Learning Using Partially Available Privileged Information and Label Uncertainty: Application in Detection of Acute Respiratory Distress Syndrome

Elyas Sabeti, *Member, IEEE*, Joshua Drews, Narathip Reamaroon, Elisa Warner, Michael Sjoding, Jonathan Gryak, and Kayvan Najarian, *Senior, IEEE*

***Abstract*— Acute respiratory distress syndrome (ARDS) is a fulminant inflammatory lung injury that develops in patients with critical illnesses, affecting 200,000 patients in the United States annually. However, a recent study suggests that most patients with ARDS are diagnosed late or missed completely and fail to receive life-saving treatments. This is primarily due to the dependency of current diagnosis criteria on chest x-ray, which is not necessarily available at the time of diagnosis. In machine learning, such an information is known as Privileged Information - information that is available at training but not at testing. However, in diagnosing ARDS, privileged information (chest x-rays) are sometimes only available for a portion of the training data. To address this issue, the Learning Using Partially Available Privileged Information (LUPAPI) paradigm is proposed. As there are multiple ways to incorporate partially available privileged information, three models built on classical SVM are described. Another complexity of diagnosing ARDS is the uncertainty in clinical interpretation of chest x-rays. To address this, the LUPAPI framework is then extended to incorporate label uncertainty, resulting in a novel and comprehensive machine learning paradigm - Learning Using Label Uncertainty and Partially Available Privileged Information (LULUPAPI). The proposed frameworks use Electronic Health Record (EHR) data as regular information, chest x-rays as partially available privileged information, and clinicians' confidence levels in ARDS diagnosis as a measure of label uncertainty. Experiments on an ARDS dataset demonstrate that both the LUPAPI and LULUPAPI models outperform SVM, with LULUPAPI performing better than LUPAPI.**

***Index Terms*— ARDS, Chest X-Ray, EHR Data, SVM, Partially Available Privileged Information, Label Uncertainty**

This work was partially supported by the National Science Foundation under Grant No. 1722801 and by the National Institute of Health under Grant NHLBI K01HL136687.

E. Sabeti is with the Michigan Institute for Data Science (MIDAS), University of Michigan, Ann Arbor, MI, 48109 USA (e-mails: sabeti@umich.edu).

J. Drews, N. Reamaroon and E. Warner are with the Department of Computational Medicine and Bioinformatics, University of Michigan, Ann Arbor, MI, 48109 USA (e-mails: {joshdr,nreamaro,elisawa,gryakj}@umich.edu).

M. Sjoding is with the Department of Internal Medicine and Michigan Center for Integrative Research in Critical Care (MCIRCC), University of Michigan, Ann Arbor, MI, 48109 USA (e-mail: msjoding@umich.edu).

J.Gryak is with the Department of Computational Medicine and Bioinformatics and the Michigan Institute for Data Science (MIDAS), University of Michigan, Ann Arbor, MI, 48109 USA (email: gryakj@umich.edu).

K. Najarian is with the Department of Computational Medicine and Bioinformatics, Department of Emergency Medicine, Department of Electrical Engineering and Computer Science, and Michigan Center for Integrative Research in Critical Care (MCIRCC), University of Michigan, Ann Arbor, MI, 48109 USA (e-mail: kayvan@umich.edu).

I. Introduction

ACUTE respiratory distress syndrome (ARDS) is a fulminant inflammatory lung injury that develops in patients with critical illnesses including sepsis, pneumonia, and trauma [1], [2]. Each year, 200,000 patients in the United States suffer from ARDS, with mortality rate of 30-40% [3]. While simple interventions such as reducing ventilator tidal volume have been shown to improve patient outcomes [4], physician recognition of ARDS ranges from 50 to 80 percent. Consequently, many patients do not receive these life-saving treatments [5]. Hence, the development of a clinical decision support system that employs machine learning to flag patients at risk for ARDS and prompt clinicians to administer treatment is of immediate interest.

Current ARDS diagnostic criteria require chest x-ray results as an input, as they provide critical information about whether ARDS is present [6], [7]. However, chest x-rays may not always be available, particularly at the early stages of care. Moreover, clinicians may be equivocal or even disagree about the diagnosis in some patients using chest x-ray [8], which may result in incorrect labels being provided by an expert. Despite these concerns, if a chest x-ray is available, it is informative and should be integrated into the model training process. An additional challenge when developing an ARDS detection systems is there exists no gold standard available to determine which patients develop ARDS. The diagnosis of ARDS is made based on clinical criteria [9], but even clinical experts may disagree or have uncertainty about the diagnosis in some patients [8]. Information about the uncertainty level of an ARDS diagnostic label provided by clinical experts can also prove useful when training a system for ARDS detection [10]. Additionally, one potentially effective yet underutilized method of assisting physicians in the recognition of ARDS is the analysis of electronic health record (EHR) data. To our knowledge, EHR data is currently underutilized during the design and training of decision support systems for ARDS diagnosis. As such, the focus in this manuscript is on developing a machine learning framework that: 1) can diagnose ARDS using EHR data only; 2) uses available chest x-rays in training to fine-tune the model; and 3) accounts for diagnostic label uncertainty.

In machine learning terminology, any source of information that is available at the time of training but not at testing (such as a chest x-ray in the current study) is called *privileged information*, as introduced by Vapnik et al. [11]–[13], while the learning task is called Learning Using Privileged Information (LUPI). Another frequently occurring problem in machine learning is that of Label Uncertainty (e.g., the ARDS diagnosis uncertainty considered here), in which class labels are corrupted by some uniform or non-uniform noise distribution [14]. Since these two machine learning problems are integral parts of the proposed method, Sections I-A and I-B are dedicated to short reviews of these problems. Finally, Section I-C outlines the proposed framework.

A. Learning Using Privileged Information

Since its introduction, LUPI has undergone many theoretical advancements. Vapnik et al. [11]–[13] modified Support Vector Machines (SVM) to accommodate privileged information, calling this new formulation SVM+. In [15], Lapin proved that the solutions for SVM+ are inclusive of the solutions for weighted SVM, although there is no well-defined method to reformulate an arbitrary SVM+ problem into its equivalent weighted SVM problem. Cai et al. extended SVM+ to multitask learning [16]. LUPI has also been incorporated into many other machine learning paradigms. Chen et al. generated weak classifiers based on privileged information for boosting algorithms [17]. In metric learning, it was shown that the input space metric can be more precisely designed using privileged information [18]. In addition, privileged information can be incorporated into a three-node Bayesian network [19]. By estimating the probabilities behind privileged information, features can be localized in a regression forest model [20]. Chen et al. enhanced each layer of a convolutional neural network with privileged information to better generalize the model [21]. The sparsity problem in learning to hash methods can be ameliorated with privileged information [22]. Privileged information improves unsupervised hierarchical text clustering [23].

SVM+ has also proven useful in several applications. Sharmanska et al. improved computer vision tasks with SVM+ using privileged information such as bounding boxes or text descriptions [24]. Descriptions of facial features used as privileged information in a computer vision model helped predict human age [25]. SVM+ outperformed SVM in bankruptcy prediction [26]. To our knowledge, SVM+, as a standard convex quadratic problem with linear equalities and bounding constraints, is currently the only SVM-based model built to address the LUPI scenario [11]–[13], [27]. Hence, in this paper the SVM+ formulation is the starting point of the proposed model development process.

B. Label Uncertainty

The problem of label uncertainty, and potential solutions to it, have been explored for a number of machine learning models. Van Hulse et al. found that label noise degraded all the classifiers studied in their work; however, connectionist methods like SVM and neural networks fared worse than simpler methods like naive Bayes [28]. Likewise, certain loss functions are more robust to label noise than SVM [29]. One method to account for label noise in SVMs was provided by Claesen et al., who created an ensemble of SVMs using bootstrapping and found it to result in a substantial improvement in some cases [30]. Many approaches have been proposed to better train on datasets with label noise for a variety of machine learning paradigms. Frenay et al. created a taxonomy of label noises, then surveyed methods that account for each case [14], [31]. Label noise in deep learning algorithms was considered in [32], [33], [34] and [35]. In active learning, a committee of models can be created that eliminates suspicious points [36]. The Imprecise Information Gain Ratio is reported to be more robust against label noise in decision trees [37]. Tomasev et al. leveraged hubness-based fuzzy k-Nearest Neighbors classification as a label noise robust alternative to kNN [38]. AdaBoost is made more robust to label noise by using a loss function that restricts the penalties on misclassified examples [39].

As mentioned previously, label uncertainty can be incorporated as weights within a SVM model. These weights determine the relative penalty of misclassification of training set points. There are two types of weights that could be used in this scenario - weights that are learned from the data itself, and weights that are provided by the user. For an example of the former, see [40], in which importance re-weighting is described as the method of determining the weights. For the later we have shown in previous work that incorporating these weights improves the performance of a standard SVM model [10].

C. Outline of the Proposed Approach

One limitation of SVM+ is that it assumes that privileged information is available for all training samples during parameter estimation. This is not the case in this study, as chest x-rays are not necessarily available for all the training samples. As such, this paper begins with the proposal of an SVM-based formulation, called SVMp+, to address the issue of learning using partially available privileged information (LUPAPI). Note that the need to formulate such a learning paradigm was first suggested by Vapnik in [12], but to our knowledge it has not been developed further, though Wang et al. [41] also studied partial availability of privileged information using a non-SVM-based model.

Another improvement considered here is to incorporate privilege information with label uncertainty, since there are many real-world machine learning scenarios in which the simultaneous use of privileged information and label uncertainty have the potential to improve model performance. Motivated by the potential benefit for such an integration into a unified paradigm, label uncertainty is incorporated into the LUPAPI paradigm and attendant SVMp+ formulation, resulting in a general framework of learning using label uncertainty and partially available privileged information (LULUPAPI).

As there are multiple ways in which to incorporate partially available privileged information into SVM, in Section II three models are considered: 1) *Vapnik's model* [12], a natural extension of SVM+; 2) the *mixture model*, an SVMp+ formulation with a mixture of slack variables and a correcting function; and 3) the *symmetric mixture model*, an SVMp+ formulation with a mixture of slack variables and a correcting function with label coefficients. The end of this section describes how label uncertainty can be integrated into the SVMp+ formulation (LULUPAPI).

II. SVM-Based Models for LUPAPI and LULUPAPI Paradigms

This section describes three models within the LUPAPI framework. The first is Vapnik's model [12], while the other two models are different realizations of the SVMp+ formulation: the mixture model and the symmetric mixture model. These formulations provide alternative methodologies for incorporating partially available privileged information into an SVM model, and the relative superiority of one model over another may vary from one machine learning application to another. Finally, the SVMp+ framework is modified to incorporate label uncertainty into the LUPAPI paradigm, resulting in the LULUPAPI framework.

A. Vapnik's Model: An Initial Model for Partial Availability of Privileged Information

Fundamentally, the problem of partial availability of privileged information can be addressed using a combination of the classical SVM and standard SVM+. In other words, one can consider slack variables for the samples without privileged information and the correcting function for the samples with privileged information. This model was proposed by Vapnik et al. [12] within the LUPI framework, but not explored further.

