Phenotypical Ontology Driven Framework for Multi-Task Learning

Mohamed Ghalwash, Zijun Yao, Prithwish Chakraborty, James Codella, Daby Sow
IBM Research
Yorktown Heights, NY 10598

Abstract

The increasing availability of rich longitudinal Electronic Health Records (EHR) has driven the development of AI methods aimed at generating valuable clinical insights. Despite the large coverage of patients in EHRs, cohorts extracted to derive outcome predictive models for specific phenotypes often result in small and imbalanced data sets that are of limited practical use for AI modeling. In this paper, we propose an Ontology driven Multi-Task Learning framework called OMTL that is able to cope with these data limitations for EHR modeling by leveraging external knowledge. The model is especially designed for situations where we want to study patients for a set of related disease categorizations, and interested in potentially different outcomes corresponding to different categories. More specifically, OMTL leverages phenotype similarity according to a predefined ontology to learn the outcome prediction tasks. The model structure allows for an increased sharing of data representations across phenotype profiles such that the information sharing is regulated by the ontological and patient level similarities. We evaluate the benefit of leveraging an ontology to relate tasks in a multi-task learning settings and demonstrate the efficacy of OMTL on several real outcome prediction problems on patients from the MIMIC III database over state-of-the-art multi-task learning schemes.

1 Introduction

The increasing availability of rich longitudinal patient records from large EHR has enabled novel data-driven analysis on such observational data towards clinical hypothesis generation and testing. EHR patient data can cover several aspects of patient history such as diagnosis, medication, and lab tests resulting in a diverse set of patient attributes.

Despite the large number of patients covered in these data sets, formulating a query to extract a cohort for the outcome of interest often results in a very modest number of data points being returned, hindering the effectiveness and reproducibility of deep learning techniques on these data sets. Let us consider a scernario where we want to study patients who have experienced Rheumatic disease of hearth valve and Mitral valve stenosis on the well-known MIMIC III data set [10]. Such a query only returns 210 ICU stays, down from the original 42000 stays present in the complete data set. In another case, on an EHR database spanning more than 55 million patients, querying for patients within the age group $21 - 40$ and having experienced concussions that led to mild cognitive impairments returns fewer than 1100 patients. Further characterization of this cohort to focus only on patients that eventually experienced dementia takes that number below 200.

Furthermore, such histories are often sparse and difficult to be directly utilized for simple analysis. Indeed, being observational in nature, EHR data sets often exhibit widely varying regularity of encounters as well as diseases coverage. This data imbalance can manifest itself as a 'vanishing data'

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

problem during cohort analysis of specific diseases of interest with low coverage. Any analysis on such data sets can be impractical and suffer from significant lack of robustness.

Interestingly, a significant amount of domain knowledge relating medical concepts is readily available in several medical ontology. Phenotypes corresponding to observable physical properties of patients [1], including diseases have been organized in ontologies and expressed within EHR systems. We postulate that leveraging phenotypical relations within EHR data sets to tie and jointly learn multiple learning outcomes in a multi-task learning setup could provide solutions to this vanishing data problem and lead to more robust prediction models.

In this paper, we propose OMTL, a novel phenotypical ontology-driven framework for multi-task learning. The main tenant of our approach is to ground the structure of the proposed multi-task framework on phenotyical medical concepts relations between patients from different medical cohorts. Using OMTL allows us to cope with the lack of data for training specific outcomes by leveraging existing knowledge encapsulated in ontologies. Learning outcomes for different phenotypes may borrow representation expressiveness during training for the prediction of several outcomes. This sharing allows OMTL to generate prediction models for patient phenotyical cohorts even for phenotypes with little amount of patient observations by leveraging data from other phenotypical cohorts.

Moreover, OMTL enables sharing information among phenotypes based on the ontology which allows the model to decouple the outcome learning from the representation learning of similar diseases. In this case, the model allows studying patients with related diseases with different outcomes. For example, it can learn about mortality prediction for diabetic patients and readmission prediction for patients with hypertension by sharing information between diabetes and hypertension.

The contributions of this paper are 3 folds:

1. We proposed a systematic way to tie different phenotypical outcome using existing domain knowledge in the form of ontologies of medical concepts (e.g. SNOMED CT [4]).
2. We propose a novel hierarchical deep learning architecture using gating mechanisms mirroring the structure of a domain ontology.
3. We demonstrate the efficacy of the proposed approach on the prediction of several outcomes on patients from the MIMIC III database.

In the remainder of the paper, we start with a review of related work before exposing in details the proposed method OMTL. We then present the experimental results geared towards the evaluation of the gain achieved by leveraging domain ontologies to link together multiple tasks in OMTL.

