EXPERTNET: A Deep Learning Approach to Combined Risk Modeling and Subtyping in Intensive Care Units

Shivin Srivastava, Vaibhav Rajan*

***Abstract*— Risk models play a crucial role in disease prevention, particularly in intensive care units (ICU). Diseases often have complex manifestations with heterogeneous subpopulations, or subtypes, that exhibit distinct clinical characteristics. Risk models that explicitly model subtypes have high predictive accuracy and facilitate subtype-specific personalization. Such models combine clustering and classification methods but do not effectively utilize the inferred subtypes in risk modeling. Their limitations include tendency to obtain degenerate clusters and cluster-specific data scarcity leading to insufficient training data for the corresponding classifier. In this paper, we develop a new deep learning model for simultaneous clustering and classification, EXPERTNET , with novel loss terms and network training strategies that address these limitations. The performance of EXPERTNET is evaluated on the tasks of predicting risk of (i) sepsis and (ii) acute respiratory distress syndrome (ARDS), using two large electronic medical records datasets from ICUs. Our extensive experiments show that, in comparison to state-of-the-art baselines for combined clustering and classification, EXPERTNET achieves superior accuracy in risk prediction for both ARDS and sepsis; and comparable clustering performance. Visual analysis of the clusters further demonstrates that the clusters obtained are clinically meaningful and a knowledge-distilled model shows significant differences in risk factors across the subtypes. By addressing technical challenges in training neural networks for simultaneous clustering and classification, EXPERTNET lays the algorithmic foundation for the future development of subtype-aware risk models.**

***Index Terms*— Clinical Risk Prediction, Sepsis, Acute Respiratory Distress Syndrome, Subtyping**

I. INTRODUCTION

MUCH of clinical practice is reactive – aiming to intervene to inhibit or abate the progression of diagnosed diseases [1]. A preventive approach not only provides the greatest health benefits, but is also the most cost-effective [1], [2]. Risk models play a crucial role in enabling such prevention, particularly in intensive care units (ICU), where patients are vulnerable to several complications. When informed of potential risks, clinical staff may be able to avert the disease or be better prepared to combat them. Many risk models have been developed to aid clinical decision-making, e.g., [3]–[7]. Within ICUs rule-based models are commonly used to assess patient severity and risk of specific diseases, e.g., [8]–[10]. In contrast to rule-based models, machine learning models can effectively utilize more diverse sources of information [11], and are often more accurate, e.g., [3], [12]–[21]. The impact of risk models on patient care and costs can be significant even with marginal improvements in predictive accuracy. For instance, just 1% reduction in sepsis cases through better prevention can approximately reduce hospital costs in the US by roughly $450M (by 2016 estimates) [22].

Analysis of large-scale electronic medical records (EMR) has revealed tremendous heterogeneity within patient populations (e.g., in sepsis [12]). We find distinct subpopulations, called *subtypes*, having relatively homogeneous clinical characteristics within the larger population. Most previous approaches for risk modeling learn a single population-based model using available historical patient data. Such a 'one-size-fits-all' model may consistently underestimate or overestimate risks for specific subtypes [23]. Risk models that account for subtypes are found to be more effective than population-based approaches [15], [24]–[26], providing improved accuracy and informative, subtype-specific features [12], [27]. Thus, they facilitate personalization as subsequent clinical decisions can be tailored to each subtype [28].

Previous works that account for subtypes during risk modeling broadly fall into the following three categories.

1) Most previous works adopt a 'cluster-then-predict' approach, where clustering is first done independently to find subtypes and then predictive models (e.g., binary classifiers) are trained for each cluster. E.g., in [15], autoencoders are used to learn patient representations which are clustered to find subtypes, and a mortality predictor is trained for each subtype via multi-task learning; In [12], data is clustered to find cluster-specific features to predict sepsis risk and the most important features are combined, from each cluster, to build a population-level model. Since clustering is performed independent of classifier training, such approaches may not discover latent cluster structures that benefit subsequent classification.
2) Some recent works have developed combined clustering and classification models that enforce stratification through homogeneous risk outcomes for each subtype. In [29], an Actor Critic Approach for Temporal Predictive

Date Submitted: February 7, 2023. "This work was supported by Singapore MoE Academic Research Fund [A-8000874-00-00], PI: VR"

Shivin Srivastava is with Department of Information Systems and Analytics, School of Computing, National University of Singapore (e-mail: shivin@comp.nus.edu.sg).

Vaibhav Rajan is with Department of Information Systems and Analytics, School of Computing, National University of Singapore (e-mail: vaibhav.rajan@nus.edu.sg);*Corresponding Author.

Clustering (AC-TPC) is designed wherein latent representations from patient data are clustered and predictor networks are trained per cluster. In [30], Deep Significance Clustering (DICE) is developed where patient representations are clustered, and cluster membership values are used to train a classifier while ensuring significant association between the risk and cluster membership. However, subtypes may be homogeneous with respect to any clinical characteristic, not necessarily (only) the disease risk. Hence, such models are limited in the kinds of subtypes they find.

3) A risk model on general subtypes (i.e., not risk-stratified clusters) was proposed in [27]. In their Deep Mixture Neural Networks (DMNN) neural representations are clustered using softmax gating and a mixture of expert neural networks, weighted by the gating values, is used to predict risk. As discussed in [27], the problems with this approach are (a) The deep network can easily overfit leading to poor generalization and (b) The gating mechanism can easily degenerate leading to all data points collapsing into a single cluster (thus, no subtypes are found).

In addition, an unaddressed problem that can occur with all these approaches is that if any of the inferred clusters has very few data points, the corresponding local classifier does not have sufficient data to learn from.

Our Contributions

In this paper, we develop EXPERTNET, a new deep learning based subtype-aware risk model. EXPERTNET's architecture and training procedure are carefully designed to address limitations of previous approaches. The architecture of EXPERTNET combines the autoencoder architecture of advanced neural clustering methods (e.g., [31]–[33]) with a neural mixture-of-experts, where each *local expert* is a neural network trained to predict subtype-specific risks. Naïvely minimizing losses for both clustering and classification results in trivial clusters that do not lead to improved risk prediction. To address this problem, we propose new regularizer terms in the loss function, which are designed to prevent distortion in the latent space and control cluster sizes.

For relatively smaller clusters, cluster-specific local networks cannot learn well if only the data points within their own clusters are used for training. To address this problem, we design a novel *cluster-weighted* training strategy. Each local network is trained on *all* data points with those from its own cluster having higher weight compared to those from other clusters. We use each patient's probability of lying in a cluster as the weight, which itself is inferred during model training and is iteratively optimized to improve both the clustering and risk prediction. Thus, clustering impacts the risk model not just by grouping the data for each local expert but also informs the magnitude of signal each local expert uses for data samples. The mixture of experts are both trained together on shared data, with different weights, and used together for prediction.

Unlike previous 'cluster-then-predict' approaches, through combined clustering and classification, EXPERTNET obtains latent patient representations that both have meaningful cluster structure and yield improved risk prediction. Further, EXPERTNET does not impose stratification requirements, thereby yielding generic subtypes. Although EXPERTNET also follows a mixture-of-experts architecture, it differs from DMNN [27], the closest related work, in the use of – (i) advanced deep learning based clustering techniques and architecture which prevent degenerate clusterw and (ii) novel loss terms and training strategies which improve its performance on clustering and risk prediction.

We evaluate EXPERTNET for the task of predicting risk of two life-threatening ICU complications: Sepsis and Acute Respiratory Distress Syndrome (ARDS). On EMR data from multiple ICU databases, we compare the performance of EXPERTNET with several competitive baselines. Our experiments show that for predicting risks of sepsis and ARDS, EXPERTNET outperforms state-of-the-art risk models based on (i) cluster-then-predict approaches (ii) simultaneous clustering and classification methods. The clustering performance of EXPERTNET is comparable (for ARDS) and superior (for Sepsis) to advanced deep learning based (pure) clustering methods in terms of metrics for cluster separability and feature discrimination. Qualitative analysis and visualizations of the clusters further demonstrate that the inferred subtypes differ significantly in their clinical characteristics.

Interpretability is an important requirement in clinical applications. While neural models often have high accuracy, they lack interpretability with respect to the reasons for predictions. To address this limitation, many Explainable AI (XAI) approaches are being developed [34], and have successfully been used in clinical contexts [35]. Any XAI method may be used with EXPERTNET. We obtain important features for prediction using one such approach, Knowledge Distillation (KD). One of the reasons to infer subtypes is to discover associations between each patient subtype and risk factors that vary across subtypes. Our KD-based analysis shows how that can be achieved and also illustrates that the subtypes inferred by EXPERTNET are clinically meaningful.

To summarize, our contributions in this paper are as follows:

- We develop EXPERTNET, a new subtype-aware risk model, that leverages the powerful representation learning ability of deep neural networks to simultaneously model the underlying heterogeneity and effectively utilize the clustered patient representations within a mixture of cluster-specific classifiers.
- We overcome limitations of previous subtype-aware risk models through the design of novel loss terms to prevent cluster degeneracy and new training strategies to address the problem of cluster-specific data scarcity for training the corresponding classifiers in EXPERTNET.
- We evaluate EXPERTNET on the tasks of predicting risk of two ICU complications – sepsis and acute respiratory distress syndrome (ARDS) using two large EMR databases. Our experiments demonstrate that EXPERTNET outperforms state-of-the-art baselines for combined clustering and classification on risk prediction, and obtains clinically meaningful subtypes with differing clinical characteristics.

II. METHOD

A. Problem Formulation

Given N datapoints $\{x_i \in X\}_{i=1}^{N}$, where each x_i represents a feature vector for a patient, class labels $y_i \in \{1, \ldots, \mathcal{B}\}$, associated with every point x_i and the number of clusters k, our aim is to simultaneously (i) cluster the N datapoints into k clusters, each represented by a centroid $\mu_j,\ j \in \{1, \ldots, k\}$, and (ii) build k distinct supervised classification models within each cluster. Note that during training, labels are used only for building the classification models and not for clustering.

B. Deep Embedded Clustering

We now briefly describe DEC's loss function [36] that is utilized in EXPERTNET. DEC uses an autoencoder architecture to obtain latent representations and cluster centroids. Initial latent representations z_i for data points x_i are found using a pretrained autoencoder. Initial cluster centroids $\{\mu_j\}$ are obtained by using k-means on the latent representations. The decoder is then removed and the representations are fine tuned by minimizing the Kullback-Leibler (KL) divergence between two distributions, $P = \{p_{ij}\}$ and $Q = \{q_{ij}\}$, defined below.

$$L_c = KL(P||Q) = \sum_{i=1}^{N}\sum_{j=1}^{N} p_{ij} \log \frac{p_{ij}}{q_{ij}}, \tag{1}$$

where q_{ij} is the probability of assigning the i^{th} embedded data point (z_i) to the j^{th} cluster (with centroid μ_j), i.e., the soft cluster assignment values for the i^{th} data point. It is measured using the similarity (via the Student's t kernel) between the embedded point and the centroid [37] as follows:

$$q_{ij} = \frac{(1 + \|z_i - \mu_j\|^2)^{-1}}{\sum_j (1 + \|z_i - \mu_j\|^2)^{-1}} \tag{2}$$

The target distribution p_{ij} is defined in terms of q_{ij} itself as:

$$p_{ij} = \frac{q_{ij}^2 / \sum_i q_{ij}}{\sum_j (q_{ij}^2 / \sum_i q_{ij})} \tag{3}$$

The term $\sum_i q_{ij}$ (that denotes cluster size) normalizes the loss contribution of each centroid to prevent large clusters from distorting the embedding space. This can be viewed as a form of self supervision as more emphasis is given to data points assigned with high confidence (high q_{ij}) and points with high confidence act as anchors and distribute other points around them more densely (leading to improved purity of clusters). The predicted cluster label of x_i is $\arg\max_j q_{ij}$.

C. EXPERTNET: Neural Architecture

Fig. 1 shows the neural architecture of EXPERTNET that consists of an encoder, a decoder and k local networks (LN_j). The encoder, which models the parameterized function $f(\mathcal{U}) : X \to Z$, is used to obtain low-dimensional latent representations (Z_i) of input datapoints (X_i), where the parameters $\mathcal{U}$ are the network weights. Cluster structure is learnt in this latent space (as described in section B above) and latent representations are used in local networks, to train k classification networks that learn the parameterized functions

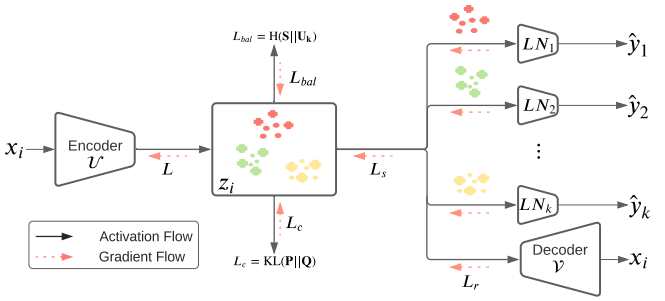


Fig. 1: Neural architecture of EXPERTNET, consisting of an encoder, a decoder and k local expert networks (LN_j). Latent representations obtained from the encoder are clustered and used as inputs to train the local networks. The entire network is trained in an end-to-end manner. During prediction, the trained encoder and local networks are used as described in the text.

