A comprehensive EHR timeseries pre-training benchmark

A Comprehensive EHR Timeseries Pre-training Benchmark

Matthew McDermott*
mmd@mit.edu
CSAIL, MIT

Bret Nestor*
bretnestor@cs.toronto.edu
University of Toronto, Vector Institute

Evan Kim
CSAIL, MIT

Wancong Zhang
New York University

Anna Goldenberg
Hospital for Sick Children, University of Toronto, Vector Institute

Peter Szolovits
CSAIL, MIT

Marzyeh Ghassemi
University of Toronto, Vector Institute

ABSTRACT

Pre-training (PT) has been used successfully in many areas of machine learning. One area where PT would be extremely impactful is over electronic health record (EHR) data. Successful PT strategies on this modality could improve model performance in data-scarce contexts such as modeling for rare diseases or allowing smaller hospitals to benefit from data from larger health systems. While many PT strategies have been explored in other domains, much less exploration has occurred for EHR data. One reason for this may be the lack of standardized benchmarks suitable for developing and testing PT algorithms. In this work, we establish a PT benchmark dataset for EHR timeseries data, establishing cohorts, a diverse set of fine-tuning tasks, and PT-focused evaluation regimes across two public EHR datasets: MIMIC-III and eICU. This benchmark fills an essential hole in the field by enabling a robust manner of iterating on PT strategies for this modality. To show the value of this benchmark and provide baselines for further research, we also profile two simple PT algorithms: a self-supervised, masked imputation system and a weakly-supervised, multi-task system. We find that PT strategies (in particular weakly-supervised PT methods) can offer significant gains over traditional learning in few-shot settings, especially on tasks with strong class imbalance. Our full benchmark and code are publicly available at https://github.com/mmcdermott/comprehensive_MTL_EHR

CCS CONCEPTS

• **Computing methodologies** → **multitask learning**; *Neural networks*; **Learning latent representations**; • **Applied computing** → **Bioinformatics**; • **Theory of computation** → *Semi-supervised learning*.

KEYWORDS

benchmarks, neural networks, multi-task learning, pre-training, electronic health records, EHR, timeseries

ACM CHIL '21, April 8–10, 2021, Virtual Event, USA

ACM ISBN 978-1-4503-8359-2/21/04.
https://doi.org/10.1145/3450439.3451877

ACM Reference Format:
Matthew McDermott*, Bret Nestor*, Evan Kim, Wancong Zhang, Anna Goldenberg, Peter Szolovits, and Marzyeh Ghassemi. 2021. A Comprehensive EHR Timeseries Pre-training Benchmark. In *ACM Conference on Health, Inference, and Learning (ACM CHIL '21), April 8–10, 2021, Virtual Event, USA.* ACM, New York, NY, USA, 23 pages. https://doi.org/10.1145/3450439.3451877

1 INTRODUCTION

Pre-training (PT) methods are instrumental in the success of machine learning in various domains, including examples such as ImageNet [10] PT in computer vision and language-model PT (e.g., ELMO [25] or BERT [11]) in natural language processing. PT has enabled ML researchers to use large, unlabelled or weakly-labeled datasets to learn a representation of a data modality such that specific fine-tuning (FT) tasks can be learned successfully even with minimal task-specific data.

One domain where PT would be particularly impactful is processing electronic health record (EHR) data in machine learning for health (ML4H). ML4H presents a prime use-case for PT in part because there are many clinical applications of ML where the ability to leverage high-capacity models effectively even on relatively small, task-specific datasets would be important. For example, clinicians at smaller health systems could leverage public PT models produced on larger, more diverse populations to produce improved models for their specific institutions via FT. Researchers could also leverage PT models to aid in the study of rare [22] or novel (e.g., COVID-19) diseases, where there may not be enough data at *any* institution to train a high-capacity model from scratch. Lastly, researchers can leverage PT models to help reduce the need for annotating large, task-specific gold-standard datasets for specific research cohorts. These examples are also shown visually in Figure 1. Simultaneously, PT is eminently feasible in the clinical domain, as the large, EHR datasets collected at the point of care serve as natural sources of PT data. While these datasets are often only weakly labeled and noisy, making them challenging to work with in the context of traditional, fully supervised ML [14], PT algorithms often use self- or weakly-supervised algorithms and thus rely less on label availability and quality.

Despite these important application areas, PT has been only minimally explored in EHR timeseries data. In part, this may be because standardized sets of benchmarks for PT/FT paradigms do not exist for EHR data. While benchmarks do exist for ML over clinical tasks in general [15, 35], these are focused on traditional

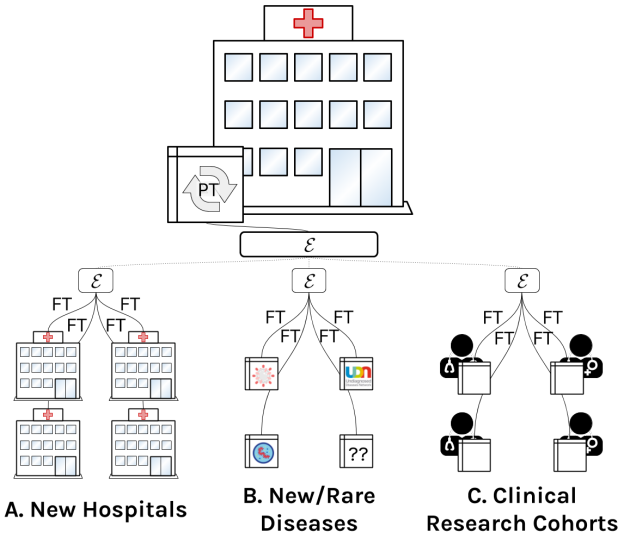

Figure 1: Pre-training (PT) an encoder $\mathcal{E}$ on a general domain, then fine-tuning (FT) it on a task specific problem fits naturally into many use-cases within ML4H. Examples include transferring a model from a large health system to smaller, community hospitals (A), specializing a model to a rare or novel disease sub-population (B), or supporting clinical research efforts which produce fully annotated datasets for select cohorts within a health system (C).

supervised learning, not PT/FT. In contrast to supervised learning, PT benchmarks are concerned primarily with how a system can optimally leverage PT data to improve performance in a disparate, secondary set of FT tasks. As one might not foresee all FT tasks at PT time, any effective benchmark must assess PT algorithms across a broad variety of tasks. Critically, we also cannot simply judge FT performance at a single FT dataset size—PT methods are exciting particularly because they enable models to be leveraged effectively even in few-shot settings, so we must judge PT algorithms over a variety of FT dataset sizes. Additionally, for PT systems in particular within ML4H, where many (though not all) use cases are multi-domain in nature, we should ensure we analyze PT system performance across multiple datasets.

In this work, we introduce the first comprehensive PT benchmark for clinical EHR timeseries data. We define a suite of FT tasks to consider across MIMIC-III [17] and eICU [26], as well as evaluation procedures for model performance across a variety of FT dataset sizes. In contrast with existing clinical benchmarks (e.g., [15]), our system includes multiple datasets, more tasks, and few-shot evaluations, all of which help support its use in analyzing PT algorithms. In addition, we provide two baselines against which the field can compare—first, a weakly-supervised, multi-task PT approach, and second, a masked-imputation based model reminiscent of a continuous analog of the BERT NLP model [11]. Based on these results, we find that while PT does not offer best-in-class performance for FT datasets at the full scale of MIMIC-III or eICU, PT can indeed be very helpful in the small-data regime, showing dramatic improvements in performance in particular on class-imbalanced, time-varying tasks across both datasets. These baseline results suggest that the gains offered by PT in clinical settings warrant future exploration, and we hope that this benchmark will help prompt those gains by enabling iterative development of PT paradigms in the clinical space.

2 RELATED WORKS

PT over EHR timeseries data has been explored only minimally, but PT on other clinical data modalities has been explored. Learning contextual representations of clinical codes, for example, has been explored via a variety of methods, often leveraging known biomedical hierarchies to improve performance [9, 29]. PT models for clinical text have also been thoroughly explored [1, 31, 40] and are regularly used in the context of clinical NLP.

Three recent examples do study topics closely related to PT over clinical timeseries data, however. In particular, Yoon et al. explored PT on tabular data via a masked-imputation based self- and semi-supervised algorithm [39], Xue et al. explore using meta-learning in a semi-supervised context to specialize PT to a specific downstream task over MIMIC-III [38], and Steinberg et al. explores a novel analog of language-modeling on discretized clinical timeseries data. Each of these three cases have slightly different foci, and thus are relevant to our work in different ways.

Yoon et al.'s work explores both self- and semi-supervised PT (of which only the self-supervised PT is relevant to us as we do not allow FT data to be leveraged at PT time); however, their primary improvements are demonstrated most soundly in semi-supervised PT, and there is minimal evidence that their algorithm offers consistent improvements in the self-supervised setting. Similarly, Xue et al.'s work is exclusively for semi-supervised learning. As a result, neither of these two works are directly comparable to our benchmark or results. Steinberg et al.'s work, however, is much more relevant. It focuses squarely on self-supervised PT, using a different analog of language model PT than our masked imputation model, and also studies clinical timeseries (albeit discretized clinical timeseries). However, their approach is tested on a private dataset, and thus is not suitable as a PT benchmark, which is our goal.

Beyond explicit PT systems, more general clinical representation learning has been explored extensively in the literature. Multi-task learning (MTL) has been explored significantly from this perspective [12, 15, 30, 36], as well as those focusing on auto-encoding, imputation, or clustering approaches [13, 34].

Benchmarks for PT paradigms are also growing in use in other domains. Rao et al. examines PT in the context of proteins, for example, and Liang et al. defines a benchmark for cross-lingual PT systems, a topic that is also of interest in clinical contexts such as diagnosing speech pathologies [2].

3 PROBLEM FORMULATION & NOTATION

Let $\boldsymbol{X}_{\text{PT}} \in \mathbb{R}^{N_{\text{PT}} \times D}$, paired with a collection of auxiliary tasks $\mathcal{T}_{\text{PT}}$ with associated labels $\boldsymbol{Y}_{\text{PT}} \in \mathbb{R}^{N_{\text{PT}} \times |\mathcal{T}_{\text{PT}}|}$ be our "pre-training" (PT) dataset. In addition, let $\mathcal{T}_{\text{FT}}$ denote our set of downstream (fine-tuning/FT) tasks and $\boldsymbol{X}_{\text{FT}} \in \mathbb{R}^{N_{\text{FT}} \times D}$, $\boldsymbol{Y}_{\text{FT}} \in \mathbb{R}^{N_{\text{FT}} \times |\mathcal{T}_{\text{FT}}|}$ denote the corresponding FT dataset. Note that $\boldsymbol{X}_{\text{FT}}$ may intersect non-trivially with $\boldsymbol{X}_{\text{PT}}$ (i.e., some data may overlap between the PT and the FT settings), *but* no tasks overlap directly between $\mathcal{T}_{\text{PT}}$ and $\mathcal{T}_{\text{FT}}$. Given

this lack of overlap in tasks, $\mathcal{T}_{\text{PT}}/Y_{\text{PT}}$ can serve as a form of weak-supervision for the ultimate FT tasks. In most practical scenarios, it will be the case that $N_{\text{FT}} \ll N_{\text{PT}}$.

Let our *pre-training* model be given by $\mathcal{M}(\boldsymbol{x}) = \mathcal{D}_{\text{PT}}(\mathcal{E}(\boldsymbol{x}))$, where $\mathcal{E}$ is an *encoder* (which we will ultimately transfer during FT) and $\mathcal{D}_{\text{PT}}$ is a *PT specific decoder* (which will not be transferred). Then, the goal of PT is to use the dataset $\boldsymbol{X}_{\text{PT}}$ (and possibly Y_{PT}) to learn the parameters of the pre-training model $\mathcal{M}$ such that $\mathcal{E}$ offers strong transfer performance for the tasks $\mathcal{T}_{\text{FT}}$, all without actually leveraging (or even knowing about) the fine-tuning labels Y_{FT} at any point during the PT process. Note that at fine-tuning time we will also train a freshly initialized decoder $\mathcal{D}_{\text{FT}}$ such that the full FT model makes predictions $\hat{\boldsymbol{y}} = \mathcal{D}_{\text{FT}}(\mathcal{E}(\boldsymbol{x}))$.

4 HIGH-LEVEL OVERVIEW

Here, we will provide an overview of the rest of the paper, to help provide a high-level grounding for the more detailed content in Section 5, which defines the benchmark's data and usage, and Section 6, which details our baseline PT experiments.

Our benchmark defines two separate cohorts: one over MIMIC-III and one over eICU (Section 5.1). Cohorts consist of timeseries of labs, vitals, & treatments. In addition, we also define a set of 10 clinically meaningful downstream tasks which we use as FT tasks (Section 5.2) to judge PT algorithms within our benchmark.

PT systems using our benchmark must fall into one of two categories: self-supervised, in which case they can only leverage the labs, vitals, and treatment dataset during PT, or weakly-supervised, in which case they can also leverage "off-target" tasks during PT as auxiliary labels (Section 5.4). After PT, models are fine-tuned under two distinct transfer regimes (Section 5.5) and across datasets ranging in size (Section 5.6) to simulate extreme $\frac{N_{\text{PT}}}{N_{\text{FT}}}$ ratios.

Ultimately, PT systems are judged on their final FT scores across all tasks, datasets, and $\frac{N_{\text{PT}}}{N_{\text{FT}}}$ ratios. In particular, to profile a PT system on our benchmark, one simply downloads the provided cohorts and the 5 standardized train/validation/test splits, tunes hyperparameter and trains their PT model according to the appropriate procedures (for either self- or weakly-supervised methods), then fine-tunes the model against our 10 downstream tasks at all dataset sizes. This usage procedure is detailed more in Section 5.7.

To demonstrate this use in practice, and establish baseline results for further research, we profile one self-supervised and one weakly-supervised PT method against our tasks in the manner described above (Section 6). Ultimately, even with simple PT methods, we see important improvements in the few-shot context (Section 6.4).

5 PRE-TRAINING BENCHMARK

5.1 Data Cohorts & Pre-processing

MIMIC-III Cohort Selection. Our MIMIC-III [17] cohort is extracted via the MIMIC-Extract pipeline [35], with missingness threshold set to 2% and otherwise default parameters. This pipeline extracts a cohort of ICU stay records corresponding to the first ICU stay of patients over age 15, extracting labs, vitals, and treatments, with labs & vitals aggregated into clinically meaningful buckets to produce a more robust representation [23]. ICD codes, comfort-measures-only (CMO)/do-not-resuscitate (DNR) codes, and records of death, discharge, and readmission are also extracted via novel extraction code primarily as task labels, not input signals, though CMO/DNR codes that are present or added during an input window are incorporated as features as well. Lastly, we also extract static, demographic data at a per-patient level. Appendix Table 6 reports the set of all labs & vitals we consider in this work along with their relative measurement rate. Treatments studied include various forms of ventilation, vasopressors, or fluid boluses (See Appendix Section A.1 for a full list). Static data includes age, gender, ethnicity, insurance type, admission type, and first care unit. Basic dataset statistics are shown in Table 1.

eICU Cohort Selection. To extract the eICU [26] data, we attempt to mimic the structure of our MIMIC-III cohort wherever possible. This cohort also extracts labs and vitals (See Appendix Table 6), as well as static demographic data (age, gender, ethnicity, and unit type). We also extract records of death and discharge to form our downstream tasks. This cohort contains only patients over age 15 and only labs & vitals measured for at least 5% of all observed time-points are included. In addition, as the eICU dataset is multi-institution, we also restrict our data to correspond only to institutions with at least 500 patients in the dataset. Extraction code for our eICU extraction system will be released publicly after publication. Basic dataset statistics are shown in Table 1.