Suppose the training data has m samples with privileged information and $n-m$ samples without privileged information:

$$\left(\mathbf{x}_1,\mathbf{x}_1^*,y_1\right),\ldots,\left(\mathbf{x}_m,\mathbf{x}_m^*,y_m\right),\left(\mathbf{x}_{m+1},y_{m+1}\right),\ldots,\left(\mathbf{x}_n,y_n\right)$$
$$\mathbf{x}_i\in X,\ \mathbf{x}_i^*\in X^*,\ y_i\in\{-1,1\}$$

The decision rule, the slack variables, and the correcting function hyperplane parameters are achieved simultaneously by the following optimization:

$$\begin{aligned}
&\min_{\mathbf{w},b,\mathbf{w}^*,b^*,\xi}\frac{1}{2}\|\mathbf{w}\|_2^2+\frac{\gamma}{2}\|\mathbf{w}^*\|_2^2+C\sum_{i=m+1}^{n}\xi_i \qquad (1)\\
&\qquad +C^*\sum_{i=1}^{m}\left(\mathbf{w}^*\cdot\mathbf{z}_i^*+b^*\right)\\
&\text{s.t. }\forall 1\le i\le m \quad y_i\left(\mathbf{w}\cdot\mathbf{z}_i+b\right)\ge 1-\left(\mathbf{w}^*\cdot\mathbf{z}_i^*+b^*\right)\\
&\quad\ \forall 1\le i\le m \quad \mathbf{w}^*\cdot\mathbf{z}_i^*+b^*\ge 0\\
&\forall m+1\le i\le n \quad y_i\left(\mathbf{w}\cdot\mathbf{z}_i+b\right)\ge 1-\xi_i\\
&\forall m+1\le i\le n \quad \xi_i\ge 0
\end{aligned}$$

where $C>0$, $C^*>0$, and $\gamma>0$ are the hyperparameters. This cost function is the most natural extension of the LUPI model. The dual optimization problem of (1) can be written as:

$$\begin{aligned}
&\max_{\boldsymbol{\alpha},\boldsymbol{\beta}} D\left(\boldsymbol{\alpha},\boldsymbol{\beta}\right)=\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_jy_iy_jK_{i,j} \qquad (2)\\
&-\frac{1}{2\gamma}\sum_{i,j=1}^{m}\left(\alpha_i+\beta_i-C^*\right)\left(\alpha_j+\beta_j-C^*\right)K_{i,j}^*\\
&\text{s.t.}\qquad \sum_{i=1}^{n}y_i\alpha_i=0 \qquad (3)\\
&\qquad \sum_{i=1}^{m}\left(\alpha_i+\beta_i-C^*\right)=0 \qquad (4)\\
&\qquad \forall m+1\le i\le n,\quad 0\le\alpha_i\le C \qquad (5)\\
&\qquad \forall 1\le i\le m,\quad 0\le\alpha_i,\ 0\le\beta_i \qquad (6)
\end{aligned}$$

where $K_{i,j}^*\triangleq K^*\left(\mathbf{z}_i^*,\mathbf{z}_j^*\right)$ is a kernel in the correcting space and $K_{i,j}\triangleq K\left(\mathbf{z}_i,\mathbf{z}_j\right)$ is the kernel in the decision space with the decision function

$$f\left(\mathbf{z}\right)=\mathbf{w}\cdot\mathbf{z}+b=\sum_{i=1}^{n}y_i\alpha_iK\left(\mathbf{z}_i,\mathbf{z}\right)+b. \qquad (7)$$

Since the aforementioned Vapnik's model was never explored further, in this paper an optimization procedure was also developed and tested for this formulation.

B. The Proposed LUPAPI Framework: SVMp+ Formulations

In this section, two realizations of the SVMp+ formulation of LUPAPI are provided: the mixture model and the symmetric mixture model.

1) Mixture Model: This formulation of SVMp+ can be thought of as SVM+ with the mixture model of slacks as:

$$\xi_i^{'}=\left(\mathbf{w}^*\cdot\mathbf{z}_i^*+b^*\right)+\rho\xi_i^* \qquad \forall 1\le i\le n \qquad (8)$$

In this case, the slack variables are considered for all training samples, and the correcting function only for those samples with privileged information. The decision rule, the slack variables, and the correcting function hyperplane parameters are achieved simultaneously by the following optimization:

$$\begin{aligned}
&\min_{\mathbf{w},b,\xi,\mathbf{w}^*,b^*,\xi^*}\frac{1}{2}\|\mathbf{w}\|_2^2+\frac{\gamma}{2}\|\mathbf{w}^*\|_2^2+C\sum_{i=m+1}^{n}\xi_i \qquad (9)\\
&\qquad +\rho C^*\sum_{i=1}^{m}\xi_i^*+C^*\sum_{i=1}^{m}\left(\mathbf{w}^*\cdot\mathbf{z}_i^*+b^*\right)\\
&\text{s.t. }\forall 1\le i\le m \quad y_i\left(\mathbf{w}\cdot\mathbf{z}_i+b\right)\ge 1-\left(\mathbf{w}^*\cdot\mathbf{z}_i^*+b^*\right)-\xi_i^*\\
&\quad\ \forall 1\le i\le m \quad \mathbf{w}^*\cdot\mathbf{z}_i^*+b^*\ge 0\\
&\quad\ \forall 1\le i\le m \quad \xi_i^*\ge 0\\
&\forall m+1\le i\le n \quad y_i\left(\mathbf{w}\cdot\mathbf{z}_i+b\right)\ge 1-\xi_i\\
&\forall m+1\le i\le n \quad \xi_i\ge 0
\end{aligned}$$

The dual optimization problem of (9) can be formulated as:

$$\max_{\boldsymbol{\alpha},\boldsymbol{\beta}} D(\boldsymbol{\alpha},\boldsymbol{\beta}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} \alpha_i\alpha_j y_i y_j K_{i,j}$$

$$-\frac{1}{2\gamma}\sum_{i,j=1}^{m}(\alpha_i+\beta_i-C^*)(\alpha_j+\beta_j-C^*)K^*_{i,j} \quad (10)$$

$$\text{s.t.} \quad \sum_{i=1}^{n} y_i\alpha_i = 0 \quad (11)$$

$$\sum_{i=1}^{m}(\alpha_i+\beta_i-C^*) = 0 \quad (12)$$

$$\forall\, m+1 \le i \le n, \quad 0 \le \alpha_i \le C \quad (13)$$

$$\forall\, 1 \le i \le m, \quad 0 \le \alpha_i \le \rho C^*, \quad 0 \le \beta_i \quad (14)$$

2) Symmetric Mixture Model: In this model, the goal is to better transfer the knowledge obtained in the privileged information space to the decision space by allowing the privileged information and the training label to interact, as suggested in [13]. Instead of the mixture model of slacks in (8), consider the following mixture for the LUPAPI model:

$$\xi_i^{'} = y_i(\mathbf{w}^*\cdot\mathbf{z}_i^*+b^*)+\rho\xi_i^* \qquad \forall\, 1\le i\le m. \quad (15)$$

The problem can then be written as:

$$\min_{\mathbf{w},b,\xi,\mathbf{w}^*,b^*,\xi^*} \frac{1}{2}\|\mathbf{w}\|_2^2+\frac{\gamma}{2}\|\mathbf{w}^*\|_2^2 + C\sum_{i=m+1}^{n}\xi_i \quad (16)$$

$$+\rho C^*\sum_{i=1}^{m}\xi_i^* + C^*\sum_{i=1}^{m} y_i(\mathbf{w}^*\cdot\mathbf{z}_i^*+b^*)$$

$$\begin{aligned} \text{s.t. } \forall\, 1\le i\le m \quad & y_i(\mathbf{w}\cdot\mathbf{z}_i+b)\ge 1-y_i(\mathbf{w}^*\cdot\mathbf{z}_i^*+b^*)-\xi_i^* \\ \forall\, 1\le i\le m \quad & y_i(\mathbf{w}^*\cdot\mathbf{z}_i^*+b^*)\ge 0 \\ \forall\, 1\le i\le m \quad & \xi_i^*\ge 0 \\ \forall\, m+1\le i\le n \quad & y_i(\mathbf{w}\cdot\mathbf{z}_i+b)\ge 1-\xi_i \\ \forall\, m+1\le i\le n \quad & \xi_i\ge 0 \end{aligned}$$

The corresponding dual problem can be formulated as:

$$\max_{\boldsymbol{\alpha},\boldsymbol{\beta}} D(\boldsymbol{\alpha},\boldsymbol{\beta}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} \alpha_i\alpha_j y_i y_j K_{i,j}$$

$$-\frac{1}{2\gamma}\sum_{i,j=1}^{m}(\alpha_i+\beta_i-C^*)(\alpha_j+\beta_j-C^*)y_i y_j K^*_{i,j} \quad (17)$$

$$\text{s.t.} \quad \sum_{i=1}^{n} y_i\alpha_i = 0 \quad (18)$$

$$\sum_{i=1}^{m} y_i(\alpha_i+\beta_i-C^*) = 0 \quad (19)$$

$$\forall\, m+1 \le i \le n, \quad 0 \le \alpha_i \le C \quad (20)$$

$$\forall\, 1 \le i \le m, \quad 0 \le \alpha_i \le \rho C^*, \quad 0 \le \beta_i \quad (21)$$

This model is referred to as *symmetric* because the y_i coefficients are considered for the hyperplane in the privileged space as well. This model differs from that proposed in [13] in two ways - it incorporates partially available privileged information and allows for two separate sets of slack variables for the training and privileged spaces.

C. LULUPAPI: Incorporating Label Uncertainty within the SVMp+ Formulations

In this section, label uncertainty is integrated into the SVMp+ formulation of LUPAPI, yielding the Learning Using Label Uncertainty and Partially Available Privileged Information (LULUPAPI) model. To avoid repetition, only the mixture model of LUPAPI (described in Section II-B.1) is considered. In order to incorporate label uncertainty, one can vary the parameter C for training samples in proportion to their respective label confidence.

