Deep Patient Representation of Clinical Notes via Multi-Task Learning for Mortality Prediction

Yuqi Si, MS[1], Kirk Roberts, PhD[1]
[1]School of Biomedical Informatics,
The University of Texas Health Science Center at Houston
Houston, TX, USA

Abstract

We propose a deep learning-based multi-task learning (MTL) architecture focusing on patient mortality predictions from clinical notes. The MTL framework enables the model to learn a patient representation that generalizes to a variety of clinical prediction tasks. Moreover, we demonstrate how MTL enables small but consistent gains on a single classification task (e.g., in-hospital mortality prediction) simply by incorporating related tasks (e.g., 30-day and 1-year mortality prediction) into the MTL framework. To accomplish this, we utilize a multi-level Convolutional Neural Network (CNN) associated with a MTL loss component. The model is evaluated with 3, 5, and 20 tasks and is consistently able to produce a higher-performing model than a single-task learning (STL) classifier. We further discuss the effect of the multi-task model on other clinical outcomes of interest, including being able to produce high-quality representations that can be utilized to great effect by simpler models. Overall, this study demonstrates the efficiency and generalizability of MTL across tasks that STL fails to leverage.

Introduction

A significant amount of patient information is documented in unstructured data within Electronic Health Records (EHRs) such as discharge summaries, lab reports, radiology reports, and nursing notes. Many efforts via machine learning approaches have focused on mining and extracting patient information from these document resources to provide meaningful encoded representations. Such approaches, however, largely focus on just a single task and fail to realize the possible shared information between related tasks. While a good representation for a specific task does not automatically imply good performance on similar tasks, the sharing of information between models could result in performance improvements for all tasks. Despite this, there has been relatively few studies attempting to utilize multiple tasks on clinical notes.

Multi-task learning (MTL) is a subfield of machine learning in which multiple related tasks are trained simultaneously and the learned parameters are partly shared[1]. By sharing information between related tasks, MTL improves the generalization and performance of the model by leveraging the hidden information of related tasks, when compared to training individual tasks. MTL intuitively makes sense due to its power of helping the model focus attention and acting as a regularization by introducing an inductive bias[1]. Various MTL approaches differ in terms of model structure, including optimization techniques, and levels of information sharing (see Figure 1).

In this study, we propose a deep learning-based MTL approach that encodes a common patient representation by leveraging clinical note information on multiple mortality prediction tasks. Our method falls into the MTL type of *joint training* where different predictions are regarded as different tasks and loss functions of individual tasks are optimized simultaneously. We investigate the method on tasks related to patient mortality and length of stay prediction using data from the public MIMIC-III[2] intensive care database. Specifically, we adopt a multi-level Convolutional Neural Network (CNN)[3] to train the shared representation from clinical notes jointly on all prediction tasks. Then we use the representation to predict additional tasks which are achieved by a simple neural network model with single dense layer. Our ultimate goal is to build a general-purpose patient representation by incorporating multiple resources and predicting interesting clinical outcomes. This representation will be an effective tool to support a variety of clinical research problems from quality care improvement (e.g., readmission prediction) to clinical prediction (e.g., mortality prediction, early diagnosis detection).

The remainder of this paper is structured as follows. First, we describe the relevant prior work, notably MTL applications in both artificial intelligence and biomedicine. Next, we introduce the model with its objective function. Then we describe our experiments on mortality prediction. After that, we present our results by empirically evaluating the model. Finally, we conclude with a discussion, including the limitations of the method and directions for future work.

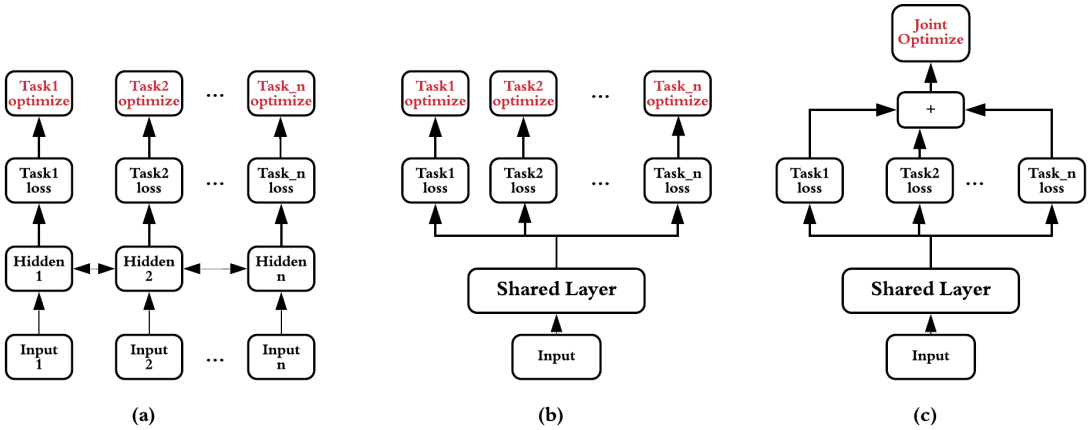


Figure 1: MTL architecture. a: Soft Parameter Sharing Training. b: Alternate Training. c: Joint Training