Dataset Post-processing. Both datasets are standardized to hourly granularity and represented as numerical timeseries with missingness. Treatment records are also standardized hourly and concatenated to the numerical series via a one-hot encoding. Static data are duplicated and appended to each hour of the series. To form a pre-training or fine-tuning sample, we first sample a random ICU stay from the record, then a random end-time T within that stay, and treat all data for that stay prior to T as the *input window* for this sample, and the task labels corresponding either to the end of the patient's overall stay (for static tasks) or within a prescribed prediction window after T (for time-varying tasks) as the *labels* for this sample (see Section 5.2 for more details on task labels). Users may choose to featurize this input window however they like—in our baselines, for example, rather than processing the entire input window $[0, T]$, we use a fixed size window ranging from 12 - 96 hours ending at T for computational efficiency. Note that in evaluating rolling or time-varying tasks, whose labels will vary throughout patients' stays, we sample multiple random endpoints and aggregate evaluation results in a per-patient manner across those different endpoints to approximate the expected performance of such tasks at a per-patient level.

Dataset Splits & Release. To capture all relevant sources of variance, our benchmark consists of 5 random train/tuning/test splits (split by patient and ICU stay), so that a given PT system can separately undergo hyperparameter tuning, training, and evaluation (including fine-tuning training/hyperparameter tuning) across 5 different data splits. These splits are publicly available with the rest of our benchmark.

5.2 Benchmark Fine-tuning Tasks

Our benchmark consists of 10 FT tasks that span a variety of traditional ML4H targets as well as several new tasks. In the interest

Table 1: Dataset Statistics for our MIMIC-III and eICU Cohorts. Values are aggregated across the 5 random splits of our dataset, shown in the format "[mean] ± [standard deviation] ([min] - [max])." If only a single number is shown, the quantity does not vary enough to show any difference at the presented precision. Though the MIMIC-III cohort does include patients with very short stays, in practice we restrict our analyses to only those with sufficient data to encompass a single input window (at least 12 hours).

Dataset	Split	# Stays	# Patients	# Patient-Hrs	Hrs/Patient
MIMIC-III	Train	17.5K	17.5K	1.48M ± 2.22K (1.47M - 1.48M)	84.3 ± 47.2 (3 - 239)
	Tuning	2.19K	2.19K	183K ± 1.5K (182K - 186K)	83.9 ± 46.7 (6 - 239)
	Held-out Test	2.19K	2.19K	184K ± 2.33K (182K - 188K)	84.3 ± 47.2 (3 - 239)
eICU	Train	58.1K ± 34.4 (58.1K - 58.2K)	51.5K	4.1M	70.6 ± 45.8 (25 - 242)
	Tuning	7.27K ± 16.2 (7.26K - 7.3K)	6.44K	517K ± 3.99K (511K - 522K)	71.1 ± 46.3 (25 - 242)
	Held-out Test	7.28K ± 34.2 (7.26K - 7.34K)	6.44K	518K ± 2.46K (516K - 522K)	71.1 ± 46.4 (25 - 242)

of ensuring our set of tasks is sufficiently diverse so as to be as generalizable as possible over FT use cases, and in order to capture the variety of task definitions commonly used in the literature, we formulate many of our tasks in a multi-label format, with distinct labels within a single task spanning different possible configurations of the task. For example, our "Imminent Mortality" task encompasses a *multi-label* prediction of both mortality within 24 hours and mortality within 48 hours, both with different gap times. We'll use the term "task" to refer to the overarching learning target (e.g., "Imminent Mortality") and "label" to refer to an individual prediction target (e.g., binary prediction of mortality within 24 hours, or, using an example target from a different task, the presence of a particular ICD code in the record). In addition, we also will use the term "Rolling" to correspond to tasks with time-varying labels (e.g., prediction of mortality within the next 24 hours), "Static" to correspond to tasks that have a single, fixed label corresponding to the end of the patient's stay (e.g., predicting overall ICD codes), or "Autoregressive" to correspond to a task that explicitly is involved with forecasting the future state of the patient *in the feature-space used by our model*. Note that the focus in our task selection is first on utility for an ML benchmark, and second on direct clinical utility. Where the latter is certainly important, we choose to focus here on including a broad variety of tasks, on examining tasks that are well represented in the current ML literature, and on tasks can be defined at scale over MIMIC-III and/or eICU, such that we can easily examine the performance of models across various dataset sizes within MIMIC-III without anchoring ourselves to a particular set of clinical cohorts that already have gold-standard labels.

A full table of the tasks we use, across both datasets, is given in Table 2. In the remainder of this section, we will walk through each task in more detail. For each task, we will report a formal definition, over which cohorts the task is defined in this benchmark, over what input windows the task is predicted (e.g., either throughout the patient's stay or only based on the first 24 hours), a brief pointer to any relevant prior literature for the task, and more detailed majority class statistics for the tasks/labels. When reporting task definitions, we will frame our rolling tasks relative to the last measured timepoint in the sample's input window—e.g., if an input sample corresponds to the ICU stay record of patient p up to time T, and the task is defined over a prediction window of 24 hours, with a gap time of 2 hours, then the task will capture instances of a label within the time window $[T + 2, T + 24]$ for patient p. Note that we include the gap time to ensure both that the learning task isn't biased by any potential temporal leakage in the data and that any superficial signals that would be already known to clinicians during the input window (which is more likely when the task event, e.g., mortality, would take place just after T) don't overwhelm the learning objective. Task statistics will be reported at a *per-patient* level (i.e., rolling tasks will have labels first aggregated within a patient's record, then across patients, so as not to be biased by the behavior of patients with longer overall stays), and aggregated over the train set of all 5 standardized train/tuning/test splits in our benchmark. All statistics (as well as some not reported here) are also available in table form in the appendix, in Supplementary Tables 4, 5, 6, and 7).

5.2.1 *Imminent Mortality: MOR.*

Definition: We predict whether the patient's recorded time of death is within the subsequent 24/48 hours, with a 2/6 hour gap time.
Cohort: This task is available on both cohorts.
Input Window: Throughout the entire stay.
Prior Art: Prediction of imminent mortality has been studied extensively as a silver learning signal for more general physiological decompensation [15].
Statistics: 24h/48h mortality is false for 97.6 ± 9.91%/95.4 ± 17.36% and 97.9 ± 10.26%/96.2 ± 16.41% of hours per-patient for MIMIC-III and eICU.

5.2.2 *Comfort Measures: CMO.*

Definition: "Comfort Measures Only" (CMO) orders indicate that the (usually terminally ill) patient has requested to receive care *only* designed to provide comfort, not treatment, and otherwise the course of illness should be allowed to progress (typically to mortality). We predict whether a patient will add a CMO flag to their record over the next 24/48 hours, using a 2/6 hour gap time.
Cohort: This task is only available on MIMIC-III.
Input Window: Throughout the entire stay.
Prior Art: In the traditional ML4H community, CMO prediction is somewhat understudied. The only work we know of to study this prediction task is [20], which uses natural language processing over clinical notes and structured data to predict CMO codes and do not resuscitate (DNR) codes.

Table 2: The set of tasks defined in our benchmark dataset. These tasks are used both as FT tasks or for signals of weak-supervision during PT. Average (macro) majority class accuracy across train folds is reported for all classification tasks to give an estimate of the relative level of class imbalance of the task. More detailed MCA statistics are reported in Appendix Table 4. Both Future Treatment Sequence (FTS), and our more granular Final Acuity (ACU) task are novel tasks.
***Abbreviations: AR*: Auto-regressive, *Bin.*: binary classification, *ML:* binary multi-label classification, *MC:* multi-class classification, *SMC*: sequential decoding multi-class classification, *Reg.*: regression.**

Task	Abbr.	Specific Labels	Temporal	Gap	Pred.	Type	In eICU?	Rel. Work	Majority Class Acc.	
									MIMIC-III	eICU
Imminent Mortality	MOR	Mortality (24h)	Rolling	2h	24h	Bin.	✓	[15]	97%	97%
		Mortality (48h)		6h	48h		✓			
Comfort Measures	CMO	CMO added (24h)	Rolling	2h	24h	Bin.		[20]	98%	
		CMO added (48h)		6h	48h					
DNR Ordered	DNR	DNR added (24h)	Rolling	2h	24h	Bin.		[20]	96%	
		DNR added (48h)		6h	48h					
Imminent Discharge	DIS	Discharge (24h)	Rolling	2h	24h	MC	✓	[4]	43%	44%
		Discharge (48h)		6h	48h		✓			
ICD Code Prediction	ICD	Appendix Table 7	Static	12h	N/A	ML		[12, 15]	67%	
Long Length-of-Stay	LOS		Static	12h	N/A	Bin.	✓	[15, 23, 35]	53%	67%
30 Day ICU Readmission	REA		Static	N/A	N/A	Bin.		[15]	95%	
Final Acuity	ACU		Static	12h	N/A	MC	✓	[6, 30, 35]	25%	60%
Next Timepoint	WBM	Appendix Table 6	AR	0h	1h	ML		[5]	92%	88%
Future Treatment Sequence	FTS		AR	N/A	N/A	SMC		[24, 37]	97%	

Statistics: 24h/48h CMO status is false for 99.1 ± 5.55%/98.5 ± 9.59% of hours per-patient.

5.2.3 *DNR Ordered: DNR.*

Definition: "Do Not Resuscitate" (DNR) orders indicate that the patient has requested to not receive resuscitation care (e.g., cardiopulmonary resuscitation a.k.a. CPR). We predict whether a patient will add a DNR flag to their record over the next 24/48 hours, using a 2/6 hour gap time.
Cohort: This task is only available on MIMIC-III.
Input Window: Throughout the entire stay.
Prior Art: To the best of our knowledge this task has only been studied within the ML4H community in [20].
Statistics: 24h/48h DNR status is false for 96.6 ± 16.16%/96.1 ± 18.03% of hours per-patient.

5.2.4 *Imminent Discharge: DIS.*

Definition: We predict whether the patient will be discharged, *and if so to where* (e.g., discharged home vs. to a skilled nursing facility), within the next 24/48 hours, using a 2/6 hour gap time. Unlike the prior tasks, this task is both multi-label (across prediction/gap windows) and multi-class (across discharge locations).
Cohort: This task is available on both MIMIC-III and eICU.
Input Window: Throughout the entire stay.
Prior Art: Imminent discharge has been primarily predicted in operational contexts, rather than for use as a signal of acuity, e.g. [4].
Statistics: Within 24 hours, the patients are more commonly not discharged than they are discharged to any other possible individual discharge location (57.7 ± 24.68% and 48.2 ± 26.67% of hours per-patient for MIMIC-III and eICU). Within 48 hours, MIMIC-III patients are again most commonly not discharged (27.6 ± 26.58%), but eICU patients are most commonly discharged home (40.2 ± 35.95%). A full list of possible discharge locations, and their prevalence per-hour, per-patient, is shown in Appendix Tables 8,9 for the MIMIC-III and eICU cohorts.

5.2.5 *ICD Code Prediction: ICD.*

Definition: We predict the multi-label presence of each of the 19 major ICD category under the categorization of Slee.
Cohort: ICD codes are available only on the MIMIC-III dataset.
Input Window: The first 24 hours of data.
Prior Art: Prediction of ICD codes is commonly studied in ML4H as a phenotyping task [12, 15].
Statistics: Per-label majority class accuracies are shown in Supplementary Table 7. Macro-averaged across all categories, the majority class accuracy of this task is 67.0 ± 18.07%.

5.2.6 *Long Length-of-Stay: LOS.*

Definition: We predict via binary classification whether a patient's total length-of-stay will be longer than 3 days or not.
Cohort: LOS is available on both cohorts.
Input Window: The first 24 hours of data.
Prior Art: Long LOS has been predicted numerous times, both in a classification sense for 3-day LOS [35] and 7-day LOS [15].
Statistics: Patient LOS is longer than 3 days 47.1 ± 0.11% and 33.3 ± 0.04% of the time on MIMIC-III and eICU.

5.2.7 *30 Day ICU Readmission: REA.*

Definition: We predict whether or not patients *who are successfully*

discharged will be readmitted *to the ICU*[1] within 30 days.
Cohort: This task is defined only on the MIMIC-III cohort. As MIMIC-Extract extracts a cohort only of patients' *first* ICU stays [35], this task also has the bias of only being analyzed on the first ICU visit for a patient.
Input Window: 30-day ICU readmission is predicted over the entirety of the patient's data, up until discharge. In practice this often means that it will be predicted over a fixed size window of, e.g., 48 hours prior to discharge.
Prior Art: Rajkomar et al. [27] examine overall hospital readmission in their work.
Statistics: 95.1 ± 0.09% of patients aren't readmitted within 30 days.

5.2.8 *Final Acuity: ACU.*

Definition: We predict, in a multi-class manner, whether the patient will die—and if so, when (e.g., In-ICU v. In-Hospital)—or be discharged—and if so, to where (e.g., Home, a Skilled Nursing Facility)—at the end of their stay.
Cohort: This task is defined over both cohorts.
Input Window: The first 24 hours of data.
Prior Art: Various sub-forms of this task have been explored historically. In-ICU and in-hospital mortality, for example, have been explored as separate, binary classification tasks in numerous ways [7, 15, 35]. Challenging the model to predict death (including location of mortality) and the final discharge location jointly is novel, to the best of our knowledge.
Statistics: Prevalences of all classes for the final acuity task are shown in Supplementary Tables 11 and 10, for the MIMIC-III and eICU cohorts. The macro averaged majority class accuracy for this task, however, is 25.3 ± 0.24% of patients being discharged to a home health care system and 59.7 ± 0.1% being discharged to home for MIMIC-III and eICU.

5.2.9 *Next Timepoint Will be Measured: WBM.*

Definition: We predict which labs & vitals will be measured in the next hour via multi-label binary classification.
Cohort: This task is defined over both cohorts.
Input Window: Throughout the patient's stay.
Prior Art: Imputation and forecasting over clinical data has been explored extensively in the past, both as a necessary technical preprocessing step for large pipelines and as a vehicle for direct use in other clinical tasks, such as anomaly detection [21]. The classification formulation is somewhat less common than the regression formulation, but the analysis of measurement observation patterns in clinical data in general has been explored in a number of contexts beyond just prediction [5].
Statistics: The labs & vitals over which we predict, along with their observed measurement rates, are shown in Appendix Table 6 for both the MIMIC-III and eICU cohorts. Macro averaged majority class accuracy per-hour, per-patient for this task is 92.1 ± 9.18% on MIMIC-III and 88.0 ± 19.61% on eICU.

5.2.10 *Future Treatment Sequence: FTS.*

Definition: We predict the sequence of combinations of ventilation, vasopressor, and fluid bolus treatments the patient will receive over the remainder of their stay (bucketed to an hourly granularity), in a duration agnostic manner, meaning this task does not differentiate between a patient who is ventilated for one hour, followed by receiving vasopressors for two hours and a patient who is ventilated for two hours, followed by receiving vasopressors for one hour—in both cases, the task labels would simply be the sequence "ventilation, vasopressors."