As the slack variables ξ_i (or the correcting function) permit some misclassification with penalty parameter C to establish soft-margin decision boundaries, data with high label confidence can be given more weight and subsequent influence on the decision boundary. This yields the LULUPAPI paradigm, which requires the training samples

$$(\mathbf{x}_1,\mathbf{x}_1^*,y_1,\pi_1),\ldots,(\mathbf{x}_m,\mathbf{x}_m^*,y_m,\pi_m),(\mathbf{x}_{m+1},y_{m+1},\pi_{m+1}),$$
$$(\mathbf{x}_{m+2},y_{m+2},\pi_{m+2}),\ldots,(\mathbf{x}_n,y_n,\pi_n)$$
$$\mathbf{x}_i\in X,\ \mathbf{x}_i^*\in X^*,\ y_i\in\{-1,1\},\pi_i\ge 0$$

where π_i is a quantitative measure of uncertainty in the labels. In this case, the LULUPAPI mixture model is

$$\min_{\mathbf{w},b,\xi,\mathbf{w}^*,b^*,\xi^*} \frac{1}{2}\|\mathbf{w}\|_2^2+\frac{\gamma}{2}\|\mathbf{w}^*\|_2^2 + C\sum_{i=m+1}^{n}\pi_i\xi_i$$

$$+\rho C^*\sum_{i=1}^{m}\pi_i\xi_i^* + C^*\sum_{i=1}^{m}(\mathbf{w}^*\cdot\mathbf{z}_i^*+b^*) \quad (22)$$

$$\begin{aligned} \text{s.t. } \forall\, 1\le i\le m \quad & y_i(\mathbf{w}\cdot\mathbf{z}_i+b)\ge 1-(\mathbf{w}^*\cdot\mathbf{z}_i^*+b^*)-\xi_i^* \\ \forall\, 1\le i\le m \quad & \mathbf{w}^*\cdot\mathbf{z}_i^*+b^*\ge 0 \\ \forall\, 1\le i\le m \quad & \xi_i^*\ge 0 \\ \forall\, m+1\le i\le n \quad & y_i(\mathbf{w}\cdot\mathbf{z}_i+b)\ge 1-\xi_i \\ \forall\, m+1\le i\le n \quad & \xi_i\ge 0 \end{aligned}$$

and the dual optimization problem is

$$\max_{\boldsymbol{\alpha},\boldsymbol{\beta}} D(\boldsymbol{\alpha},\boldsymbol{\beta}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} \alpha_i\alpha_j y_i y_j K_{i,j}$$

$$-\frac{1}{2\gamma}\sum_{i,j=1}^{m}(\alpha_i+\beta_i-C^*)(\alpha_j+\beta_j-C^*)K^*_{i,j} \quad (23)$$

$$\text{s.t.} \quad \sum_{i=1}^{n} y_i\alpha_i = 0$$

$$\sum_{i=1}^{m}(\alpha_i+\beta_i-C^*) = 0$$

$$\forall\, m+1 \le i \le n, \quad 0 \le \alpha_i \le \pi_i C$$

$$\forall\, 1 \le i \le m, \quad 0 \le \alpha_i \le \rho\pi_i C^*, \quad 0 \le \beta_i$$

III. An Alternating SMO Algorithm for SVMp+ Formulations of LUPAPI and LULUPAPI Paradigms

A widely used algorithm for solving conventional SVM and SVM+ is Sequential Minimal Optimization (SMO) [42]. SMO-style algorithms iteratively maximize the dual cost function by selecting the best *maximally sparse feasible direction* in each iteration and updating the corresponding α_i and β_j such that the dual constraints are also satisfied.

A variant of SMO called Alternating SMO for solving SVM+ was introduced in [43], [44]. Inspired by this optimization method, an alternating SMO-style algorithm for SVMp+ is proposed. The SVMp+ dual optimization problems of (2), (10), (17), and (23) can be considered as the general form of

$$\max_{\boldsymbol{\theta}\in\mathcal{F}} D\left(\boldsymbol{\theta}\right),$$

where $\boldsymbol{\theta}\in\mathbb{R}^k$, $D:\mathbb{R}^k\to\mathbb{R}$ is a concave quadratic function, and $\mathcal{F}$ is a convex compact set defined by linear equalities and inequalities.

In order to achieve an alternating SMO algorithm for SVMp+, all feasible directions for each model must be determined. Feasible and maximally sparse feasible directions were defined in [43] as follows:

Definition 1. A direction $\mathbf{u}\in\mathbb{R}^k$ is *feasible* at the point $\boldsymbol{\theta}\in\mathcal{F}$ if there exists $\lambda>0$ such that $\boldsymbol{\theta}+\lambda\mathbf{u}\in\mathcal{F}$.

Definition 2. A direction $\mathbf{u}_1\in\mathbb{R}^k$ with $n_1<k$ zero elements is *maximally sparse feasible* if any $\mathbf{u}_2\in\mathbb{R}^k$ with $n_2<k$ zero elements such that $n_1<n_2$ is not feasible.

The cost function in equations (2), (10), (17), and (23) have $n+m$ variables: $\{\alpha_i\}_{i=1}^n$ and $\{\beta_i\}_{i=1}^m$. These can be combined into a single $(n+m)$-variable vector $\boldsymbol{\theta}$ by concatenating the $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ variables: $\boldsymbol{\theta}\triangleq(\boldsymbol{\alpha},\boldsymbol{\beta})^T$. Thus, each maximally sparse feasible direction is $\mathbf{u}\in\mathbb{R}^{n+m}$. It can be verified that the cost functions in equations (2), (10), and (23) have 9 sets of such directions, and (17) has 10. Following [43], each set of feasible directions is denoted by I_i. The detailed descriptions of the feasible directions and other optimization information for Vapnik's model (2), the mixture model (10), the symmetric mixture model (17), and the LULUPAPI mixture model (23) formulations can be found in Appendices I, II, III, and IV, respectively.

A. Optimization Process

Similar to the SMO algorithm for the LUPI model ([43], [44]), the recursive step in the proposed optimization for the LUPAPI and LULUPAPI paradigms is finding $\boldsymbol{\theta}=\boldsymbol{\theta}^{\text{old}}+\lambda^*(\mathbf{s})\mathbf{u}_\mathbf{s}$ such that $\mathbf{u}_\mathbf{s}\in\cup I_i$ of the corresponding feasible directions and the step size $\lambda^*(\mathbf{s})$ maximize the corresponding cost function $\psi\left(\lambda\right)=D\left(\boldsymbol{\theta}^{\text{old}}+\lambda\mathbf{u}_\mathbf{s}\right)$, while satisfying the constraints. Hence, given the cost function, its constraints, and the corresponding feasible directions, the recursive optimization process of the proposed alternating SMO-style algorithm is the same in both the LUPAPI and LULUPAPI contexts.

Let $\mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)$ and H respectively be the gradient at point $\boldsymbol{\theta}^{\text{old}}$ and the Hessian of the cost function. Using the Taylor expansion of $\psi\left(\lambda\right)$ at point $\lambda=0$ yields

$$\lambda^{'}\left(\boldsymbol{\theta}^{\text{old}},\mathbf{s}\right)=\arg\max_{\lambda\geq0}\psi\left(\lambda\right)=-\left.\frac{\frac{\partial\psi(\lambda)}{\partial\lambda}}{\frac{\partial^2\psi(\lambda)}{\partial\lambda^2}}\right|_{\lambda=0} =-\frac{\mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)^T\mathbf{u}_\mathbf{s}}{\mathbf{u}_\mathbf{s}^\mathbf{T}H\mathbf{u}_\mathbf{s}} \tag{24}$$

Let $\tau>0$ be small constant. Define $I=\{\mathbf{u}_\mathbf{s}|\mathbf{u}_\mathbf{s}\in\bigcup I_i,\ \mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)^T\mathbf{u}_\mathbf{s}>\tau\}$. If $I=\emptyset$, then the algorithm stops. Suppose $I\neq\emptyset$, define:

$$\widetilde{I}_i=\{\mathbf{u}_\mathbf{s}|\mathbf{u}_\mathbf{s}\in I_i,\ \mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)^T\mathbf{u}_\mathbf{s}>\tau\}.$$

For each non-empty $\widetilde{I}_i$, find the vector $\mathbf{u}_{\mathbf{s}^{(\mathrm{i})}}\in\widetilde{I}_i$ that has the minimal angle with $\mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)$ among all the candidates in $\widetilde{I}_i$:

$$\mathbf{s}^{(i)}=\arg\max_{\mathbf{s}:\mathbf{u}_\mathbf{s}\in\widetilde{I}_i}\mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)^T\mathbf{u}_\mathbf{s}. \tag{25}$$

In the next step, for the directions containing pairs, if $\mathbf{s}^{(i)}=\left(s_1^{(i)},s_2^{(i)}\right)\neq\emptyset$, fix the value of $s_1^{(i)}$ and find $\mathbf{s}^{'(i)}=\left(s_1^{(i)},s_2^{'(i)}\right)$ such that $\mathbf{u}_{\mathbf{s}^{'(\mathrm{i})}}\in\widetilde{I}_i$ and

$$\begin{aligned}\mathbf{s}^{'(i)}&=\arg\max_{\substack{\mathbf{t}:\mathbf{t}=\left(s_1^{(i)},t_2\right)\\ \mathbf{u}_\mathbf{t}\in\widetilde{I}_i}}D\left(\boldsymbol{\theta}^{\text{old}}+\lambda^{'}\left(\boldsymbol{\theta}^{\text{old}},\mathbf{t}\right)\mathbf{u}_\mathbf{t}\right)-D\left(\boldsymbol{\theta}^{\text{old}}\right)\\ &=\arg\max_{\substack{\mathbf{t}:\mathbf{t}=\left(s_1^{(i)},t_2\right)\\ \mathbf{u}_\mathbf{t}\in\widetilde{I}_i}}\frac{-\left(\mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)^T\mathbf{u}_\mathbf{t}\right)^2}{\mathbf{u}_\mathbf{t}^\mathbf{T}H\mathbf{u}_\mathbf{t}}\end{aligned} \tag{26}$$

where the last equality is achieved by substituting $\lambda^{'}\left(\boldsymbol{\theta}^{\text{old}},\mathbf{s}\right)$ of equation (24) into $D\left(\boldsymbol{\theta}^{\text{old}}+\lambda^{'}\left(\boldsymbol{\theta}^{\text{old}},\mathbf{t}\right)\mathbf{u}_\mathbf{t}\right)-D\left(\boldsymbol{\theta}^{\text{old}}\right)$. Similarly, for the directions containing triplets, if $\mathbf{s}^{(i)}=\left(s_1^{(i)},s_2^{(i)},s_3^{(i)}\right)\neq\emptyset$, fix the value of $s_1^{(i)}$ and $s_3^{(i)}$, and find $\mathbf{s}^{'(i)}=\left(s_1^{(i)},s_2^{'(i)},s_3^{(i)}\right)$ such that $\mathbf{u}_{\mathbf{s}^{'(\mathrm{i})}}\in\widetilde{I}_i$ and

$$\mathbf{s}^{'(i)}=\arg\max_{\substack{\mathbf{t}:\mathbf{t}=\left(s_1^{(i)},t_2,s_3^{(i)}\right)\\ \mathbf{u}_\mathbf{t}\in\widetilde{I}_i}}\frac{-\left(\mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)^T\mathbf{u}_\mathbf{t}\right)^2}{\mathbf{u}_\mathbf{t}^\mathbf{T}H\mathbf{u}_\mathbf{t}} \tag{27}$$

Among all the possible directions from $\mathbf{u}_{\mathbf{s}^{'(\mathrm{i})}}\in\cup\widetilde{I}_i$, the optimal direction that maximizes the cost function is chosen:

$$\mathbf{s}^{*(i)}=\arg\max_{\mathbf{s}^{'(i)}\neq\emptyset}\frac{-\left(\mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)^T\mathbf{u}_\mathbf{t}\right)^2}{\mathbf{u}_\mathbf{t}^\mathbf{T}H\mathbf{u}_\mathbf{t}} \tag{28}$$

Having chosen the optimal direction $\mathbf{s}^{*(i)}$, the value of $\lambda^{'}\left(\boldsymbol{\theta}^{\text{old}},\mathbf{s}^{*(i)}\right)$ should be clipped such that it satisfies the upper/lower bound constraints on $\{\alpha_i\}_{i=1}^n$ and $\{\beta_i\}_{i=1}^m$. Clipping functions are specific to the dual problems of each SVMp+ formulation and can be found in the Appendices.