$h(\mathcal{W}_j) : Z \to \hat{y}_j$ for $j = 1 \ldots k$, where $\mathcal{W}_j$ are the corresponding network weights. In addition, there is a decoder which models the parameterized function $g(\mathcal{V}) : Z \to X$, where the parameters $\mathcal{V}$ are the network weights. The decoder is used to reconstruct the input from the latent representations. The network weights: $\mathcal{U}, \mathcal{V}, \{\mathcal{W}_j\}_{j=1}^{k}$ are learnt by optimizing a combination of losses as described in the following.

D. EXPERTNET: Loss Function

The overall loss function L is a weighted combination, with coefficients $\beta, \gamma, \delta \geq 0$, of the reconstruction loss L_r, clustering loss L_c, cluster balance loss L_{bal} and classification loss L_s:

$$L = L_r + \beta \cdot L_c + \gamma \cdot L_s + \delta \cdot L_{bal} \tag{4}$$

L_c is the KL divergence loss (Eq. 1), where the cluster membership distribution Q (Eq. 2) uses representations z_i and cluster centroids μ_j inferred during EXPERTNET training. As suggested in [31], [32], to prevent distortion in the latent space and improve clustering performance, we add the reconstruction loss measured by mean squared error:

$$L_r = \sum_{i=1}^{N} \|x_i - g(f(x_i))\|^2 \tag{5}$$

DEC and its variants are centroid-based and have objectives similar to that of k-means. In such approaches, the encoder can map the centroids to a single point to make the loss zero and thus collapse the clusters resulting in trivial solutions. To address this problem in EXPERTNET we design a novel cluster balance loss to discourage unevenly distributed cluster sizes. We define the 'soft' size of a cluster C_j as $|C_j| = \sum_i q_{ij}$ and cluster support counts $\mathbf{S} = [|C_1|, |C_2|, \ldots, |C_k|]$ as a k-dimensional probability distribution. Let $\mathbf{U_k} = (1/k)\mathbf{U}(0, 1)$ denote the k-dimensional uniform distribution function. We use the Hellinger distance (H), which measures the dissimilarity between two distributions, as the loss $L_{bal} = H(\mathbf{S}||\mathbf{U_k}) = \frac{1}{2}\|\sqrt{S} - \sqrt{U_k}\|_2$. L_s is a weighted cross-entropy classification loss described below.

E. EXPERTNET: Training

To initialize the parameters $(\mathcal{U}, \mathcal{V})$, we first pre-train the encoder and decoder with the input data using only the recon-

struction loss L_r. This is followed by k-means clustering on $\{z_i = f(x_i;\mathcal{U})\}_{i=1}^{N}$ to obtain cluster centroids $\{\mu_i\}_{i=1}^{k}$ which are used to calculate cluster membership and target distributions, Q, P. After initialization, we use mini-batch stochastic gradient descent to train the entire network, using the loss L. To stabilize training we update P only after every epoch.

Since the embeddings get updated progressively in every iteration, we train the LNs for a larger number of *sub-iterations* within every iteration of the main training loop. The number of sub-iterations gradually increases (by 1 for every 5 epochs until a max-limit which we set to 10). This enables the LNs to learn better from the stabilized clustered embeddings than from the intermediate representations.

Further, we use a novel strategy called *Cluster Weighted Training* in each sub-iteration, enabling each LN to learn from data in other clusters. This addresses the problem of insufficient training data for relatively smaller clusters. Each LN is trained on the *entire* training data, but with higher weights for intra-cluster data points and lower weights for other data points. We leverage the probabilistic definition of cluster membership to obtain individual weights for each training point with respect to each LN. This enables points within each cluster to have a relatively higher weight compared to points outside the cluster. The local loss utilizes these weights to calculate the final loss function. The final classification loss is a weighted cross entropy (CE) loss:

$$L_s = \sum_{j=1}^{k} \sum_{p \in C_j} q_{p,j}\, \mathrm{CE}(y_p, h_j(x_p; \mathcal{V}_j)). \tag{6}$$

The LNs are trained in this manner for $T-1$ sub-iterations *without* backpropagating the error to the encoder. The individual errors are collected from all the k LNs only at the last (T^{th}) iteration and backpropagated to the encoder to adjust the cluster representations accordingly. After training, the encoder network is frozen and the local networks are finetuned, using the cluster-weighted classification loss as described above, on the latent embeddings to further improve LN performance. Algorithm 1 summarizes the entire training procedure.

Training Time Complexity: We assume that the training data has N data points, the encoder, decoder, and the k local networks in EXPERTNET have the same network architecture, with L layers, and layer l has s_l number of neurons. The overall time complexity of Algorithm 1, when run for E epochs, is $O\left(N \cdot E \cdot T_{iter} \cdot k \sum_{\ell=1}^{L} s_\ell s_{\ell-1} + E \cdot Nk\right)$. The detailed derivation is presented in Appendix VI.

F. EXPERTNET: Predictions

Prediction requires only the encoder and local networks. For a test point $\hat{x}_p$, the soft cluster probabilities ($\hat{q}_{pj}$) are calculated from Eq. 2 using the cluster centroids (learnt after training) and embedding of the test point (from the encoder). All the local networks are used to predict the class label using the cluster membership probabilities: $\hat{y}_p = \sum_{j=1}^{k} \hat{q}_{pj} h_j(\hat{x}_p)$.

Algorithm 1: EXPERTNET Training

Input: Training Data: $X \in \mathbb{R}^{n \times d}$, labels $y^{n\times 1} \in [\mathcal{B}]^n$, k, $f(\cdot;\mathcal{U}), g(\cdot;\mathcal{V})$ and $\{h_j(\cdot;\mathcal{W}_j)\}_{j=1}^{k}$, $\mathcal{J}$

▷ **Initialization**
Pre-train $f(\cdot;\mathcal{U})$ & $g(\cdot;\mathcal{V})$ via back-propagating loss in eq. 5
Compute $\mu(C_j) \forall\ C_j\ s.t.\ j \in [k]$
Compute matrices Q and P according to eqs. 2 and 3
▷ **Algorithm**
while *Validation AUC increases* **do**
  for *every mini-batch* $\mathcal{X}_b$ **do**
    Calculate latent embeddings by encoder $\mathcal{U}$
    for T_{iter} *sub-iterations* **do**
      Train local classifiers $\{LN_j\}_{j=1}^{k}$
    Backpropagate $L = L_r + \beta \cdot L_c + \gamma \cdot L_s + \delta \cdot L_{bal}$ and update $\mu, \mathcal{U}, \mathcal{V}$ and $\{LN\}_{j=1}^{k}$
  Update P via eq. (3)
▷ **Fine tune Local Classifiers**
For every cluster C_j, train a classifier h_j on $(\mathcal{X}_j, \mathcal{Y}_j)$.
Output Trained EXPERTNET model. Cluster centroids $\mu = \{\mu_j\}_{j=1}^{k}$

G. Interpretability by Distilling EXPERTNET

We obtain important features for prediction using Knowledge Distillation (KD), that has been used previously for risk prediction in critical care (e.g., in [27], [38]). KD "distills" the learnt knowledge in a complex machine learning model into a relatively simpler model that may be more interpretable without significant loss in performance [39], [40]. The general procedure is to first train the neural network, then use training data features and predictions of the network as the input data to train the simpler model. We choose Random Forest (RF) to distill EXPERTNET because RF has sufficient capacity to model non-linear interactions and also provides feature importance values.

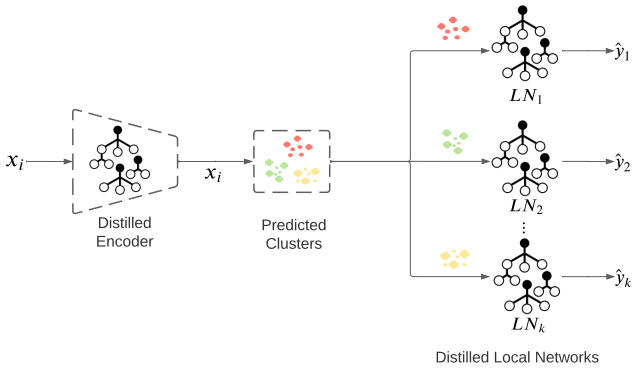


Fig. 2: Distilled EXPERTNET model.

We train multiple KD models (see Fig. 2), one for distilling the encoder and k separate distilled models for the local networks. Each KD model is a RF with 100 base decision trees. The RF for distilling the encoder learns to predict the

cluster labels $\hat{c}_i = \arg\max_j q_{ij}$ values (from EXPERTNET's encoder), with features x_i as inputs. The k local RF models are trained on the predictions of EXPERTNET's k LNs. The trained KD models can be used for prediction as follows. First, the distilled encoder RF predicts the cluster membership for a test data point, which determines the LN to use; the corresponding distilled LN then predicts the class label.

III. EXPERIMENTS

A. Experimental Setting

1) Risks: Our experimental evaluation is on two important disease risks: Sepsis and Acute Respiratory Distress Syndrome (ARDS). Sepsis is a life-threatening condition that occurs when the body's response to infection causes tissue damage, organ failure, or death. In the US, about 1.7 million develop sepsis and 270,000 die of sepsis per year; in fact, over 1/3rd of people who die in U.S. hospitals have sepsis [41]. Globally, in 2017, around 48.9 million developed sepsis, and there were 11 million sepsis-related deaths [42]. The costs for managing sepsis in U.S. hospitals – at USD 24 billion annually (13% of U.S. healthcare expenses) – exceed those for any other health condition [22]. ARDS often manifests as respiratory failure characterized by rapid onset of widespread inflammation in the lungs. Globally, ARDS affects more than 3 million people annually, contributing to about 10% of ICU admissions; with high mortality of 35-46% depending on severity at onset [43]–[45]. The recent COVID-19 outbreak has also led to large number of ARDS cases [46].

2) Data: We use de-identified patient data from publicly available ICU databases. For ARDS prediction, we use the MIMIC III dataset [47] comprising 33,798 unique patients' data from the Beth Israel Deaconess Medical Center, USA. For sepsis prediction, we use the dataset, from [48], of 40,366 patients from MIMIC III and Emory University Hospital, USA. Sepsis patients are identified using the Sepsis-3 criteria [49]–[51], as described in [48]. ARDS patients are identified, using the Berlin criteria [52], [53]. Thus, hourly binary labels indicating the presence/absence of the conditions are determined for each patient.

3) Prediction Setting: We use the first 24 hours of data to predict risk (of each condition, separately) in the remaining ICU stay, as recommended in [54]. All patients whose length of ICU stay is less than 24 hours and who develop the condition within 24 hours of ICU stay are excluded (Fig. 3). After these exclusions, we have 30,661 and 22,450 patients in the Sepsis and ARDS datasets out of which 1,844 and 1,701 patients have a diagnosis of sepsis and ARDS, respectively, 24 hours after ICU entry. Note that our datasets are highly imbalanced: 6% and 7.5% positive cases in Sepsis and ARDS datasets respectively.

4) Features: Routinely available clinical variables are used for prediction. For the sepsis dataset, we use all the variables given in [48] and the SOFA score [9]. For the ARDS dataset, we include the variables used in previous studies [55]–[57]. Only those variables with at least one non-missing value in all included patients are considered. Tables V and VI in Appendix I list all the variables and their summary statistics

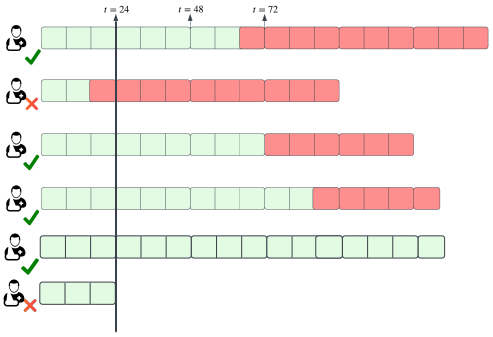


Fig. 3: Prediction Setting: Green color indicates time periods before the diagnosis of complication (Sepsis/ARDS). Red color indicates time periods at and after the diagnosis. Clinical data up to 24 hours from ICU admission is used to predict the risk of complication later. Patients with less than 24 hours of ICU stay and those with diagnosis within the first 24 hours are excluded.

in our ARDS and sepsis datasets respectively. Standard preprocessing steps are performed to obtain patient-wise feature vectors. Categorical variables are converted to binary vectors using one-hot encoding. For temporal variables (i.e., variables with repeated measurements), we use the first, last, median, minimum and maximum values in the first 24 hours as features. Real-valued variables for each patient are normalized to have mean 0 and standard deviation 1. Static features (e.g., age) are used directly. The final feature vector dimensions for ARDS and Sepsis data are 486 and 209 respectively.