As this task is a sequential decoding task, predictions for FTS must use more specialized prediction heads and training regimes than on our other tasks; our baselines, for example, rely on LSTM RNN decoders and teacher forcing [18], but other users may attempt different strategies. We evaluate this task in an autoregressive manner also using teacher forcing [18].
Cohort: This task is defined only on the MIMIC-III cohort.
Input Window: Throughout the patient's stay.
Prior Art: While this task formulation is novel, researchers have investigated learning optimal control policies for applications of treatments, including ventilators or vasopressors [16, 24, 37].
Statistics: We show the relative frequency of the various treatment combinations in Appendix Figure 5. The majority class accuracy of this task at a per-patient, per-hour level is 97.4 ± 2.77%.

5.3 Pre-training vs. Fine-tuning Data

For both cohorts, we leverage the full dataset (excluding separate hyperparameter tuning and held-out sets) as our PT data $\boldsymbol{X}_{\text{PT}}$. Naturally, this also means that our fine-tuning datasets $\boldsymbol{X}_{\text{FT}}$ will overlap with our PT data. While this renders our benchmark less reflective of cases where one would like to deploy a PT model on a disjoint FT dataset, there are also many use-cases where these two datasets will overlap.

5.4 Pre-training Regimes

Our benchmark supports two styles of PT: self-supervised and weakly-supervised. Under self-supervision, only a single PT model is pre-trained, which is then used to assess FT performance directly on each downstream task (through separate FT runs, all transferring from the single PT model). Under weak-supervision, we permit the user to leverage a portion of our provided downstream tasks at pre-training time while still ensuring there is no task leakage from PT to FT via a "leave-one-task-out" (LOTO) framework. If our total set of downstream tasks is given by $\mathcal{T}$, then the LOTO framework requires pre-training a separate encoder $\mathcal{E}_t$ per downstream task $t \in \mathcal{T}$ such that $\mathcal{E}_t$ is transferred only to FT task $\mathcal{T}_{\text{FT}} = \{t\}$ for evaluation and leverages only tasks $\mathcal{T}_{\text{PT}} = \{t' \in \mathcal{T} | t' \neq t\}$ for PT weak supervision signals.

5.5 Fine-tuning Regimes

We analyzed two different styles of FT transfer: fine-tuning, decoder-only (FTD), and fine-tuning, full (FTF).

In the fine-tuning decoder-only (FTD) setting, the encoder $\mathcal{E}$ is frozen after PT, and only the decoder $\mathcal{D}$ is allowed to change during the FT stage. In the fine-tuning full (FTF) setting, the entire model, including the PT encoder $\mathcal{E}$ and the FT decoder $\mathcal{D}$ (which is not initialized during PT), can be updated during FT. This setting allows greater capacity, at the expense of a risk of over-fitting during FT. In addition, we naturally also encourage users to profile traditional,

[1]Hospital readmission would be both a more natural and more actionable task in practice; however, the granularity of our input data only permits ICU readmission, so we use this as a proxy for the more traditional hospital readmission task.

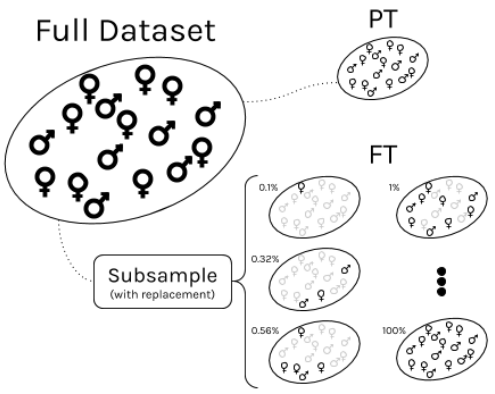

Figure 2: We always pre-train on the full available dataset, but additionally assess our models' ability to fine-tune in a few-shot context by randomly subsampling (with replacement) a variety of smaller FT datasets for each experiment.

non-PT, single-task (ST) models of the same architectures over these tasks, to establish baseline performance levels.

5.6 Few-shot Analyses

In addition to comparing FTF vs. FTD performance, we also assess FT systems across various FT dataset sizes to judge models across a wide range of $N_{\text{PT}}/N_{\text{FT}}$ disparities. These few-shot analysis datasets are formed by taking a series of random subsets (with replacement) of our overarching dataset $\boldsymbol{X}_{\text{PT}}$ corresponding to 14 different sampling rates ranging on a logarithmic scale from 0.03% to 100%. This process is shown in Figure 2. Note that as all samples are taken randomly, our benchmark currently does not support PT/ FT in a setting with domain shift. This is obviously also an important challenge as well, that we hope to explore in future work.

5.7 Benchmark Utilization Protocol

First, the encoder must be pre-trained on the MIMIC-III and eICU cohorts. For a self-supervised PT system, hyperparameter tuning and pre-training are performed once (per random train/test split). For a weakly-supervised PT system, a separate round of pre-training must be performed per task t such that the pre-trained encoder $\mathcal{E}_t$ is trained to optimize task performance on all tasks *except* for task t, which is reserved for fine-tuning evaluation. To ease the hyperparameter tuning burden for weakly-supervised systems, it also is possible to perform a single round of PT hyperparameter tuning using the entire set of tasks, risking a small amount of task leakage at the gain of a significant reduction in compute cost (though of course actual PT must still be repeated for each model $\mathcal{E}_t$ with the proper subdivision of tasks after hyperparameter tuning is complete).

Next, fine-tuning is performed on task-specific models across all cohorts, sub-sampled datasets, and tasks. To assess the self-supervised system, all fine-tuning models will transfer from the same pre-trained source model, whereas for the weakly-supervised system, following LOTO, each encoder $\mathcal{E}_t$ must be fine-tuned on only task t to ensure no overlap between PT and FT tasks. This fine-tuning procedure is repeated across both the FTF and FTD transfer settings defined in Section 5.5.

Finally, fully-supervised, single-task (ST) models of the same base architecture are hyperparameter tuned and trained from scratch for each task to provide a baseline score.

The output of this process will yield one score per task, cohort, sub-sampled-dataset, PT algorithm, and FT transfer regime. This process is then repeated across the random splits within the benchmark to assess variance. Based on these results, the user can judge if either of these PT algorithms offer robust benefits across all cohorts and tasks, if one fine-tuning transfer style is preferred over another, or any number of other questions.

6 BASELINE EXPERIMENTS

6.1 Baseline-specific Data Post-processing

Our baseline models featurize the timeseries into fixed-size input windows of anywhere from 12 - 96 hours (chosen via hyperparameter search). Within these fixed *input* windows (and not taking into account any data from the prediction windows), any missing features are linearly interpolated between their previous and subsequent measurements. If a measurement is only observed on one side of the value (e.g., there are no future measurements or no previous measurements within the input window), values are carried forward or backward, respectively, and if no measurements are observed, they are imputed to the feature's mean value over the train dataset. In addition, time-since-last-measured ordinal indicators (up to 8 hours) are added to capture how long it has been at any given time-point since a specific feature was last measured. Both to simplify our shared code base, and as a form of data augmentation, all training is done across random time-points throughout the patient's stay, regardless of the specific details of the task's prescribed evaluation input window, though those relationships are, of course, respected during evaluation. For example, though ICD code prediction will only be evaluated using the first 24 hours of a patient's stay, during training we will train this model on inputs throughout the patient's stay.

6.2 Models

6.2.1 Encoder Architecture. All models in this work use a GRU model [8] as their encoder $\mathcal{E}$. Early experiments suggested this model outperformed other architectures, including a simpler, linear baseline, a convolutional neural network architecture, and a transformer model, and it is a commonly used model in the literature, so it is a reasonable choice for a baseline architecture here. Input data is projected down to a unified numerical representation, then run through a (potentially) multi-layer, bidirectional GRU (GRU parameters are determined via hyperparameter tuning) to yield a final encoder. This encoded representation is then passed through a task-specific decoder, which is either (1) a LSTM based sequential decoder for the FTS task, or (2) a simple linear layer up to the appropriate dimensionality of the task, followed by an appropriate classification activation (e.g., sigmoid or softmax) for all other tasks. For multi-label tasks, activations and losses are computed in a per-label manner and losses are then averaged.

6.2.2 Supervised, Single-task (ST) Models. We perform fully supervised, single-task (ST) training, with no PT, on each task separately, to provide baselines in comparisons with our PT/FT methods. These runs use the same GRU architecture as our other experiments, and are hyperparameter tuned separately for each downstream task.

6.2.3 Pre-training Algorithms. We profile two distinct PT systems on our benchmark: A weakly-supervised, multi-task (MT) PT model, and a self-supervised, masked-imputation (MI) model. For a visual overview of both of these methods, see Figure 3.

Weakly-supervised, Multi-task (MT) Pre-training. In multi-task (MT) PT, we use the "leave-one-task-out" method described in Section 5.4 to ensure our MT PT approach does not leak task information between FT and PT contexts. Our multi-task approach is very straightforward: all tasks in the learning ensemble (e.g., all tasks save the eventual fine-tuning target) will be jointly trained via a model whose encoder $\mathcal{E}$ is shared across all tasks but with separate decoders $\mathcal{D}_t$ per task. No loss weighting or task-alignment is used.

Masked-Imputation (MI) Pre-training. Masked-imputation (MI) PT is inspired by the successes of models such as BERT [11] in NLP. To adapt the ideas of BERT to a continuous domain with missingness, we choose at random approximately 15% of the time-points in the input window to "mask," (replace with all zeros and augment with a bit indicating masking took place). Then, the model is tasked with predicting within this masked time-point which labs/vitals were actually measured via a classification task and what their values were via a continuous regression task. At fine-tuning time, the model is no longer asked to perform masked imputation, and no masking is applied. For this PT task, we limit our GRU models to unidirectional GRUs to avoid leaking information from future time-points[2], and models are hyperparameter tuned to maximize the mean of the classification task's macro AUROC and the regression task's R^2 value.

Fine-tuning. For both PT systems, after PT is complete, the model is fine-tuned by initializing an untrained decoder layer and training the system according to the loss criteria appropriate to the type of task at hand (e.g., binary cross-entropy loss or a negative log likelihood loss depending on the task type). Tasks that are multi-label in nature are trained by averaging the losses together for all labels. As described in Section 5.5, we profile both FTF and FTD transfer styles in this baseline, and we report across all sub-sampled dataset sizes as described in Section 5.6.

6.3 Hyperparameter Tuning

Hyperparameter tuning was performed to optimize the underlying architecture via the PT task with random search, via the Bayesian Hyperopt Library [3]. No FT specific hyperparameter tuning was performed, as the majority of the details of the architecture (e.g., the GRU depth and dimensionality) are fixed by the pre-training algorithm. ST models were naturally tuned based on output task performance, as there is no PT stage for these models.

[2]This is especially important in the context of our imputation procedure, which directly encodes how long it has been since any given lab/vital was measured.

Specific model hyperparameters were chosen to maximize the appropriate score on the validation fold over a random search drawn from a customized hyperparameter distribution. For MT PT models, this search was done once across all tasks simultaneously in a MT manner, optimizing for the average AUROC across all tasks in the ensemble, then used for all PT/FT experiments. This represents a possible source of very mild task leakage, but yielded significant computational savings. For MI models, hyperparameter tuning was done once in a task-independent manner (as masked imputation is a self-supervised, rather than weakly-supervised PT method). Final hyperparameters were chosen based on the full dataset, and were not repeated at smaller training set sizes for the few-shot experiments, which may represent another possible source of bias. Additional details on the hyperparameter search can be found in Appendix Section B.

6.4 Results

In this section, we will highlight a subset of the most relevant results found in our baseline experiments. Figure 4 shows two things of interest: First, it shows a graph cataloging over what fraction of tasks a particular PT/FT model type offers best performance as a function of dataset size. This allows us to see quickly, for example, that for a wide variety of dataset sizes, MT FTF offers significnat improvements over other strategies on a significant portion of the tested tasks. To show these relationships in more detail, Figure 4 also shows more complete results for 3 of our 10 tasks across both cohorts and all dataset fractions, comparing specifically both varieties of the FTF models and the ST model. In addition, the results corresponding to the 1%, 10%, and full-data scales for both cohorts are shown in Table 3. In this view, we see that at the 1% setting, the Multi-task (MT) FTF setting performs best in 7/10 settings, whereas ST never offers best-in-class performance. At the 10% setting, MT FTF excels 6/10 times, and ST performs best in only one task. Finally, at the 100% (e.g., full) data scale, ST always performs best. This demonstrates a strong trend between the performance benefit offered by MT-FTF PT and the severity of the N_{FT} v. N_{PT} imbalance. Full results can be found in the Appendix, in Figures 6,7 for MIMIC-III and 8,9 for eICU.

6.5 Discussion

Pre-training does not offer benefits at full data scale. Table 3 shows that PT does not offer any gains over traditional supervised learning at full MIMIC or eICU dataset scales. In some cases, PT actually harms the final results. This is not necessarily surprising; while PT can help compensate for too little data and provide (indirect) access to additional data in the case of larger N_{FT} / N_{PT} discrepancies, when $\boldsymbol{X}_{\text{FT}} = \boldsymbol{X}_{\text{PT}}$ the risks that PT simply serves as a distraction from the (comparatively direct) supervised learning signal is large.

Pre-training can offer significant benefits in few-shot settings. Our benchmark reveals that PT in few-shot settings is helpful. Figure 4 shows that across a significant fraction of the dataset fractions for both the MIMIC-III and eICU cohorts, MT FTF offers significant benefits over other approaches. In Table 3, we see more concretely that in the 1% FT dataset setting, some form of PT/FT offers best in class performance across all tasks in both cohorts except for LOS on the eICU cohort, with performance improvements ranging as

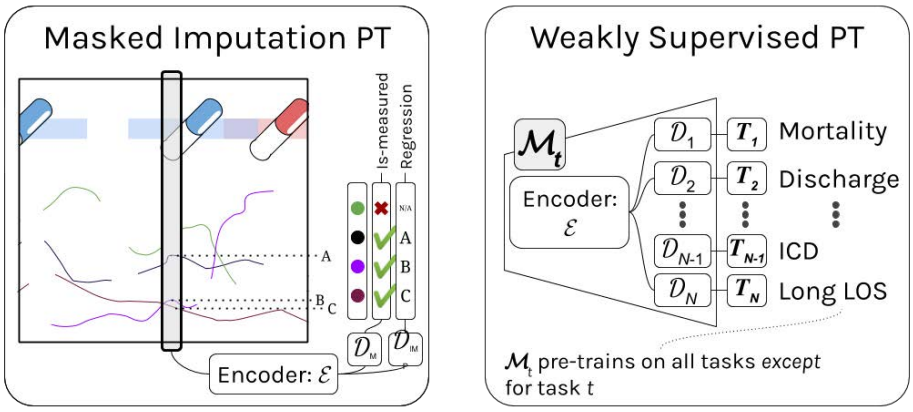

Figure 3: We profile both a self-supervised, masked-imputation PT system and a weakly-supervised multi-task PT system.