2 Related Work

Multi-task learning: Multi-task learning has received a great deal of attention in the last few years. More notably, [12] adapts Mixture-of-Experts (MoE) structure to jointly learn multiple outcomes by sharing a single expert, backed by gating networks to specialize for each outcome and learn the relationships jointly. In [8, 2], authors focus on using hierarchical relationships to learn multiple outcomes. Specifically, a two-level hierarchy has been used to regularize a learned model to share information among related outcomes. [21] extended these approaches with multi-level hierarchies. They proposed a cascade approach to progressively learn outcomes at prior levels to boost learning performance on outcomes in immediately subsequent levels.

The model proposed in [14] learns intermediate representations for predictive tasks at multiple scales for the same input via a cascade of connected layers. In [15], the authors proposed a way to incorporate a set of *carefully* selected semantic NLP tasks into the multi-task model. The tasks that are “easy” to learn are supervised at the lower levels of the architecture while more complex interactions are kept at deeper layers. A soft-ordering of the shared layers was proposed in [13] to learn both the layers and the order of layers for each task.

OMTL is motivated by such efforts to learn phenotype specific predictive models in a multi-task setting to share information across outcomes. It is also important to note that OMTL uses an ontology based hierarchy, but more crucially, decouples tasks from hierarchies such that representations learned for different phenotypes can be further specialized w.r.t. different tasks such as readmission and mortality prediction. In most of the cases, the shared layers are integrated in parallel orders, where the outcome of one layer is fed into the next layer.

Knowledge graph: Structured domain knowledge, often in the form of graphs, have been applied in various applications to achieve more robust inferences over only data-driven models. [18] presents a framework to enable the use of various kinds of external knowledge bases to retrieve answers relevant to a given question. It provides a graph based model which maps the text phrases to concepts in the external knowledge graph with several strategies: concepts only, one-hop, and two-hop. Finally, the extracted graph is used as an input to a neural network model for entailment.

Outcome prediction from EHR: Over the years, many researchers have developed predictive models on EHR for the prediction of various adverse events such as early identification of heart failure [16], readmission prediction [19], and recently acute kidney injury prediction [17]. Other notable works include [20] where the authors presented a multi-level attention mechanism to derive patient specific representations. Specifically, the authors split the trajectory of patients into subsequences and use within-subsequence and between-subsequence attention levels sequentially to arrive at patient representations in order to handle various data issues such as data sparsity and noise. Of note is also the work by [7] where the authors present four clinical prediction benchmarks using data derived from the publicly available Medical Information Mart for Intensive Care (MIMIC-III) database, including In-hospital mortality, Decompensation, Length of stay, and Acute care phenotype classification. In addition, it proposes a deep learning model called multi-task Recurrent Neural Networks to empirically validate the four prediction benchmarking tasks. This benchmark has been used as the basis for evaluating the effectiveness of OMTL.

3 Methods

To tackle the vanishing data problem, OMTL uses augmented patients having *similar* phenotypical profiles to the original cohort being studied. Specifically, OMTL builds on the recently published work of [12] using a multi-task mixture-of-experts model to share information across different learning tasks. This model essentially learns different shared patient data representations (called experts) fed as inputs for each individual learning outcomes. OMTL takes this multi-task learning approach further by leveraging the relationship and ontological distance among phenotypes to share information between learning tasks.

We assume the existence of an ontology for the domain, such as SNOMED [4] and CSS [6], that represents the semantic phenotypical relationships among the medical concepts describing the data fields of interest (e.g. diagnosis codes). We represent this ontology with a directed acyclic graph over of a set of nodes (e.g. diagnosis codes) and edges expressing parent-child *semantic* relationships between the nodes. Let $X = \{x_i \in \mathbb{R}^d; \forall\, i\}$ be the EHR data where x_i is the data record i (e.g. an intensive care unit visit or an admission for a patient), and d is the number of features[1].

For each data record, we assume the presence of all phenotypical medical concepts tied to the record. Using that information, we construct a subgraph $\mathcal{G}$ of the ontology with only the nodes corresponding to medical concepts that are present in X. Within $\mathcal{G}$, the set of user-specified phenotypes of interest corresponding to the cohort under study (collection of medical concepts) are represented by a subset of nodes, which we call core nodes $\mathcal{C}$ (blue nodes in $\mathcal{G}$ depicted in Figure 1). The process for obtaining such a graph is described in details in the supplementary materials.