5) Evaluation Details: In each case of sepsis and ARDS, the entire dataset is divided into train-test split of 85:15. Within each train split, 15% of the data is used for validation and the remaining for model training. All experiments are run on 21 such random splits for a robust evaluation and the average test results are reported. Binary classification metrics are used to evaluate performance on risk prediction: (i) Area under the ROC Curve (AUC) and (ii) Area under the Precision Recall Curve (AUPRC). Statistical significance of the performance improvement is measured using Student's t-test. To quantitatively evaluate subtypes, we use (i) silhouette score (SIL) to measure the density and separation of clusters and (ii) a novel metric called the **H**ypothesis **T**esting based **F**eature **D**iscrimination (HTFD) to measure feature discrimination across inferred clusters described below.

6) Hypothesis testing based Feature Discrimination (HTFD): A common practice in subtyping studies (e.g., [58]) is to check, for each feature, if there is a statistically significant difference, indicated by a low p-value (< 0.05), in its distribution across the inferred subtypes. We quantify this through an aggregation over all features in the HTFD metric. Let X_i^f denote the values of feature f for data points in the i^{th} cluster C_i and X^f denote the feature values of data points in all clusters except C_i (stacked together). F denotes the set of all features and $|F|$ is the total number of features. For cluster C_i, we define:

$$\text{HTFD}(C_i) = \frac{1}{|F|} \sum_{f \in F} -\ln\left(\text{p-value}(X_i^f, X^f)\right) \qquad (7)$$

where p-values are obtained via Student's t-test. The negative logarithm of p-value is added and normalized by the number

TABLE I: Classification (AUC and AUPRC) scores for EXPERTNET and baselines on ARDS and Sepsis dataset for $k = 2, 3, 4, 5$ (row-wise). "-" indicates no results due to an empty cluster. Within each row, the best result is in bold and the second best result is underlined. $p^* < 0.05$, $p^{**} < 0.01$ indicates significantly better performance of EXPERTNET compared to the baseline.

Dataset (Metric)	k	KM-Z -	DCN-Z (2017)	IDEC-Z (2017)	DMNN (2020)	AC-TPC (2020)	DICE (2021)	EXPERTNET (Ours)
ARDS (AUPRC)	2	$0.134 \pm 0.031^{**}$	$0.234 \pm 0.04^{**}$	$0.215 \pm 0.04^{**}$	$0.263 \pm 0.02^{**}$	–	$0.256 \pm 0.021^{**}$	$\mathbf{0.291 \pm 0.023}$
	3	$0.109 \pm 0.033^{**}$	$0.198 \pm 0.053^{**}$	$0.187 \pm 0.061^{**}$	$0.261 \pm 0.024^{*}$	$0.156 \pm 0.039^{**}$	0.263 ± 0.025	$\mathbf{0.28 \pm 0.035}$
	4	$0.126 \pm 0.037^{**}$	$0.203 \pm 0.042^{**}$	$0.179 \pm 0.048^{**}$	$0.245 \pm 0.017^{**}$	$0.142 \pm 0.03^{**}$	$0.266 \pm 0.018^{*}$	$\mathbf{0.284 \pm 0.033}$
	5	$0.094 \pm 0.03^{**}$	$0.177 \pm 0.051^{**}$	$0.199 \pm 0.043^{**}$	$0.249 \pm 0.034^{**}$	$0.147 \pm 0.029^{**}$	$0.192 \pm 0.031^{**}$	$\mathbf{0.29 \pm 0.02}$
Sepsis (AUPRC)	2	$0.091 \pm 0.018^{**}$	$0.206 \pm 0.102^{**}$	$0.278 \pm 0.068^{**}$	$0.291 \pm 0.031^{**}$	–	$0.335 \pm 0.038^{**}$	$\mathbf{0.426 \pm 0.03}$
	3	$0.113 \pm 0.019^{**}$	$0.195 \pm 0.082^{**}$	$0.265 \pm 0.06^{**}$	$0.287 \pm 0.03^{**}$	$0.19 \pm 0.024^{**}$	$0.218 \pm 0.022^{**}$	$\mathbf{0.414 \pm 0.035}$
	4	$0.087 \pm 0.023^{**}$	$0.173 \pm 0.045^{**}$	$0.266 \pm 0.033^{**}$	$0.301 \pm 0.022^{**}$	$0.192 \pm 0.04^{**}$	$0.332 \pm 0.048^{**}$	$\mathbf{0.418 \pm 0.038}$
	5	$0.089 \pm 0.018^{**}$	$0.181 \pm 0.069^{**}$	$0.26 \pm 0.033^{**}$	$0.291 \pm 0.028^{**}$	$0.209 \pm 0.037^{**}$	$0.243 \pm 0.033^{**}$	$\mathbf{0.409 \pm 0.038}$
ARDS (AUC)	2	$0.577 \pm 0.066^{**}$	$0.726 \pm 0.040^{**}$	$0.707 \pm 0.041^{**}$	$0.755 \pm 0.014^{**}$	–	$0.715 \pm 0.013^{**}$	$\mathbf{0.785 \pm 0.015}$
	3	$0.524 \pm 0.081^{**}$	$0.681 \pm 0.061^{**}$	$0.658 \pm 0.083^{**}$	0.76 ± 0.011	$0.638 \pm 0.009^{**}$	$0.732 \pm 0.012^{**}$	$\mathbf{0.766 \pm 0.034}$
	4	$0.554 \pm 0.070^{**}$	$0.692 \pm 0.050^{**}$	$0.656 \pm 0.057^{**}$	$0.747 \pm 0.014^{**}$	$0.606 \pm 0.040^{**}$	$0.736 \pm 0.010^{**}$	$\mathbf{0.772 \pm 0.020}$
	5	$0.489 \pm 0.066^{**}$	$0.646 \pm 0.081^{**}$	$0.658 \pm 0.068^{**}$	$0.746 \pm 0.02^{**}$	$0.626 \pm 0.037^{**}$	$0.691 \pm 0.020^{**}$	$\mathbf{0.784 \pm 0.013}$
Sepsis (AUC)	2	$0.576 \pm 0.048^{**}$	$0.713 \pm 0.128^{**}$	$0.784 \pm 0.057^{**}$	$0.824 \pm 0.011^{**}$	–	$0.770 \pm 0.015^{**}$	$\mathbf{0.861 \pm 0.017}$
	3	$0.614 \pm 0.048^{**}$	$0.728 \pm 0.081^{**}$	$0.768 \pm 0.079^{**}$	$0.818 \pm 0.02^{**}$	$0.718 \pm 0.043^{**}$	$0.731 \pm 0.040^{**}$	$\mathbf{0.854 \pm 0.023}$
	4	$0.546 \pm 0.071^{**}$	$0.719 \pm 0.045^{**}$	$0.768 \pm 0.035^{**}$	$0.83 \pm 0.011^{**}$	$0.689 \pm 0.071^{**}$	$0.800 \pm 0.030^{**}$	$\mathbf{0.853 \pm 0.017}$
	5	$0.566 \pm 0.058^{**}$	$0.719 \pm 0.061^{**}$	$0.740 \pm 0.047^{**}$	$0.821 \pm 0.016^{**}$	$0.706 \pm 0.053^{**}$	$0.78 \pm 0.041^{**}$	$\mathbf{0.847 \pm 0.019}$

TABLE II: Clustering (Silhouette and HTFD) scores for EXPERTNET and baselines on ARDS and Sepsis dataset for $k = 2, 3, 4, 5$ (row-wise). "-" indicates no results due to an empty cluster. Within each row, the best result is in bold and the second best result is underlined. $p^* < 0.05$, $p^{**} < 0.01$ indicates significantly different (better/worse) performance of baseline compared to EXPERTNET.

Dataset (Metric)	k	KM-Z -	DCN-Z (2017)	IDEC-Z (2017)	DMNN (2020)	AC-TPC (2020)	DICE (2021)	EXPERTNET (Ours)
ARDS (SIL)	2	0.414 ± 0.188	0.439 ± 0.119	0.432 ± 0.117	$0.069 \pm 0.173^{**}$	–	$\mathbf{0.514 \pm 0.031}^{**}$	0.404 ± 0.134
	3	$0.212 \pm 0.098^{**}$	$0.272 \pm 0.148^{**}$	$0.296 \pm 0.121^{**}$	$0.02 \pm 0.09^{**}$	$0.17 \pm 0.097^{**}$	$0.355 \pm 0.053^{*}$	$\mathbf{0.423 \pm 0.11}$
	4	$0.144 \pm 0.091^{**}$	$0.198 \pm 0.122^{**}$	$0.197 \pm 0.143^{**}$	$0.0 \pm 0.0^{**}$	$0.212 \pm 0.08^{**}$	0.313 ± 0.033	$\mathbf{0.373 \pm 0.134}$
	5	$0.122 \pm 0.086^{**}$	$0.09 \pm 0.094^{**}$	0.205 ± 0.138	$0.088 \pm 0.186^{**}$	0.196 ± 0.093	$\mathbf{0.272 \pm 0.032}$	0.244 ± 0.117
Sepsis (SIL)	2	$0.485 \pm 0.022^{**}$	0.554 ± 0.185	0.581 ± 0.147	$0.074 \pm 0.16^{**}$	–	$0.48 \pm 0.028^{**}$	$\mathbf{0.649 \pm 0.13}$
	3	$0.218 \pm 0.112^{**}$	$0.319 \pm 0.221^{**}$	0.474 ± 0.262	$0.056 \pm 0.136^{**}$	$0.148 \pm 0.063^{**}$	$0.437 \pm 0.018^{**}$	$\mathbf{0.635 \pm 0.127}$
	4	$0.131 \pm 0.054^{**}$	$0.197 \pm 0.171^{**}$	$0.435 \pm 0.261^{**}$	$0.073 \pm 0.154^{**}$	$0.151 \pm 0.064^{**}$	$0.424 \pm 0.026^{**}$	$\mathbf{0.623 \pm 0.13}$
	5	$0.157 \pm 0.085^{**}$	$0.126 \pm 0.102^{**}$	$0.401 \pm 0.228^{**}$	$0.059 \pm 0.136^{**}$	$0.159 \pm 0.073^{**}$	$0.126 \pm 0.102^{**}$	$\mathbf{0.668 \pm 0.125}$
ARDS (HTFD)	2	$1.069 \pm 0.379^{**}$	$1.12 \pm 0.118^{**}$	$1.135 \pm 0.13^{**}$	0.978 ± 0.874	–	$1.118 \pm 0.03^{**}$	$\mathbf{1.31 \pm 0.097}$
	3	$0.961 \pm 0.364^{**}$	$0.89 \pm 0.336^{**}$	$0.994 \pm 0.437^{*}$	0.858 ± 0.9	$0.7 \pm 0.251^{**}$	$0.976 \pm 0.116^{**}$	$\mathbf{1.22 \pm 0.09}$
	4	0.876 ± 0.373	0.891 ± 0.335	1.017 ± 0.253	0.772 ± 0.892	$0.263 \pm 0.327^{**}$	1.025 ± 0.097	$\mathbf{1.049 \pm 0.288}$
	5	$0.741 \pm 0.37^{**}$	$0.612 \pm 0.331^{*}$	$0.886 \pm 0.314^{**}$	0.863 ± 0.896	$0.078 \pm 0.241^{**}$	$\mathbf{0.899 \pm 0.058}$	0.895 ± 0.436
Sepsis (HTFD)	2	$1.466 \pm 0.015^{**}$	$1.462 \pm 0.064^{**}$	$1.496 \pm 0.069^{**}$	$1.155 \pm 0.783^{*}$	–	$1.465 \pm 0.008^{**}$	$\mathbf{1.554 \pm 0.038}$
	3	1.414 ± 0.327	$1.4 \pm 0.133^{**}$	1.48 ± 0.037	$0.686 \pm 0.875^{**}$	$1.24 \pm 0.138^{**}$	$1.435 \pm 0.06^{**}$	$\mathbf{1.489 \pm 0.048}$
	4	1.334 ± 0.304	$1.291 \pm 0.104^{**}$	1.434 ± 0.044	$0.642 \pm 0.84^{**}$	$0.715 \pm 0.483^{**}$	$1.262 \pm 0.059^{**}$	$\mathbf{1.457 \pm 0.041}$
	5	1.161 ± 0.387	$1.09 \pm 0.305^{*}$	$\mathbf{1.391 \pm 0.071}$	$0.519 \pm 0.812^{**}$	$0.178 \pm 0.368^{**}$	$1.09 \pm 0.305^{*}$	1.341 ± 0.302

of features to obtain a measure where higher values indicate better feature discrimination across the i^{th} cluster and remaining clusters. A single value for the entire clustering is obtained by the average over each cluster's HTFD values. We multiply by a positive constant 0.05 (a monotonic transformation that does not change the ordering of the values) to obtain average HTFD values close to 1, for ease of interpretation.