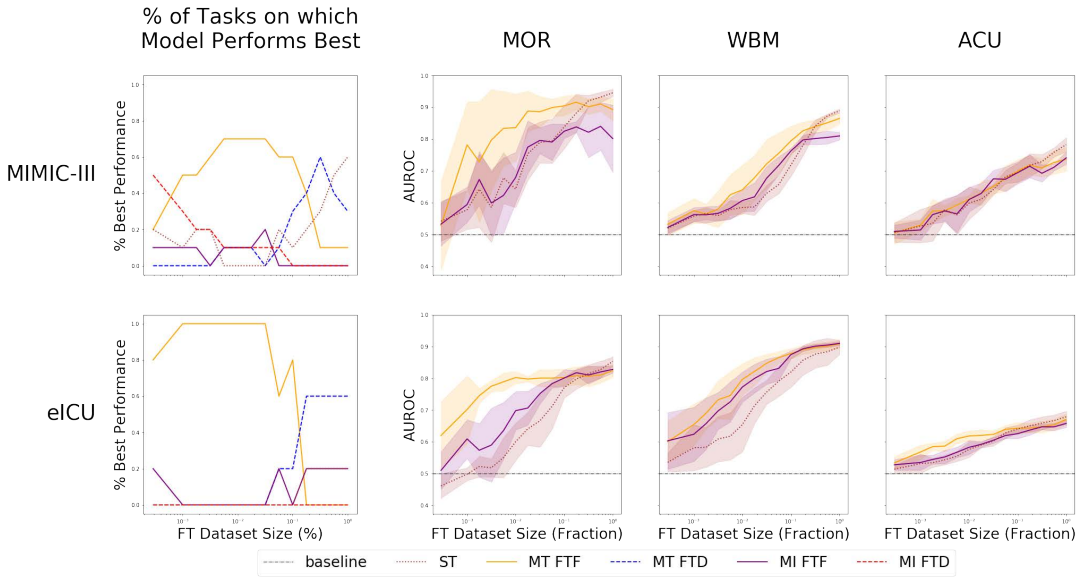

Figure 4: *(left column)* For what % of tasks (y-axis) does a given PT/FT regime (linestyle) perform better than all other PT/FT regimes, as a function of dataset fraction (x-axis). *(right 3 columns)* Performance in macro-averaged AUROC (y-axis) of various PT/FT models (linestyle) across various FT dataset sub-sampling rates (x-axis), over 3 sample FT tasks (subplots).

high as an AUROC improvement of 0.2/0.24 for mortality prediction in MIMIC-III/eICU. In the 10% dataset size setting, some form of PT/FT still offers best-in-class performance on all tasks save LOS for the eICU cohort and ICD for the MIMIC cohort. The margins of improvement are no longer as high, but offer consistent gains across a variety of other tasks such as with CMO, DNR, MOR, WBM, and FTS all offering AUROC improvements of up to 0.1 on MIMIC-III (improvements are much smaller on eICU at this threshold). These findings provide evidence to affirm that EHR PT/FT strategies could enable more effective modelling even given only very small task-specific datasets, thus potentially offering a vehicle to help train models for novel or rare diseases.

We also observe that the tasks which tend to show the largest improvements with PT (MOR, WBM, CMO, DNR, and FTS) are all *rolling tasks* with substantial class imbalance. Across both cohorts these tasks report majority class accuracies greater than or equal to 88% (with most $\geq$ 95%). No other tasks in our benchmark meet these criteria, suggesting there may be stronger benefits from PT (in particular, from MT PT) on rolling, imbalanced tasks.

Weakly-supervised pre-training out-performs self-supervised pre-training. In general, MT PT is superior to MI, even at small data scales, suggesting that simple adaption of the masked language modeling idea is not sufficient for the clinical domain. Despite this, in the few-shot domain, MI FTF training still does outperform traditional ST modelling, just not by as much as MT FTF does. For

Table 3: GRU Results (AUROC) subdivided among different PT regimes, under both the full-data fine-tuning setting and few-shot (1%, 10%) settings, on both the MIMIC-III and eICU cohorts. Bolded results indicate top performing result per each task/evaluation setting.

Dataset Size	Task	MIMIC-III					eICU				
		MI FTD	MI FTF	MT FTD	MT FTF	ST	MI FTD	MI FTF	MT FTD	MT FTF	ST
Few-shot (1%)	MOR	0.55 ± 0.08	0.68 ± 0.08	0.57 ± 0.17	**0.84 ± 0.11**	0.64 ± 0.06	0.57 ± 0.04	0.7 ± 0.06	0.52 ± 0.14	**0.8 ± 0.02**	0.6 ± 0.06
	CMO	0.6 ± 0.09	0.65 ± 0.05	0.52 ± 0.17	**0.76 ± 0.18**	0.58 ± 0.11					
	DNR	0.59 ± 0.04	0.57 ± 0.06	0.55 ± 0.08	**0.63 ± 0.1**	0.55 ± 0.07					
	ICD	0.49 ± 0.03	**0.56 ± 0.03**	0.52 ± 0.02	0.56 ± 0.03	0.56 ± 0.02					
	LOS	0.51 ± 0.09	0.62 ± 0.03	0.6 ± 0.08	**0.67 ± 0.03**	0.58 ± 0.02	0.51 ± 0.02	0.55 ± 0.02	0.53 ± 0.06	**0.59 ± 0.02**	0.54 ± 0.04
	REA	**0.54 ± 0.03**	0.51 ± 0.02	0.5 ± 0.04	0.54 ± 0.03	0.51 ± 0.03					
	DIS	0.52 ± 0.02	0.57 ± 0.05	0.54 ± 0.03	**0.58 ± 0.03**	0.54 ± 0.01	0.51 ± 0.01	0.56 ± 0.01	0.53 ± 0.03	**0.58 ± 0.01**	0.56 ± 0.01
	ACU	0.51 ± 0.03	0.61 ± 0.04	0.56 ± 0.05	**0.61 ± 0.01**	0.6 ± 0.05	0.51 ± 0.01	0.58 ± 0.02	0.55 ± 0.05	**0.62 ± 0.02**	0.58 ± 0.02
	WBM	0.53 ± 0.03	0.61 ± 0.03	0.53 ± 0.03	**0.64 ± 0.08**	0.58 ± 0.02	0.59 ± 0.04	0.77 ± 0.05	0.53 ± 0.02	**0.8 ± 0.05**	0.65 ± 0.09
	FTS	0.61 ± 0.05	0.61 ± 0.04	**0.62 ± 0.05**	0.62 ± 0.06	0.6 ± 0.04					
Few-shot (10%)	MOR	0.61 ± 0.12	0.82 ± 0.02	0.82 ± 0.17	**0.9 ± 0.03**	0.84 ± 0.07	0.62 ± 0.09	0.8 ± 0.03	0.77 ± 0.07	**0.8 ± 0.03**	0.77 ± 0.03
	CMO	0.62 ± 0.08	0.74 ± 0.03	0.76 ± 0.15	**0.85 ± 0.06**	0.77 ± 0.09					
	DNR	0.6 ± 0.04	0.75 ± 0.03	0.76 ± 0.09	**0.82 ± 0.03**	0.71 ± 0.1					
	ICD	0.53 ± 0.05	0.65 ± 0.01	0.6 ± 0.02	0.64 ± 0.01	**0.67 ± 0.03**					
	LOS	0.55 ± 0.09	0.6 ± 0.02	**0.69 ± 0.02**	0.65 ± 0.03	0.66 ± 0.02	0.52 ± 0.03	0.6 ± 0.03	0.61 ± 0.02	**0.61 ± 0.02**	0.61 ± 0.0
	REA	0.54 ± 0.04	0.53 ± 0.03	0.54 ± 0.05	**0.57 ± 0.04**	0.57 ± 0.03					
	DIS	0.56 ± 0.03	0.63 ± 0.02	**0.67 ± 0.04**	0.66 ± 0.04	0.64 ± 0.03	0.53 ± 0.02	0.6 ± 0.01	0.61 ± 0.04	**0.62 ± 0.01**	0.61 ± 0.01
	ACU	0.58 ± 0.06	0.69 ± 0.03	0.68 ± 0.04	**0.7 ± 0.04**	0.69 ± 0.03	0.56 ± 0.03	0.63 ± 0.02	**0.67 ± 0.03**	0.64 ± 0.02	0.64 ± 0.01
	WBM	0.57 ± 0.04	0.76 ± 0.01	0.65 ± 0.06	**0.79 ± 0.05**	0.72 ± 0.04	0.72 ± 0.03	0.87 ± 0.01	0.66 ± 0.05	**0.88 ± 0.02**	0.82 ± 0.05
	FTS	0.74 ± 0.09	0.77 ± 0.08	**0.82 ± 0.05**	0.81 ± 0.05	0.73 ± 0.06					
Full Data	MOR	0.74 ± 0.09	0.8 ± 0.11	0.94 ± 0.01	0.89 ± 0.03	**0.95 ± 0.01**	0.72 ± 0.07	0.83 ± 0.01	**0.86 ± 0.01**	0.82 ± 0.02	0.85 ± 0.01
	CMO	0.72 ± 0.06	0.77 ± 0.05	**0.92 ± 0.01**	0.85 ± 0.04	0.91 ± 0.02					
	DNR	0.72 ± 0.09	0.74 ± 0.01	**0.87 ± 0.02**	0.78 ± 0.06	0.87 ± 0.02					
	ICD	0.65 ± 0.05	0.67 ± 0.01	0.67 ± 0.01	0.68 ± 0.03	**0.74 ± 0.01**					
	LOS	0.61 ± 0.04	0.58 ± 0.03	0.69 ± 0.02	0.64 ± 0.04	**0.71 ± 0.01**	0.54 ± 0.04	0.64 ± 0.02	0.62 ± 0.02	0.63 ± 0.05	**0.65 ± 0.0**
	REA	0.57 ± 0.04	0.56 ± 0.02	0.6 ± 0.04	0.57 ± 0.02	**0.61 ± 0.02**					
	DIS	0.64 ± 0.05	0.68 ± 0.04	**0.75 ± 0.02**	0.72 ± 0.02	0.74 ± 0.03	0.58 ± 0.03	0.64 ± 0.01	**0.66 ± 0.02**	0.64 ± 0.01	0.65 ± 0.0
	ACU	0.7 ± 0.05	0.74 ± 0.02	0.75 ± 0.04	0.74 ± 0.04	**0.78 ± 0.02**	0.62 ± 0.02	0.66 ± 0.01	**0.7 ± 0.02**	0.67 ± 0.02	0.68 ± 0.02
	WBM	0.68 ± 0.07	0.81 ± 0.01	0.77 ± 0.02	0.86 ± 0.02	**0.89 ± 0.01**	0.79 ± 0.04	**0.91 ± 0.01**	0.79 ± 0.02	0.91 ± 0.01	0.9 ± 0.02
	FTS	0.86 ± 0.02	0.89 ± 0.01	0.89 ± 0.01	**0.9 ± 0.0**	0.9 ± 0.01					

example, at 1% on MIMIC-III, MI FTF outperforms ST in all but one case, and at 10% it does in all but four cases (though on eICU the situation is murkier).

FTF is in general preferred over FTD. Consistent across both MI and MT PT is that FTF models are preferred to FTD models. This is true across datasets and sub-sampling rate, and suggests that despite the increased risk of overfitting offered by fine-tuning the encoder as well as the decoder, this strategy may be integral to obtaining strong PT/FT results in this modality.

7 CONCLUSION

In this work, we present a novel benchmark for PT systems over EHR time-series data. We define a suite of FT task targets, including several novel tasks, over both MIMIC-III and eICU, and establish evaluation procedures for examining a PT system's performance both across various FT dataset sizes. We then establish three baseline systems on this benchmark, including a traditional, non-pre-trained single-task baseline, a weakly-supervised multi-task PT baseline, and a fully self-supervised masked-imputation based PT baseline. These baselines demonstrate that weakly-supervised, multi-task PT can offer substantial improvements in few-shot contexts for tasks suffering from significant class imbalance. In addition, they suggest important findings on the viability of different styles of PT and FT; in particular that masked-imputation based PT currently is not competitive with multi-task PT, and that fine-tuning both model encoders and decoders is necessary for ensuring strong FT performance.

While significant future work remains, including assessing additional PT systems on this benchmark as well as augmenting this benchmark to assess the impact PT has on fine-tuning under domain shift such as pre-training on one hospital and fine-tuning on another, or subpopulation shift in fairness applications, we believe that this benchmark can be an invaluable tool for the ML4H community. By standardizing PT/FT training and evaluation procedures, including few-shot evaluation analyses and the inclusion of a sufficiently diverse set of tasks to assess the utility of PT schemes in general, rather than merely on a isolated, highly specific subset of tasks, this benchmark offers the possibility of greatly increasing the efficiency of PT research on EHR data. This benchmark will help enable iterative analysis and development of PT strategies in this domain and lead to the release of PT encoders that enable easy specialization and deployment in clinical settings.

ACKNOWLEDGEMENTS

This work was funded in part by New Frontiers in Research Fund - NFRFE-2019-00844 and a CIFAR AI Chair. Resources used in preparing this research were provided, in part, by the Province of Ontario, the Government of Canada through CIFAR, and companies sponsoring the Vector Institute www.vectorinstitute.ai/partners. In addition, BN is supported in part by CIHR, AG is supported by Varma Chair, CIHR, NSERC and CIFAR, and MBAM is supported in part by NIH grant LM013337 and by a collaborative research agreement with IBM.

REFERENCES

[1] Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. 2019. Publicly Available Clinical BERT Embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*. Association for Computational Linguistics, Minneapolis, Minnesota, USA,

72–78. https://doi.org/10.18653/v1/W19-1909

[2] Aparna Balagopalan, Jekaterina Novikova, Matthew B A Mcdermott, Bret Nestor, Tristan Naumann, and Marzyeh Ghassemi. 2020. Cross-Language Aphasia Detection using Optimal Transport Domain Adaptation. In *Proceedings of the Machine Learning for Health NeurIPS Workshop (Proceedings of Machine Learning Research)*, Adrian V. Dalca, Matthew B.A. McDermott, Emily Alsentzer, Samuel G. Finlayson, Michael Oberst, Fabian Falck, and Brett Beaulieu-Jones (Eds.), Vol. 116. PMLR, 202–219. http://proceedings.mlr.press/v116/balagopalan20a.html

[3] James Bergstra, Daniel Yamins, and David Cox. 2013. Making a Science of Model Search: Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures. In *Proceedings of the 30th International Conference on Machine Learning (Proceedings of Machine Learning Research)*, Sanjoy Dasgupta and David McAllester (Eds.), Vol. 28. PMLR, Atlanta, Georgia, USA, 115–123. http://proceedings.mlr.press/v28/bergstra13.html

[4] Dimitris Bertsimas, Jean Pauphilet, Jennifer Stevens, and Manu Tandon. 2020. Predicting inpatient flow at a major hospital using interpretable analytics. *medRxiv* (2020).