Any outcome prediction on X can be formulated as a mapping from X into an outcome set Y. To enable sharing across outcomes, we break down this mapping as follows. We first define an encoding function $\mathcal{E}$ generating representations as:

$$\mathcal{E} : X \rightarrow Z \tag{1}$$

where $Z = \{z_i \in \mathbb{R}^{d'}\}$, d' is the dimension of the representation, and z_i represents the representation for record i. For this representation to be useful, we enforce an unsupervised reconstruction process by learning a reconstruction function $\mathcal{R}$:

$$\mathcal{R} : Z \rightarrow X \tag{2}$$

[1] For simplicity we assume that $x_i \in \mathbb{R}^d$ but it is straightforward to apply the method to temporal data where $x_i \in \mathbb{R}^{T \times d}$

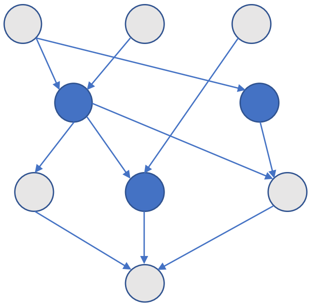


Figure 1: A directed acyclic subgraph that encompasses nodes represented in the dataset. The nodes in blue (i.e. core nodes) are of specific interest to the user.

The generated representation can be used as an input to a standard supervised machine learning algorithm to predict the outcome of interest $Y = \{y_i; \forall\ i\}$. This can be achieved by learning an outcome prediction function $\mathcal{E}_S$:

$$\mathcal{E}_S : Z \longrightarrow Y \tag{3}$$

Learning the parameters for the mappings $\mathcal{E}$, $\mathcal{R}$ and $\mathcal{E}_S$ can be achieved by minimizing the following objective function:

$$\mathcal{L} = \lambda \underbrace{\sum_i \|\mathcal{R}\big(\mathcal{E}(x_i)\big) - x_i\|_2}_{\mathcal{L}_1} + \underbrace{\sum_i \ell\Big(\mathcal{E}_S\big(\mathcal{E}(x_i)\big), y_i\Big)}_{\mathcal{L}_2} \tag{4}$$

where, $\mathcal{L}_1$ is the reconstruction loss function and $\mathcal{L}_2$ is the total outcome prediction loss that aggregates the losses ℓ (e.g. cross entropy for classification and mean square error for regression) at the node levels corresponding to the potentially different outcomes. $\lambda \in \mathbb{R}^+$ is a hyper-parameter that weighs the importance of the two losses.

Learning a representation and a predictive model for one task is relatively straightforward by optimizing Equation (4). However, learning such representations for multiple patient cohorts with different phenotypical profiles and multiple predictive tasks simultaneously is a challenging problem. OMTL tackles the problem by using a structured hierarchical network that mimics $\mathcal{G}$ and account for dissimilar coverage of patients across nodes. We formalize OMTL by first describing a multi-task learner and, subsequently, complete the formalization by describing how we include the hierarchical ontological structure as part of the model.

3.1 Multi-gate Mixture of Experts for Multi-task

Inspired by [12] a multi-gate mixture of experts approach can be used to learn $\mathcal{E}$ in a multi-task settings as in Figure 2 (a). Input data flows from the top into a mixture of E *expert nodes* responsible for clustering the input data subspace to facilitate representation computations. We denote these expert nodes as $E_e(\cdot), 1 \leq e \leq E$. These expert nodes are neural networks producing a representation $E_e(x_i)$ given input x_i.

The outputs of these expert nodes are consumed by an array of P representation nodes denoted $P_p(.), 1 \leq p \leq P$. Each P_p maps directly into one medical concept (i.e. the number of representation nodes is equal to the number of nodes in the subgraph $\mathcal{G}$ defined in the previous section). Each of these P_p nodes ingest experts outputs via a gating function G_p to learn how to combine information from all expert nodes for the generation of representations for the current medical concept. The gating node G_p controls the combination of expert nodes to produce the input $\mathcal{M}_p$ to the representation node P_p. These are typically very simple networks that produce combination weights for the experts at each representation node via following Equation:

$$\begin{aligned} G_p(x_i) &= \text{Softmax}(x_i \cdot W_p + b_p) \in \mathbb{R}^E \\ \mathcal{M}_p(x_i) &= \sum_{e=1}^{E} G_p^e(x_i) E_e(x_i) \in \mathbb{R}^{d_e} \end{aligned} \tag{5}$$