Background

With the emergence of deep learning, there has been renewed interest in utilizing MTL to improve learning efficiency and prediction accuracy. A recent study by Zhang and Yang[4] detailed the theoretical foundations and future trends of MTL. Ruder[5] presented a comprehensive overview of MTL with deep neural networks and concluded that deep learning largely speeds up the computation and enhances the chances of achieving MTL. Numerous studies in artificial intelligence have successfully attempted to apply MTL to existing tasks, including computer vision[6–10], natural language processing[11–16], and speech recognition[17]. It is now well-established that MTL can enhance learning efficiency if the model is able to selectively share information in a manner that avoids negative effects between related tasks[18]. Even when confronting such challenges, MTL tends to provide insights into tasks, and it has been further applied to biomedical settings where the main goal is scientific discovery, e.g., biological functions[19,20] and drug discoveries[21].

Even without MTL, researchers have demonstrated the competence of multilabel disease modeling on EHR data[22–24]. Currently, multi-task EHR learning has been utilized as a strategy on clinical events where certain model parameters can be shared and certain parameters can be specialized. Those events pose a challenging problem in medicine due to the complexity of associated conditions and multiple modalities of data from heterogeneous sources[25]. Futoma et al.[26,27] proposed Multitask Gaussian Process (MGP) Recurrent Neural Network (RNN) classifier to detect early sepsis with physiological variables including vitals and laboratory values. Nagpal[28] concatenated clinical report word embedding with ICD-9 code embedding as the input towards a late fusion multi-task network to predict certain clinical conditions, which outperformed simple feature representation. Harutyunyan et al.[29] formulated a heterogeneous MTL architecture using time series variables to predict four benchmark clinical tasks including in-hospital mortality, decompensation, forecasting length of stay and phenotype classification. Ngufor et al.[30,31] explored strategies of how to cluster similar tasks in MTL to enhance cross-transfer of shared knowledge by predicting blood transfusion procedure outcomes. Razavian at al.[32] compared three MTL neural networks including two CNN variants and one Long Short Term Memory network (LSTM) variant with single task learning (STL) baselines to predict the onset of chronic kidney disease based solely on longitudinal lab test values. Wiens et al.[33] adapted MTL to learn models for patient risk stratification where different patient populations are considered as related tasks. Wang et al.[34] proposed a MTL algorithm for joint disease onset prediction using ICD-9 codes, which outperformed STL models. Nori et al.[35] achieved higher performance of several variants of multi-task models than single task models in predicting ICU patient mortality using variables from patient demographic information and ICD-10 codes. Lopez-Martinez et al.[36,37] made use of physiological signals such as skin conductance and electrocardiogram results and implemented a MTL technique to leverage information across patient populations in pain recognition predictions. Overall, these studies highlight the trend for adopting MTL based on lab tests or physiological variables in clinical settings.

Meanwhile, several studies have developed patient representations from free-text clinical data. Sushil et al.[38] learned patient representations using unsupervised methods and evaluated the representation in multiple supervised setups including patient gender, mortality, diagnostic, and procedural categories. Dubois et al.[39] generated patient representations from a source task of visit diagnosis prediction and used target tasks to evaluate. However, little research has

investigated the usage of MTL to develop the patient representation itself. To the best of our knowledge, this is the first study that explicitly focuses on building an encoded patient vector representation from publicly accessible clinical notes via multi-task deep learning.

Method

1 Data and Prediction settings

Following the preprocessing steps in Grnarova et al.[3], we retrieve clinical notes associated with patient metadata from MIMIC-III[2]. We remove neonates and patients with more than one hospital admission, then exclude the discharge summary and all notes written post-discharge. Each patient with a single admission has multiple clinical notes and each note has a category that indicates the specific type such as "ECG" or "Nursing". We tokenize the notes using regular expressions, keeping only the 300K most frequent words and then pre-train word embeddings for the network input. The word embeddings are trained using the Continuous Bag-of-Words (CBOW) model with gensim[40]. Additionally, the category of the clinical note is concatenated with the sentence representations (following Grnarova et al.[3]) as the input of a sentence-level network where the initial category embedding is randomly assigned but dynamically altered during training.

We focus on two issues of common concern in ICU prediction scenarios: patient mortality prediction and forecasting length of stay. We use these issues as either source tasks to learn the representation or target tasks to evaluate the representation. Mortality predictions includes in-hospital mortality and mortality at certain time post-discharge. For in-hospital mortality, the death time is always the same as the discharge time. For mortality post-discharge, since MIMIC-III uses the social security master death index[2], we are able to calculate the exact death date for post-discharge mortality. Following these steps, we avoid the use of any future information to predict the current situations and derive the ground-truth label of each task from structured data of MIMIC-III[2].