[5] Chun-Hao Chang, Mingjie Mai, and Anna Goldenberg. 2019. Dynamic Measurement Scheduling for Event Forecasting using Deep RL. In *Proceedings of the 36th International Conference on Machine Learning (Proceedings of Machine Learning Research)*, Kamalika Chaudhuri and Ruslan Salakhutdinov (Eds.), Vol. 97. PMLR, Long Beach, California, USA, 951–960. http://proceedings.mlr.press/v97/chang19a.html

[6] Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. 2018. Recurrent Neural Networks for Multivariate Time Series with Missing Values. *Scientific Reports* 8, 1 (17 Apr 2018), 6085. https://doi.org/10.1038/s41598-018-24271-9

[7] Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. 2018. Recurrent Neural Networks for Multivariate Time Series with Missing Values. *Scientific Reports* 8, 1 (April 2018), 6085. https://doi.org/10.1038/s41598-018-24271-9

[8] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. 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)*. Association for Computational Linguistics, Doha, Qatar, 1724–1734. https://doi.org/10.3115/v1/D14-1179

[9] Edward Choi, Cao Xiao, Walter Stewart, and Jimeng Sun. 2018. MiME: Multilevel Medical Embedding of Electronic Health Records for Predictive Healthcare. In *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Eds.), Vol. 31. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2018/file/934b535800b1cba8f96a5d72f72f1611-Paper.pdf

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255. https://doi.org/10.1109/CVPR.2009.5206848

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, Minneapolis, Minnesota, 4171–4186. https://doi.org/10.18653/v1/N19-1423

[12] Daisy Yi Ding, Chloé Simpson, Stephen Pfohl, Dave C Kale, Kenneth Jung, and Nigam H Shah. 2019. The Effectiveness of Multitask Learning for Phenotyping with Electronic Health Records Data.. In *Biocomputing 2019*. World Scientific, 18–29. https://doi.org/10.1142/9789813279827_0003

[13] Vincent Fortuin, Matthias Hüser, Francesco Locatello, Heiko Strathmann, and Gunnar Rätsch. 2018. SOM-VAE: Interpretable Discrete Representation Learning on Time Series. *arXiv:1806.02199 [cs, stat]* (June 2018). http://arxiv.org/abs/1806.02199 arXiv: 1806.02199.

[14] Marzyeh Ghassemi, Tristan Naumann, Peter Schulam, Andrew L Beam, Irene Y Chen, and Rajesh Ranganath. 2019. Practical guidance on artificial intelligence for health-care data. *The Lancet Digital Health* 1, 4 (2019), e157–e159.

[15] Hrayr Harutyunyan, Hrant Khachatrian, David C Kale, Greg Ver Steeg, and Aram Galstyan. 2019. Multitask learning and benchmarking with clinical time series data. *Scientific data* 6, 1 (2019), 1–18.

[16] George Hripcsak, David J Albers, and Adler Perotte. 2015. Parameterizing time in electronic health record studies. *Journal of the American Medical Informatics Association* 22, 4 (02 2015), 794–804. https://doi.org/10.1093/jamia/ocu051 arXiv:https://academic.oup.com/jamia/article-pdf/22/4/794/25894265/ocu051.pdf

[17] Alistair E. W. Johnson, Tom J. Pollard, Lu Shen, Li-wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. 2016. MIMIC-III, a freely accessible critical care database. *Scientific Data* 3, 1 (May 2016), 1–9. https://doi.org/10.1038/sdata.2016.35

[18] Alex M Lamb, Anirudh Goyal Alias Parth Goyal, Ying Zhang, Saizheng Zhang, Aaron C Courville, and Yoshua Bengio. 2016. Professor forcing: A new algorithm for training recurrent networks. In *Advances In Neural Information Processing Systems*, Vol. 29. 4601–4609. https://proceedings.neurips.cc/paper/2016/file/16026d60ff9b54410b3435b403afd226-Paper.pdf

[19] Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Xiaodong Fan, Ruofei Zhang, Rahul Agrawal, Edward Cui, Sining Wei, Taroon Bharti, Ying Qiao, Jiun-Hung Chen, Winnie Wu, Shuguang Liu, Fan Yang, Daniel Campos, Rangan Majumder, and Ming Zhou. 2020. XGLUE: A New Benchmark Datasetfor Cross-lingual Pretraining, Understanding and Generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 6008–6018. https://doi.org/10.18653/v1/2020.emnlp-main.484

[20] Sharon L Lojun, Christina J Sauper, Mitchell Medow, William J Long, Roger G Mark, and Regina Barzilay. 2010. Investigating resuscitation code assignment in the intensive care unit using structured and unstructured data. In *AMIA Annual Symposium Proceedings*, Vol. 2010. American Medical Informatics Association, 467.

[21] Yuan Luo, Peter Szolovits, Anand S Dighe, and Jason M Baron. 2016. Using machine learning to predict laboratory test results. *American journal of clinical pathology* 145, 6 (2016), 778–788.

[22] Aya A Mitani and Sebastien Haneuse. 2020. Small Data Challenges of Studying Rare Diseases. *JAMA Network Open* 3, 3 (2020), e201965–e201965.

[23] Bret Nestor, Matthew B. A. McDermott, Willie Boag, Gabriela Berner, Tristan Naumann, Michael C. Hughes, Anna Goldenberg, and Marzyeh Ghassemi. 2019. Feature Robustness in Non-stationary Health Records: Caveats to Deployable Model Performance in Common Clinical Machine Learning Tasks. In *Proceedings of the 4th Machine Learning for Healthcare Conference (Proceedings of Machine Learning Research)*, Finale Doshi-Velez, Jim Fackler, Ken Jung, David Kale, Rajesh Ranganath, Byron Wallace, and Jenna Wiens (Eds.), Vol. 106. PMLR, Ann Arbor, Michigan, 381–405. http://proceedings.mlr.press/v106/nestor19a.html

[24] Xuefeng Peng, Yi Ding, David Wihl, Omer Gottesman, Matthieu Komorowski, Li-wei H Lehman, Andrew Ross, Aldo Faisal, and Finale Doshi-Velez. 2018. Improving sepsis treatment strategies by combining deep and kernel-based reinforcement learning. In *AMIA Annual Symposium Proceedings*, Vol. 2018. American Medical Informatics Association, 887.

[25] Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. 2227–2237.

[26] Tom J Pollard, Alistair EW Johnson, Jesse D Raffa, Leo A Celi, Roger G Mark, and Omar Badawi. 2018. The eICU Collaborative Research Database, a freely available multi-center database for critical care research. *Scientific data* 5 (2018), 180178.

[27] Alvin Rajkomar, Eyal Oren, Kai Chen, Andrew M Dai, Nissan Hajaj, Michaela Hardt, Peter J Liu, Xiaobing Liu, Jake Marcus, Mimi Sun, et al. 2018. Scalable and accurate deep learning with electronic health records. *NPJ Digital Medicine* 1, 1 (2018), 18.

[28] Roshan Rao, Nicholas Bhattacharya, Neil Thomas, Yan Duan, Peter Chen, John Canny, Pieter Abbeel, and Yun Song. 2019. Evaluating protein transfer learning with tape. In *Advances in Neural Information Processing Systems*. 9689–9701.

[29] Junyuan Shang, Tengfei Ma, Cao Xiao, and Jimeng Sun. 2019. Pre-training of Graph Augmented Transformers for Medication Recommendation. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization, 5953–5959. https://doi.org/10.24963/ijcai.2019/825

[30] Yuqi Si and Kirk Roberts. 2019. Deep Patient Representation of Clinical Notes via Multi-Task Learning for Mortality Prediction. *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science* 2019 (06 May 2019), 779–788. https://pubmed.ncbi.nlm.nih.gov/31259035 31259035[pmid].

[31] Yuqi Si, Jingqi Wang, Hua Xu, and Kirk Roberts. 2019. Enhancing clinical concept extraction with contextual embeddings. *Journal of the American Medical Informatics Association* 26, 11 (2019), 1297–1304.

[32] Vergil N Slee. 1978. The International classification of diseases: ninth revision (ICD-9). *Annals of internal medicine* 88, 3 (1978), 424–426.

[33] Ethan Steinberg, Ken Jung, Jason A. Fries, Conor K. Corbin, Stephen R. Pfohl, and Nigam H. Shah. 2021. Language models are an effective representation learning technique for electronic health record data. *Journal of Biomedical Informatics* 113 (2021), 103637. https://doi.org/10.1016/j.jbi.2020.103637

[34] Harini Suresh, Nathan Hunt, Alistair Johnson, Leo Anthony Celi, Peter Szolovits, and Marzyeh Ghassemi. 2017. Clinical Intervention Prediction and Understanding using Deep Networks. In *Proceedings of the 2nd Machine Learning for Healthcare Conference (Proceedings of Machine Learning Research)*, Finale Doshi-Velez, Jim Fackler, David Kale, Rajesh Ranganath, Byron Wallace, and Jenna Wiens (Eds.), Vol. 68. PMLR, Boston, Massachusetts, 322–337. http://proceedings.mlr.press/v68/suresh17a.html

[35] Shirly Wang, Matthew B. A. McDermott, Geeticka Chauhan, Marzyeh Ghassemi, Michael C. Hughes, and Tristan Naumann. 2020. MIMIC-Extract: A Data Extraction, Preprocessing, and Representation Pipeline for MIMIC-III. In *Proceedings of the ACM Conference on Health, Inference, and Learning (CHIL 2020)*. Association for Computing Machinery, New York, NY, USA, 222–235. https://doi.org/10.1145/3368555.3384469

[36] Xiang Wang, Fei Wang, and Jianying Hu. 2014. A multi-task learning framework for joint disease risk prediction and comorbidity discovery. In *2014 22nd International Conference on Pattern Recognition*. IEEE, 220–225.

[37] Mike Wu, Marzyeh Ghassemi, Mengling Feng, Leo A Celi, Peter Szolovits, and Finale Doshi-Velez. 2016. Understanding vasopressor intervention and weaning: risk prediction in a public heterogeneous clinical time series database. *Journal of the American Medical Informatics Association* 24, 3 (10 2016), 488–495. https://doi.org/10.1093/jamia/ocw138 arXiv:https://academic.oup.com/jamia/article-pdf/24/3/488/25421894/ocw138.pdf

[38] Yuan Xue, Nan Du, Anne Mottram, Martin Seneviratne, and Andrew M Dai. 2020. Learning to Select Best Forecast Tasks for Clinical Outcome Prediction. *Advances in Neural Information Processing Systems* 33 (2020).

[39] Jinsung Yoon, Yao Zhang, James Jordon, and Mihaela van der Schaar. 2020. VIME: Extending the Success of Self-and Semi-supervised Learning to Tabular Domain. *Advances in Neural Information Processing Systems* 33 (2020).

[40] Henghui Zhu, Ioannis Ch Paschalidis, and Amir Tahmasebi. 2018. Clinical concept extraction with contextual word embedding. *arXiv preprint arXiv:1810.10566* (2018).

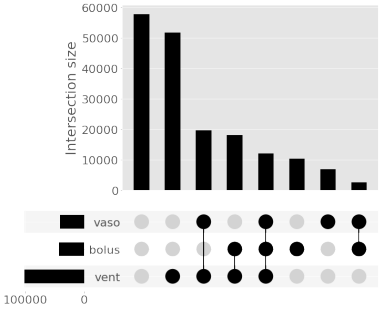

Figure 5: A sample Upset plot showing the frequency of relative combinations of our three treatment types: Vasopressors (vaso), Ventilation (vent), and Fluid Bolus administration (bolus) on the MIMIC-III cohort.

A ADDITIONAL DATA/TASK INFORMATION

A.1 Additional Dataset Details

MIMIC-III Cohort Treatment Data. In the MIMIC-III cohort, we incorporate as inputs treatments including adenosine, colloid bolus, crystalloid bolus, dobutamine, dopamine, epinephrine, isuprel, milrinone, nivdurations, norepinephrine, phenylephrine, vaso (other vasopressor application), vasopressin, and vent (ventilation).

A.2 Task Details and Statistics

Imminent Discharge: DIS. The below two tables (Table 8, 9) capture the overall prevalence of all DIS classes observed across both cohorts and all labels.

Final Acuity: ACU. The below two tables (Table 11, 10) capture the overall prevalence of all ACU classes observed across both cohorts.

Next Timepoint: WBM. Table 6 shows the majority class accuracy for all labs & vitals used in this work for the WBM task.

Future Treatment Sequence: FTS. Figure 5 shows which combinations of treatments are most commonly observed over MIMIC-III.

B HYPERPARAMETER SEARCH ANALYSIS

B.1 Search Space

For our hyperparameter search procedure, we searched over a wide variety of parameters, including number of epochs, batch size, learning rate, learning rate decay paradigms, L2 regularization penalty, dropout, the maximum length of a patients record included, the size, number, and configuration of various hidden layers, pooling and fully connected stack parameters, and various other model-specific options. All search distributions are shown in Table 12. Various numbers of samples were run for each experiment. Universally, at least 100 random samples per search were run. Runs that had more than 100 samples were almost universally single-task runs, not PT/FT runs.

C FINAL RESULTS

In Figures 6,9 we show the absolute performance of all models on the MIMIC-III, eICU cohorts, and Figures 7,9 show the relative performance of all model types as compared to a ST baseline for the MIMIC-III, eICU cohorts. We note that the eICU results for the ST LOS task appear anomalous—while all runs reported here have gone through internal validation, this oddity warrants further investigation in future work.

D SAMPLES COMPLETED

Below are the full experiment counts for all results reported in this work. Note that an extra rotation was also run on the MIMIC-III MT results. This was unintentional, but as all rotations here are separate random train/test splits, we chose to retain the result as it simply improves the quality of our estimates of variance and should add no bias to the results or comparisons.

E NEGATIVE TRANSFER ANALYSES

We can also leverage these experiments to perform a robust analysis of negative transfer within EHR timeseries multi-task learning. In particular, by comparing our multi-task pre-training results, which are trained over all but one task (as one task is withheld for use in fine-tuning) vs. our hyperparameter search results as well as our single-task results vs. our full MT hyperparameter search results. First, at a lgobal scale, we see in Table 15 that there is no general apparent preference between ST and MT runs. This suggests that we see no evidence of either global positive or negative transfer.

Examining the transfer utility on a local, per-task level, we can examine how the performance on a particular task is affected by removing a single other task from the full multi-task ensemble or, in a transpose fashion, how including a given task in the learning ensemlbe effects the performance of other downstream tasks. These results are shown visually in Figure 10. There, we see that, like our global finding, there is minimal evidence of any universal positive or negative transfer; instead, we see examples of both positive and negative transfer, which, in aggregate, offer no consistent effect. These results suggest that negative transfer is quite likely in a generic MT setting without careful consideration.

Table 4: Macro-averaged (train-set) majority class accuracy aggregated across all folds / labels for all tasks.

Task	MIMIC-III			eICU		
	Train	Tuning	Held-out Test	Train	Tuning	Held-out Test
MOR	96.5 ± 14.14%	96.5 ± 14.23%	96.3 ± 14.67%	97.0 ± 13.67%	97.1 ± 13.58%	97.1 ± 13.60%
CMO	98.8 ± 7.83%	98.8 ± 7.84%	98.7 ± 8.03%			
DNR	96.3 ± 17.12%	96.3 ± 17.13%	96.4 ± 17.13%			
WBM	92.1 ± 9.18%	92.2 ± 9.19%	92.1 ± 9.21%	88.0 ± 19.61%	88.0 ± 19.63%	88.0 ± 19.65%
DIS	42.6 ± 25.65%	42.6 ± 25.53%	42.6 ± 25.71%	44.2 ± 31.65%	44.1 ± 31.72%	44.1 ± 31.69%
ICD	67.0 ± 18.07%	70.2 ± 18.12%	70.2 ± 18.19%			
LOS	52.9 ± 0.11%	53.0 ± 1.26%	52.8 ± 1.02%	66.6 ± 0.04%	66.5 ± 0.26%	66.6 ± 0.32%
REA	95.1 ± 0.09%	95.1 ± 0.62%	95.0 ± 0.61%			
ACU	25.3 ± 0.24%	25.2 ± 1.34%	25.3 ± 1.15%	59.7 ± 0.10%	59.2 ± 0.39%	59.4 ± 0.64%
FTS	97.4 ± 2.77%	97.5 ± 2.91%	97.4 ± 2.96%	97.0 ± 3.75%	97.0 ± 3.75%	97.0 ± 3.75%

Table 5: Per-label majority class accuracies for all tasks aside from WBM and LOS, which are shown separately.