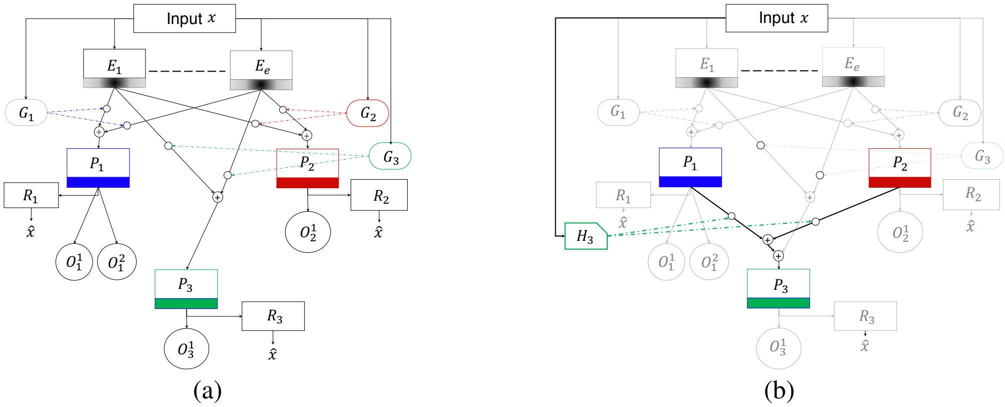


Figure 2: (a) Multi-gate mixture of experts architecture. (b) OMTL: The model is explained using only 3 nodes P_1, P_2, P_3 (medical concepts) to minimize confusion. The modules that are the same as in the baseline model are faded out for clarity, while the new additional modules are plotted in bold.

where $W_p \in \mathbb{R}^E, b_p \in \mathbb{R}$ are parameters to be learned by the gate network during training, and $G_p^e(x_i)$ is the entry e in the vector $G_p(x_i)$. The input $\mathcal{M}_p(x_i)$ is then passed to the representation node P_p to produce representations

$$\mathcal{E}_p(x_i) = P_p(\mathcal{M}_p(x_i)) \tag{6}$$

for the current medical concept p.

The representations are then passed as input to the *outcome nodes*. The outcome nodes leverage the representations to predict these outcomes. It is quite important to note that the model can be optimized for multiple outcomes of interest *and* each representation node can have different and multiple outcomes, e.g. we can associate mortality prediction for certain medical concepts and readmission prediction for another (possibly overlapping) subset of medical concepts. For example, in Figure 2 (a), the medical concept P_1 is associated with two outcomes O_1^1 and O_1^2 while the medical concept P_3 is associated with the outcome O_3^1.

This model is trained in a multi-task setting allowing expert nodes and representation nodes to share learned representations and support outcome nodes in their prediction tasks. The loss function to be optimized is

$$\mathcal{L} = \underbrace{\lambda \frac{1}{N.P} \sum_i \sum_{\substack{p \\ x_i \text{ has } p}} \big(\mathcal{R}_p(\mathcal{E}_p(x_i)) - x_i\big)^2}_{\mathcal{L}_1} + \underbrace{\frac{1}{N.P} \sum_i \sum_{\substack{p \in \mathcal{C} \\ x_i \text{ has } p}} \sum_{\substack{o \\ o \text{ assoc. with } p}} \ell\big(O_p^o(\mathcal{E}_p(x_i)), y_i^o\big)}_{\mathcal{L}_1} \tag{7}$$

where $\mathcal{R}_p$ is the reconstruction network that reconstructs the input x_i from the representation $\mathcal{E}_p(x_i)$, ℓ is the prediction loss function (e.g. cross entropy for classification) and y_i^o is the label for the outcome o for the visit i. The second sum in $\mathcal{L}_1$ runs over all representation nodes (medical concepts) that x_i expresses forcing the representation from all nodes to be useful for the reconstruction of the original input. While the second sum in $\mathcal{L}_2$ runs over all *core* nodes $\mathcal{C}$ that x_i expresses, based on the user interest in specific phenotypes, the third sum runs over all outcome nodes o that are associated with the representation nodes p.

3.2 Hierarchical Multi-task Approach with OMTL

Optimizing Equation (7) can be used to solve the multi-task problem outlined above. However, depending on the choice of the phenotypes of interest, there can be a significant variation in the number of patients spanned by the representation nodes. For example, nodes corresponding to a higher level concept will typically span a much greater number of patients than the ones spanned by concepts at a leaf level of the ontology where concepts are very specialized. Interestingly, such nodes are semantically related via the medical concept ontology. Therefore, the generated representations across different medical concepts should share similar semantic relationships. OMTL aims to increase the information sharing among nodes using the ontology.

