of the patients with extra features, we extract those patients who have been enrolled in the UKB assessment within two years before their initial diabetes diagnosis, as the assessment introduces additional features.

3. Here the size of $\boldsymbol{W}_{q_i}$ is $\mathbb{R}^{1\times d_h}$ since we only need to generate an aggregated embedding for each node/hyperedge.

Table 3: The 25 pre-defined phenotypes in the MIMIC-III dataset.

Phenotype	Type
Acute and unspecifed renal failure	acute
Acute cerebrovascular disease	acute
Acute myocardial infarction	acute
Cardiac dysrhythmias	mixed
Chronic kidney disease	chronic
Chronic obstructive pulmonary disease	chronic
Complications of surgical/medical care	acute
Conduction disorders	mixed
Congestive heart failure; nonhypertensive	mixed
Coronary atherosclerosis and related	chronic
Diabetes mellitus with complications	mixed
Diabetes mellitus without complication	chronic
Disorders of lipid metabolism	chronic
Essential hypertension	chronic
Fluid and electrolyte disorders	acute
Gastrointestinal hemorrhage	acute
Hypertension with complications	chronic
Other liver diseases	mixed
Other lower respiratory disease	acute
Other upper respiratory disease	acute
Pleurisy; pneumothorax; pulmonary collapse	acute
Pneumonia	acute
Respiratory failure; insufficiency; arrest	acute
Septicemia (except in labor)	acute
Shock	acute

MIMIC-III. Table 3 presents a detailed list of the 25 pre-defined phenotypes, which are identified using Clinical Classifications Software (CCS) from the Healthcare Cost and Utilization Project (HCUP)[4].

Appendix C. Baselines

We consider the following baselines in this work:

- Logistic Regression (LR, Keyhani et al. (2008)): It first transforms each EHR visit into a multi-hot vector, and then uses a linear layer to perform prediction.
- XGBoost (Chen and Guestrin, 2016): It optimizes the model's performance through gradient descent and regularization techniques.
- Transformer (Li et al., 2020): It directly uses a self-attention structure for modeling EHR visits.

4. https://hcup-us.ahrq.gov/toolssoftware/ccs/AppendixASingleDX.txt

- PT-FT (Xu et al., 2022): It adopts the standard pretrain-then-finetune pipeline, which first pretrain on patients with basic features, then finetune on patients with extra features without any other training strategies.

- Reweight (Li et al., 2023): It adaptively adjusts the weight between the patients between basic and extra features.

- AUX-TS (Han et al., 2021): It uses the cosine similarity between gradients of loss to balance the weight between basic and extra features.

- G-Adv (Dai et al., 2022; Liu et al., 2021b): It uses adversarial training during the fine-tuning stage to improve the model's robustness.

- ForkMerge (Jiang et al., 2023a): It is the most recent work on transfer learning, which forks the model into multiple branches and dynamically merges branches to enhance auxiliary-target generalization. To adapt it to our setting, we set two branches to encode the update in pretraining and fine-tuning, respectively.

We recognize that traditional missing data imputation (MDI) approaches seem to be applicable as baselines. However, we highlight the significant gap between MDI problems and our setting, which involves learning with both basic and extra features: (1) MDI approaches often make strong assumptions about the data distribution (e.g., missing at random), whereas in our setting, for patients with only basic features, the extra features are entirely missing. (2) MDI approaches typically lack the flexibility to handle mixed data types, including both continuous and categorical variables, as found in the EHR data used in our study. (3) In our setting, the volume of missing values is too substantial for missing data imputation methods to be feasible. Taking the UK Biobank as an example, according to Table 1, 1,140 out of 1,629 patients have missing values for the entire 1,371 out of 2,013 features. Given such a large proportion of missing data, applying MDI techniques would introduce significant bias in the imputed values.

Consequently, we believe that these traditional approaches are not directly adaptable to our setting without significant modifications.

Appendix D. Long-term Impact

`HTP-Star` introduces a novel approach to EHR modeling with several potential long-term impacts. Firstly, it enables more comprehensive and personalized healthcare decision-making by seamlessly integrating diverse patient data, including basic and additional features from local medical institutions. This could drive advancements in real-world clinical decision-making, healthcare delivery, and improved patient outcomes through better utilization of rich patient data. Additionally, it inspires future research into transfer learning and domain adaptation techniques for effective knowledge transfer across different patient populations. Moreover, it may also inspire follow-up work such as incorporating different data modalities, enhancing scalability, or applying the approach to other datasets.