Task	Label	Majority Class	MIMIC-III			eICU		
			Train	Tuning	Held-out Test	Train	Tuning	Held-out Test
MOR	24H	0	97.6 ± 9.91%	97.6 ± 10.07%	97.5 ± 10.53%	97.9 ± 10.20%	97.9 ± 10.09%	97.9 ± 10.06%
	48H	0	95.4 ± 17.36%	95.4 ± 17.42%	95.2 ± 17.88%	96.2 ± 16.41%	96.2 ± 16.34%	96.2 ± 16.39%
CMO	24H	0	99.1 ± 5.55%	99.1 ± 5.58%	99.1 ± 5.77%			
	48H	0	98.5 ± 9.59%	98.4 ± 9.58%	98.4 ± 9.77%			
DNR	24H	0	96.6 ± 16.16%	96.6 ± 16.18%	96.6 ± 16.22%			
	48H	0	96.1 ± 18.03%	96.1 ± 18.03%	96.1 ± 18.00%			
DIS	24H	No Discharge	57.7 ± 24.68%	57.7 ± 24.57%	57.7 ± 24.79%	48.2 ± 26.67%	48.2 ± 26.73%	48.2 ± 26.71%
	48H	Home				40.2 ± 35.95%	39.9 ± 36.03%	40.0 ± 35.98%
		No Discharge	27.6 ± 26.58%	27.5 ± 26.45%	27.6 ± 26.60%			
LOS		0	52.9 ± 0.11%	53.0 ± 1.26%	52.8 ± 1.02%	66.6 ± 0.04%	66.5 ± 0.26%	66.6 ± 0.32%
REA		0	95.1 ± 0.09%	95.1 ± 0.62%	95.0 ± 0.61%			
ACU		Home				59.7 ± 0.10%	59.2 ± 0.39%	59.4 ± 0.64%
		Home Health Care	25.3 ± 0.24%	25.2 ± 1.34%	25.3 ± 1.15%			
FTS		0	97.4 ± 2.77%	97.5 ± 2.91%	97.4 ± 2.96%	97.0 ± 3.75%	97.0 ± 3.75%	97.0 ± 3.75%

Table 6: Per-label majority class accuracies for the WBM task

Task	Label	Majority Class	MIMIC-III			eICU		
			Train	Tuning	Held-out Test	Train	Tuning	Held-out Test
WBM	Anion Gap	0	91.4 ± 4.95%	91.5 ± 4.87%	91.5 ± 4.96%			
	Bedside Glucose	0				86.0 ± 16.68%	85.9 ± 16.75%	85.9 ± 16.77%
	Bicarbonate	0	91.1 ± 4.88%	91.1 ± 4.81%	91.1 ± 4.90%			
	Blood Urea Nitrogen	0	91.0 ± 4.87%	91.0 ± 4.78%	91.0 ± 4.92%			
	Bun	0				94.8 ± 3.29%	94.8 ± 3.29%	94.8 ± 3.32%
	Calcium	0	92.8 ± 5.05%	92.8 ± 4.94%	92.8 ± 5.05%			
	Calcium Ionized	0	95.2 ± 7.01%	95.3 ± 6.87%	95.2 ± 7.00%			
	Cardiac Index	0	96.7 ± 10.73%	96.6 ± 11.02%	96.7 ± 10.84%			
	Cardiac Output Thermodilution	0	97.2 ± 9.92%	97.1 ± 10.23%	97.2 ± 9.99%			
	Central Venous Pressure	0	82.2 ± 26.70%	82.1 ± 26.67%	82.0 ± 26.75%			
	Chloride	0	90.3 ± 5.68%	90.4 ± 5.55%	90.3 ± 5.66%			
	Co2	0	96.5 ± 3.52%	96.4 ± 3.54%	96.4 ± 3.53%			
	Co2 (Etco2, Pco2, Etc.)	0	92.3 ± 9.05%	92.4 ± 8.89%	92.3 ± 8.97%			
	Creatinine	0	90.9 ± 4.91%	91.0 ± 4.81%	91.0 ± 4.95%	94.8 ± 3.30%	94.8 ± 3.30%	94.8 ± 3.33%
	Diastolic Blood Pressure	1	88.0 ± 12.70%	88.1 ± 12.61%	88.0 ± 12.77%			
	Fraction Inspired Oxygen	0	95.9 ± 8.12%	96.0 ± 8.12%	96.0 ± 8.15%			
	Fraction Inspired Oxygen Set	0	94.1 ± 9.67%	94.1 ± 9.73%	94.0 ± 9.79%			
	Glascow Coma Scale Total	0	82.4 ± 17.85%	82.0 ± 18.10%	82.1 ± 17.82%			
	Glucose	0	77.0 ± 15.35%	77.0 ± 15.38%	77.2 ± 15.20%	94.7 ± 3.62%	94.7 ± 3.61%	94.7 ± 3.67%
	Hct	0				94.7 ± 3.37%	94.7 ± 3.37%	94.7 ± 3.41%
	Heart Rate	0				80.3 ± 29.21%	80.3 ± 29.29%	80.3 ± 29.27%
		1	91.1 ± 11.41%	91.2 ± 11.44%	91.0 ± 11.61%			
	Hematocrit	0	88.2 ± 6.81%	88.3 ± 6.63%	88.3 ± 6.66%			
	Hemoglobin	0	90.6 ± 4.89%	90.7 ± 4.70%	90.7 ± 4.80%			
	Lactate	0	97.2 ± 3.97%	97.3 ± 3.88%	97.3 ± 3.92%			
	Lactic Acid	0	97.6 ± 4.04%	97.7 ± 4.00%	97.7 ± 3.99%			
	Magnesium	0	91.5 ± 4.79%	91.6 ± 4.67%	91.6 ± 4.78%			
	Mean Blood Pressure	1	87.5 ± 13.31%	87.7 ± 13.21%	87.5 ± 13.43%			
	Mean Corpuscular Hemoglobin	0	93.5 ± 2.44%	93.6 ± 2.40%	93.6 ± 2.47%			
	Mean Corpuscular Hemoglobin Concentration	0	93.5 ± 2.44%	93.6 ± 2.40%	93.6 ± 2.47%			
	Mean Corpuscular Volume	0	93.5 ± 2.44%	93.6 ± 2.40%	93.6 ± 2.47%			
	Noninvasive Diastolic	1				79.7 ± 24.12%	79.7 ± 24.16%	79.6 ± 24.21%
	Noninvasive Mean	1				79.8 ± 24.12%	79.8 ± 24.17%	79.8 ± 24.22%
	Noninvasive Systolic	1				79.7 ± 24.12%	79.7 ± 24.16%	79.6 ± 24.21%
	Oxygen Saturation	1	86.8 ± 15.09%	86.9 ± 14.99%	86.7 ± 15.18%			
	Partial Pressure Of Carbon Dioxide	0	92.3 ± 9.05%	92.4 ± 8.89%	92.3 ± 8.97%			
	Partial Pressure Of Oxygen	0	96.0 ± 6.79%	96.1 ± 6.63%	96.0 ± 6.76%			
	Partial Thromboplastin Time	0	93.7 ± 5.11%	93.8 ± 4.96%	93.8 ± 4.98%			
	Peak Inspiratory Pressure	0	95.4 ± 6.80%	95.3 ± 6.90%	95.3 ± 6.85%			
	Ph	0	91.4 ± 9.74%	91.5 ± 9.62%	91.4 ± 9.67%			
	Phosphate	0	94.4 ± 3.11%	94.4 ± 3.07%	94.4 ± 3.15%			
	Phosphorous	0	94.6 ± 3.30%	94.6 ± 3.29%	94.6 ± 3.30%			
	Plateau Pressure	0	97.2 ± 4.70%	97.1 ± 4.78%	97.1 ± 4.75%			
	Platelets	0	91.4 ± 4.50%	91.5 ± 4.34%	91.5 ± 4.42%			
	Positive End-Expiratory Pressure Set	0	93.5 ± 8.45%	93.4 ± 8.55%	93.4 ± 8.49%			
	Potassium	0	89.4 ± 6.15%	89.4 ± 6.06%	89.4 ± 6.20%	94.8 ± 4.66%	94.8 ± 4.67%	94.8 ± 4.71%
	Potassium Serum	0	96.7 ± 4.12%	96.8 ± 4.08%	96.8 ± 4.13%			
	Prothrombin Time Inr	0	94.0 ± 4.67%	94.1 ± 4.53%	94.1 ± 4.60%			
	Prothrombin Time Pt	0	94.0 ± 4.67%	94.1 ± 4.53%	94.1 ± 4.60%			
	Pulmonary Artery Pressure Mean	0	97.2 ± 12.56%	97.0 ± 13.02%	97.1 ± 12.81%			
	Pulmonary Artery Pressure Systolic	0	91.3 ± 19.70%	91.0 ± 19.93%	91.2 ± 19.77%			
	Red Blood Cell Count	0	93.5 ± 2.44%	93.6 ± 2.41%	93.5 ± 2.47%			
	Respiratory Rate	0				82.1 ± 28.33%	82.0 ± 28.43%	82.1 ± 28.38%
		1	89.6 ± 13.74%	89.7 ± 13.76%	89.5 ± 14.04%			
	Respiratory Rate Set	0	95.8 ± 6.52%	95.8 ± 6.60%	95.7 ± 6.57%			
	Sao2	0				81.6 ± 28.19%	81.6 ± 28.25%	81.6 ± 28.27%
	Sodium	0	89.7 ± 5.89%	89.9 ± 5.74%	89.8 ± 5.87%			
	St1	0				92.2 ± 20.06%	92.3 ± 19.98%	92.2 ± 20.03%
	St2	0				91.8 ± 20.66%	91.9 ± 20.59%	91.8 ± 20.67%
	St3	0				92.4 ± 19.86%	92.4 ± 19.76%	92.4 ± 19.83%
	Systemic Vascular Resistance	0	96.8 ± 10.43%	96.7 ± 10.68%	96.8 ± 10.55%			
	Systolic Blood Pressure	1	88.0 ± 12.69%	88.1 ± 12.60%	88.0 ± 12.77%			
	Temperature	0	70.6 ± 13.32%	70.5 ± 13.47%	70.7 ± 13.33%			
	Tidal Volume Observed	0	94.3 ± 8.13%	94.3 ± 8.18%	94.3 ± 8.15%			
	Tidal Volume Set	0	96.1 ± 6.13%	96.0 ± 6.20%	96.0 ± 6.16%			
	Tidal Volume Spontaneous	0	97.3 ± 4.55%	97.3 ± 4.59%	97.2 ± 4.59%			
	Weight	0	97.3 ± 2.76%	97.3 ± 2.58%	97.3 ± 2.67%			
	White Blood Cell Count	0	91.7 ± 4.27%	91.8 ± 4.12%	91.8 ± 4.24%			

Table 7: ICD task per-label majority class accuracies. As the ICD task is defined only on MIMIC-III, this table is specific to that cohort.

Task	Label	Class	Train	Tuning	Held-out Test
ICD	Blood	0	52.5 ± 0.16%	54.3 ± 0.77%	54.7 ± 0.48%
	Circulatory	1	72.0 ± 0.14%	78.8 ± 1.45%	78.7 ± 1.14%
	Congenital	0	91.4 ± 0.15%	94.8 ± 0.51%	95.1 ± 0.56%
	Defined	0	50.8 ± 0.41%	53.3 ± 1.30%	53.6 ± 1.17%
	Digestive	0	51.2 ± 0.35%	54.0 ± 1.34%	54.0 ± 1.34%
	Endocrine	1	63.5 ± 0.27%	67.6 ± 1.30%	67.1 ± 1.18%
	Genitourinary	0	51.9 ± 0.10%	52.8 ± 1.77%	52.5 ± 1.26%
	Infection	0	58.0 ± 0.24%	64.4 ± 0.70%	65.5 ± 1.61%
	Injury	0	50.1 ± 0.04%	51.8 ± 0.10%	51.1 ± 0.63%
		1	50.2 ± 0.13%	48.8 ± 1.16%	48.8 ± 1.05%
	Mental	0	57.0 ± 0.19%	61.4 ± 0.88%	61.1 ± 0.80%
	Musculoskeletal	0	66.7 ± 0.08%	73.5 ± 0.28%	73.8 ± 0.22%
	Neoplasms	0	70.3 ± 0.32%	76.1 ± 1.03%	75.7 ± 0.32%
	Nervous	0	59.3 ± 0.16%	63.9 ± 0.62%	64.0 ± 0.41%
	Perinatal	0	100.0 ± 0.01%	100.0 ± 0.03%	100.0 ± 0.03%
	Pregnancy	0	98.8 ± 0.04%	99.3 ± 0.30%	99.5 ± 0.24%
	Respiratory	1	53.3 ± 0.15%	53.6 ± 0.87%	53.3 ± 0.81%
	Skin	0	76.7 ± 0.25%	85.1 ± 1.50%	85.3 ± 1.13%
	Unknown	0	100.0 ± 0.01%	100.0 ± 0.03%	100.0 ± 0.03%

Table 8: All discharge locations we predict on the MIMIC-III cohort, along with the percent of patient-hours in the train set across all 5 splits.