The learning architecture for OMTL is depicted in Figure 2 (b). Compared to the architecture shown in Figure 2 (a), the representation nodes are now connected by mirroring the sub-graph $\mathcal{G}$. The representation nodes without ancestors in $\mathcal{G}$ can be computed using the same gating principle outlined in Equation (5). However, for a node with ancestors (such as P_3 in Figure 2 (b)), the model shares both the representation from the expert nodes and its parents. Assuming that P_j has n parents, the resultant representation $\mathcal{E}_j$ at P_j is an aggregation of two vectors: (i) $\mathcal{M}_j$ computed using the combination of experts as given by Equation (5) and, (ii) $\mathcal{P}_j$ computed using the combination of the representations from the parents as:

$$\begin{aligned} H_j(x_i) &= \text{Softmax}(x_i \cdot W_j + b_j) \\ \mathcal{P}_j(x_i) &= \sum_{k=1}^{n} H_j^k(x_i)\mathcal{E}_k(x_i) \end{aligned} \tag{8}$$

The input to the node P_j will be the aggregation of $\mathcal{M}_j$ and $\mathcal{P}_j$. The outcomes $\mathcal{E}_j(x_i)$ can be computed using the same procedure outlined in Equation (6). OMTL encourages information sharing among the nodes using the graphical structure informed by the ontology. More specifically, the ancestors of a representation node will be less affected by the 'vanishing data' problem and the structure thereby acts as a forcing mechanism to help the network to learn the representation nodes and outcome with higher efficacy. During the forward and backward pass of the training, data x_i is only routed to representation nodes corresponding to phenotypes expressed in x_i.

4 Experiments

In this section, we present an experimental evaluation of OMTL focusing on the impact provided by leveraging an ontology in a multi-task framework. We first describe the data used in the experiments including the construction of patient cohorts. We then present the experimental setup before presenting experimental results.

4.1 Data Description

We train and evaluate all models on data extracted from the publicly available MIMIC-III (Medical Information Mart for Intensive Care) data set [10], a large, single-center database of patients admitted to critical care units at a large tertiary hospital. While MIMIC-III includes several facets of patient medical records, we focus our experiments on the vital signs, observations and diagnosis codes from the cohort construction proposed in [7].

Since MIMIC-III tracks diagnosis codes in the ICD-9 format, only preserving a shallow two-level hierarchical structure of medical concepts, we map all the observed ICD-9 diagnosis codes to SNOMED CT codes, and construct a graph with a sufficiently deep hierarchy of medical concepts. As a result, we obtain a complex knowledge graph where each node is associated with a set of ICU stays. Next, we exploit a data augmentation algorithm[2] to extract cohorts for training our models as follows. Given the knowledge graph and a few core nodes (e.g., specific SNOMED nodes identified by domain experts for a certain phenotype), we aim to include more SNOMED nodes to augment the ICU stays for analysis. The data augmentation algorithm starts from this set of nodes and grows the set by choosing a series of predecessors using an MCMC sampler. We apply this node growth for a predefined number of iterations to obtain connected graphs with sufficient nodes and stays for training. In this manner, we obtain 3 data sets for different cohorts: liver disorder, heart disorder, and kidney disorder for which our predictive outcomes are phenotypes "Disorders of lipid metabolism", "Acute myocardial infarction', and "Chronic kidney disease", respectively, and in-hospital mortality prediction on both cohorts. The extracted graphs for all data sets are shown in Figure 4.

More statistical details on the selected cohorts can be found in Table 1. The corresponding data are multivariate time series for each ICU stay. As typical with real-world data, these data contain missing values. To address this issue, we replace these missing values with an average of carry-forward, carry-backward and mean imputation methods. We then flatten the time series data to a single time point using the mean values of each feature and each ICU stay. For each visit, we extracted 41 features for analysis.

[2] The details for the data augmentation algorithm used are described in the supplementary material.

Table 1: Statistics of datasets.

Dataset	# Core nodes	# Nodes	# Visits (+)
Liver Mortality	5	26	4767 (837)
Liver Pheno	5	26	9073 (1474)
Heart Mortality	3	21	13067 (1867)
Heart Pheno	3	21	24135 (4039)
Kidney Mortality	1	78	20316 (2749)
Kidney Pheno	1	78	39108 (5588)

4.2 Experimental Setup

As mentioned in the Methods section, our goal is to share information among outcomes of interest at a few core nodes of interest $\mathcal{C}$. To realize the benefits of the augmented nodes without restricting the outcomes to be homegenous across the nodes, we train OMTL by optimizing for outcome loss at the core nodes along with reconstruction loss at all nodes. Under this setup, our primary objective is to test the hypothesis that adding the structure information from the ontology helps the model to produce more accurate predictions. To test this hypothesis, we propose to compare OMTL against the multi-task mixture of expert models [12] trained using the loss structure outline in Equation 7. This model referred as MMOE in the rest of this section, uses mixture of experts [9] to share information among outcomes without using any prior knowledge about the semantic outcome-relationship. In addition, we compare the proposed method to mixture of experts (MOE) [5] where mulitple experts were utilized but no gates, and the most commonly used shared-bottom (SB) multi-task deep learning structure [3] where there is only one expert.