2 Network Architecture

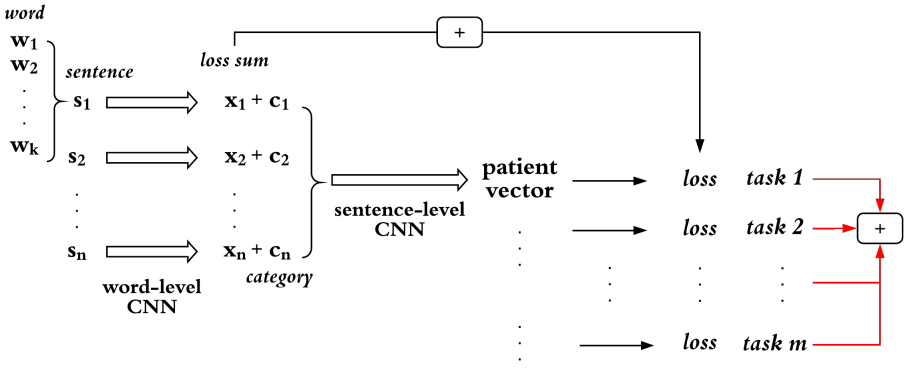


Figure 2: Model architecture overview

The basic structure of the model for MTL training on source task is a Convolutional Neural Network (CNN). We propose an architecture that includes two levels: a word-level CNN and a sentence-level CNN (hereafter, referred to in combination as a multi-level CNN). As shown in Figure 2, word embeddings are first aggregated into a sentence representation (denoted as the word-level CNN), then sentence representations are used to generate a single patient representation (denoted as the sentence-level CNN). We concatenate multiple notes of one patient into one document, thus the model does not consider any temporal information. To consider the relations between sentences, we add the loss of each sentence derived from the word-level CNN with the loss function of the sentence-level CNN (known as target replication[3]). For each task, we apply a softmax function after the final layer and each task is optimized for cross entropy loss. To achieve joint multi-task learning, we sum up losses from all the tasks and optimize the final loss. The loss function the MTL task is defined as follows:

$$\mathcal{L} = \sum_{j=1}^{m} [\, l_j(y, y^*) + \frac{\lambda}{n} \sum_{i=1}^{n} l_i(y, y^*)] \,, y \in [0, 1] \; y^* \in \{0, 1\} \tag{1}$$

where *m* represents the task number, *n* represents the number of sentences in a single note, *y* is the final probability output of the neural network while y^* is the gold standard label, λ is the parameter determines the strength of target replication. $l_i(y, y^*)$ and $l_j(y, y^*)$ represents respectively the losses from the i^{th} sentence representation and the patient representation in the j^{th} task.

In terms of the model structure for target task, we adopt a neural network with single dense layer. The input of this network is the pre-trained patient vector-based representation. After a matrix computation on the input vector, the network outputs a binary classification using softmax function with the optimization of cross entropy loss. This model can save a great deal of time and effort compared with the CNN model in the source task.

3 Experiment and Evaluation

We pretrained 100-dimension word embeddings for MIMIC-III clinical notes. The word-level CNN uses 50 convolutional filters each of sizes [3, 4, 5] with stride of 1 and valid padding. The initial category embedding had 10 dimensions and the parameter of target replication λ was set to 5. The sentence-level CNN uses 50 filters each of size 3 with the same stride and padding format as the word-level CNN. We applied a dropout probability of 0.8 after max-pooling layer to prevent overfitting. All tasks use the cross entropy loss function after the softmax layer along with the Adam optimizer (batch size: 64 patients; learning rate: 0.01; decay: 0.99). The neural network was implemented in Tensorflow[41] on an NVidia Tesla GPU with the cuDNN library. Our code is publicly available at `https://github.com/Yuqi92/deep-patient-representation-mimiciii-multitask`.

The entire dataset after preprocessing was split into training, testing, and development (dev) sets with a ratio of 8:1:1, respectively. We trained the model on the training set, evaluated the trained model on the dev set for early stopping before overfitting, and evaluated the final performance on the test set. The descriptive statistics of data information is shown in Table 1. Table 1(a) shows the partition of patient data into the respective subsets. As observed from Table 1(b), data is imbalanced with a large number of negative samples, thus we use the area under the ROC curve (AUROC) score as the evaluation metric as it is insensitive to the class distribution.

Table 1: Descriptive statistics of dataset

(a) Partitioning of patient data

	# Patients	# Notes
train	30,668	969,574
dev	3,833	125,559
test	3,833	122,502
total	**38,334**	**1,217,635**

(b) Positive-class distribution

	# Patients	Overall %
died in hospital	4,063	10.60
died in 30 days after discharge	1,202	3.14
died in 1 year after discharge	2,560	6.68

As the clinical outcome of interest, mortality predictions (including in-hospital mortality and mortality at certain days post-discharge) were extracted and trained as source tasks. We selected 3 common tasks (in-hospital, 30-day and 1-year mortality) to constitute the multi-task model in the first experiment, which we refer to as the 3-task model. The entire feature representations were fed into the multi-level CNN network, trained with shared hyperparameters, then split to each individual task, and finally returned a binary classification output for each task. Further, in the second and third experiment, we respectively implemented a 5-task and a 20-task model with the same training process. To create 5- and 20-task datasets, we split the distribution of patient mortality days into roughly equal-sized groups of 5-quantiles and 20-quantiles, respectively. This resulted in the 5-task model consisting of [in-hospital, 30-day, 3-month, 1-year, 3-year] single-task models, and the 20-task model consisting of [in-hospital, 5-, 14-, 30-, 43-, 68-, 103-, 142-, 196-, 269-, 366-, 453-, 573-, 711-, 893-, 1092-, 1342-, 1626-, 1997-, 2548-day] single-task models. We also trained a separate model for each individual task, namely, in-hospital, 30-day and 1-year mortality prediction. Lastly, we

reported and compared the performance of MTL and STL models.