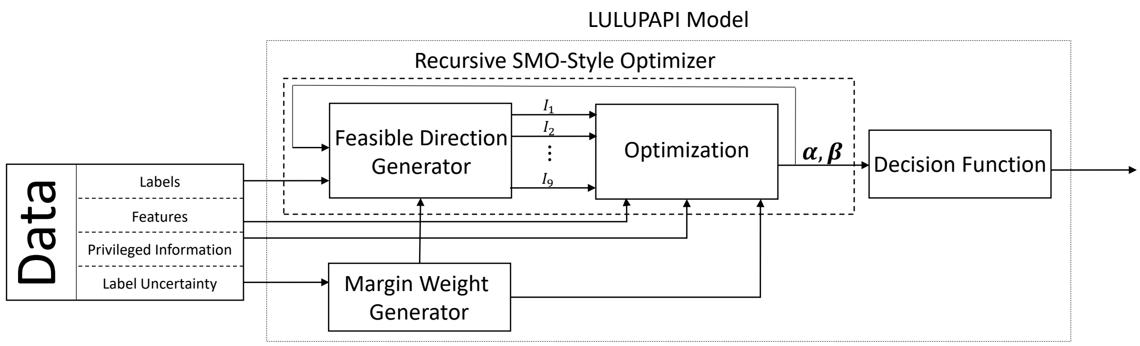


Fig. 1. Alternating SMO-style optimizer for the LULUPAPI model

B. Algorithm

Having described the framework for the proposed alternating SMO-style algorithm that solves the SVMp+ dual cost functions of the LUPAPI and LULUPAPI models, the resultant algorithm is codified in Algorithm 1. Note that most of the calculations for I_i can be performed once, rather than in each iteration, as they depend largely on the label and indices of the training data samples. One can consider various initial conditions for the $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ variables. Since the feasible directions and clipping function ensure the satisfaction of the dual problem conditions, a satisfactory initial condition guarantees the fulfillment of these conditions in each iteration. In all variants of the LUPAPI and LULUPAPI models, the simplest initial condition that satisfies all of the constraints is $\alpha_i^{(0)} = 0$ and $\beta_i^{(0)} = C^*$.

As mentioned earlier, while the proposed optimization process for the LUPAPI and LULUPAPI models is the same given the dual cost function and the corresponding feasible directions, the LULUPAPI model is always the most comprehensive model and any given algorithm for LULUPAPI can easily be modified to realize a LUPAPI version. For instance, any algorithm specifically designed for the LULUPAPI mixture model can be made into a LUPAPI mixture model by simply replacing the uncertainty coefficients with unity. A general schematic diagram of the LULUPAPI iterative optimizer is depicted in Figure 1.

Algorithm 1 Alternating SMO-style Optimization for SVMp+ formulations of LUPAPI and LULUPAPI

Require: Training data, training labels, $\tau > 0$, $\gamma > 0$, $C > 0$, $C^* > 0$ and $0 < \epsilon \ll 1$.
Calculate: Kernels K and K^*, Hessian H.
Initialize: $\theta_i^{(0)}$(i.e., $\alpha_i^{(0)}$ and $\beta_i^{(0)}$)
Initialize: I_i for each feasible direction based on the indexes and training labels.
while exists a maximally sparse feasible direction $\mathbf{u_s}$ s.t. $\mathbf{g}\left(\boldsymbol{\theta}^{\text{new}}\right)^T \mathbf{u_s} > \tau$ and $\left(D\left(\boldsymbol{\theta}^{\text{new}}\right) - D\left(\boldsymbol{\theta}^{\text{old}}\right)\right) > \epsilon$ **do**
  $\boldsymbol{\theta}^{\text{old}} = \boldsymbol{\theta}^{\text{new}}$
  Calculate $\mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)$
  Update I_i for all i based on $\boldsymbol{\theta}^{\text{old}}$
  Calculate $\widetilde{I}_i$ if $I_i \neq \emptyset$
  Calculate $\mathbf{s}^{(i)}$ if $\widetilde{I}_i \neq \emptyset$ using (25)
  Calculate $\mathbf{s}^{'(i)}$ if $\widetilde{I}_i \neq \emptyset$ using (26) or (27)
  Calculate $\mathbf{s}^{*(i)}$ if $\cup \widetilde{I}_i \neq \emptyset$ using (28)
  Calculate λ^* using the corresponding clipping function
  Update $\boldsymbol{\theta}^{\text{new}} = \boldsymbol{\theta}^{\text{old}} + \lambda^* u_{\mathbf{s}^*}$
end while

IV. Experiments and Results

A number of experiments were performed to test the performance of the three SVMp+ models against standard SVM, in both the LUPAPI and LULUPAPI contexts. Recall that as SVM+ requires the availability of privileged information for all training samples, SVM+ cannot be utilized. The performance of the LUPAPI and LULUPAPI models were tested against SVM using a dataset of patients with and without ARDS. The dataset contains privileged information and label uncertainty related to the ARDS diagnosis, which allows for a direct comparison of the performance of the LUPAPI and LULUPAPI models.

A. ARDS

The ARDS dataset used in this study consisted of 485 patients with either moderate hypoxia or acute hypoxic respiratory failure, treated at the University of Michigan Hospital. We received institutional review board from the University of Michigan to collect data for the study (HUM00104714) with a waiver of informed consent among study participants. Each case was independently reviewed by multiple expert clinicians for the diagnosis of ARDS, with their confidence in that diagnosis recorded (note that for label uncertainty, the average confidence was used). Multiple experts reviewed the cases as there can be disagreement between doctors reviewing the same patients for the diagnosis of ARDS [5]. Clinical experts also identified the time of ARDS onset for those patients who were deemed to have developed the condition. Patients who developed ARDS were labeled as negative before the time of onset and positive for ARDS after. The non-privileged information consisted of 25 clinical variables (features) extracted at two-hour intervals from the patient's Electronic Health Record (EHR). The clinical features were temperature, heart rate, respiratory rate, systolic blood pressure, diastolic blood pressure, positive end-expiratory pressure (PEEP), plateau pressure, mean airway pressure, white blood cell count level, lactate

TABLE I
LUPAPI RESULT FOR ARDS CLASSIFICATION

	Train				Test			
	Accuracy	Sensitivity	Specificity	AUC	Accuracy	Sensitivity	Specificity	AUC
SVM	88.61	80.43	91.76	86.10	89.27	76.58	90.33	83.46
Vapnik's Model	86.01	78.51	88.91	83.71	88.42	75.62	89.50	82.56
Mixture Model	88.09	81.28	90.72	86.00	88.78	**82.23**	89.34	**85.78**
Symmetric Mixture Model	88.56	81.59	91.26	86.42	88.94	**81.13**	89.60	**85.37**

TABLE II
ARDS CLASSIFICATION RESULTS USING SVM, SVM WITH LABEL UNCERTAINTY, LUPAPI MIXTURE MODEL, AND LULUPAPI MIXTURE MODEL.

	Train				Test				
	Accuracy	Sensitivity	Specificity	AUC	F1 Score	Accuracy	Sensitivity	Specificity	AUC
SVM	88.61	80.43	91.76	86.10	79.73	89.27	76.58	90.33	83.46
SVM with LU	87.56	78.74	90.96	84.85	79.60	89.52	80.03	90.32	85.17
LUPAPI	88.09	81.28	90.72	86.00	79.17	88.78	**82.23**	89.34	**85.78**
LULUPAPI	87.96	83.59	89.65	86.62	79.46	88.38	**85.40**	88.63	**87.01**

TABLE III
MCNEMAR $\mathcal{X}^2$ TEST ASSESSMENT OF STATISTICAL SIGNIFICANCE OF PERFORMANCE IMPROVEMENTS EXCLUSIVELY AMONG ARDS PATIENTS. IN THIS TABLE, EACH ROW REPRESENTS THE *null* CLASSIFIER, AND EACH COLUMN REPRESENTS THE *alternative* CLASSIFIER. FOR EXAMPLE, LULUPAPI VERSUS SVM HAS THE MCNEMAR TEST STATISTIC $\mathcal{X}^2 = 56.70$, WHICH IS EXTREMELY IN FAVOR OF LULUPAPI (p-VALUE $<< 0.001$), WHILE LUPAPI VERSUS SVM WITH LABEL UNCERTAINTY RESULTS IN THE MCNEMAR TEST STATISTIC $\mathcal{X}^2 = 10.22$ (p-VALUE $= 0.0014$). IF THE NULL CLASSIFIER OUTPERFORMS THE ALTERNATIVE CLASSIFIER, THE VALUE IS REPRESENTED WITH AN "X".

	SVM	SVM with LU	LUPAPI	LULUPAPI
SVM	0	12.80	39.02	56.70
SVM with LU	X	0	10.22	33.58
LUPAPI	X	X	0	15.61
LULUPAPI	X	X	X	0

acid level, bicarbonate level, carbon dioxide level, pH level, brain natriuretic peptide level (BNP), troponin level, albumin level, and pulse oximetry value. Privileged information for each patient consisted of the average of scores among multiple clinical experts reviewing chest radiographs performed during the hospitalization. Each clinician gave each chest x-ray a rating of 1-8, scoring their belief that the x-ray was consistent with ARDS (8 for high-confidence ARDS and 1 for high-confidence non-ARDS). As such, privileged information is the average of these scores if the chest x-ray is available.

In the experiment for classifying ARDS, the dataset was first split into training and testing sets as shown in Figure 2. In order to avoid bias toward patients, all samples from the same patient were kept exclusively in either training or testing. This yielded 323 patients in the training dataset, and the rest in the testing set. Also, due to the strong inter-dependency between samples of longitudinal patient data, the IID (independent and identically distributed) assumption was invalid. Therefore, the time-series sampling method proposed in [10] was performed to reduce inter-correlation among the longitudinal clinical data from each patient used in model training. After sampling, there were 4661 samples in the training dataset, with 1298 positive for ARDS. Since there was no sampling in the testing dataset, there were 9362 samples in the test dataset.

Within the training set, 5-fold cross-validation was performed. Five folds were formed, split by patients. At each round, four folds were used for training and one for testing. 4-dimensional hyperparameter optimization with a linear kernel was performed separately for SVM and each of the LUPAPI models, with the hyperparameters ranging over the following intervals: $[0.1, 5]$ for C and C^*, $[1, 5]$ for ρ, and $[0.5, 2]$ for γ. The results are depicted in Table I. Based on these results, the mixture model and the symmetric mixture model outperformed SVM while Vapnik's model underperformed.

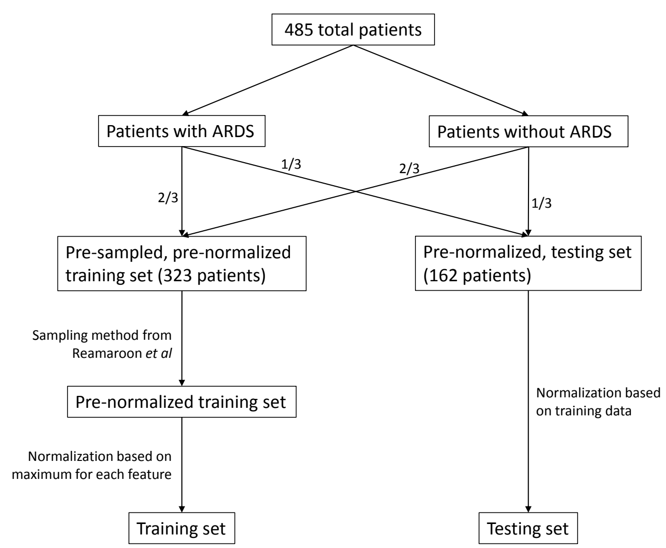


Fig. 2. Flowchart of this study's protocol with 5-fold cross-validation as suggested in [10]. Hyperparameter optimization was performed by grid search.