OMTL augments MMOE with a gating mechanism $\{H_j; \forall j\}$ enforcing the ontological structural relationship across representation nodes as described in the Methods section. To train OMTL, we first train it without considering the hierarchy to learn the experts and their corresponding gates G. We freeze these components and subsequently fine-tune for the outcomes and representations at the nodes by admitting the hierarchical structure.

Implementation details: Each module (expert, representation, reconstruction, outcome) is implemented as a one layer feed-forward neural network. We used 3 experts for MOE, MMOE, and OMTL. The size of the expert, reconstruction, representation modules are 41×5, 5×41, 5×5, respectively. The activation functions are Leaky ReLU for experts, Softplus for the representation layers, and ReLU for the reconstruction. Each expert has a dropout layer for regularization. The size of the gate G_p is 41×3 and the size of the gate H_j is $41 \times n_j$ where n_j is the number of parents for the representation node j. We performed hyper-parameter tuning and found good performance and stability across all models by setting the batch size to 64, learning rate to 0.001, dropout to 0.5, λ to 0.0001, and using Adam as the optimizer [11]. We evaluated the models using stratified 5-fold cross validation. the cross validation was done so as to ensure we have the same distribution of positive and negative examples in each node. All models were evaluated on exactly the same folds for fair comparison.

4.3 Multi-task Outcome Results

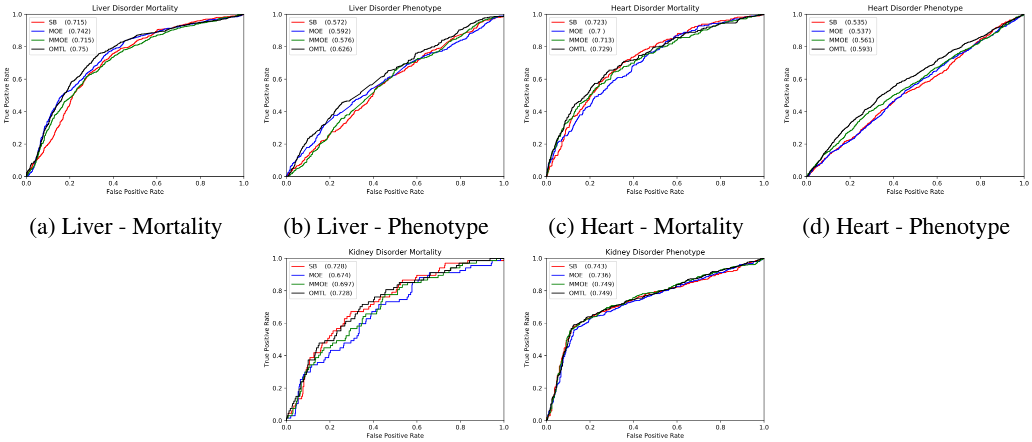
We compare all baselines with OMTL using AUC. The results are shown in Table 2 and Figure 3.

Table 2: AUC under ROC for all models.

Dataset	SB	MOE	MMOE	OMTL
Liver Disease Mortality	0 .715	0.742	0.715	**0.750**
Liver Disease Phenotype	0 .572	0.592	0.576	**0.626**
Heart Disease Mortality	0.723	0.700	0.713	**0.770**
Heart Disease Phenotype	0.535	0.537	0.561	**0.593**
Kidney Disease Mortality	**0.728**	0.674	0.697	**0.728**
Kidney Disease Phenotype	0.743	0.736	**0.749**	**0.749**

Table 3: AUC under PRC for all models.

Dataset	SB	MOE	MMOE	OMTL
Liver Disease Mortality	0.414	0.448	0.429	**0.478**
Liver Disease Phenotype	0.086	0.106	0.084	**0.109**
Heart Disease Mortality	0.211	0.221	0.242	**0.262**
Heart Disease Phenotype	0.180	0.182	0.202	**0.214**
Kidney Disease Mortality	0.252	0.219	0.233	**0.267**
Kidney Disease Phenotype	**0.806**	0.798	**0.806**	**0.806**



(a) Liver - Mortality

(b) Liver - Phenotype

(c) Heart - Mortality

(d) Heart - Phenotype

(e) Kidney - Phenotype

(f) Kidney - Phenotype

Figure 3: ROC for all experiments.

In general, we can see from Table 2 that the AUC gain that the structure is adding to MMOE ranges from 5%-9% for the liver disorder data with ontological graph shown in Figure 4 (a). It is worth noting that the core nodes in this graph are well-connected to each other. We hypothesize that the benefit of using the structural information among such well-connected core nodes helps OMTL to attain a significant boost in sharing information and hence provides more accurate predictions than MMOE.