In addition, we extracted 50-dimension patient representations from above multi-task models, and evaluated the representations on a target task for predicting patient mortality in 60 days. The target task prediction uses a neural network model with single dense layer, which is vastly simpler than the multi-level CNN presented above (in both input and structure) and thus must rely heavily on the given patient representation for accurate classification. We further apply t-SNE visualization on the patient representation trained by MTL models. This representation visualization contains all the patient samples from test subset and each representation associates with its label. The label is referred to the patient mortality date and stratified into different subgroups based on the design of MTL models.

Results

We report the performances comparing the single-task model with three multi-task models on different scenarios. As is shown in Table 2, at least one multi-task model outperforms the single-task model in each case (scores in bold). The gains are small but generally consistent with known trends in MTL. Specifically, the 3-task model achieved the best performance on in-hospital and 30-day mortality predictions with AUC scores of 94.57% and 93.24% respectively. The 20-task model performs the best on the task of 1-year mortality prediction with an AUC score of 90.59%. By incorporating 30-day and 1-year predictions, the performance of in-hospital mortality rises by 0.67% with a statistically significant difference on 95% confidence level. Similar improvements respectively appear on the 30-day and 1-year predictions by 0.19% and 0.20%. Accordingly, MTL does not negatively affect learning efficiency and achieves a well-matched performance among all tasks, which is in line with our assumption that since the tasks are relevant, the multi-task model enables the generalizability by sharing a robust intermediate representation.

Table 2: Performance comparison of MTL and STL models

	AUROC (%) of different models on each task			
Task	single-task	3-task	5-task	20-task
In-hospital	93.90[a]	**94.57**	94.07	93.41
30-day	93.05[b]	**93.24**	93.07	92.35
1-year	90.39[c]	89.58	90.56	**90.59**

[a] 95% Confidence Interval: [93.63 - 94.16]
[b] 95% Confidence Interval: [92.76 - 93.33]
[c] 95% Confidence Interval: [90.06 - 90.71]

MTL models have the potential to learn more generalized representations that may be of use for additional tasks beyond those they are trained on. To explore this, we extract three patient vector representations from the 3-task, 5-task and 20-task models and evaluate these representations on the target task of 60-day mortality prediction. We also train a single-task model on 60-day mortality and compare the performance of pre-trained representations with the single-task model on the same task. The result is shown in Table 3. We observe that the efficiency of 5-task patient representation outperforms other approaches with an AUC score of 92.42%. Overall, the performances of multi-task models on the target task are well-matched with the single-task model, which is consistent with the results of source tasks. Further, the total training time is dramatically reduced while the representations still achieving comparable performances. Training a complex STL model for 60-day mortality takes almost 20 hours, while using pre-trained patient representation only takes less than 10 minutes owing to the simplicity of the single dense layer model. In this way, we show the high efficiency of general-purpose patient representation.

We produced t-SNE visualization on the representations to explore the patient distribution in the embedding space. Due to the page limit, we only display one visualization in Figure 3, which is extracted from the patient representation of 3-task model. An obvious trend of the patient distribution reveals the majority of patients cluster into several individual clouds. Patients who died in the hospital form into several distinct groups which are–not unexpectedly–similar to patients that died within 30 days of discharge. Patients who died within 1 year are similar to those that survived for at least a year (again, not surprising). We note that since the vast majority of patients lived at least 1 year, this data imbalance results in such patients being dispersed throughout the embedding space.

Table 3: Evaluation of patient representations on target task

2	**AUROC (%) on 60-day task**	**Total time of training**
single-task	92.32[a]	18.45 hours
3-task patient vector	91.97	3.53 mins
5-task patient vector	**92.42**	4.24 mins
20-task patient vector	92.12	5.12 mins

[a] 95% Confidence Interval: [92.02 - 92.61]