Class	24H			48H		
	Train	Tuning	Held-out Test	Train	Tuning	Held-out Test
Long Term Care Hospital	1.0 ± 6.07%	1.0 ± 5.96%	1.0 ± 5.80%	1.9 ± 10.96%	1.9 ± 10.68%	1.8 ± 10.55%
Rehab/Distinct Part Hosp	3.8 ± 11.04%	3.8 ± 11.12%	3.9 ± 11.23%	7.0 ± 20.01%	7.1 ± 20.06%	7.1 ± 20.20%
Home Health Care	8.6 ± 16.23%	8.6 ± 16.23%	8.6 ± 16.09%	15.9 ± 29.41%	15.9 ± 29.40%	15.8 ± 29.22%
Disc-Tran Cancer/Chldrn H	0.5 ± 4.25%	0.5 ± 4.16%	0.5 ± 4.14%	0.9 ± 7.58%	0.8 ± 7.40%	0.8 ± 7.37%
Short Term Hospital	0.3 ± 3.66%	0.4 ± 3.73%	0.4 ± 3.83%	0.6 ± 6.55%	0.6 ± 6.61%	0.7 ± 6.75%
Icf	0.0 ± 1.50%	0.0 ± 1.33%	0.0 ± 1.47%	0.1 ± 2.61%	0.1 ± 2.26%	0.1 ± 2.48%
Disc-Tran To Federal Hc	0.0 ± 0.62%	0.0 ± 0.01%	0.0 ± *nan*%	0.0 ± 1.19%	0.0 ± 0.03%	0.0 ± *nan*%
Disch-Tran To Psych Hosp	0.4 ± 4.20%	0.3 ± 3.86%	0.4 ± 4.13%	0.7 ± 7.29%	0.6 ± 6.80%	0.7 ± 7.22%
Other Facility	0.0 ± 1.30%	0.0 ± 1.19%	0.0 ± 0.01%	0.1 ± 2.36%	0.1 ± 2.13%	0.0 ± 0.03%
Home With Home Iv Providr	0.0 ± 1.42%	0.1 ± 1.60%	0.1 ± 1.72%	0.1 ± 2.50%	0.1 ± 2.90%	0.1 ± 3.01%
Home	9.1 ± 17.60%	9.3 ± 17.76%	9.2 ± 17.66%	16.3 ± 30.90%	16.6 ± 31.15%	16.5 ± 31.05%
Left Against Medical Advi	0.2 ± 2.66%	0.2 ± 2.72%	0.2 ± 2.74%	0.3 ± 4.61%	0.3 ± 4.73%	0.3 ± 4.75%
Hospice-Medical Facility	0.1 ± 1.86%	0.1 ± 1.65%	0.1 ± 1.66%	0.2 ± 3.36%	0.2 ± 3.06%	0.2 ± 3.02%
No Discharge	57.7 ± 24.68%	57.7 ± 24.57%	57.7 ± 24.79%	27.6 ± 26.58%	27.5 ± 26.45%	27.6 ± 26.60%
Snf	5.5 ± 13.24%	5.5 ± 13.32%	5.3 ± 13.09%	10.0 ± 23.95%	10.1 ± 24.02%	9.9 ± 23.79%
Hospice-Home	0.3 ± 3.18%	0.3 ± 3.04%	0.2 ± 2.97%	0.5 ± 5.73%	0.5 ± 5.52%	0.5 ± 5.47%
Snf-Medicaid Only Certif	0.0 ± 0.32%	*nan* ± *nan*%	*nan* ± *nan*%	0.0 ± 0.61%	*nan* ± *nan*%	*nan* ± *nan*%

Table 9: All discharge locations we predict for the eICU cohort, along with the percent of patient-hours in the train set across all 5 splits.

Class	24H			48H		
	Train	Tuning	Held-out Test	Train	Tuning	Held-out Test
	0.3 ± 3.93%	0.3 ± 3.88%	0.3 ± 3.92%	0.5 ± 5.74%	0.5 ± 5.79%	0.5 ± 5.81%
Other	1.4 ± 8.63%	1.5 ± 8.84%	1.4 ± 8.75%	2.1 ± 12.08%	2.2 ± 12.18%	2.1 ± 12.00%
Other External	1.5 ± 8.56%	1.5 ± 8.47%	1.6 ± 8.64%	2.4 ± 12.55%	2.4 ± 12.44%	2.5 ± 12.63%
Other Hospital	0.9 ± 6.68%	0.9 ± 6.56%	0.9 ± 6.53%	1.5 ± 9.91%	1.4 ± 9.68%	1.4 ± 9.67%
Skilled Nursing Facility	5.6 ± 15.51%	5.6 ± 15.52%	5.6 ± 15.56%	8.8 ± 22.67%	8.9 ± 22.79%	8.8 ± 22.71%
No Discharge	48.2 ± 26.67%	48.2 ± 26.73%	48.2 ± 26.71%	21.6 ± 24.60%	21.7 ± 24.74%	21.7 ± 24.68%
Nursing Home	0.4 ± 4.20%	0.4 ± 4.30%	0.4 ± 4.22%	0.6 ± 6.13%	0.6 ± 6.38%	0.6 ± 6.36%
Home	27.8 ± 28.14%	27.6 ± 28.13%	27.6 ± 28.06%	40.2 ± 35.95%	39.9 ± 36.03%	40.0 ± 35.98%
Rehabilitation	2.0 ± 9.46%	1.9 ± 9.30%	2.0 ± 9.30%	3.2 ± 14.15%	3.1 ± 13.96%	3.2 ± 14.00%

Table 10: The prevalence for the various classes for our "Final Acuity" (ACU) task on the eICU cohort, averaged over all 5 rotations.

Label	Class	Train	Tuning	Held-out Test
		0.8 ± 0.01%	0.8 ± 0.10%	0.8 ± 0.11%
	Other	3.4 ± 0.04%	3.4 ± 0.14%	3.3 ± 0.33%
	In-Hospital Mortality	3.5 ± 0.05%	3.6 ± 0.29%	3.6 ± 0.32%
	Other External	4.1 ± 0.03%	4.1 ± 0.16%	4.2 ± 0.10%
	Other Hospital	2.6 ± 0.06%	2.5 ± 0.31%	2.5 ± 0.31%
	In-ICU Mortality	4.7 ± 0.01%	4.8 ± 0.15%	4.8 ± 0.17%
	Skilled Nursing Facility	14.7 ± 0.03%	14.9 ± 0.47%	14.7 ± 0.56%
	Home	59.7 ± 0.10%	59.2 ± 0.39%	59.4 ± 0.64%
	Nursing Home	1.0 ± 0.01%	1.1 ± 0.12%	1.1 ± 0.12%
	Rehabilitation	5.5 ± 0.01%	5.5 ± 0.31%	5.6 ± 0.30%

Table 11: The prevalence for the various classes for our "Final Acuity" (ACU) task on the MIMIC-III cohort, averaged over all 5 rotations.

Label	Class	Train	Tuning	Held-out Test
	Long Term Care Hospital	3.7 ± 0.07%	3.7 ± 0.41%	3.6 ± 0.38%
	Rehab/Distinct Part Hosp	13.2 ± 0.08%	13.4 ± 0.84%	13.4 ± 0.59%
	Disc-Tran Cancer/Chldrn H	1.5 ± 0.05%	1.5 ± 0.31%	1.5 ± 0.28%
	Home Health Care	25.3 ± 0.24%	25.2 ± 1.34%	25.3 ± 1.15%
	Short Term Hospital	1.1 ± 0.05%	1.1 ± 0.22%	1.1 ± 0.18%
	Icf	0.1 ± 0.01%	0.1 ± 0.06%	0.1 ± 0.06%
	Disc-Tran To Federal Hc	0.0 ± 0.00%	0.1 ± 0.03%	0.0 ± *nan*%
	Disch-Tran To Psych Hosp	1.0 ± 0.03%	0.9 ± 0.13%	1.0 ± 0.23%
	In-Hospital Mortality	3.7 ± 0.07%	3.4 ± 0.23%	3.7 ± 0.55%
	In-ICU Mortality	7.4 ± 0.05%	7.5 ± 0.48%	7.5 ± 0.42%
	Home	24.0 ± 0.08%	24.3 ± 0.38%	24.0 ± 0.50%
	Left Against Medical Advi	0.4 ± 0.03%	0.4 ± 0.16%	0.4 ± 0.16%
	Home With Home Iv Providr	0.1 ± 0.01%	0.2 ± 0.11%	0.2 ± 0.08%
	Other Facility	0.1 ± 0.01%	0.1 ± 0.06%	0.1 ± 0.04%
	Hospice-Medical Facility	0.3 ± 0.03%	0.3 ± 0.16%	0.3 ± 0.16%
	Snf	17.3 ± 0.12%	17.2 ± 0.75%	16.9 ± 0.53%
	Hospice-Home	0.9 ± 0.03%	0.8 ± 0.14%	0.8 ± 0.19%
	Snf-Medicaid Only Certif	0.0 ± 0.00%	*nan* ± *nan*%	*nan* ± *nan*%

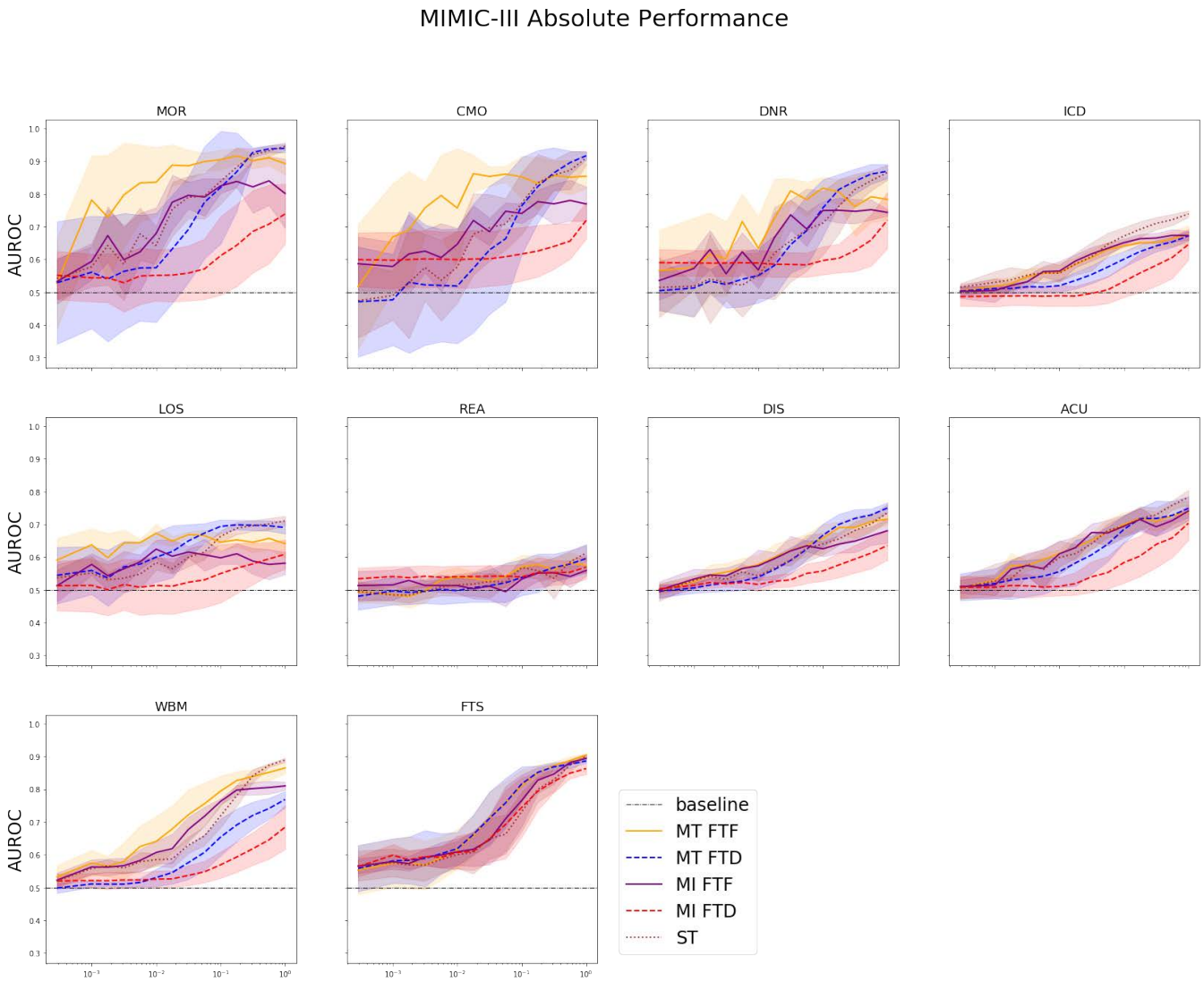

Figure 6: Performance in macro-averaged AUROC (y-axis) of various PT/FT models (linestyle) across various FT dataset subsampling rates (x-axis), over all FT tasks (subplots) for MIMIC-III.

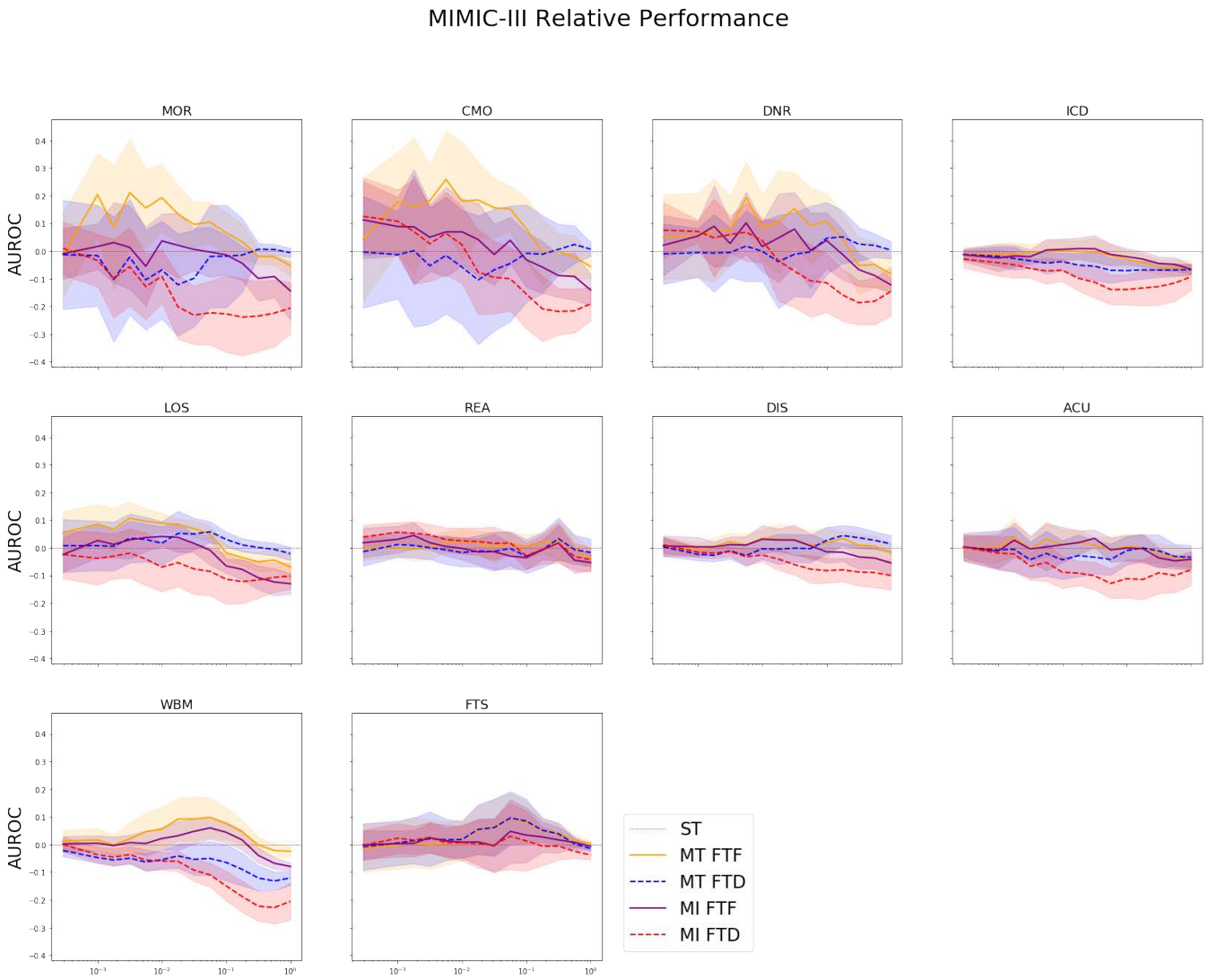

Figure 7: The difference between various FT modes and ST results on MIMIC-III.