7) Baselines: We compare the performance of EXPERTNET with two types of baseline algorithms.

1) 'Cluster-then-predict' approaches where clustering is first independently performed and then neural network classifiers are trained on each cluster (denoted by **-Z**). We compare with 3 clustering methods: (i) k-means (**KM**), where we use an autoencoder to get embeddings which are then clustered, (ii) Deep Clustering Network (**DCN**) [32] and (iii) Improved Deep Embedded Clustering (**IDEC**) [31].
2) Simultaneous clustering and classification methods: (i) Deep Mixture of Neural Networks (**DMNN**) [27] (ii) Actor Critic Temporal Predictive Clustering (**AC-TPC**) [29] and (iii) Deep Significance Clustering (**DICE**) [30].

All the methods require the number of clusters k as input. We report results for $k = 2, 3, 4, 5$. For a fair comparison, neural network architectures are identical for (i) local networks in EXPERTNET and classifiers in 'cluster-then-predict' baselines, (ii) encoder in EXPERTNET and feedforward networks used to obtain latent representations in AC-TPC and DICE. We use UMAP [59] to visualize the clusters and feature distribution in data embeddings found by EXPERTNET in 2 dimensions. The axes represent the 2 dimensions to which data is projected. Additional details on hyperparameters are in Appendix II.

8) Sensitivity Analysis and Ablation Studies: We evaluate the sensitivity of EXPERTNET on hyperparameters β, γ and δ (loss weights in Eq. 4). We individually vary each hyperparameter while setting the other two to 0 and measure the classification and clustering performance. Note that a value of 0 implies that the corresponding term is not used and thus ablates the term. Further, our cluster weighted loss approach can be optionally used independently during training and prediction. As another ablation study, we evaluate the effect of all four combinations (see Fig. 8). We denote the combinations by TT, TF, FT and

TABLE III: Clinical variables specific to subtypes in ARDS and Sepsis datasets for $k = 3$. S-1, subtype 1; S-2, subtype 2; S-3, subtype 3

(a) Clinical variables significantly specific to ARDS subtypes 1, 2 and 3						
Clinical Variables **Cluster Details**	**Mean or % Subtype 1** ($\|C_1\| = 3598, \%D = 12.5$)	**Mean or % Subtype 2** ($\|C_2\| = 11293, \%D = 4.7$)	**Mean or % Subtype 3** ($\|C_3\| = 1328, \%D = 17.0$)	S-1	S-2	S-3
Ventilator	0.727 ± 0.446	0.029 ± 0.166	0.158 ± 0.365	Y	Y	
Mean Airway Pressure	9.308 ± 3.012	13.0 ± 0.033	12.433 ± 1.88	Y	Y	
PIP	18.197 ± 7.852	12.0 ± 0.0	13.106 ± 4.163	Y	Y	
PEEP	4.53 ± 2.161	3.511 ± 0.182	3.684 ± 0.97	Y	Y	
Plateau Pressure	24.149 ± 5.718	28.0 ± 0.0	27.399 ± 2.56	Y	Y	
PaO2	161.227 ± 93.522	109.698 ± 36.592	109.892 ± 62.967	Y	Y	
Fibrinogen	266.799 ± 70.605	280.14 ± 48.002	360.616 ± 191.886		Y	Y
Bilirubin Total	0.678 ± 0.927	0.823 ± 1.568	2.947 ± 6.759			Y
Blood urea nitrogen	19.754 ± 14.631	24.675 ± 20.156	40.603 ± 31.292	Y		Y
Blood culture	-69136.57 ± 48851.355	-52641.595 ± 54762.433	-17131.559 ± 53005.115	Y		Y
(b) Clinical variables significantly specific to Sepsis subtypes 1, 2 and 3						
Cluster Details	($\|C_1\| = 7511, \%D = 9.3$)	($\|C_2\| = 4978, \%D = 6.7$)	($\|C_3\| = 9662, \%D = 3.0$)	S-1	S-2	S-3
pH	7.373 ± 0.395	4.154 ± 3.664	0.14 ± 1.007	Y	Y	Y
PaCO2	40.296 ± 9.293	22.105 ± 21.1	0.0 ± 0.0	Y	Y	Y
FiO2	0.418 ± 0.255	0.265 ± 0.283	0.032 ± 0.127	Y	Y	
SaO2	61.132 ± 45.52	37.328 ± 45.875	0.785 ± 7.547	Y		Y
Sofa_O2	2.589 ± 1.379	3.188 ± 1.198	3.998 ± 0.062	Y	Y	Y
MAP	85.824 ± 13.628	62.391 ± 8.828	89.397 ± 14.442	Y	Y	Y
SBP	127.914 ± 23.468	100.234 ± 20.918	128.628 ± 25.895		Y	Y
DBP	58.729 ± 22.106	38.27 ± 22.96	53.316 ± 31.819	Y	Y	
Unit2	0.428 ± 0.495	0.374 ± 0.484	0.185 ± 0.388	Y	Y	Y
HCO3	15.95 ± 11.84	16.011 ± 11.869	7.75 ± 11.708	Y		Y

TABLE IV: Top 10 most important features for ARDS and Sepsis prediction in for 3 clusters. Features common in two clusters are highlighted in yellow, those common in three clusters are highlighted in blue, while the rest are unique to their respective clusters. $|C_i|$ indicates the size of the i^{th} cluster. $\%D$ indicates the number of positive class labels, i.e., patients diagnosed with the disease (sepsis/ARDS), in the cluster.

ARDS			Sepsis		
$\|C_1\| = 3598$ $\%D = 12.5$	$\|C_2\| = 11293$ $\%D = 4.7$	$\|C_3\| = 1328$ $\%D = 17.0$	$\|C_1\| = 7511$ $\%D = 9.3$	$\|C_2\| = 4978$ $\%D = 6.7$	$\|C_3\| = 9662$ $\%D = 3.0$
GCS Verbal	Red blood cell count	Platelets	Temp	SBP	SBP
Extubated	Hematocrit	Respiratory rate	HR	Temp	HospAdmTime
GCS Total	White blood cell count	Heart Rate	HospAdmTime	HospAdmTime	Temp
GCS Eye	Platelets	Glucose	SBP	Age	Age
GCS Motor	PTT	Mean corpuscular hemoglobin	WBC	Glucose	Glucose
Age	Respiratory rate	PaO2	Platelets	WBC	HR
PaO2	Hemoglobin	Blood urea nitrogen	Age	Resp	WBC
Red blood cell count	Mean blood pressure	GCS Verbal	Glucose	HR	Creatinine
Heart Rate	Heart Rate	Weight	Resp	Platelets	BUN
PIP	Temperature	Urine output	Creatinine	Phosphate	Hct

FF, where the first and second positions indicate training and prediction respectively. T indicates the use of our approach while F indicates that it is not used. All the results are averages over 5 runs.

9) Code and Online Appendix: The source code for EXPERTNET is available at https://github.com/shivin9/ExpertNet. The online appendix is available here.

B. Results

1) Risk Prediction: Table I shows the performance of EXPERTNET and all the baseline methods. The performance of both cluster-then-predict algorithms (KM-Z, DCN-Z and IDEC-Z) and simultaneous clustering and classification models (DMNN, AC-TPC and DICE) are significantly inferior to that of EXPERTNET for all values of k tested. The margin of improvement, in both Sepsis and ARDS, is higher for AUPRC, which is considered a better metric in cases of class imbalance, as AUPRC correlates better with positive predictive value and reflects the discrimination of the features better [60]. Also note that AUPRC values are dataset-dependent and the baseline value is the fraction of positive samples, as discussed in [61]. Hence, in our highly imbalanced datasets, the baseline values are 0.06 and 0.075 respectively for Sepsis and ARDS. Thus, it is not surprising to find AUPRC values in the range of 0.1 – 0.5.

2) Subtyping: Table II shows our quantitative evaluation of clustering performance. On Sepsis, EXPERTNET outperforms all the baseline methods in all cases with respect to Silhouette score and, in 3 out of 4 cases, for HTFD (in the 4th case, it has the second best value, which is not significantly different from the best). On ARDS, there is no single algorithm that has the best performance across all values of k. With respect to Silhouette score, EXPERTNET has the best score for $k = 3, 4$, and the second best score for $k = 5$. For HTFD,

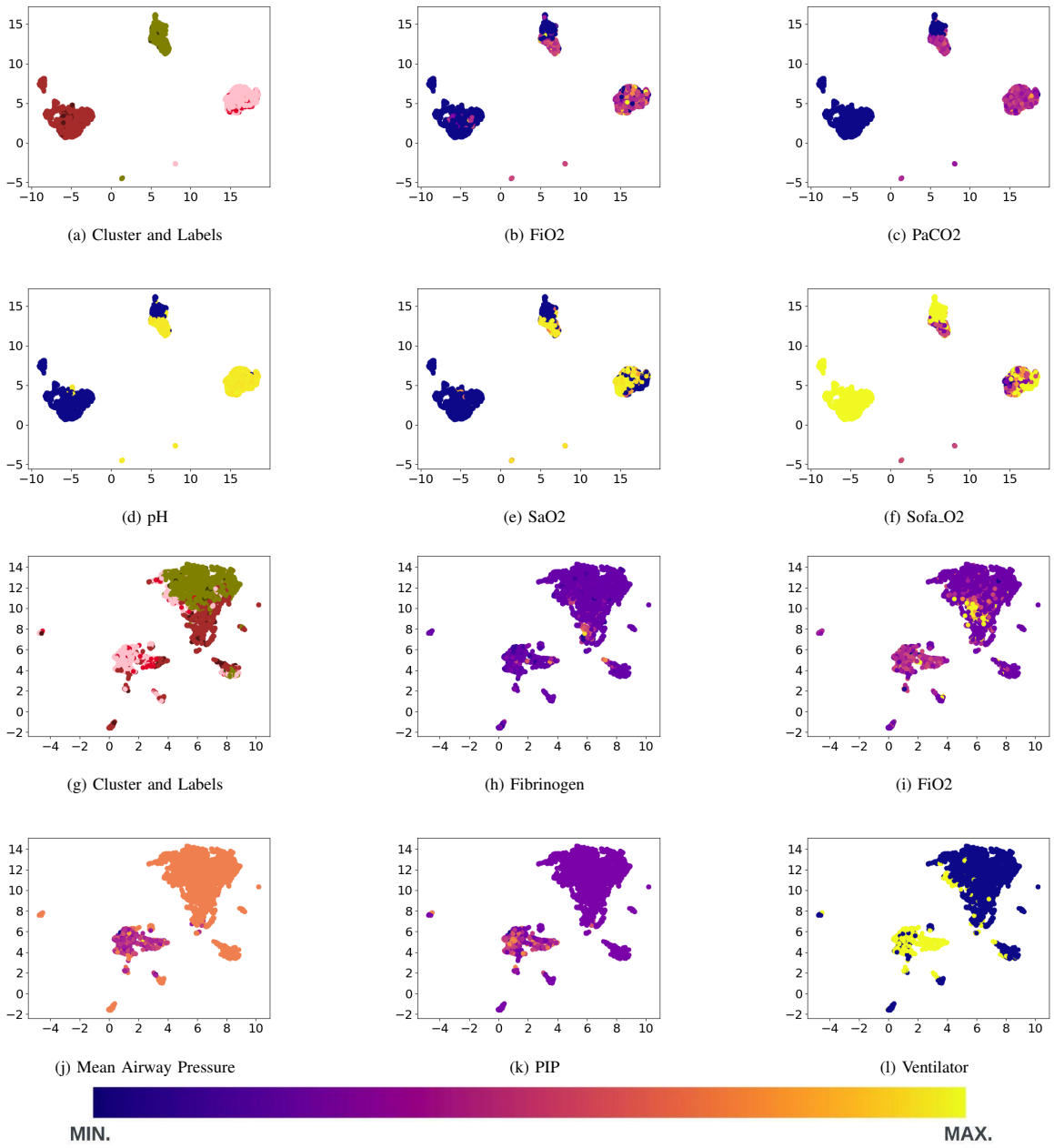
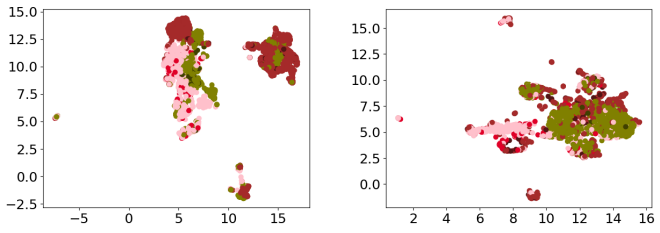


Fig. 4: Top (a-f) for Sepsis, Bottom (g-l) for ARDS. (a) and (g) show the 2-dimensional UMAP embeddings in representation space depicting overall clusters for $k = 3$ (Green: Cluster 1, Pink: Cluster 2, Brown: Cluster 3. Dark green, pink and brown points represent patients who get Sepsis/ARDS in their respective cohorts.) Other figures show variation of selected features (from Table III) for Sepsis (b-f) and for ARDS (g-l) data. Each point is color coded according to the scale presented at the bottom (the minimum value is shaded blue while the maximum value is shaded yellow). Image best viewed in color.