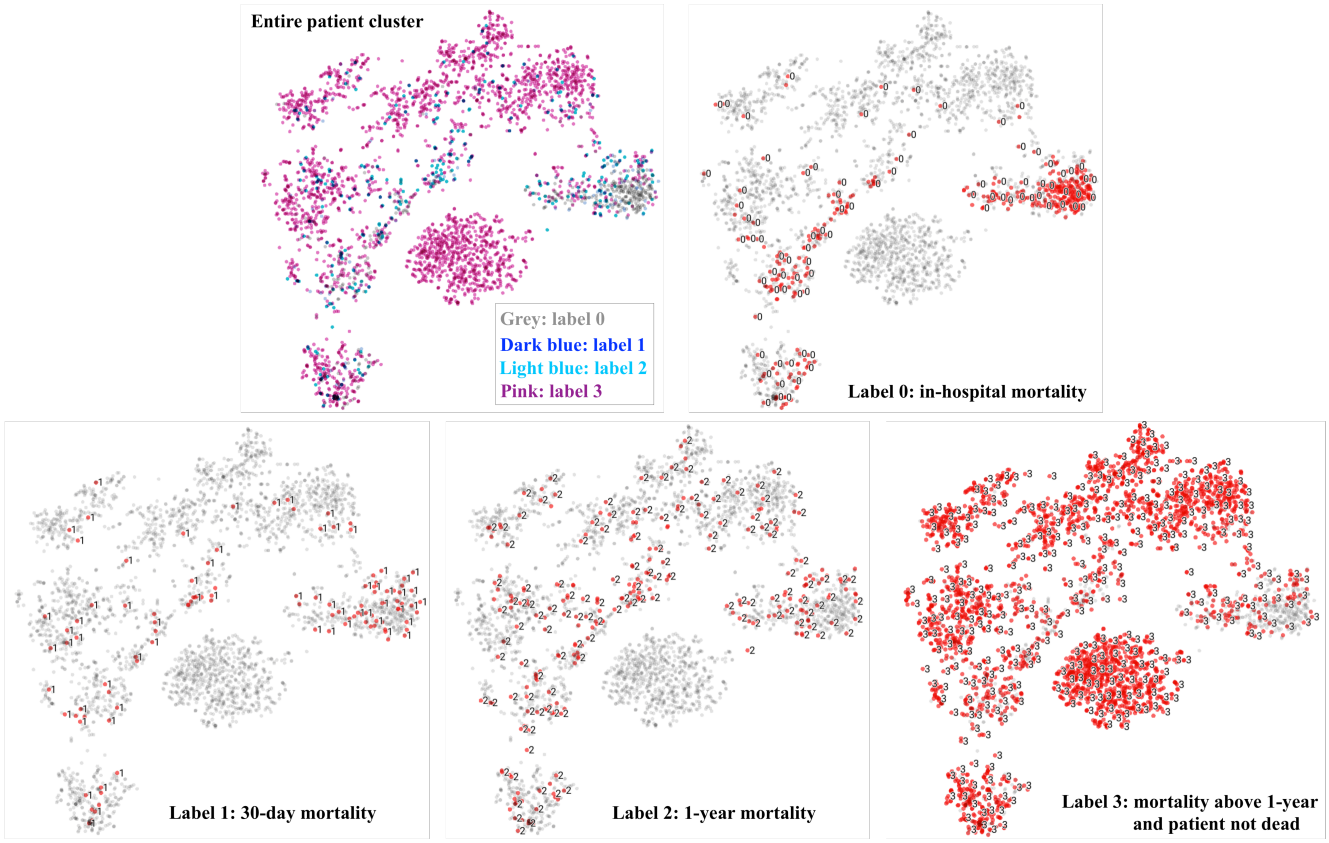


Figure 3: tSNE visualization of 3-task patient distributed representation.
Label represents: 0: in-hospital mortality, 1: 30-day mortality, 2: 1-year mortality, 3: mortality above 1-year or patient not dead

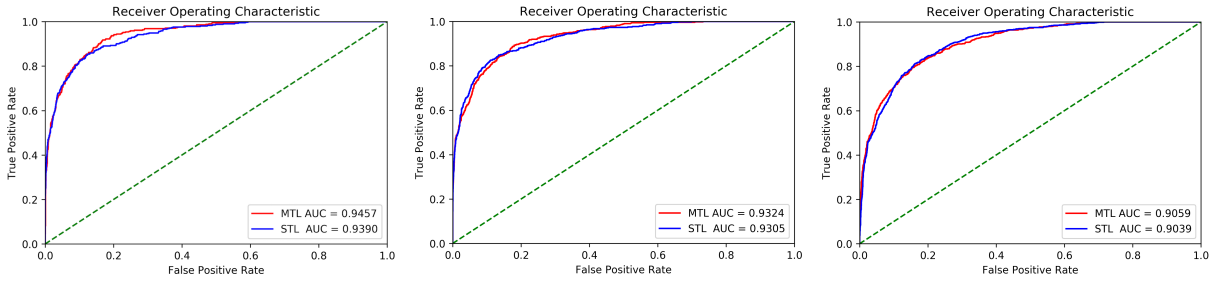
Discussion

Our network structure is similar to Grnarova et al.[3] in terms of mortality prediction, though they focus on single-task learning. They have also reported the AUROC scores of 0.963, 0.858 and 0.853 for in-hospital, 30-day, 1-year mortality predictions respectively, compared to our STL scores of 0.939, 0.931, and 0.904 (Table 2). A direct comparison of STL model performance is difficult, as we do not have the access to their preprocessing details and exact train/test splits. We also note that they have excluded patients that died in the hospital when it comes to 30-day and 1-year mortality. As single-task approaches, the three models were separately trained and thus failed to make use of the potential correlations between three tasks. In another similar work to ours, Sushil et al[38] utilized unsupervised method such as bag-of-word (BoW), doc2vec[42], and stacked denoising autoencoder (SDAE) to learn patient representations from clinical notes and evaluated on multiple predictions. They presented the best models associated with AUROC scores of 0.9457 on in-hospital mortality using BoW, 0.8113 on 30-day mortality, and 0.8302 on 1-year mortality using

doc2vec with SDAE-BoW. However, we assume the representations with solely unsupervised learning would lose information, and we argue for using multi-task supervised learning to encode as much medically-relevant information as the representation enables. The most similar experiment to ours is Dubois et al[39], who applied MTL and extracted clinical note features to make phenotype predictions using several RNN variant models. Although they incorporated all diagnosis codes as the multi-task supervision, they chose six prevalent diagnosis codes as the target tasks to evaluate. It is not clear whether the improvement is statistically significant when they transfer the representation to other tasks.