In order to analyze the effect of incorporating label uncertainty into the LUPAPI model, the LULUPAPI model of Section II-C was used to classify ARDS. Clinicians reported the level of confidence in their diagnosis, denoted by l_i, using a 1-8 scale in which 1 is not ARDS with high confidence and 8 is ARDS with high confidence. To quantitatively measure uncertainty in the labels, i.e., π_is in equation (22), a margin weight generator $\pi_i = (|l_i - p_1| - p_2)p_3 + p_4$ was used with $p_1 = 4.5$, $p_2 = 3$, $p_3 = 0.2$, and $p_1 = 0.9$. This scaled l_is from the range 1-8 into the range 0.4-1 such that high-

TABLE IV
COMPARISON OF ARDS CLASSIFICATION RESULTS USING LUPAPI MIXTURE MODEL, LULUPAPI MIXTURE MODEL AND DEEP LEARNING METHODS.

	Train				Test			
	Accuracy	Sensitivity	Specificity	AUC	Accuracy	Sensitivity	Specificity	AUC
LUPAPI	88.09	81.28	90.72	86.00	88.78	82.23	89.34	85.78
LULUPAPI	87.96	83.59	89.65	86.62	88.38	**85.40**	88.63	**87.01**
Shallow NN (2 layers, 10 nodes)	87.52	86.65	87.77	87.31	79.98	84.99	79.56	82.23
Shallow NN (2 layers, 50 nodes)	87.89	84.81	89.08	86.94	83.41	82.97	89.05	85.39
Shallow NN (2 layers, 100 nodes)	87.41	82.43	89.33	85.88	87.97	66.67	89.76	78.22
LSTM (25 layers, 10 nodes)	90.94	83.10	93.97	88.53	88.39	64.33	90.41	77.34
LSTM (25 layers, 50 nodes)	91.99	84.67	94.82	89.74	88.91	70.39	90.47	80.43
LSTM (25 layers, 100 nodes)	92.62	85.79	95.25	90.52	87.40	71.76	88.71	80.24

confidence cases l_i=1 and l_i=8 were mapped to $\pi_i = 1$ and low-confidence cases l_i=4 and l_i=5 were mapped to $\pi_i = 0.4$. Table II summarizes the results of SVM, SVM with label uncertainty, the LUPAPI mixture model, and the LULUPAPI mixture model. As can be seen, while separate incorporation of label uncertainty and privileged information improved the SVM result (SVM with label uncertainty and LUPAPI in Table II), the best test AUC was achieved by simultaneous use of label uncertainty and privileged information (LULUPAPI in Table II). As can be seen in these tables, any improvement in AUC is correlated with improvement of sensitivity. This is mainly due to the imbalanced nature of data (fewer ARDS cases), which also results in a slight drop in accuracy.

The McNemar test [45] was employed to assess the statistical significance of improvements in performance of the proposed models over SVM. Since this test is insensitive to the proportion of positive versus negative cases [46], the test was applied exclusively to positive cases. Table III summarizes the results of the McNemar tests and verifies the statistical significance of incorporating both label uncertainty and partial available privileged information in detection of patients with ARDS.

Beyond comparisons with SVM-based methods, the LUPAPI and LULUPAPI models were also benchmarked against multiple popular deep learning methods. A "shallow" neural network (two-layer feedforward network) with one hidden layer of either 10, 50, or 100 nodes was trained to create a less complex neural network more suitable for this type of data. In addition, a long short-term memory (LSTM) network, a specialized type of artificial recurrent neural network for time-series sequential data [47], was also trained to provide a performance comparison to a state-of-the-art deep learning algorithm. The LSTM network was composed of 25 layers with either 10, 50, or 100 hidden units. Both the shallow neural network (Shallow NN) and LSTM models were implemented with the Keras deep learning library using the Adam optimizer algorithm [48] with 500 epochs (mini-batch size of 32) and cross entropy as the loss function. Table IV summarizes the results of this experiment. As can be seen, LUPAPI and LULUPAPI outperformed the deep learning methods.

V. DISCUSSION

For the ARDS dataset in the previous section, a LUPAPI or LULUPAPI formulation was the best performing model. In the LUPAPI experiments on the ARDS dataset as depicted in Table I, the mixture model was the best performing model, achieving an AUC of 85.78, a 2.8% improvement over SVM. Using the same mixture model, but incorporating label uncertainty, the LULUPAPI formulation in Table II achieved an AUC of 87.01, a 4.3% improvement over SVM. The statistical tests in III verify the statistical significance of improvements in performance. Additionally, Table IV shows that LULUPAPI formulation achieved 2.39% improvement over the most competitive deep learning method.

Though the LUPAPI and LULUPAPI frameworks improved performance overall, Vapnik's model underperformed the mixture models and SVM on the ARDS dataset. Based on the experiments, the primary reason for such performance is due to the offset parameter b of the decision function (the detailed calculation of which can be found in the Appendices). For the mixture models and SVM, the set N (defined in the Appendices) includes fewer α_is, corresponding to fewer support vectors that would be used to calculate the offset parameter. This is in turn due to the consideration of slack variables for all samples, regardless of privileged information availability. However, for Vapnik's model the set N consists of more α_is that negatively effects the offset parameter precision.

With respect to time complexity, the proposed alternating SMO-style algorithm for the LUPAPI model is $O(n)$. This is similar to the SMO algorithm for conventional SVM and the alternating SMO algorithm for the LUPI model (the SVM+ formulation), and results from the feasible direction vectors having a constant number of nonzero components. However, the experiments showed that given the same parameters and stoppage criterion, Vapnik's model required more iterations for convergence.

The experimental results also support the claim that if the hyperparameter optimization is performed thoroughly, performance of the mixture model (9) is always lower-bounded by SVM. This claim can also be verified using the dual forms, noting the bound on α_i and the inclusion of SVM feasible directions in the feasible directions of (9).

VI. CONCLUSION

In this paper, a unified framework for handling machine learning tasks in which privileged learning is partially available is presented, while simultaneously correcting for label uncertainty. As there are multiple means of incorporating partially available privileged information into SVM, three models were considered: Vapnik's model [12]; and two new SVMp+ formulations, the mixture and symmetric mixture models. An alternating SMO-style optimization algorithm was provided that solves all model formulations.

While Vapnik's model (1) is a natural extension of the SVM+ framework, the experiments showed that only the two SVMp+ models (9) and (16) outperform SVM on the real-world ARDS dataset. Moreover, on the ARDS dataset, which contained both partially available privileged information and label uncertainty, the LULUPAPI model incorporating both outperformed the LUPAPI model that solely considered privileged information. Even though these models were developed with clinical decision support systems in mind, the proposed models can be applied to many other machine learning applications in healthcare and other domains.

APPENDIX I
VAPNIK'S MODEL

Let $\boldsymbol{\theta}$ be a $(n+m)$-variable vector as the concatenation of the $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ variables: $\boldsymbol{\theta} \triangleq (\boldsymbol{\alpha}, \boldsymbol{\beta})^T$.

A. Feasible Directions and the Clipping Function

It can be verified that the cost function in equation (2) has 9 sets of *maximally sparse feasible directions* (defined in Section III) as follows:

Direction 1: $I_1 \triangleq \{\mathbf{u_s} | \mathbf{s} = (s_1, s_2),\ n+1 \le s_1, s_2 \le n+m,\ s_1 \neq s_2;\ u_{s_1} = 1,\ u_{s_2} = -1,\ \theta_{s_2} > 0,\ \forall i \notin \mathbf{s}\, u_i = 0\}$.

Direction 2: $I_2 \triangleq \{\mathbf{u_s} | \mathbf{s} = (s_1, s_2),\ 1 \le s_1, s_2 \le m,\ s_1 \neq s_2,\ y_{s_1} = y_{s_2};\ u_{s_1} = 1,\ u_{s_2} = -1,\ \theta_{s_2} > 0,\ \forall i \notin \mathbf{s}\ u_i = 0\}$.

Direction 3: $I_3 \triangleq \{\mathbf{u_s} | \mathbf{s} = (s_1, s_2),\ m+1 \le s_1, s_2 \le n,\ s_1 \neq s_2,\ y_{s_1} = y_{s_2};\ u_{s_1} = 1,\ \theta_{s_1} < C,\ u_{s_2} = -1,\ \theta_{s_2} > 0,\ \forall i \notin \mathbf{s}\, u_i = 0\}$.

Direction 4: $I_4 \triangleq \{\mathbf{u_s} | \mathbf{s} = (s_1, s_2),\ m+1 \le s_1, s_2 \le n,\ s_1 \neq s_2,\ y_{s_1} \neq y_{s_2},\ \forall i \notin \mathbf{s}\, u_i = 0;\ u_{s_1} = u_{s_2} = 1,\ \theta_{s_1} < C,\ \theta_{s_2} < C \quad \text{or} \quad u_{s_1} = u_{s_2} = -1,\ \theta_{s_1} > 0,\ \theta_{s_2} > 0\}$.

Direction 5: $I_5 \triangleq \{\mathbf{u_s} | \mathbf{s} = (s_1, s_2, s_3),\ 1 \le s_1, s_2 \le m,\ n+1 \le s_3 \le n+m,\ s_1 \neq s_2,\ y_{s_1} \neq y_{s_2},\ \forall i \notin \mathbf{s}\, u_i = 0;\ u_{s_1} = u_{s_2} = 1,\ u_{s_3} = -2,\ \theta_{s_3} > 0 \quad \text{or} \quad u_{s_1} = u_{s_2} = -1,\ \theta_{s_1} > 0,\ \theta_{s_2} > 0,\ u_{s_3} = 2\}$.

Direction 6: $I_6 \triangleq \{\mathbf{u_s} | \mathbf{s} = (s_1, s_2, s_3),\ 1 \le s_1 \le m,\ m+1 \le s_2 \le n,\ n+1 \le s_3 \le n+m,\ y_{s_1} = y_{s_2},\ \forall i \notin \mathbf{s}\, u_i = 0;\ u_{s_1} = 1,\ u_{s_2} = -1,\ \theta_{s_2} > 0,\ u_{s_3} = -1,\ \theta_{s_3} > 0 \quad \text{or} \quad u_{s_1} = -1,\ \theta_{s_1} > 0,\ u_{s_2} = 1,\ \theta_{s_2} < C,\ u_{s_3} = 1\}$.

Direction 7: $I_7 \triangleq \{\mathbf{u_s} | \mathbf{s} = (s_1, s_2, s_3),\ 1 \le s_1 \le m,\ m+1 \le s_2 \le n,\ n+1 \le s_3 \le n+m,\ y_{s_1} \neq y_{s_2},\ \forall i \notin \mathbf{s}\, u_i = 0;\ u_{s_1} = u_{s_2} = 1,\ \theta_{s_2} < C,\ u_{s_3} = -1,\ \theta_{s_3} > 0 \quad \text{or} \quad u_{s_1} = u_{s_2} = -1,\ \theta_{s_1} > 0,\ \theta_{s_2} > 0,\ u_{s_3} = 1\}$.

Direction 8: $I_8 \triangleq \{\mathbf{u_s} | \mathbf{s} = (s_1, s_2, s_3),\ 1 \le s_1, s_2 \le m,\ m+1 \le s_3 \le n,\ s_1 \neq s_2,\ y_{s_1} \neq y_{s_2},\ y_{s_3} = y_{s_2},\ \forall i \notin \mathbf{s}\, u_i = 0;\ u_{s_1} = 1,\ u_{s_2} = -1,\ \theta_{s_2} > 0,\ u_{s_3} = 2,\ \theta_{s_3} < C \quad \text{or} \quad u_{s_1} = -1,\ \theta_{s_1} > 0,\ u_{s_2} = 1,\ u_{s_3} = -2,\ \theta_{s_3} > 0\}$.