EXPERTNET has the best score for $k = 2, 3, 4$, and the second best score for $k = 5$. Both the second best scores, for Silhouette and HTFD, are not significantly different from the corresponding best scores. IDEC, a neural network based pure clustering method performs well and the performance of EXPERTNET (which uses a similar clustering technique) is better or comparable in many cases. Note that low means and high standard deviation in Silhouette and HTFD scores of DMNN are due to degenerate or empty clusters (see Appendix VII for a detailed discussion).

We qualitatively analyze the subtypes (clusters) for $k = 2$ (Table VII in Appendix IV) and $k = 3$ (Table III). Tables III(a) and III(b) present the clinical variables that are significantly different across the subtypes for ARDS and Sepsis respectively. A 'Y' for a feature under the S-j column indicates that that feature is significantly different in cluster C_j when compared with the rest of the samples in the other two clusters. They clearly indicate meaningful clustering with differences in clinical characteristics across patient subtypes.

Visualization of the clusters demonstrates the discriminatory power of the learnt embeddings. Fig. 4a and 4g, for sepsis and ARDS respectively, show the embeddings of the clusters and



(a) Sepsis Dataset (b) ARDS Dataset

Fig. 5: Plots of 2-dimensional UMAP embeddings of Sepsis and ARDS data (in feature space) colored based on clusters found by EXPERTNET for $k = 3$ (color coded Green for Cluster 1, Pink for Cluster 2 and Brown for Cluster 3. Dark green and Dark Pink points represent patients who get Sepsis/ARDS in their respective cohorts). Compare with Fig. 4 (a) and (g).

also depict the class labels within each cluster. We observe that the clusters are well separated and within each cluster, there are members of both classes (those with and without the diseases). Similar visualizations for $k = 2$ are in Fig. 12 in Appendix IV. In contrast, the feature space does not show an apparent cluster structure, especially for sepsis. Fig. 5 visualizes the feature space with the colors (pink, green, and brown) depicting clusters found by EXPERTNET for $k = 3$. The cluster separation in feature space is clearer for ARDS than for Sepsis. This suggests an explanation for why the relative improvement in performance (on Silhouette scores) is lower for ARDS. For sepsis, EXPERTNET is better at disentangling the mingled features in the latent space and achieves relatively better results.

3) Subtype-specific Risk Factors: Table IV shows the top 10 risk factors linked to each subtype for predicting ARDS and Sepsis respectively inferred via knowledge distilled RF models (for $k = 3$). In the case of sepsis, although the three subtypes differ significantly in terms of the feature distributions, important risk factors are found to be similar across the subtypes with Phosphate being an important risk factor solely in subtype 2 while BUN and Hct solely in subtype 3. In the case of ARDS, the risk factors differ in larger numbers across the subtypes. Subtype 1 has several Glasgow Coma Scores (GCS) scores as important risk predictors. This subtype is associated with larger ventilator time. Thus, our analysis suggests that for patients with a longer time on ventilator, monitoring GCS is particularly important for predicting the onset of ARDS. The results for $k = 2$ can be found in Table VIII in Appendix V.

4) Sensitivity Analysis and Ablation Studies: Figures 6 and 7 show the results for ARDS and Sepsis respectively. We observe that both the AUC and AUPRC values are fairly robust to changes in β, γ, δ, for a large range of their values. For optimum performance, hyperparameter tuning is highly desirable. Note that the value $\delta = 0$ which indicates that the cluster balance loss is not used gives lower AUC and AUPRC values compared to most other non-zero values, particularly for Sepsis. This shows that addition of this loss function term improves performance. We study the sensitivity to these hyperparameters on clustering performance in Appendix III. Figures 8a and 8b show the performance on ARDS; and Figures 8c and 8d show the performance on Sepsis datasets for all 4 combinations, TT, FT, TF and FF. The best performance, in 7 out of 8 cases, is achieved when the approach is used both in training and prediction (TT, blue). The advantages of our proposed loss terms and cluster-weighted training approach are thus empirically supported by our ablation studies.

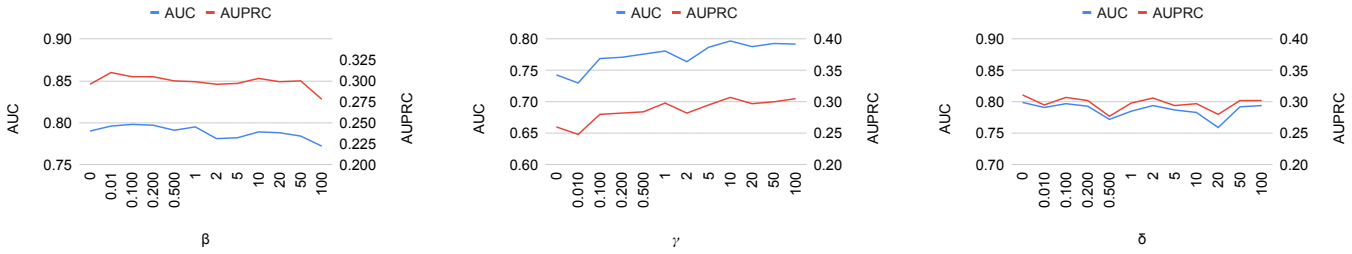
IV. DISCUSSION AND CONCLUSION

Our principal contribution is a neural model, EXPERTNET, for subtype-aware risk prediction. Leveraging the representation learning power of deep networks, EXPERTNET finds latent well clustered patient representations, and, simultaneously, cluster-specific classification networks are trained to predict risk outcomes. Standard techniques for training the network yield trivial clusterings or insufficient intra-cluster training data for classification – we address these challenges through a new loss function and training strategy that lead to improved predictive accuracy and clinically meaningful clusters.

Our model can be used in decision support systems within ICUs to potentially prevent complications, which in turn can improve patient outcomes and reduce the clinical and economic burden of these diseases. To demonstrate its practical utility, we evaluate the performance of EXPERTNET for predicting two important ICU complications, sepsis and ARDS. In our experiments, EXPERTNET outperforms state-of-the-art approaches for simultaneous clustering and classification as well as cluster-then-predict methods, on risk prediction. The clusters obtained by EXPERTNET are clinically meaningful and, in terms of cluster separability and feature discrimination, are comparable or better than those from competitive clustering methods. We show how subtype-specific risk factors can be determined using knowledge distillation. Our ablation studies demonstrate the benefits of our novel loss terms and training strategy. In addition, we show that such training strategies can also benefit other models – we discuss the case of DMNN in Appendix VII.

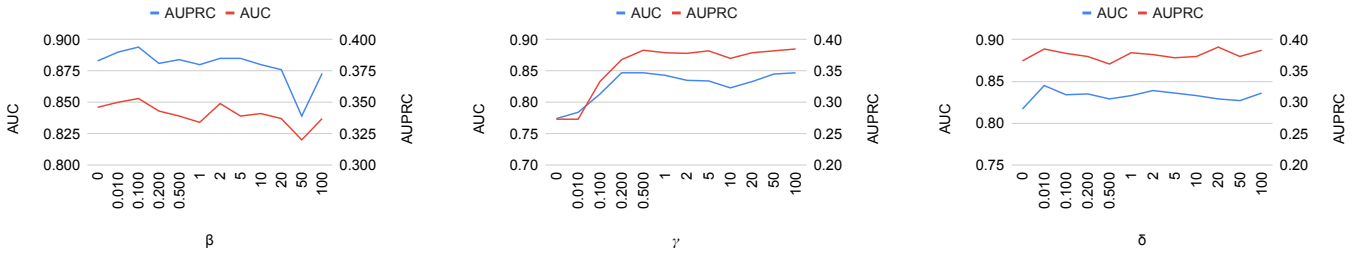
Our proposed approach has some limitations. First, similar to many clustering algorithms, the number of clusters has to be determined a priori. Often, a range of values for k is used and the best value is chosen based on application-specific requirements. The choice can be made using scores such as Silhouette score or Calinski-Harabasz Index. Second, our model is based on neural networks and is not inherently interpretable. Post-hoc XAI techniques can be used to interpret the model's predictions [34]. We have described one such technique, Knowledge Distillation [40], in the paper. Many other techniques exist such as LIME [62] and SHAP [63] may also be used. Third, in terms of evaluation, we have demonstrated performance gains on two complications only, using data from two hospitals. Generalizability to other complications, prediction settings, and hospitals needs to be evaluated in future work.

Future studies can extend this work in many ways. The neural architecture of EXPERTNET allows it to be generalized to other datatypes, without changing the remaining architecture which only utilizes latent representations. This may be evaluated, e.g., using text, images, sequential or multimodal data found in ICUs. Techniques to infer number of clusters during model training can be explored. Future work can also evaluate the efficacy of EXPERTNET in other contexts that require modeling underlying heterogeneous sub-populations.



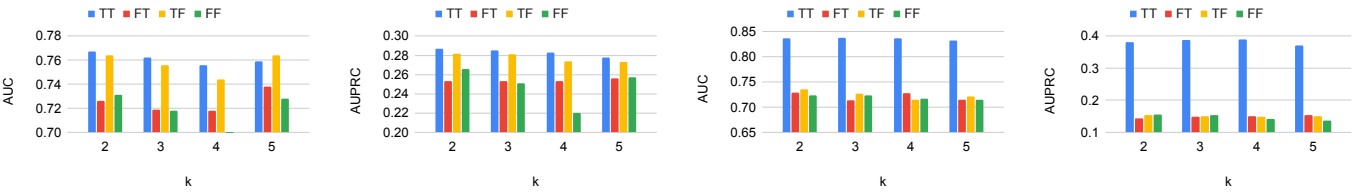
(a) AUC/AUPRC vs. β (b) AUC/AUPRC vs. γ (c) AUC/AUPRC vs. δ

Fig. 6: Sensitivity Analysis on the ARDS dataset



(a) AUC/AUPRC vs. β (b) AUC/AUPRC vs. γ (c) AUC/AUPRC vs. δ

Fig. 7: Sensitivity Analysis on the Sepsis dataset



(a) AUC vs. k (b) AUPRC vs. k (c) AUC vs. k (d) AUPRC vs. k

Fig. 8: Ablation Studies for Cluster weighted training approach on ARDS dataset (a–b) and Sepsis dataset (c–d)

REFERENCES

[1] S. A. Waldman and A. Terzic, "Healthcare evolves from reactive to proactive," *Clinical Pharmacology and Therapeutics*, vol. 105, no. 1, p. 10, 2019.

[2] R. Z. Goetzel, "Do prevention or treatment services save money? the wrong debate," *Health Affairs*, vol. 28, no. 1, pp. 37–41, 2009.

[3] K. E. Henry, D. N. Hager, P. J. Pronovost *et al.*, "A targeted real-time early warning score (TREWScore) for septic shock," *Science Translational Medicine*, vol. 7, no. 299, 2015.

[4] T. A. Carmo, I. B. Ferreira, R. C. Menezes *et al.*, "Derivation and validation of a novel severity scoring system for pneumonia at intensive care unit admission," *Clinical Infectious Diseases*, vol. 72, no. 6, pp. 942–949, 2021.

[5] G. J. Escobar, J. C. LaGuardia, B. J. Turk *et al.*, "Early detection of impending physiologic deterioration among patients who are not in intensive care: development of predictive models using data from an automated electronic medical record," *Journal of Hospital Medicine*, vol. 7, no. 5, pp. 388–395, 2012.

[6] M. Berman, A. Stamler, G. Sahar *et al.*, "Validation of the 2000 bernstein-parsonnet score versus the euroscore as a prognostic tool in cardiac surgery," *The Annals of Thoracic Surgery*, vol. 81, no. 2, pp. 537–540, 2006.

[7] L. L. Kirkland, M. Malinchoc, M. O'Byrne *et al.*, "A clinical deterioration prediction tool for internal medicine patients," *American Journal of Medical Quality*, vol. 28, no. 2, pp. 135–142, 2013.

[8] R. Morgan, F. Williams, and M. Wright, "An early warning scoring system for detecting developing critical illness," *Clin Intensive Care*, vol. 8, no. 2, p. 100, 1997.