In this study, we provide insights into the feasibility of learning effective patient representations solely from clinical notes based on supervised multi-task deep learning. We employ a MTL neural network architecture to predict interesting clinical problems in the ICU. This architecture contains a unified word-level and sentence-level CNN with a joint loss function that sums the individual task losses. We obtain the encoded patient representation from the pre-trained neural network and use the representation to predict patient mortality of certain days with a neural network with single dense layer. We report the performance of those models to demonstrate the efficiency of MTL and present the t-SNE visualization of the patient representation. It has been shown that the performances of proposed multi-task models slightly outperform that of single-task, which means the multi-task models are able to extract meaningful information from the clinical note that single-task models fails to leverage. Meanwhile multi-task models enlarge the generalizability of the trained patient representation. The 20-task MTL model even transfers the classification into a roughly regression-like model due to its division of patient mortality into 20 stratified subgroups ranging from in-hospital to the death maximum date. Our pre-trained patient representation is convenient for other researchers who wish to develop and build machine learning models for patients upon our work since the dataset in this experiment is directly from a freely-accessible database. The patient representation can be fed into the input of a lightweight machine learning model with small size of parameters and acquire a decent performance score (e.g., logistic regression).

We reviewed the samples of test set to check whether they reveal any systematic trend and clinical findings. We found that MTL models, relatively speaking tend to recognize negative samples (e.g., patients who were not dead) while failing to identify positive ones (e.g., patient who died). Fortunately, these multi-task models have higher positive predictive values, which means that one can be more confident in samples labeled as positive. This finding is consistent with the trend shown in the ROC curve of in-hospital mortality (Figure 4(a)).



(a) In hospital mortality **(b)** 30 day mortality **(c)** One year mortality

Figure 4: ROC curves

Apart from the capability of using the patient representation to predict mortality, we further explore how the representation performs on other tasks. Following the approach described in Method section, we first train a 3-task binary classification model containing 30-day, 1-year mortality and 6-day length-of-stay (LOS) prediction. Considering the differences between these two kinds of tasks (that the mortality prediction is a classification problem and LOS forecasting is normally a regression task), we differentiate the model for this experiment from the previous multi-level CNN network by adding another fully-connected layer ahead of the softmax output. Thus, the tasks are trained on two fully-connected layers after the maxpooling of the sentence-level CNN. This difference is intended to distinguish the characteristics of individual tasks since the patient mortality and length-of-stay are not entirely related. Similarly, we report the performance of this 3-task model based on AUROC score in Table 4(a). We also evaluate the pre-trained patient representation from the above tasks to predict the classification of 14-day LOS using the neural network model with one dense layer and the result is shown in Table 4(b). It is surprisingly observed that the patient vector extracted from the 3-task model of 30-day, 1-year mortality and 6-day LOS reached out the best performance of an AUROC

score of 90.39% on the target task of 14-day LOS prediction. By incorporating mortality information, the performance of 3-task model on the target task rises by 1.39% over the single-task model with a statistically significant difference on 99% confidence level. Besides, patient vector encoded with both mortality and LOS information (3-task[c]: 90.39%) significantly improved the performance of LOS prediction comparing with the vector encoded with only mortality information (3-task[b]: 74.71%). In this manner, the generalizability of the multi-task representation can be achieved among different clinical outcomes.

Table 4: Performance evaluation of MTL model on patient mortality and LOS task

(a) Performance of 3-task model on each task

	AUROC(%) on each task
30-day	92.91
1-year	90.85
6-day LOS task	88.61

(b) Pre-trained representation on 14-day LOS task

	AUROC(%) on 14-day LOS task
single-task	89.00[a]
3-task[b] patient vector	74.71
3-task[c] patient vector	**90.39**
5-task patient vector	74.41
20-task patient vector	74.91

[a] 99% Confidence Interval: [88.58 - 89.41]
[b] tasks of in-hospital, 30-day, 1-year mortality
[c] tasks of 30-day, 1-year mortality, 6-day LOS

A limitation of this study is that we currently limit our predictions to solely mortality and length-of-stay problems. However, our work establishes the foundation path for further research in multi-task supervised learning of clinical data. In the future, we intend to incorporate multiple modalities of data, including structured information such as observational variables, to contribute the efficiency of representation learning. As for the prediction tasks, it would be interesting to consider specific phenotypes or other clinical outcomes. We hope this would not merely improve the performance on the given prediction task but also bring forward new research insights for patient treatment. Those complex problems are more challenging in that they need complicated model architectures to selectively share information. To achieve this, neural networks like Bi-LSTM can be used to emphasize the long-term dependencies and correlations into multi-task predictions. In addition, we intend to further investigate the challenges and opportunities of different MTL architectures. This includes applying attention mechanisms and learning attention weights across multiple tasks to avoid negative impacts and make the most of shared information. Another future direction on MTL architecture would be the application of fusion techniques[43]. Specifically, the late fusion approach concatenates the attention weights across tasks. Based on different requirements, the weight coefficient of each individual task can be set uniformly, proportional to one task, or tuned with a classifier. However, this manner triggers the trade-off between computing complexity and model performance.