Direction 9: $I_9 \triangleq \{\mathbf{u_s} | \mathbf{s} = (s_1, s_2, s_3),\ 1 \le s_1, s_2 \le m,\ m+1 \le s_3 \le n,\ s_1 \neq s_2,\ y_{s_1} \neq y_{s_2},\ y_{s_3} = y_{s_1},\ \forall i \notin \mathbf{s}\, u_i = 0;\ u_{s_1} = 1,\ u_{s_2} = -1,\ \theta_{s_2} > 0,\ u_{s_3} = -2,\ \theta_{s_3} > 0 \quad \text{or} \quad u_{s_1} = -1,\ \theta_{s_1} > 0,\ u_{s_2} = 1,\ u_{s_3} = 2,\ \theta_{s_3} < C\}$.

Generally, a move from an old feasible point $\boldsymbol{\theta}^{\text{old}}$ to a new feasible point $\boldsymbol{\theta}^{\text{old}} = \boldsymbol{\theta}^{\text{old}} + \lambda \mathbf{u_s}$ in the direction of $\mathbf{u_s} \in \cup I_i$ will satisfy all the constraints corresponding to the dual problem if the step size λ fulfills the bounding constraints (5) and (6). In Section III-A it was shown how the best direction and the corresponding step size parameter λ are chosen. After determining the direction and step size, the clipping function, equation (29), ensures that the aforementioned boundary conditions of the dual form are satisfied.

B. Offset Parameter of the Decision Function

In order to calculate the offset parameter b of the decision function, suppose $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are the solution of the SVMp+ dual problem (2). Define the two sets $N \triangleq \{i | 1 \le i \le m,\ \alpha_i > 0\}$ and $N' \triangleq \{i | m+1 \le i \le n,\ 0 < \alpha_i < C\}$. By the conditions in SVMp+ (1), for the support vectors the Karush-Kuhn-Tucker (KKT) conditions state [44], [49]:

$$\forall i \in N \qquad y_i \left(\mathbf{w} \cdot \mathbf{z}_i + b \right) = 1 - \left(\mathbf{w}^* \cdot \mathbf{z}_i^* + b^* \right) \tag{31}$$

$$\forall i \in N' \qquad y_i \left(\mathbf{w} \cdot \mathbf{z}_i + b \right) = 1 \tag{32}$$

Define:

$$F_i \triangleq \mathbf{w} \cdot \mathbf{z}_i |_{i \in N} = \sum_{j=1}^{n} y_j \alpha_j K_{ij} |_{i \in N}$$

$$F_i' \triangleq \mathbf{w} \cdot \mathbf{z}_i |_{i \in N'} = \sum_{j=1}^{n} y_j \alpha_j K_{ij} |_{i \in N'}$$

$$f_i \triangleq \gamma \mathbf{w}^* \cdot \mathbf{z}_i^* |_{i \in N} = \sum_{j=1}^{m} \left(\alpha_j + \beta_j - C^* \right) K_{ij}^* |_{i \in N}$$

The equalities in equations (31) and (32) can be rewritten as:

$$\begin{cases} b + b^* = 1 - \frac{f_i}{\gamma} - F_i & \forall i \in N,\ y_i = 1 \\ b - b^* = -1 + \frac{f_i}{\gamma} - F_i & \forall i \in N,\ y_i = -1 \\ b = 1 - F_i' & \forall i \in N',\ y_i = 1 \\ b = -1 - F_i' & \forall i \in N',\ y_i = -1 \end{cases}$$

Define $N_+ = \{i | i \in N,\ y_i = 1\}$ and $S_+ = \sum_{i \in N_+} \left(1 - \frac{f_i}{\gamma} - F_i \right)$, $N_- = \{i | i \in N,\ y_i = -1\}$ and $S_- = \sum_{i \in N_-} \left(-1 + \frac{f_i}{\gamma} - F_i \right)$, $N_+' = \{i | i \in N',\ y_i = 1\}$ and $S_+' = \sum_{i \in N_+'} \left(1 - F_i' \right)$, $N_-' = \{i | i \in N',\ y_i = -1\}$ and $S_-' = \sum_{i \in N_-'} \left(-1 - F_i' \right)$. Solving the four equations gives two possible answers: $b = \frac{1}{2} \left(\frac{S_+}{|N_+|} + \frac{S_-}{|N_-|} \right)$ and $b = \frac{1}{2} \left(\frac{S_+'}{|N_+'|} + \frac{S_-'}{|N_-'|} \right)$. The following average for the offset parameter was used:

$$b = \left[\frac{|N|}{|N| + |N'|} \left(\frac{S_+}{|N_+|} + \frac{S_-}{|N_-|} \right) + \frac{\left| N' \right|}{|N| + |N'|} \left(\frac{S_+'}{|N_+'|} + \frac{S_-'}{|N_-'|} \right) \right]$$

APPENDIX II
MIXTURE MODEL

Let $\boldsymbol{\theta}$ be a $(n+m)$-variable vector as the concatenation of the $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ variables: $\boldsymbol{\theta} \triangleq (\boldsymbol{\alpha}, \boldsymbol{\beta})^T$.

$$\lambda^*\left(\boldsymbol{\theta}^{\text{old}},\mathbf{s}^{*(i)}\right)=\min_{\substack{k\in\mathbf{s}^{*(i)},\,u_k>0\\ m+1\le k\le n}}\left\{\frac{C-\theta_k^{\text{old}}}{u_k},\ \min_{\substack{j\in\mathbf{s}^{*(i)}\\ u_j<0}}\left(\lambda^{'}\left(\boldsymbol{\theta}^{\text{old}},\mathbf{s}^{*(i)}\right),\left|\frac{\theta_j^{\text{old}}}{u_j}\right|\right)\right\} \tag{29}$$

$$\lambda^*\left(\boldsymbol{\theta}^{\text{old}},\mathbf{s}^{*(i)}\right)=\begin{cases}\min\left\{\frac{\rho C^*-\theta_k^{\text{old}}}{u_k},\ \min\limits_{\substack{j\in\mathbf{s}^{*(i)}\\ u_j<0}}\left(\lambda^{'}\left(\boldsymbol{\theta}^{\text{old}},\mathbf{s}^{*(i)}\right),\left|\frac{\theta_j^{\text{old}}}{u_j}\right|\right)\right\} & 1\le k\le m,\ k\in\mathbf{s}^{*(i)},\ u_k>0\\ \min\left\{\frac{C-\theta_k^{\text{old}}}{u_k},\ \min\limits_{\substack{j\in\mathbf{s}^{*(i)}\\ u_j<0}}\left(\lambda^{'}\left(\boldsymbol{\theta}^{\text{old}},\mathbf{s}^{*(i)}\right),\left|\frac{\theta_j^{\text{old}}}{u_j}\right|\right)\right\} & m+1\le k\le n,\ k\in\mathbf{s}^{*(i)},\ u_k>0\end{cases} \tag{30}$$

A. Feasible Directions and the Clipping Function

For the cost function in equation (10) and its constraints, the sets of feasible directions are as follows:

Direction 1: $I_1 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2)\,,\ n+1\le s_1,s_2\le n+m,\ s_1\neq s_2;\ u_{s_1}=1,\ u_{s_2}=-1,\ \theta_{s_2}>0,\ \forall i\notin \mathbf{s}\,u_i=0\}$.

Direction 2: $I_2 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2)\,,\ 1\le s_1,s_2\le m,\ s_1\neq s_2,\ y_{s_1}=y_{s_2};\ u_{s_1}=1,\ \theta_{s_1}<\rho C^*,\ u_{s_2}=-1,\ \theta_{s_2}>0,\ \forall i\notin \mathbf{s}\,u_i=0\}$.

Direction 3: $I_3 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2)\,,\ m+1\le s_1,s_2\le n,\ s_1\neq s_2,\ y_{s_1}=y_{s_2};\ u_{s_1}=1,\ \theta_{s_1}<C,\ u_{s_2}=-1,\ \theta_{s_2}>0,\ \forall i\notin \mathbf{s}\,u_i=0\}$.

Direction 4: $I_4 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2)\,,\ m+1\le s_1,s_2\le n,\ s_1\neq s_2,\ y_{s_1}\neq y_{s_2},\ \forall i\notin \mathbf{s}\,u_i=0;\ u_{s_1}=u_{s_2}=1,\ \theta_{s_1}<C,\ \theta_{s_2}<C\ \text{ or }\ u_{s_1}=u_{s_2}=-1,\ \theta_{s_1}>0,\ \theta_{s_2}>0\}$.

Direction 5: $I_5 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2,s_3)\,,\ 1\le s_1,s_2\le m,\ n+1\le s_3\le n+m,\ s_1\neq s_2,\ y_{s_1}\neq y_{s_2},\ \forall i\notin \mathbf{s}\,u_i=0;\ u_{s_1}=u_{s_2}=1,\ \theta_{s_1}<\rho C^*,\ \theta_{s_2}<\rho C^*,\ u_{s_3}=-2,\ \theta_{s_3}>0\ \text{ or }\ u_{s_1}=u_{s_2}=-1,\ \theta_{s_1}>0,\ \theta_{s_2}>0,\ u_{s_3}=2\}$.

Direction 6: $I_6 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2,s_3)\,,\ 1\le s_1\le m,\ m+1\le s_2\le n,\ n+1\le s_3\le n+m,\ y_{s_1}=y_{s_2},\ \forall i\notin \mathbf{s}\,u_i=0;\ u_{s_1}=1,\ \theta_{s_1}<\rho C^*,\ u_{s_2}=-1,\ \theta_{s_2}>0,\ u_{s_3}=-1,\ \theta_{s_3}>0\ \text{ or }\ u_{s_1}=-1,\ \theta_{s_1}>0,\ u_{s_2}=1,\ \theta_{s_2}<C,\ u_{s_3}=1\}$.

Direction 7: $I_7 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2,s_3)\,,\ 1\le s_1\le m,\ m+1\le s_2\le n,\ n+1\le s_3\le n+m,\ y_{s_1}\neq y_{s_2},\ \forall i\notin \mathbf{s}\,u_i=0;\ u_{s_1}=u_{s_2}=1,\ \theta_{s_1}<\rho C^*,\ \theta_{s_2}<C,\ u_{s_3}=-1,\ \theta_{s_3}>0\ \text{ or }\ u_{s_1}=u_{s_2}=-1,\ \theta_{s_1}>0,\ \theta_{s_2}>0,\ u_{s_3}=1\}$.

Direction 8: $I_8 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2,s_3)\,,\ 1\le s_1,s_2\le m, m+1\le s_3\le n, s_1\neq s_2, y_{s_1}\neq y_{s_2}, y_{s_3}=y_{s_2}, \forall i\notin \mathbf{s}u_i=0;\ u_{s_1}=1,\ \theta_{s_1}<\rho C^*,\ u_{s_2}=-1,\ \theta_{s_2}>0,\ u_{s_3}=2,\ \theta_{s_3}<C\ \text{ or }\ u_{s_1}=-1,\ \theta_{s_1}>0,\ u_{s_2}=1,\ \theta_{s_2}<\rho C^*,\ u_{s_3}=-2,\ \theta_{s_3}>0\}$.