[9] J.-L. Vincent, R. Moreno, J. Takala *et al.*, "The SOFA (Sepsis-related Organ Failure Assessment) score to describe organ dysfunction/failure," *Intensive Care Medicine*, vol. 22, pp. 707–710, 1996.

[10] M. M. Pollack, K. M. Patel, and U. E. Ruttimann, "Prism iii: an updated pediatric risk of mortality score," *Critical care medicine*, vol. 24, no. 5, pp. 743–752, 1996.

[11] A. Meyer, D. Zverinski, B. Pfahringer *et al.*, "Machine learning for real-time prediction of complications in critical care: a retrospective study," *The Lancet Respiratory Medicine*, vol. 6, no. 12, pp. 905–914, 2018.

[12] Z. M. Ibrahim *et al.*, "On classifying sepsis heterogeneity in the icu: insight using machine learning," *Journal of the American Medical Informatics Association*, vol. 27, no. 3, pp. 437–443, 2020.

[13] M. Y. Yan, L. T. Gustad, and Ø. Nytrø, "Sepsis prediction, early detection, and identification using clinical text for machine learning: a systematic review," *Journal of the American Medical Informatics Association*, vol. 29, no. 3, pp. 559–575, 2022.

[14] S. Bhattacharya, V. Rajan, and H. Shrivastava, "ICU mortality prediction: a classification algorithm for imbalanced datasets," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, no. 1, 2017.

[15] H. Suresh, J. J. Gong, and J. V. Guttag, "Learning tasks for multitask learning: Heterogenous patient populations in the ICU," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 802–810.

[16] Y. Sun, R. Kaur, S. Gupta *et al.*, "Development and validation of high definition phenotype-based mortality prediction in critical care units," *JAMIA open*, vol. 4, no. 1, p. ooab004, 2021.

[17] Z. Wang and B. Yao, "Multi-branching temporal convolutional network for sepsis prediction," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 2, pp. 876–887, 2022.

[18] S. Liu *et al.*, "Dynamic sepsis prediction for intensive care unit patients using xgboost-based model with novel time-dependent features," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 8, pp. 4258–4269, 2022.

[19] Z. Xiao, X. Xu *et al.*, "A federated learning system with enhanced feature extraction for human activity recognition," *Knowledge-Based Systems*, vol. 229, p. 107338, 2021.

[20] Z. Xiao *et al.*, "Rtfn: a robust temporal feature network for time series classification," *Information sciences*, vol. 571, pp. 65–86, 2021.

[21] H. Xing, Z. Xiao, R. Qu, Z. Zhu, and B. Zhao, "An efficient federated distillation learning system for multitask time series classification," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–12, 2022.

[22] C. J. Paoli *et al.*, "Epidemiology and costs of sepsis in the United States — an analysis based on timing of diagnosis and severity level," *Critical Care Medicine*, vol. 46, no. 12, p. 1889, 2018.

[23] R. Snyderman, "Personalized health care: from theory to practice," *Biotechnology Journal*, vol. 7, no. 8, pp. 973–979, 2012.

[24] S. B. Cho, S. C. Kim, and M. G. Chung, "Identification of novel population clusters with different susceptibilities to type 2 diabetes and their impact on the prediction of diabetes," *Scientific Reports*, vol. 9, no. 1, pp. 1–9, 2019.

[25] K. Ng *et al.*, "Personalized predictive modeling and risk factor identification using patient similarity," in *AMIA Summits on Translational Science*, 2015.

[26] Z. Huang and W. Dong, "Adversarial mace prediction after acute coronary syndrome using electronic health records," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 5, pp. 2117–2126, 2019.

[27] X. Li, D. Zhu, and P. Levy, "Predicting clinical outcomes with patient stratification via deep mixture neural networks," in *AMIA Summits on Translational Science*, 2020.

[28] S. Saria and A. Goldenberg, "Subtyping: What it is and its role in precision medicine," *IEEE Intelligent Systems*, vol. 30, pp. 70–75, 2015.

[29] C. Lee and M. Van Der Schaar, "Temporal phenotyping using deep predictive clustering of disease progression," in *Thirty-seventh International Conference on Machine Learning (ICML)*, 2020.

[30] Y. Huang, Y. Liu, P. A. Steel *et al.*, "Deep significance clustering: a novel approach for identifying risk-stratified and predictive patient subgroups," *Journal of the American Medical Informatics Association*, vol. 28, no. 12, pp. 2641–2653, 2021.

[31] X. Guo, L. Gao, X. Liu, and J. Yin, "Improved deep embedded clustering with local structure preservation," in *International Joint Conference on Artificial Intelligence (IJCAI)*, 2017.

[32] B. Yang, X. Fu *et al.*, "Towards k-means-friendly spaces: Simultaneous deep learning and clustering," in *34th International Conference on Machine Learning (ICML)*, 2017.

[33] Q. Ma, J. Zheng, S. Li, and G. W. Cottrell, "Learning representations for time series clustering," *Advances in neural information processing systems*, vol. 32, 2019.

[34] C. Molnar, *Interpretable Machine Learning*, 2022. [Online]. Available: https://christophm.github.io/interpretable-ml-book

[35] H. Hakkoum, I. Abnane, and A. Idri, "Interpretability in the medical field: A systematic mapping and review study," *Applied Soft Computing*, vol. 117, p. 108391, 2021.

[36] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *33rd International Conference on Machine Learning (ICML)*, 2016.

[37] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.

[38] Z. Che, S. Purushotham, R. Khemani, and Y. Liu, "Interpretable deep models for ICU outcome prediction," in *AMIA Annual Symposium Proceedings*, 2016, p. 371.

[39] G. Hinton, O. Vinyals, J. Dean *et al.*, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, vol. 2, no. 7, 2015.

[40] L. Wang and K.-J. Yoon, "Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 6, pp. 3048–3068, 2021.

[41] CDC, "What is sepsis?" https://www.cdc.gov/sepsis/what-is-sepsis.html, 2021.

[42] K. E. Rudd *et al.*, "Global, regional, and national sepsis incidence and mortality, 1990–2017: analysis for the global burden of disease study," *The Lancet*, vol. 395, no. 10219, pp. 200–211, 2020.

[43] G. Bellani *et al.*, "Epidemiology, patterns of care, and mortality for patients with acute respiratory distress syndrome in intensive care units in 50 countries," *JAMA*, vol. 315, no. 8, pp. 788–800, 2016.

[44] T. Pham and G. D. Rubenfeld, "Fifty years of research in ards. the epidemiology of acute respiratory distress syndrome. a 50th birthday review," *American Journal of Respiratory and Critical Care Medicine*, vol. 195, no. 7, pp. 860–870, 2017.

[45] E. Fan, D. Brodie, and A. S. Slutsky, "Acute respiratory distress syndrome: advances in diagnosis and treatment," *JAMA*, vol. 319, no. 7, pp. 698–710, 2018.

[46] K. W. Hendrickson, I. D. Peltan, and S. M. Brown, "The epidemiology of acute respiratory distress syndrome before and after coronavirus disease 2019," *Critical Care Clinics*, vol. 37, no. 4, pp. 703–716, 2021.

[47] A. E. Johnson, T. J. Pollard, L. Shen *et al.*, "MIMIC-III, a freely accessible critical care database," *Scientific Data*, vol. 3, no. 1, pp. 1–9, 2016.

[48] M. A. Reyna, C. Josef, S. Seyedi *et al.*, "Early prediction of sepsis from clinical data: the physionet/computing in cardiology challenge 2019," no. 2, 2020, pp. 210–217.

[49] C. W. Seymour, V. X. Liu, T. J. Iwashyna *et al.*, "Assessment of clinical criteria for sepsis: for the third international consensus definitions for sepsis and septic shock (sepsis-3)," *JAMA*, vol. 315, no. 8, pp. 762–774, 2016.

[50] M. Shankar-Hari, G. S. Phillips, M. L. Levy *et al.*, "Developing a new definition and assessing new clinical criteria for septic shock: for the third international consensus definitions for sepsis and septic shock (sepsis-3)," *JAMA*, vol. 315, no. 8, pp. 775–787, 2016.

[51] M. Singer, C. S. Deutschman, C. W. Seymour *et al.*, "The third international consensus definitions for sepsis and septic shock (sepsis-3)," *JAMA*, vol. 315, no. 8, pp. 801–810, 2016.

[52] N. D. Ferguson, E. Fan, L. Camporota *et al.*, "The berlin definition of ards: an expanded rationale, justification, and supplementary material," *Intensive Care Medicine*, vol. 38, no. 10, pp. 1573–1582, 2012.

[53] G. Rubenfeld *et al.*, "Acute respiratory distress syndrome. The Berlin definition," *JAMA*, vol. 307, no. 23, pp. 2526–2533, 2012.

[54] D. Bertsimas *et al.*, "Predicting inpatient flow at a major hospital using interpretable analytics," *Manufacturing & Service Operations Management*, vol. 24, no. 6, pp. 2809–2824, 2022.

[55] P. Yang *et al.*, "A new method for identifying the acute respiratory distress syndrome disease based on noninvasive physiological parameters," *PloS One*, vol. 15, no. 2, 2020.

[56] Z. Zhang, "Identification of three classes of acute respiratory distress syndrome using latent class analysis," *PeerJ*, vol. 6, p. e4592, 2018.

[57] N. Dimitrova *et al.*, "Latent class analysis of ards subphenotypes: analysis of data from two randomized controlled trials carolyn," *Lancet Respiratory Medicine*, vol. 32, pp. 736–740, 2017.

[58] L. Li et al., "Identification of type 2 diabetes subgroups through topological analysis of patient similarity," *Science Translational Medicine*, vol. 7, no. 311, 2015.

[59] L. McInnes, J. Healy, and J. Melville, "UMAP: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, 2018.

[60] B. Ozenne, F. Subtil, and D. Maucort-Boulch, "The precision–recall curve overcame the optimism of the receiver operating characteristic curve in rare diseases," *Journal of clinical epidemiology*, vol. 68, no. 8, pp. 855–859, 2015.

[61] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets," *PLOS ONE*, vol. 10, no. 3, pp. 1–21, 03 2015. [Online]. Available: https://doi.org/10.1371/journal.pone.0118432

[62] M. T. Ribeiro, S. Singh, and C. Guestrin, "" why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 1135–1144.

[63] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems*, vol. 30, 2017.

APPENDIX I
FEATURE LISTS

See tables V, VI for full feature lists for ARDS and Sepsis datasets.