The gain in the heart disorder graph, which is shown in Figure 4 (b), ranges from 2% to 6%. The AUC gain is not as significant as in the liver disorder case. A possible explanation for this is that the core nodes are less connected to each other. In such situation, the training may benefit by leveraging more outcome nodes. Similar conclusion are seen in the kidney graph. In the next section, we investigate this hypothesis.

Since the datasets are highly imbalanced, we also show the AUC under the precision recall curve (PRC) as shown in Figure 5 and Table 3. The OMTL model outperforms MMOE in the precision which indicates that it is able to predict the rare positive class more accurately than MMOE.

5 Conclusion

We proposed OMTL, a novel multi-task learning approach to analyze EHR data. OMTL models outcomes by incorporating existing ontology of phenotypical information in the multi-task setup. We have demonstrated the utility of this approach by comparing against state-of-the-art knowledge agnostic multi-task framework over a set of datasets. Our ongoing efforts are aimed at deploying the framework for large scale EHR analysis.

References

[1] Definition phenotype. https://www.nature.com/scitable/definition/phenotype-phenotypes-35, 2019. Accessed: 2019-05-22.

[2] A. Ahmed, A. Das, and A. J. Smola. Scalable hierarchical multitask learning algorithms for conversion optimization in display advertising. In *Proceedings of the 7th ACM international conference on web search and data mining*, pages 153–162. ACM, 2014.

[3] R. Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.

[4] K. Donnelly. SNOMED-CT: The advanced terminology and coding system for e-health. *Studies in health technology and informatics*, 121:279, 2006.

[5] D. Eigen, M. Ranzato, and I. Sutskever. Learning factored representations in a deep mixture of experts. *arXiv preprint arXiv:1312.4314*, 2013.

[6] A. Elixhauser, C. Steiner, and L. Palmer. Clinical classifications software (CCS). *Book Clinical Classifications Software (CCS)(Editor edˆ eds)*, 2014.

[7] H. Harutyunyan, H. Khachatrian, D. C. Kale, and A. Galstyan. Multitask learning and benchmarking with clinical time series data. In *ML4H: Machine Learning for Health workshop, held at Advances in Neural Information Processing Systems*, 2017.

[8] J. He and Y. Zhu. Hierarchical multi-task learning with application to wafer quality prediction. In *2012 IEEE 12th International Conference on Data Mining*, pages 290–298. IEEE, 2012.

[9] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, G. E. Hinton, et al. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

[10] A. E. Johnson, T. J. Pollard, L. Shen, H. L. Li-wei, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. A. Celi, and R. G. Mark. MIMIC-III, a freely accessible critical care database. *Scientific data*, 3:160035, 2016.

[11] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[12] J. Ma, Z. Zhao, X. Yi, J. Chen, L. Hong, and E. H. Chi. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1930–1939. ACM, 2018.

[13] E. Meyerson and R. Miikkulainen. Beyond shared hierarchies: Deep multitask learning through soft layer ordering. In *6th International Conference on Learning Representations (ICLR)*, 2018.

[14] R. Sanabria and F. Metze. Hierarchical multi task learning with CTC. In *Workshop On Spoken Language Technology, held at IEEE SLT*. IEEE, 2018.

[15] V. Sanh, T. Wolf, and S. Ruder. A hierarchical multi-task approach for learning embeddings from semantic tasks. In *AAAI*, 2019.

[16] J. Sun et. al. Combining knowledge and data driven insights for identifying risk factors using electronic health records. In *AMIA Annual Symposium Proceedings*, volume 2012, page 901. American Medical Informatics Association, 2012.