Direction 9: $I_9 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2,s_3)\,,\ 1\le s_1,s_2\le m,\ m+1\le s_3\le n,\ s_1\neq s_2,\ y_{s_1}\neq y_{s_2},\ y_{s_3}=y_{s_1},\ \forall i\notin \mathbf{s}\,u_i=0;\ u_{s_1}=1,\ \theta_{s_1}<\rho C^*,\ u_{s_2}=-1,\ \theta_{s_2}>0,\ u_{s_3}=-2,\ \theta_{s_3}>0\ \text{ or }\ u_{s_1}=-1,\ \theta_{s_1}>0,\ u_{s_2}=1,\ \theta_{s_2}<\rho C^*,\ u_{s_3}=2,\ \theta_{s_3}<C\}$.

It can be verified that when moving from any feasible point $\boldsymbol{\theta}^{\text{old}}$ in the direction of $\mathbf{u_s}\in\cup I_i$ and applying the clipping function of equation (30), the constraints corresponding to dual problems are satisfied.

B. Offset Parameter of the Decision Function

In order to calculate the offset parameter b of the decision function, suppose $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are the solution of the SVMp+ dual problem (10). Define the two sets $N\triangleq\{i|1\le i\le m,\ 0<\alpha_i<\rho C^*\}$ and $N^{'}\triangleq\{i|m+1\le i\le n,\ 0<\alpha_i<C\}$. The rest of the calculations are similar to the previous case in Appendix I.

Appendix III
Symmetric Mixture Model

Let $\boldsymbol{\theta}$ be a $(n+m)$-variable vector as the concatenation of the $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ variables: $\boldsymbol{\theta}\triangleq(\boldsymbol{\alpha},\boldsymbol{\beta})^T$.

A. Feasible Directions and the Clipping Function

Using the cost function in equation (17), the sets of feasible directions are

Direction 1: $I_1 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2)\,,\ n+1\le s_1,s_2\le n+m,\ s_1\neq s_2,\ y_{s_1-n}=y_{s_2-n};\ u_{s_1}=1,\ u_{s_2}=-1,\ \theta_{s_2}>0,\ \forall i\notin \mathbf{s}\,u_i=0\}$.

Direction 2: $I_2 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2)\,,\ n+1\le s_1,s_2\le n+m,\ s_1\neq s_2,\ y_{s_1-n}\neq y_{s_2-n},\ \forall i\notin \mathbf{s}\,u_i=0;\ u_{s_1}=u_{s_2}=1\ \text{ or }\ u_{s_1}=u_{s_2}=-1,\ \theta_{s_1}>0,\ \theta_{s_2}>0\}$.

Direction 3: $I_3 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2)\,,\ 1\le s_1,s_2\le m,\ s_1\neq s_2,\ y_{s_1}=y_{s_2};\ u_{s_1}=1,\ \theta_{s_1}<\rho C^*,\ u_{s_2}=-1,\ \theta_{s_2}>0,\ \forall i\notin \mathbf{s}\,u_i=0\}$.

Direction 4: $I_4 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2)\,,\ 1\le s_1,s_2\le m,\ s_1\neq s_2, y_{s_1}\neq y_{s_2}, \forall i\notin \mathbf{s}u_i=0; u_{s_1}=u_{s_2}=1, \theta_{s_1}<\rho C^*, \theta_{s_2}<\rho C^*\ \text{ or }\ u_{s_1}=u_{s_2}=-1,\ \theta_{s_1}>0,\ \theta_{s_2}>0\}$.

Direction 5: $I_5 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2)\,,\ m+1\le s_1,s_2\le n,\ s_1\neq s_2,\ y_{s_1}=y_{s_2};\ u_{s_1}=1,\ \theta_{s_1}<C,\ u_{s_2}=-1,\ \theta_{s_2}>0,\ \forall i\notin \mathbf{s}\,u_i=0\}$.

Direction 6: $I_6 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2)\,,\ m+1\le s_1,s_2\le n,\ s_1\neq s_2,\ y_{s_1}\neq y_{s_2},\ \forall i\notin \mathbf{s}\,u_i=0;\ u_{s_1}=u_{s_2}=1,\ \theta_{s_1}<C,\ \theta_{s_2}<C\ \text{ or }\ u_{s_1}=u_{s_2}=-1,\ \theta_{s_1}>0,\ \theta_{s_2}>0\}$.

Direction 7: $I_7 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2,s_3)\,,\ 1\le s_1\le m,\ m+1\le s_2\le n,\ n+1\le s_3\le n+m,\ y_{s_1}=y_{s_2}=y_{s_3-n},\ \forall i\notin \mathbf{s}\,u_i=0;\ u_{s_1}=-1,\ \theta_{s_1}>0,\ u_{s_2}=1,\ \theta_{s_2}<C,\ u_{s_3}=1\ \text{ or }\ u_{s_1}=1,\ \theta_{s_1}<\rho C^*,\ u_{s_2}=-1,\ \theta_{s_2}>0,\ u_{s_3}=-1,\ \theta_{s_3}>0\}$.

Direction 8: $I_8 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2,s_3)\,,\ 1\le s_1\le m,\ m+1\le s_2\le n,\ n+1\le s_3\le n+m,\ y_{s_1}=y_{s_2},\ y_{s_3-n}\neq y_{s_1},\ \forall i\notin \mathbf{s}\,u_i=0;\ u_{s_1}=-1,\ \theta_{s_1}>0,\ u_{s_2}=1,\ \theta_{s_2}<C,\ u_{s_3}=-1,\ \theta_{s_3}>0\ \text{ or }\ u_{s_1}=1,\ \theta_{s_1}<\rho C^*,\ u_{s_2}=-1,\ \theta_{s_2}>0,\ u_{s_3}=1\}$.

Direction 9: $I_9 \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2,s_3)\,,\ 1\le s_1\le m,\ m+1\le s_2\le n,\ n+1\le s_3\le n+m,\ y_{s_1}\neq y_{s_2},\ y_{s_3-n}=y_{s_1},\ \forall i\notin \mathbf{s}\,u_i=0;\ u_{s_1}=1,\ \theta_{s_1}<\rho C^*,\ u_{s_2}=1,\ \theta_{s_2}<C,\ u_{s_3}=-1,\ \theta_{s_3}>0\ \text{ or }\ u_{s_1}=-1,\ \theta_{s_1}>0,\ u_{s_2}=-1,\ \theta_{s_2}>0,\ u_{s_3}=1\}$.

Direction 10: $I_{10} \triangleq \{\mathbf{u_s}|\mathbf{s}=(s_1,s_2,s_3)\,,1\le s_1\le m, m+1\le s_2\le n, n+1\le s_3\le n+m, y_{s_1}\neq y_{s_2}, y_{s_3-n}=y_{s_2}, \forall i\notin$

$$\lambda^*\left(\boldsymbol{\theta}^{\text{old}},\mathbf{s}^{*(i)}\right)=\begin{cases}\min\left\{\rho C^*-\theta_k^{\text{old}},\ \min\limits_{\substack{j\in\mathbf{s}^{*(i)}\\u_j<0}}\left(\lambda'\left(\boldsymbol{\theta}^{\text{old}},\mathbf{s}^{*(i)}\right),\left|\frac{\theta_j^{\text{old}}}{u_j}\right|\right)\right\} & 1\le k\le m,\ k\in\mathbf{s}^{*(i)},\ u_k>0\\ \min\left\{C-\theta_k^{\text{old}},\ \min\limits_{\substack{j\in\mathbf{s}^{*(i)}\\u_j<0}}\left(\lambda'\left(\boldsymbol{\theta}^{\text{old}},\mathbf{s}^{*(i)}\right),\left|\frac{\theta_j^{\text{old}}}{u_j}\right|\right)\right\} & m+1\le k\le n,\ k\in\mathbf{s}^{*(i)},\ u_k>0\end{cases} \tag{33}$$

$$\lambda^*\left(\boldsymbol{\theta}^{\text{old}},\mathbf{s}^{*(i)}\right)=\begin{cases}\min\left\{\frac{\rho\pi_k C^*-\theta_k^{\text{old}}}{u_k},\ \min\limits_{\substack{j\in\mathbf{s}^{*(i)}\\u_j<0}}\left(\lambda'\left(\boldsymbol{\theta}^{\text{old}},\mathbf{s}^{*(i)}\right),\left|\frac{\theta_j^{\text{old}}}{u_j}\right|\right)\right\} & 1\le k\le m,\ k\in\mathbf{s}^{*(i)},\ u_k>0\\ \min\left\{\frac{C\pi_k-\theta_k^{\text{old}}}{u_k},\ \min\limits_{\substack{j\in\mathbf{s}^{*(i)}\\u_j<0}}\left(\lambda'\left(\boldsymbol{\theta}^{\text{old}},\mathbf{s}^{*(i)}\right),\left|\frac{\theta_j^{\text{old}}}{u_j}\right|\right)\right\} & m+1\le k\le n,\ k\in\mathbf{s}^{*(i)},\ u_k>0\end{cases} \tag{34}$$

$\mathbf{s}\, u_i = 0;\ u_{s_1} = 1,\ \theta_{s_1} < \rho C^*,\ u_{s_2} = 1,\ \theta_{s_2} < C,\ u_{s_3} = 1$ or $u_{s_1} = -1,\ \theta_{s_1} > 0,\ u_{s_2} = -1,\ \theta_{s_2} > 0,\ u_{s_3} = -1,\ \theta_{s_3} > 0\}$.

As described in the paper, after determining the best feasible direction and the corresponding step size, the clipping function of equation (33) ensures that the boundary conditions of the dual form are satisfied.