TABLE V: Complete Feature Set for ARDS Dataset

Feature	Mean (Std)	Feature	Mean (Std)
Alanine aminotransferase (IU/L)	66.672 (385.139)	Mean Airway Pressure (mmHg)	12.141 (2.153)
Albumin (g/dL)	3.196 (0.485)	Mean blood pressure (mmHg)	79.087 (15.890)
Alkaline phosphate (IU/L)	111.108 (95.917)	Mean corpuscular hemoglobin (picograms/cell)	29.938 (3.885)
Anion gap (mEq/L)	13.430 (3.482)	Mean corpuscular hemoglobin conc. (pg/cell)	33.837 (1.562)
Antibiotics (Minutes)	-75336.866 (43209.090)	Mean corpuscular volume (NA)	89.381 (6.708)
Asparate aminotransferase (IU/L)	64.435 (415.293)	Minute Volume (L/min)	6.873 (3.290)
Band Forms	5.993 (3.859)	Monocytes (%)	4.729 (2.864)
Base excess	-0.271 (5.782)	Neutrophils (%)	69.477 (17.137)
Basophils	0.237 (0.382)	Norepinephrine (pg/mL)	0.003 (0.027)
Bicarbonate (mEq/L)	24.271 (4.397)	PEEP (cm H_2O)	3.755 (1.081)
Bilirubin Total (mg/dL)	0.965 (2.475)	PIP (cm H_2O)	13.448 (4.633)
Blood culture	-53484.533 (54702.938)	PTT (sec)	35.220(18.613)
Blood urea nitrogen (mg/dL)	24.884 (20.811)	PaCO2 (mmHg)	40.234 (6.857)
Calcium (mg/dL)	8.474 (0.783)	PaO2 (mmHg)	120.851 (60.250)
Cardiac Index	3.279 (0.208)	Phosphate (mg/dL)	3.491 (1.184)
Central Venous Pressure (mm Hg)	5.868 (12.825)	Plateau Pressure (mm Hg)	27.104 (3.192)
Chloride mEq/L	105.039 (5.770)	Platelets ($10^9/L$)	220.524 (110.209)
Cholesterol HDL (mg/dL)	35.173 (6.190)	Potassium (mmol/L)	4.112 (1.840)
Cholesterol LDL (mg/dL)	68.632 (14.871)	RBC count ($10^6/\mu L$))	3.612 (0.718)
Cholesterol Total (mg/dL)	106.730 (25.995)	Respiratory rate (/min)	19.222 (5.312)
Creatinine (mg/dL)	1.467 (2.888)	SO_2 (%)	96.631 (7.276)
D Dimer (ng/mL)	286.833 (764.986)	Sodium (mEq/L)	138.610 (4.404)
Diastolic blood pressure (mm Hg)	61.754 (15.017)	Stroke Volume (mL)	79.412 (6.073)
Dobutamine (mg/ml)	0.023 (0.388)	Systemic Vascular Resistance Index ($dynes/sec/cm^{-5}$)	1047.545 (228.710)
Dopamine (pg/mL)	0.122 (0.980)	Systolic blood pressure (mmHg)	126.614 (794.495)
Eosinophils (10^3 cells/mcL)	1.762 (1.911)	Temperature (C)	36.816 (0.777)
Epinephrine (pg/mL)	0.000 (0.002)	Tidal Volume (mL)	507.246 (113.622)
Extubated (%)	82.7	TroponinT (ng/mL)	0.273 (1.198)
FiO2 (%)	29.1 (19.3)	Urine output (ml)	86.182 (120.205)
Fibrinogen (mg/dL)	283.299 (77.511)	Ventilator (%)	19.2 (39.4)
GCS Eye (NA)	3.790 (0.599)	Weight (kg)	80.201 (21.920)
GCS Motor (NA)	5.855 (0.628)	White blood cell count ($10^9/L$)	11.170 (8.531)
GCS Total (NA)	14.557 (1.596)	pH (NA)	7.427 (1.811)
GCS Verbal (NA)	4.526 (1.138)	pH Urine (NA)	6.104 (0.812)
Glucose (mg/dL)	134.444 (50.205)	Age (years)	63.969 (17.605)
Heart Rate (#beats/min)	83.729 (16.979)	Ethnicity_0 (Asian)	11.6%
Height	169.683 (6.837)	Ethnicity_1 (Black)	2.5%
Hematocrit (%)	32.050 (5.262)	Ethnicity_2 (Hispanic)	10.7%
Hemoglobin (gm/dL)	10.812 (1.871)	Ethnicity_3 (Other)	3.5%
INR (NA)	1.376 (1.018)	Ethnicity_4 (White)	71.5%
Lactate (mmol/L)	1.843 (1.181)	Gender_0	0
Lactate Dehydrogenase (IU/L)	297.522 (417.274)	Gender_1 (Female)	45.6%
Lymphocytes (NA)	19.108 (11.913)	Gender_2 (Male)	54.3%
Magnesium (mmol/L)	2.044 (2.734)	Gender_3	0

TABLE VI: Complete Feature Set for Sepsis Dataset

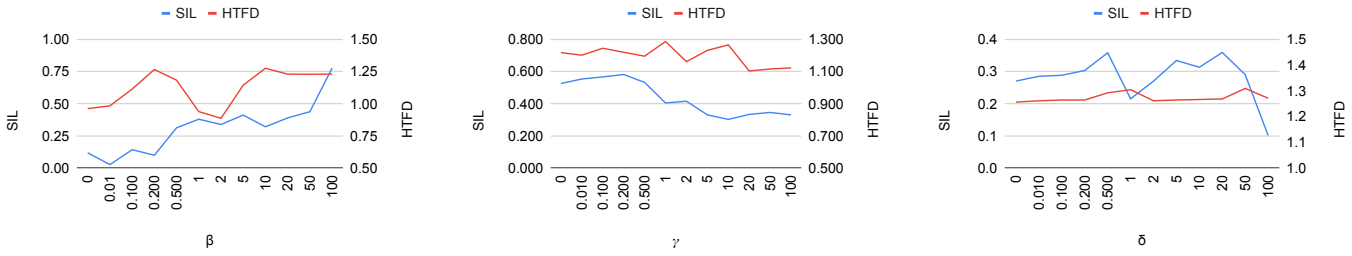
Feature	Mean (Std)	Feature	Mean (Std)
HR (beats per minute)	84.339 (17.044)	Lactate (mg/dL)	0.598 (1.230)
O2Sat (%)	96.930 (3.572)	Magnesium (mmol/dL)	1.748 (0.794)
Temp (Deg C)	36.520 (3.807)	Phosphate (mg/dL)	2.359 (1.991)
SBP (mm Hg)	121.953 (26.734)	Potassium (mmol/L)	3.903 (1.066)
MAP (mm Hg)	82.102 (16.751)	Bilirubin_total(mg/dL)	0.491 (1.892)
DBP (mm Hg)	51.670 (28.089)	TroponinI (ng/mL)	1.059 (9.348)
Resp (breaths per minute)	18.912 (5.092)	Hct (%)	29.965 (9.499)
EtCO2 (mm Hg)	2.712 (9.559)	Hgb (g/dL)	9.96 (3.327)
BaseExcess (mmol/L)	-0.081 (2.276)	PTT (seconds)	18.439 (23.943)
HCO3 (mmol/L)	12.405 (12.464)	WBC (count*$10^3/\mu L$)	10.347 (7.157)
FiO2 (%)	0.215 (0.274)	Fibrinogen (mg/dL)	32.249 (103.411)
pH	3.509 (3.691)	Platelets (count*$10^3/\mu L$)	192.579 (111.598)
PaCO2 (mmol/L)	18.645 (20.986)	Age (Years)	62.145 (16.438)
SaO2 (%)	29.473 (43.734)	Gender (F/M)	55.8%/44.2%
AST (IU/L)	50.892 (364.938)	Unit1 (ICU unit (MICU))	0.304 (0.460)
BUN (mg/dL)	21.176 (18.888)	Unit2 (ICU unit (SICU))	0.308 (0.461)
Alkalinephos (IU/L)	31.738 (76.607)	HospAdmTime (Hours)	-52.673 (139.169)
Calcium (mg/dL)	6.738 (3.423)	Sofa_O2	3.339 (1.161)
Chloride (mmol/L)	57.297 (52.772)	Sofa_MAP	0.218 (0.413)
Creatinine (mg/dL)	1.362 (1.758)	Sofa_Bilirubin	2.859 (1.716)
Bilirubin_direct (mg/dL)	0.047 (0.575)	Sofa_Creatinin	0.776 (1.299)
Glucose (mg/dL)	125.480 (51.189)	Sofa_Platelets	0.658 (1.157)

APPENDIX II
HYPERPARAMETERS

In our experiments, we set $\delta = 0.1, \beta = 10, \gamma = 5$ respectively. These were obtained after evaluating the performance of EXPERTNET for selecting values on the validation sets. Section III-A.8 has more details on sensitivity of EXPERTNET to these hyperparameters. For DMNN and EXPERTNET, the autoencoder has 3 layers of sizes $128 - 64 - 20 - 64 - 128$ and a six-layered local predictor network of size $20 - 128 - 64 - 32 - 16 - 1$. The local prediction networks use Softmax and ReLU activation functions and are not regularized. For AC-TPC, we use the hyperparameters suggested by the authors found using sensitivity analysis. The learning rate for EXPERTNET is set to $2e - 3$. The predictor, selector, and encoder networks in AC-TPC are FCNs with $64 - 64$, $64 - 64$, and $32 - 32$ neurons respectively. In UMAP, the parameters number of neighbors and min_dist (minimum distance apart that points are allowed to be in the low dimensional representation) are set to 15 and 0.1 respectively.

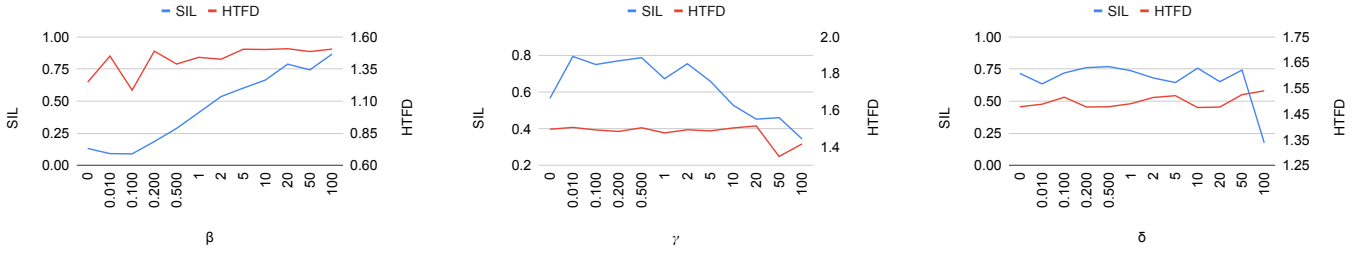
APPENDIX III
SENSITIVITY ANALYSIS

Figure 9 and 10 present sensitivity analysis plots for clustering performance measured by Silhouette (SIL) index and HTFD scores. As expected, the clustering scores improve with increasing β (clustering loss weight) and decrease with increasing γ (classification loss weight). The relationship of SIL and HTFD with δ is not linear. SIL and HTFD fall sharply at high value of δ ($\delta > 50$) but remain more or less consistent for other values. All experiments are averaged over 5-fold test sets.



(a) SIL/HTFD vs. β (b) SIL/HTFD vs. γ (c) SIL/HTFD vs. δ

Fig. 9: Sensitivity Analysis on the ARDS dataset



(a) AUC/AUPRC vs. β (b) AUC/AUPRC vs. γ (c) AUC/AUPRC vs. δ

Fig. 10: Sensitivity Analysis on the Sepsis dataset

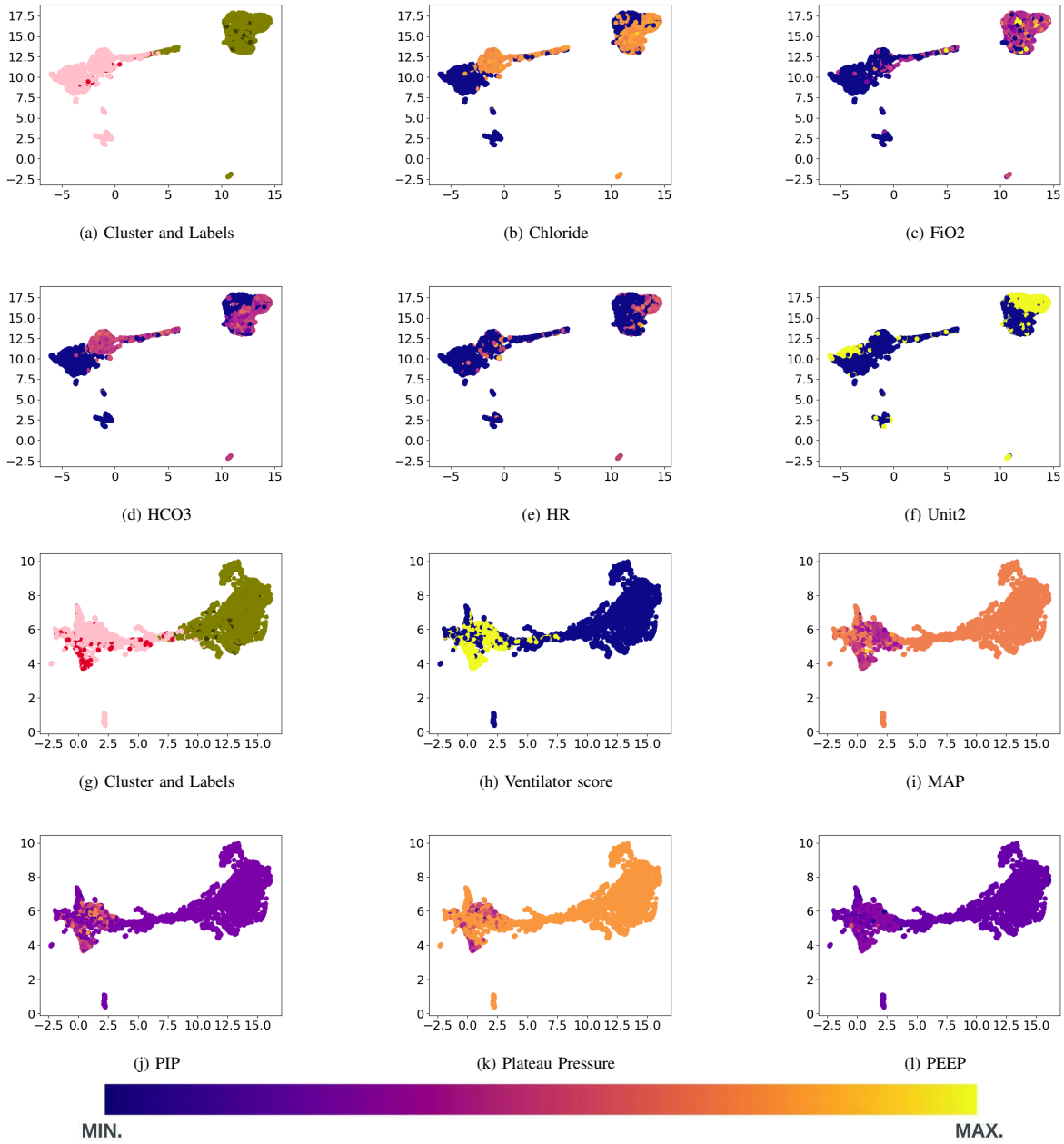
APPENDIX IV
SUBTYPING

We qualitatively analyze the subtypes (clusters) for $k = 2$ (Table VII). Tables VII(a) and VII(b) present the clinical variables that are significantly different across the subtypes for ARDS and Sepsis respectively.