Conclusion

In this study, we design a MTL architecture based on a multi-level CNN to learn mortality-focused patient representations from publicly available clinical notes. We apply different clinical outcomes of interest within the MTL framework. The initial experiments show promising results, with AUROC scores of 3-task model up to 94.57% for in-hospital mortality and 93.24% for 30-day mortality, and that of 20-task model up to 90.59% for 1-year. We also evaluate the patient representation's impact on a target task of 60-day mortality and as a result, 5-task patient representation achieves the best AUROC score of 92.42%. It has been shown that learning across multiple tasks tends to contribute to each other and leverage hidden information. Our primary goal of this research is to demonstrate the feasibility of utilizing MTL to efficiently handle patient clinical notes and to obtain a vector-based patient representation across multiple predictions. Our ultimate goal is to build a comprehensive patient representation by incorporating the information from heterogeneous resources and multiple clinical outcomes.

Acknowledgement

This work was supported by the U.S. National Library of Medicine, National Institutes of Health (NIH), under award R00LM012104, as well as the Cancer Prevention Research Institute of Texas (CPRIT), under award RP170668.

References

1. Caruana R. Multitask learning. *Machine Learning*. 1997 Jul 1;28(1):41-75.
2. Johnson AE, Pollard TJ, Shen L, Li-wei HL, Feng M, Ghassemi M, Moody B, Szolovits P, Celi LA, Mark RG. MIMIC-III, a freely accessible critical care database. *Scientific Data*. 2016 May 24;3:160035.
3. Grnarova P, Schmidt F, Hyland SL, Eickhoff C. Neural document embeddings for intensive care patient mortality prediction. *arXiv preprint arXiv:1612.00467*. 2016 Dec 1.
4. Zhang Y, Yang Q. A survey on multi-task learning. *arXiv preprint arXiv:1707.08114*. 2017 Jul 25.
5. Ruder S. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*. 2017.
6. Li S, Liu ZQ, Chan AB. Heterogeneous multi-task learning for human pose estimation with deep convolutional neural network. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops 2014* (pp. 482-489).
7. Elhoseiny M, El-Gaaly T, Bakry A, Elgammal A. A comparative analysis and study of multiview CNN models for joint object categorization and pose estimation. *International Conference on Machine Learning* 2016 Jun 11 (pp. 888-897).
8. Chowdhuri S, Pankaj T, Zipser K. Multi-Modal Multi-Task Deep Learning for Autonomous Driving. *arXiv preprint arXiv:1709.05581*. 2017 Sep 16.
9. Cao L, Li L, Zheng J, Fan X, Yin F, Shen H, Zhang J. Multi-task neural networks for joint hippocampus segmentation and clinical score regression. *Multimedia Tools and Applications*. 2018:1-8.
10. Pasunuru R, Bansal M. Multi-task video captioning with video and entailment generation. *arXiv preprint arXiv:1704.07489*. 2017 Apr 24.
11. Collobert R, Weston J. A unified architecture for natural language processing: Deep neural networks with multitask learning. *Proceedings of the 25th International Conference on Machine Learning* 2008 Jul 5 (pp. 160-167). ACM.
12. Dong D, Wu H, He W, Yu D, Wang H. Multi-task learning for multiple language translation. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing* (Volume 1: Long Papers) 2015 (Vol. 1, pp. 1723-1732).
13. Rei M. Semi-supervised multitask learning for sequence labeling. *arXiv preprint arXiv:1704.07156*. 2017 Apr 24.
14. Liu P, Qiu X, Huang X. Adversarial multi-task learning for text classification. *arXiv preprint arXiv:1704.05742*. 2017 Apr 19.
15. Crichton G, Pyysalo S, Chiu B, Korhonen A. A neural network multi-task learning approach to biomedical named entity recognition. *BMC Bioinformatics*. 2017 Dec;18(1):368.
16. McCann B, Keskar NS, Xiong C, Socher R. The Natural Language Decathlon: Multitask Learning as Question Answering. *arXiv preprint arXiv:1806.08730*. 2018 Jun 20.
17. Toshniwal S, Tang H, Lu L, Livescu K. Multitask learning with low-level auxiliary tasks for encoder-decoder based speech recognition. *arXiv preprint arXiv:1704.01631*. 2017 Apr 5.
18. Kumar A, Daume III H. Learning task grouping and overlap in multi-task learning. *arXiv preprint arXiv:1206.6417*. 2012 Jun 27.
19. Yang X, Kim S, Xing EP. Heterogeneous multitask learning with joint sparsity constraints. *Advances in Neural Information Processing Systems* 2009 (pp. 2151-2159).
20. Fa R, Cozzetto D, Wan C, Jones DT. Predicting human protein function with multi-task deep neural networks. *PLoS ONE*. 2018 Jun 11;13(6):e0198216.
21. Ramsundar B, Kearnes S, Riley P, Webster D, Konerding D, Pande V. Massively multitask networks for drug discovery. *arXiv preprint arXiv:1502.02072*. 2015 Feb 6.