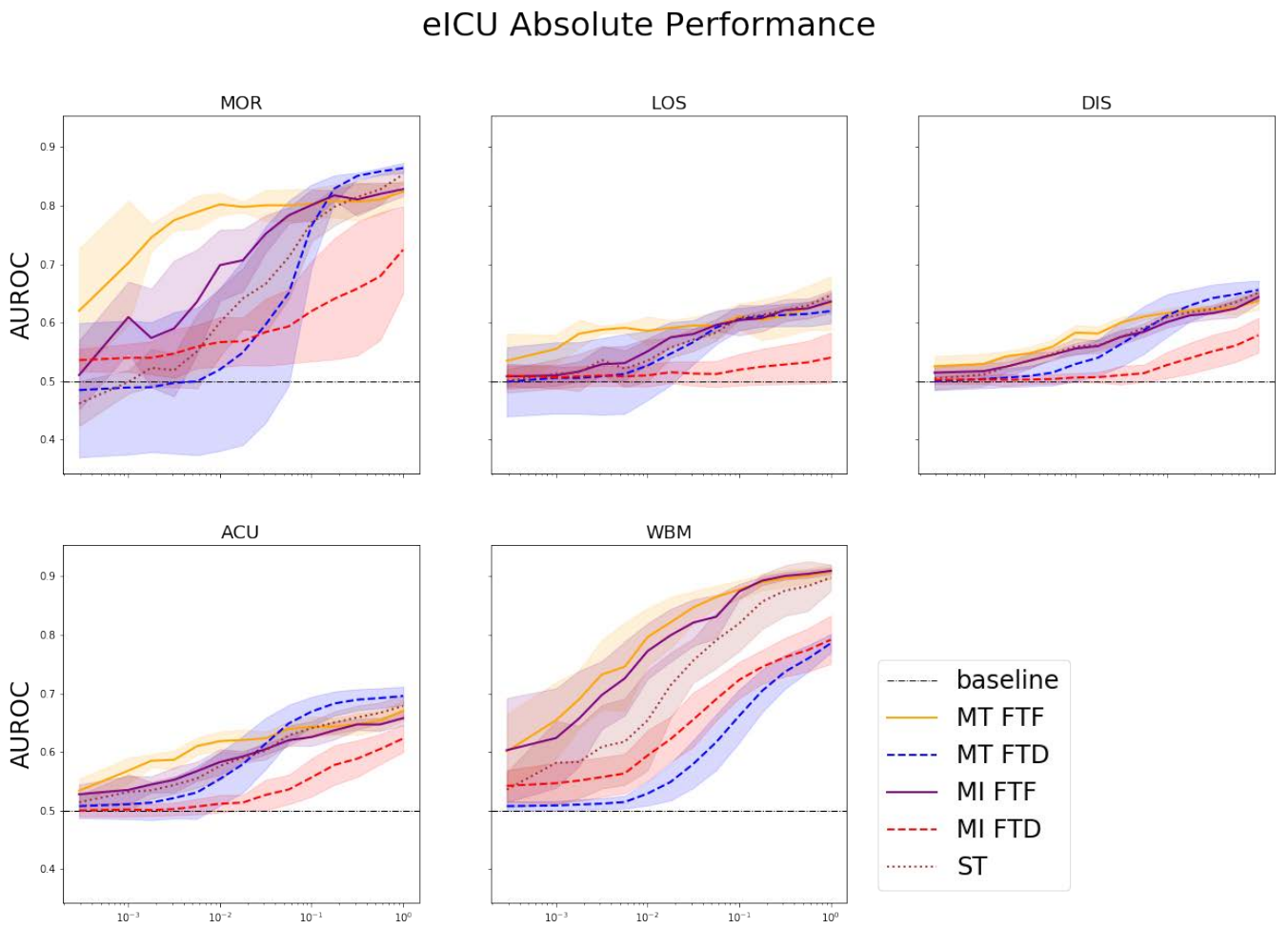

Figure 8: Performance in macro-averaged AUROC (y-axis) of various PT/FT models (linestyle) across various FT dataset subsampling rates (x-axis), over all FT tasks (subplots) for eICU.

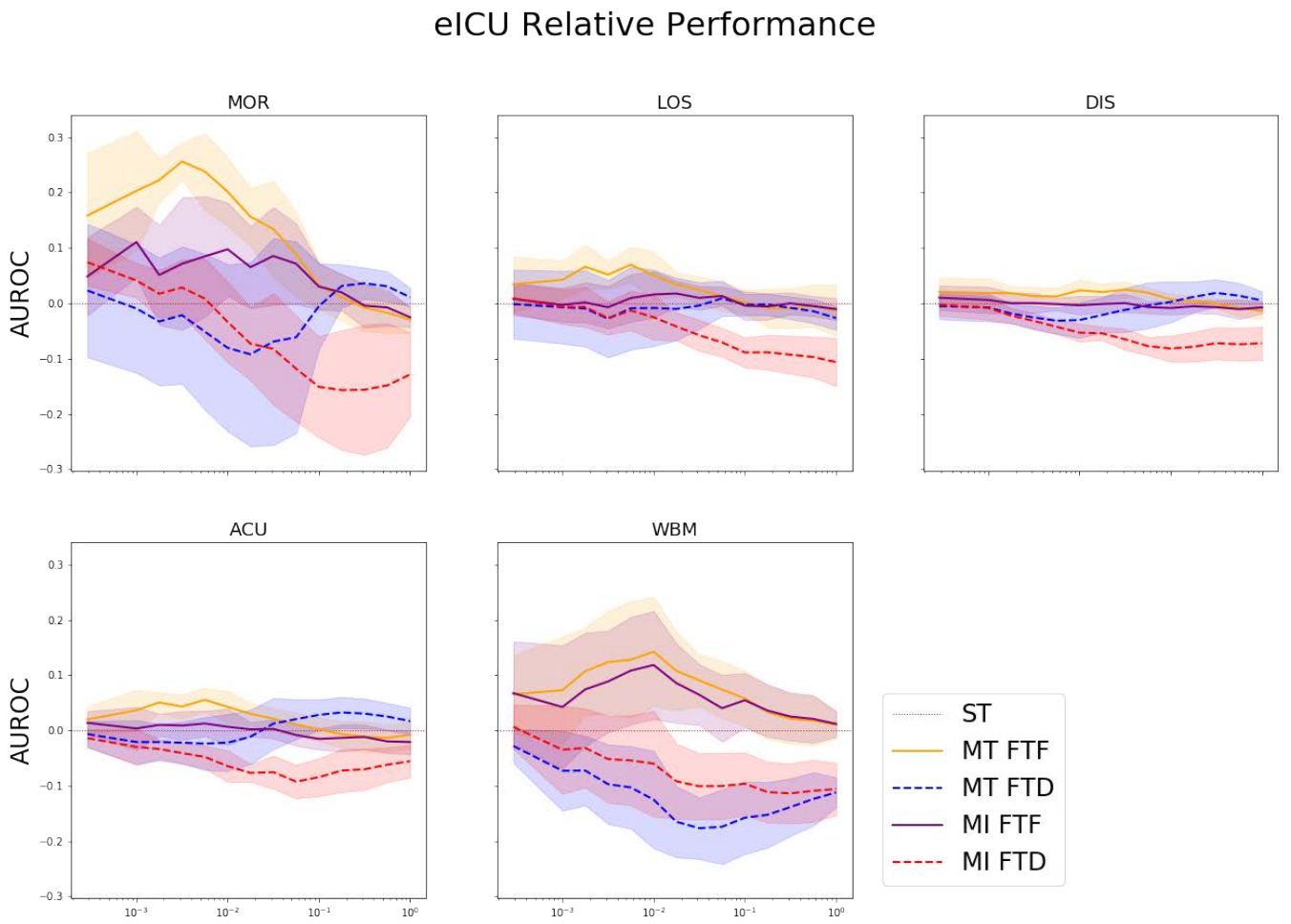

Figure 9: The difference between various FT modes and ST results on eICU.

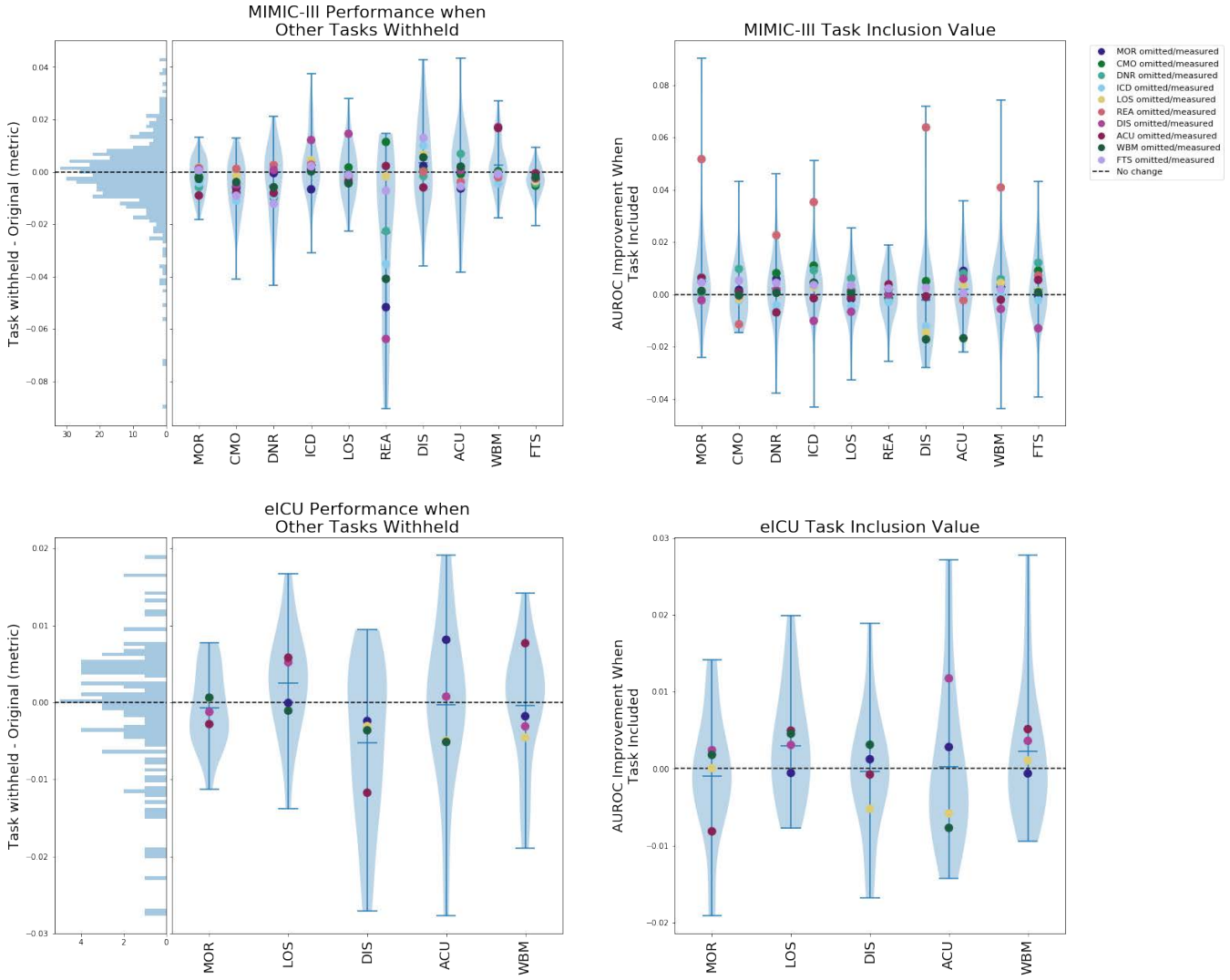

Figure 10: We examine the value either for a downstream task or by a downstream task in the context of multi-task ensemble makeup. On the left, we show, for each task on the x-axis, the performance difference *on that task* (y-axis) between a MT learning setting where a single other task (colored dot) is omitted from the ensemble vs. a full MT learning ensemble. This plot also shows an overall histogram of these discrepancies to its left. On the right, we show the transpose view – for any given task (x-axis), we plot how much performance on other tasks (colored dots) is *improved* (y-axis) by *including* the x-axis task in the learning ensemble. The same numbers are summarized in both plots, just from differing perspectives (in particular, the coordinates in the right plot are negated and transposed from those in the left). We see that, like our global finding, there is minimal evidence of any general positive or negative transfer here – instead, any relationships are highly task specific, and on average no transfer is observed one way or another. Note that while these results suggest there is no universal negative transfer, they do suggest that negative transfer is quite likely in a generic MT setting without careful consideration.

Table 12: The `Hyperopt` search space we used in this work. Distributions are noted in pseudocode, but typically refer directly to the appropriate analog in `Hyperopt` (e.g., a uniform distribution over an integral parameter maps to the quantized uniform distribution that only outputs integers).

Hyperparameter	Search Space
# Epochs	`Uniform[10, 35]`
Batch Size	`Uniform[8, 512]`
Learning Rate (LR)	`Lognormal[-7, 0.5]`
LR Decay	`Loguniform[-2.3, 0]`
Hidden Dropout	`Uniform[0, 0.5]`
Hidden Size	`Uniform[8, 256]`
Weight Decay	`Uniform[0, 1]`
Input Window Size (h)	`Uniform[12, 168]`
Bidirectional	`Choice[True, False]`
# Hidden Layers	`Uniform[1, 4]`
Encoder Hidden Layer Size	`Uniform[8, 512]`
GRU Pooling Method	`Choice[max, avg, last]`
GRU FC Layer Base Size	`Uniform[8, 512]`
GRU FC Layer Growth	`Loguniform[-1.1, 1.1]`

Table 13: How many random train/test splits are used to produce each experimental setting shown in this work for MIMIC-III. Unless otherwise stated, the same number of samples are used across all few-shot fractions under a given setting.

PT/FT Regime Task	MI FTD	MI FTF	MT FTD	MT FTF	ST
ACU	5	5	6	6	5
FTS	5	5	6	6	5
ICD	5	5	6	6	5
DIS	5	5	6	6	5
DNR	5	5	6	6	5
REA	5	5	6	6	5
MOR	5	5	6	6	5
LOS	5	5	6	6	5
CMO	5	5	6	6	5
WBM	5	5	6	6	5

Table 14: How many random train/test splits are used to produce each experimental setting shown in this work for eICU. Unless otherwise stated, the same number of samples are used across all few-shot fractions under a given setting.

PT/FT Regime Task	MI FTD	MI FTF	MT FTD	MT FTF	ST
ACU	5	5	5	5	5
DIS	5	5	5	5	5
MOR	5	5	5	5	5
LOS	5	5	5	5	5
WBM	5	5	5	5	5

	MIMIC-III	eICU
ACU	-0.02 ± 0.02	0.01 ± 0.02
CMO	0.02 ± 0.02	
DIS	0.0 ± 0.01	0.0 ± 0.01
DNR	0.01 ± 0.03	
FTS	0.0 ± 0.01	
ICD	-0.07 ± 0.04	
LOS	-0.02 ± 0.03	-0.0 ± 0.02
MOR	-0.0 ± 0.01	0.01 ± 0.01
REA	0.01 ± 0.04	
WBM	-0.08 ± 0.03	-0.01 ± 0.02

Table 15: Difference between full multi-task hyperparameter search results and single-task results across datasets and tasks. We see no systematic preference towards either multi-task or single-task results.