Visualization of the clusters demonstrates the discriminatory power of the learnt embeddings. Fig. 11a and 11g, for sepsis and ARDS respectively, show the embeddings of the clusters and also depict the class labels within each cluster. We observe that the clusters are well separated and within each cluster, there are members of both classes (those with and without the diseases). Fig. 12 shows plots of 2D UMAP embeddings plots of Sepsis and ARDS data (in feature space) colored based on clusters found by EXPERTNET for $k = 2$ (color coded Green for Cluster 1 and Pink for Cluster 2. Dark green and Dark Pink points represent patients who get Sepsis/ARDS in their respective cohorts).

TABLE VII: Clinical variables specific to subtypes in ARDS and Sepsis datasets for $k = 2$.

a) Clinical variables significantly specific to ARDS subtypes 1 and 2		
Clinical Variables **Cluster Details**	**Mean or % Subtype 1** $(\|C_1\| = 5576, \%D = 12.7)$	**Mean or % Subtype 2** $(\|C_2\| = 10643, \%D = 4.7)$
Ventilator	0.538 ± 0.499	0.014 ± 0.116
Mean Airway Pressure	10.482 ± 3.038	13.0 ± 0.0
PIP	16.262 ± 7.129	12.0 ± 0.0
PEEP	4.217 ± 1.856	3.507 ± 0.143
Plateau Pressure	25.372 ± 5.041	28.0 ± 0.0
GCS Verbal	4.048 ± 1.608	4.779 ± 0.657
Systemic Vascular Resistance Index	1138.88 ± 379.591	1001.447 ± 29.097
FiO2	0.364 ± 0.233	0.253 ± 0.143
Ethnicity_3	0.103 ± 0.304	0.0 ± 0.017
Tidal Volume	519.973 ± 167.632	499.959 ± 4.867
(b) Clinical variables significantly specific to Sepsis subtypes 1 and 2		
Cluster Details	$(\|C_1\| = 11674, \%D = 3.3)$	$(\|C_2\| = 10477, \%D = 8.9)$
pH	0.582 ± 1.994	6.74 ± 2.093
PaCO2	2.531 ± 9.809	36.571 ± 14.763
SaO2	0.93 ± 8.467	61.249 ± 45.248
FiO2	0.048 ± 0.145	0.401 ± 0.268
Sofa_O2	4.0 ± 0.019	2.601 ± 1.354
Lactate	0.062 ± 0.324	1.187 ± 1.55
Chloride	38.75 ± 50.772	77.712 ± 47.061
Unit2	0.159 ± 0.365	0.478 ± 0.5
BaseExcess	0.062 ± 0.954	-0.213 ± 3.165
HCO3	9.153 ± 12.201	15.989 ± 11.769



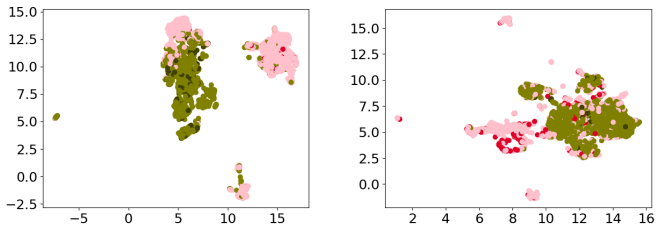
(a) Cluster and Labels (b) Chloride (c) FiO2

(d) HCO3 (e) HR (f) Unit2

(g) Cluster and Labels (h) Ventilator score (i) MAP

(j) PIP (k) Plateau Pressure (l) PEEP

Fig. 11: Top (a-f) for Sepsis, Bottom (g-l) for ARDS. (a) and (g) show the 2-dimensional UMAP embeddings in representation space depicting overall clusters for $k = 2$ (color coded Green for Cluster 1 and Pink for Cluster 2. Dark green and Dark Pink points represent patients who get Sepsis/ARDS in their respective cohorts.) Other figures show variation of selected features (from Table VII) for Sepsis (b-f) and for ARDS (g-l) data. Each point is color coded according to the scale presented at the bottom (the minimum value is shaded blue while the maximum value is shaded yellow). Image best viewed in color.



(a) Sepsis Dataset (b) ARDS Dataset

Fig. 12: Plots of 2-dimensional UMAP embeddings of Sepsis and ARDS data (in feature space) colored based on clusters found by EXPERTNET for $k = 2$ (color coded Green for Cluster 1 and Pink for Cluster 2. Dark green and Dark Pink points represent patients who get Sepsis/ARDS in their respective cohorts). Compare with Fig. 11 (a) and (g).

APPENDIX V
RISK STRATIFICATION

Risk stratification across the subtypes differ for Sepsis and ARDS and with k as shown in Table VII. In the case of Sepsis, for $k = 2$, in clusters 1 and 2, 5.7% and 6.5% develop sepsis respectively. In the case of ARDS, for $k = 2$ we observe that patients in subtype 1 are more likely to develop ARDS (11.6% as compared to 4.7% for subtype 1). Note that the risk stratification is inferred in an unsupervised manner, i.e., without explicitly adding a constraint for intra-cluster risk homogeneity. Figure 11 shows the distribution of significant variables across the two sub populations. Table VIII shows the variables that have the highest predictive power for the two sub populations.

TABLE VIII: Top 10 most important features for ARDS and Sepsis prediction in for 2 clusters. Features common in two clusters are highlighted in yellow, those common in three clusters are highlighted in blue, while the rest are unique to their respective clusters. $|C_i|$ indicates the size of the i^{th} cluster. $\%D$ indicates the number of positive class labels, i.e., patients diagnosed with the disease (sepsis/ARDS), in the cluster.

ARDS		Sepsis	
$\|C_1\| = 5576$ $\%D = 12.7$	$\|C_2\| = 10643$ $\%D = 4.7$	$\|C_1\| = 11674$ $\%D = 3.3$	$\|C_2\| = 10477$ $\%D = 8.9$
GCS Verbal	Platelets	SBP	Temp
GCS Eye	White blood cell count	Temp	HospAdmTime
GCS Total	Red blood cell count	HospAdmTime	SBP
Heart Rate	Hematocrit	WBC	Age
PaO2	Urine output	Glucose	Resp
Temperature	PTT	Age	HR
Platelets	Temperature	HR	WBC
Age	Mean blood pressure	Hct	Glucose
Extubated	Respiratory rate	Resp	Platelets
Glucose	Glucose	BUN	BUN

APPENDIX VI
EXPERTNET COMPUTATIONAL COMPLEXITY

We derive the computational time complexity of Algorithm 1 in this section. To simplify the computations, we assume that the EXPERTNET encoder, decoder, the k local networks have the same network architecture and the training algorithm terminates after E epochs. Each neural network has L layers, each having s_l number of neurons and trained on N points.

In line 3, the cluster centroids are computed by the k-means algorithm, thus the computational cost incurred is $\mathcal{O}(T_{KM} \cdot kN)$ assuming that it took T_{kM} iterations for k-means to converge.

Computing matrices P and Q in line 4 incur a cost $\mathcal{O}(kN)$.

In line 8, forward propagations through the encoder incur a cost $O\left(N \cdot E \sum_{\ell=1}^{L} s_\ell s_{\ell-1}\right)$.

In line 9-10, Training k local expert networks for T_{sub} sub-iterations for E main epochs incurs a cost of $O\left(N \cdot E \cdot T_{sub} \cdot k \sum_{\ell=1}^{L} s_\ell s_{\ell-1}\right)$

In line 8, the total loss is backpropagated through the encoder, local networks and the decoder. The computational cost incurred is $O\left(N \cdot E \cdot (k+2) \sum_{\ell=1}^{L} s_\ell s_{\ell-1}\right)$

Finally, in line 12, the cluster membership matrix P is updated in every epoch. The computational cost incurred is $O(E \cdot NK)$.

Thus the total computational cost of training EXPERTNET is

$$
\begin{aligned}
&= O\left(KN + N \cdot E \sum_{\ell=1}^{L} s_\ell s_{\ell-1} + N \cdot E \cdot T_{sub} \cdot k \sum_{\ell=1}^{L} s_\ell s_{\ell-1} + N \cdot E \cdot (k+2) \sum_{\ell=1}^{L} s_\ell s_{\ell-1} + E \cdot Nk\right) \\
&= O\left(N \cdot E \cdot (k + 3 + T_{sub} \cdot k) \sum_{\ell=1}^{L} s_\ell s_{\ell-1} + E \cdot Nk\right) \\
&= O\left(N \cdot E \cdot T_{sub} \cdot k \sum_{\ell=1}^{L} s_\ell s_{\ell-1} + E \cdot Nk\right)
\end{aligned}
$$

APPENDIX VII
COMPARISON WITH DMNN

First we recall the architecture of DMNN DMNN consists of a neural network (embedding network) to embed input features, followed by a softmax layer (gating) to indicate cluster membership. The feature embeddings are fed into k local classifiers (k is the assumed number of clusters) which are trained with losses weighted by the gating values.

The authors themselves acknowledge the problems with this approach [27]: (a) The deep network can easily overfit leading to poor generalization and (b) Gating mechanism may degenerate leading to all data points collapsing into a single cluster (thus, no subtypes are found). Their solution to overcome these problems is to first train an autoencoder, use K-Means to obtain cluster labels, then train the encoder to predict cluster labels. Thus a pre-trained encoder is used within DMNN to train the local networks. Even with this approach, DMNN has poor clustering and classification results as shown in the 'Original' column of Table IX below (and Table I of our paper).

To further study the effects of training strategies in our method on DMNN, we perform the following experiments.

We made two modifications to DMNN to investigate their effects on performance:

1) We **backpropagate** the loss from local classifiers to the **Encoder module** (as opposed to the **backpropagation to gating network** as mentioned in the original DMNN paper).
2) Next, we added the cluster balance loss (proposed in our paper) to their loss and trained DMNN.

The experiments follow the settings given in the paper (85:15 train-test split). We report averages over 5 folds for both Sepsis and ARDS datasets, for $k = 2, 3, 4, 5$.

Table IX shows the results of DMNN (without any modifications) in column 1, with our two modifications above in columns 2 and 3 respectively. In the final column we show the results for EXPERTNET for comparison. Comparing columns 1 and 2, we observe that **backpropagating** the loss from local classifiers to the **Encoder module** leads to improvement in performance. But it comes at the cost of the degeneracy in gating mechanism i.e. the gating network tends to direct all data points to one local classifier instead of distributing them amongst all the local networks.

On comparing columns 2 and 3 we see that incorporating the cluster balance loss mitigates the issue of empty clusters but the overall prediction performance of EXPERTNET is still better than that of DMNN. This improvement may be attributed to other differences in EXPERTNET such as DEC-based neural clustering and our novel cluster-weighted training strategy.

TABLE IX: Comparison of AUC, AUPRC and SIL scores for 1. DMNN, 2. DMNN with modification where we add backpropagation to encoder, 3. DMNN with backpropagation and Cluster Balance Loss, 4. ExpertNet

Dataset	**k**	**1. DMNN (w/o Backprop)**			**2. w/ Backprop**			**3. Backprop+CBL**			**4. EXPERTNET**		
		AUC	**AUPRC**	**SIL**	**AUC**	**AUPRC**	**SIL**	**AUC**	**AUPRC**	**SIL**	**AUC**	**AUPRC**	**SIL**
ards24	2	0.733	0.267	0.000	0.797	0.312	0.0	0.709	0.213	0.283	0.785	0.291	0.404
ards24	3	0.796	0.311	0.076	0.795	0.317	0.0	0.609	0.162	0.153	0.766	0.28	0.423
ards24	4	0.796	0.314	0.000	0.798	0.31	0.0	0.623	0.162	0.103	0.772	0.284	0.373
ards24	5	0.794	0.301	0.085	0.71	0.26	0.077	0.589	0.126	0.017	0.784	0.29	0.244
sepsis24	2	0.698	0.253	0.000	0.73	0.305	0.0	0.849	0.372	0.319	0.861	0.426	0.649
sepsis24	3	0.838	0.376	0.000	0.845	0.374	0.0	0.745	0.232	0.274	0.854	0.414	0.635
sepsis24	4	0.849	0.389	0.000	0.845	0.383	0.0	0.686	0.19	0.204	0.853	0.418	0.623
sepsis24	5	0.838	0.360	0.000	0.85	0.387	0.091	0.669	0.188	0.153	0.847	0.409	0.668