22. Nori N, Kashima H, Yamashita K, Ikai H, Imanaka Y. Simultaneous modeling of multiple diseases for mortality prediction in acute hospital care. *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 2015 Aug 10 (pp. 855-864). ACM.
23. Lipton ZC, Kale DC, Elkan C, Wetzel R. Learning to diagnose with LSTM recurrent neural networks. *arXiv preprint arXiv:1511.03677*. 2015 Nov 11.
24. Che Z, Kale D, Li W, Bahadori MT, Liu Y. Deep computational phenotyping. *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 2015 Aug 10 (pp. 507-516). ACM.
25. Caruana R, Baluja S, Mitchell T. Using the future to "sort out" the present: Rankprop and multitask learning for medical risk evaluation. *Advances in Neural Information Processing Systems* 1996 (pp. 959-965).
26. Futoma J, Hariharan S, Heller K. Learning to detect sepsis with a multitask gaussian process RNN classifier. *arXiv preprint arXiv:1706.04152*. 2017 Jun 13.
27. Futoma J, Hariharan S, Sendak M, Brajer N, Clement M, Bedoya A, O'Brien C, Heller K. An improved multi-output gaussian process rnn with real-time validation for early sepsis detection. *arXiv preprint arXiv:1708.05894*. 2017 Aug 19.
28. Nagpal C. Deep Multimodal Fusion of Health Records and Notes for Multitask Clinical Event Prediction.
29. Harutyunyan H, Khachatrian H, Kale DC, Galstyan A. Multitask learning and benchmarking with clinical time series data. *arXiv preprint arXiv:1703.07771*. 2017 Mar 22.
30. Ngufor C, Upadhyaya S, Murphree D, Madde N, Kor D, Pathak J. A heterogeneous multi-task learning for predicting RBC transfusion and perioperative outcomes. *Conference on Artificial Intelligence in Medicine in Europe* 2015 Jun 17 (pp. 287-297). Springer, Cham.
31. Ngufor C, Upadhyaya S, Murphree D, Kor D, Pathak J. Multi-task learning with selective cross-task transfer for predicting bleeding and other important patient outcomes. *Data Science and Advanced Analytics (DSAA)*, 2015. 36678 2015. IEEE International Conference on 2015 Oct 19 (pp. 1-8). IEEE.
32. Razavian N, Marcus J, Sontag D. Multi-task prediction of disease onsets from longitudinal laboratory tests. *Machine Learning for Healthcare Conference* 2016 Dec 10 (pp. 73-100).
33. Wiens J, Guttag J, Horvitz E. Patient risk stratification with time-varying parameters: a multitask learning approach. *The Journal of Machine Learning Research*. 2016 Jan 1;17(1):2797-819.
34. Wang X, Wang F, Hu J, Sorrentino R. Exploring joint disease risk prediction. In *AMIA Annual Symposium Proceedings* 2014 (Vol. 2014, p. 1180). American Medical Informatics Association.
35. Nori N, Kashima H, Yamashita K, Kunisawa S, Imanaka Y. Learning Implicit Tasks for Patient-Specific Risk Modeling in ICU. *AAAI* 2017 Feb 12 (pp. 1481-1487).
36. Lopez-Martinez D, Picard R. Multi-task neural networks for personalized pain recognition from physiological signals. *Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW), 2017 Seventh International Conference* on 2017 Oct 23 (pp. 181-184). IEEE.
37. Lopez-Martinez D, Rudovic O, Picard R. Physiological and behavioral profiling for nociceptive pain estimation using personalized multitask learning. *arXiv preprint arXiv:1711.04036*. 2017 Nov 10.
38. Sushil M, uster S, Luyckx K, Daelemans W. Patient representation learning and interpretable evaluation using clinical notes. *Journal of Biomedical Informatics*. 2018 Aug 1;84:103-13.
39. Dubois S, Romano N, Kale DC, Shah N, Jung K. Learning Effective Representations from Clinical Notes. *arXiv preprint arXiv:1705.07025*. 2017 May 19.
40. Rehurek R, Sojka P. Software framework for topic modelling with large corpora. *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks 2010*.
41. Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M, Kudlur M. Tensorflow: a system for large-scale machine learning. *OSDI* 2016 Nov 2 (Vol. 16, pp. 265-283).
42. Le Q, Mikolov T. Distributed representations of sentences and documents. *International Conference on Machine Learning* 2014 Jan 27 (pp. 1188-1196).
43. Yang X, Molchanov P, Kautz J. Multilayer and multimodal fusion of deep neural networks for video classification. *Proceedings of the 2016 ACM Conference on Multimedia* 2016 Oct 1 (pp. 978-987). ACM.