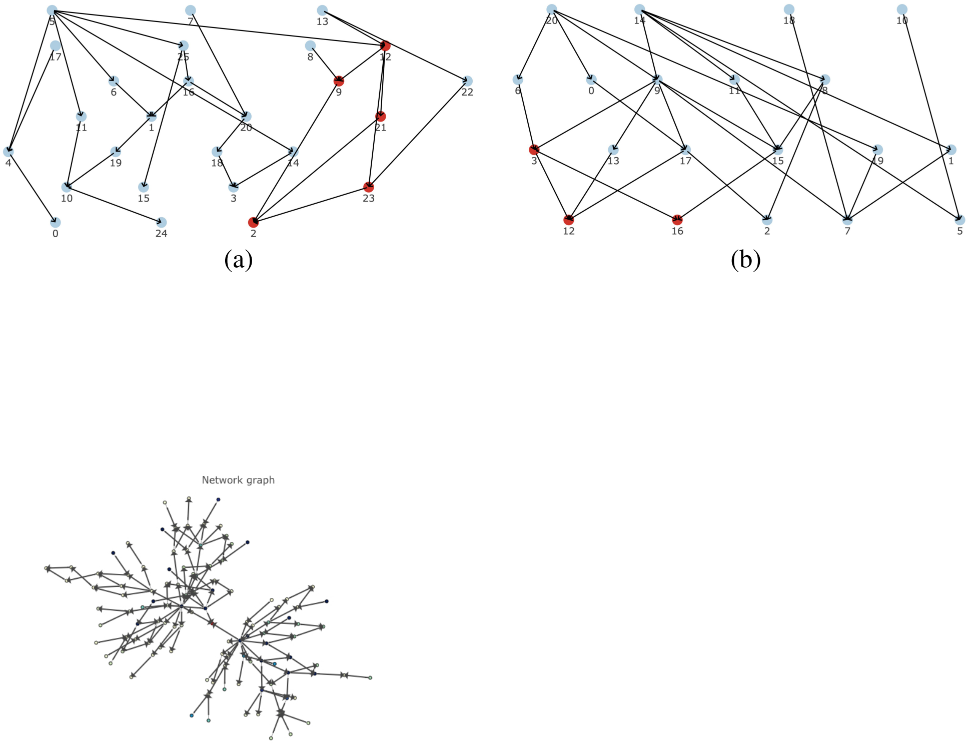
[17] N. Tomašev, X. Glorot, J. W. Rae, M. Zielinski, H. Askham, A. Saraiva, A. Mottram, C. Meyer, S. Ravuri, I. Protsyuk, et al. A clinically applicable approach to continuous prediction of future acute kidney injury. *Nature*, 572(7767):116, 2019.

[18] X. Wang, P. Kapanipathi, R. Musa, M. Yu, K. Talamadupula, I. Abdelaziz, M. Chang, A. Fokoue, B. Makni, N. Mattei, et al. Improving natural language inference using external knowledge in the science questions domain. *arXiv preprint arXiv:1809.05724*, 2018.

[19] C. Xiao, T. Ma, A. B. Dieng, D. M. Blei, and F. Wang. Readmission prediction via deep contextual embedding of clinical concepts. *PloS one*, 13(4):e0195024, 2018.

[20] J. Zhang, K. Kowsari, J. H. Harrison, J. M. Lobo, and L. E. Barnes. Patient2vec: A personalized interpretable deep representation of the longitudinal electronic health record. *IEEE Access*, 6: 65333–65346, 2018. ISSN 2169-3536. doi: 10.1109/ACCESS.2018.2875677.

[21] A. Zweig and D. Weinshall. Hierarchical multi-task learning: a cascade approach based on the notion of task relatedness. In *Theoretically Grounded Transfer Learning workshop, held at International Conference on Machine Learning (ICML)*, 2013.



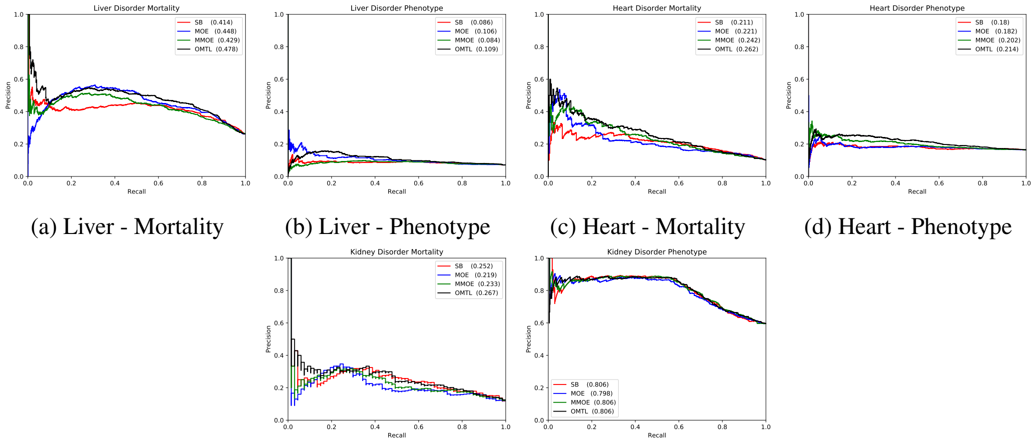
(a) (b)

(c)

Figure 4: (a) Liver disorder graph. (b) Heart disorder graph. (c) Kidney disorder graph. Core nodes are shown in red, and augmented nodes are shown in blue. The labeled indices are used to describe the details of each node in the supplementary materials.

A Data

B Precision-recall curve



(a) Liver - Mortality (b) Liver - Phenotype (c) Heart - Mortality (d) Heart - Phenotype

(e) Kidney - Mortality (f) Kidney - Phenotype

Figure 5: PRC for all experiments.