B. Offset Parameter of the Decision Function

If (17) is considered, the two sets $N \triangleq \{i|1 \le i \le m,\ 0 < \alpha_i < \rho C^*\}$ and $N' \triangleq \{i|m+1 \le i \le n,\ 0 < \alpha_i < C\}$ are defined. By the conditions for (16), for the support vectors, the KKT conditions state:

$$\begin{aligned}\forall i \in N \qquad & y_i\left(\mathbf{w}\cdot\mathbf{z}_i + b\right) = 1 - y_i\left(\mathbf{w}^*\cdot\mathbf{z}_i^* + b^*\right)\\ \forall i \in N' \qquad & y_i\left(\mathbf{w}\cdot\mathbf{z}_i + b\right) = 1\end{aligned}$$

Define (note that f_i is not the same as the previous cases in Appendices I and II):

$$\begin{aligned}F_i &\triangleq \mathbf{w}\cdot\mathbf{z}_i|_{i\in N} = \sum_{j=1}^{n} y_j\alpha_j K_{ij}|_{i\in N}\\ F_i' &\triangleq \mathbf{w}\cdot\mathbf{z}_i|_{i\in N'} = \sum_{j=1}^{n} y_j\alpha_j K_{ij}|_{i\in N'}\\ f_i &\triangleq \gamma\mathbf{w}^*\cdot\mathbf{z}_i^*|_{i\in N} = \sum_{j=1}^{m}\left(\alpha_j+\beta_j-C^*\right) y_j K_{ij}^*|_{i\in N}\end{aligned}$$

The aforementioned conditions can be written as

$$\begin{cases} b + b^* = 1 - \frac{f_i}{\gamma} - F_i & \forall i \in N,\ y_i = 1\\ b + b^* = -1 - \frac{f_i}{\gamma} - F_i & \forall i \in N,\ y_i = -1\\ b = 1 - F_i' & \forall i \in N',\ y_i = 1\\ b = -1 - F_i' & \forall i \in N',\ y_i = -1\end{cases}$$

Note that not only is f_i not the same as previous cases, but the second equation has also been changed. Define $N_+ = \{i|i \in N,\ y_i = 1\}$ and $S_+ = \sum_{i\in N_+}\left(1 - \frac{f_i}{\gamma} - F_i\right)$, $N_- = \{i|i \in N,\ y_i = -1\}$ and $S_- = \sum_{i\in N_-}\left(-1 - \frac{f_i}{\gamma} - F_i\right)$, $N_+' = \{i|i \in N',\ y_i = 1\}$ and $S_+' = \sum_{i\in N_+'}\left(1 - F_i'\right)$, $N_-' = \{i|i \in N',\ y_i = -1\}$ and $S_-' = \sum_{i\in N_-'}\left(-1 - F_i'\right)$. Solving the four equations gives two possible answers: $b = \frac{1}{2}\left(\frac{S_+}{|N_+|} + \frac{S_-}{|N_-|}\right)$ and $b = \frac{1}{2}\left(\frac{S_+'}{|N_+'|} + \frac{S_-'}{|N_-'|}\right)$. The following average for the offset parameter was used:

$$b = \left[\frac{|N|}{|N|+|N'|}\left(\frac{S_+}{|N_+|} + \frac{S_-}{|N_-|}\right) + \frac{\left|N'\right|}{|N|+|N'|}\left(\frac{S_+'}{\left|N_+'\right|} + \frac{S_-'}{\left|N_-'\right|}\right)\right]$$

Appendix IV
LULUPAPI Mixture Model

Let $\boldsymbol{\theta}$ be a $(n+m)$-variable vector as the concatenation of the $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ variables: $\boldsymbol{\theta} \triangleq (\boldsymbol{\alpha},\boldsymbol{\beta})^T$.

A. Feasible Directions and the Clipping Function

For the cost function in equation (23) and its constraints, the sets of feasible directions are as follows:

Direction 1: $I_1 \triangleq \{\mathbf{u_s}|\mathbf{s} = (s_1, s_2),\ n+1 \le s_1, s_2 \le n+m,\ s_1 \ne s_2;\ u_{s_1} = 1,\ u_{s_2} = -1,\ \theta_{s_2} > 0,\ \forall i \notin \mathbf{s}\, u_i = 0\}$.

Direction 2: $I_2 \triangleq \{\mathbf{u_s}|\mathbf{s} = (s_1, s_2),\ 1 \le s_1, s_2 \le m,\ s_1 \ne s_2,\ y_{s_1} = y_{s_2};\ u_{s_1} = 1,\ \theta_{s_1} < \rho C^*\pi_{s_1},\ u_{s_2} = -1,\ \theta_{s_2} > 0,\ \forall i \notin \mathbf{s}\, u_i = 0\}$.

Direction 3: $I_3 \triangleq \{\mathbf{u_s}|\mathbf{s} = (s_1, s_2),\ m+1 \le s_1, s_2 \le n,\ s_1 \ne s_2, y_{s_1} = y_{s_2};\ u_{s_1} = 1, \theta_{s_1} < C\pi_{s_1},\ u_{s_2} = -1, \theta_{s_2} > 0,\ \forall i \notin \mathbf{s}\, u_i = 0\}$.

Direction 4: $I_4 \triangleq \{\mathbf{u_s}|\mathbf{s} = (s_1, s_2),\ m+1 \le s_1, s_2 \le n,\ s_1 \ne s_2,\ y_{s_1} \ne y_{s_2},\ \forall i \notin \mathbf{s}\, u_i = 0;\ u_{s_1} = u_{s_2} = 1,\ \theta_{s_1} < C\pi_{s_1}, \theta_{s_2} < C\pi_{s_2}$ or $u_{s_1} = u_{s_2} = -1, \theta_{s_1} > 0, \theta_{s_2} > 0\}$.

Direction 5: $I_5 \triangleq \{\mathbf{u_s}|\mathbf{s} = (s_1, s_2, s_3),\ 1 \le s_1, s_2 \le m,\ n+1 \le s_3 \le n+m,\ s_1 \ne s_2,\ y_{s_1} \ne y_{s_2},\ \forall i \notin \mathbf{s}\, u_i = 0;\ u_{s_1} = u_{s_2} = 1,\ \theta_{s_1} < \rho C^*\pi_{s_1},\ \theta_{s_2} < \rho C^*\pi_{s_2},\ u_{s_3} = -2,\ \theta_{s_3} > 0$ or $u_{s_1} = u_{s_2} = -1,\ \theta_{s_1} > 0, \theta_{s_2} > 0,\ u_{s_3} = 2\}$.

Direction 6: $I_6 \triangleq \{\mathbf{u_s}|\mathbf{s} = (s_1, s_2, s_3),\ 1 \le s_1 \le m,\ m+1 \le s_2 \le n,\ n+1 \le s_3 \le n+m,\ y_{s_1} = y_{s_2},\ \forall i \notin \mathbf{s}\, u_i = 0;\ u_{s_1} = 1,\ \theta_{s_1} < \rho C^*\pi_{s_1},\ u_{s_2} = -1,\ \theta_{s_2} > 0,\ u_{s_3} = -1,\ \theta_{s_3} > 0$ or $u_{s_1} = -1,\ \theta_{s_1} > 0,\ u_{s_2} = 1,\ \theta_{s_2} < C\pi_{s_2},\ u_{s_3} = 1\}$.

Direction 7: $I_7 \triangleq \{\mathbf{u_s}|\mathbf{s} = (s_1, s_2, s_3),\ 1 \le s_1 \le m,\ m+1 \le s_2 \le n,\ n+1 \le s_3 \le n+m,\ y_{s_1} \ne y_{s_2},\ \forall i \notin \mathbf{s}\, u_i = 0;\ u_{s_1} = u_{s_2} = 1,\ \theta_{s_1} < \rho C^*\pi_{s_1},\ \theta_{s_2} < C\pi_{s_2},\ u_{s_3} =$

$-1,\ \theta_{s_3} > 0$ or $u_{s_1} = u_{s_2} = -1,\ \theta_{s_1} > 0,\ \theta_{s_2} > 0,\ u_{s_3} = 1\}$.

Direction 8: $I_8 \triangleq \{\mathbf{u_s} | \mathbf{s} = (s_1, s_2, s_3),\ 1 \leq s_1, s_2 \leq m,\ m+1 \leq s_3 \leq n,\ s_1 \neq s_2,\ y_{s_1} \neq y_{s_2},\ y_{s_3} = y_{s_2},\ \forall i \notin \mathbf{s}\ u_i = 0;\ u_{s_1} = 1,\ \theta_{s_1} < \rho C^* \pi_{s_1},\ u_{s_2} = -1,\ \theta_{s_2} > 0,\ u_{s_3} = 2,\ \theta_{s_3} < C\pi_{s_3}$ or $u_{s_1} = -1,\ \theta_{s_1} > 0,\ u_{s_2} = 1,\ \theta_{s_2} < \rho C^* \pi_{s_2},\ u_{s_3} = -2,\ \theta_{s_3} > 0\}$.

Direction 9: $I_9 \triangleq \{\mathbf{u_s} | \mathbf{s} = (s_1, s_2, s_3),\ 1 \leq s_1, s_2 \leq m,\ m+1 \leq s_3 \leq n,\ s_1 \neq s_2,\ y_{s_1} \neq y_{s_2},\ y_{s_3} = y_{s_1},\ \forall i \notin \mathbf{s}\ u_i = 0;\ u_{s_1} = 1,\ \theta_{s_1} < \rho C^* \pi_{s_1},\ u_{s_2} = -1,\ \theta_{s_2} > 0,\ u_{s_3} = -2,\ \theta_{s_3} > 0$ or $u_{s_1} = -1,\ \theta_{s_1} > 0,\ u_{s_2} = 1,\ \theta_{s_2} < \rho C^* \pi_{s_2},\ u_{s_3} = 2,\ \theta_{s_3} < C\pi_{s_3}\}$.

It can be verified that when moving from any feasible point $\boldsymbol{\theta}^{\text{old}}$ in the direction of $\mathbf{u_s} \in \cup I_i$ and applying the clipping function of equation (34), the constraints corresponding to dual problems are satisfied.

B. Offset Parameter of the Decision Function

In order to calculate the offset parameter b of the decision function, suppose $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are the solution of the SVMp+ dual problem (23). Define two sets $N \triangleq \{i | 1 \leq i \leq m,\ 0 < \alpha_i < \rho\pi_i C^*\}$ and $N^{'} \triangleq \{i | m+1 \leq i \leq n,\ 0 < \alpha_i < C\pi_i\}$. The rest of the calculations are then similar to the previous case in Appendix I.

Appendix V
Abbreviations

TABLE V
Abbreviation Table

Abbreviation	Explanation
SVM	Support Vector Machine
SVM+	SVM with privilege information
SVMp+	SVM with partially available privilege information
SMO	Sequential Minimal Optimization
LU	Label Uncertainty
LUPI	Learning Using Privilege Information
LUPAPI	Learning Using Partially Available Privileged Information
LULUPAPI	Learning Using Label Uncertainty and Partially Available Privileged Information
KKT	Karush-Kuhn-Tucker
AUC	Area Under the Curve
IID	Independent and Identically Distributed
NN	Neural Network
LSTM	Long Short-Term Memory
ARDS	Acute Respiratory Distress Syndrome
EHR	Electronic Health Record
PEEP	Positive End-Expiratory Pressure
BNP	Brain Natriuretic Peptide

Appendix VI
Authors' Contributions

E. Sabeti and K. Najarian designed the algorithm. E. Sabeti and J. Drews wrote the paper, wrote the codes and tested the results. M. Sjoding provided data and clinical assessment of the study. J. Gryak gathered and pre-processed the data. N. Reamaroon and E. Warner applied the algorithms on data. K. Najarian led the study.

References

[1] A. J. Walkey, R. Summer, V. Ho, and P. Alkana, "Acute respiratory distress syndrome: epidemiology and management approaches," *Clinical epidemiology*, vol. 4, p. 159, 2012.

[2] G. D. Rubenfeld, E. Caldwell, E. Peabody, J. Weaver, D. P. Martin, M. Neff, E. J. Stern, and L. D. Hudson, "Incidence and outcomes of acute lung injury," *New England Journal of Medicine*, vol. 353, no. 16, pp. 1685–1693, 2005.

[3] G. Rubenfeld, E. Caldwell, E. Peabody, J. Weaver, D. Martin, E. Stern, and L. Hudson, "Incidence and outcomes of acute lung injury," *The New England journal of medicine*, vol. 353, no. 16, pp. 1685–1693, 2005.

[4] R. Brower, M. Matthay, A. Morris, D. Schoenfeld, B. Thompson, and A. Wheeler, "Ventilation with lower tidal volumes as compared with traditional tidal volumes for acute lung injury and the acute respiratory distress syndrome. the acute respiratory distress syndrome network," *The New England journal of medicine*, vol. 342, no. 18, pp. 1301–1308, 2000.

[5] G. Bellani, J. Laffey, T. Pham, E. Fan, L. Brochard, L. Esteban, A. Gattinoni, F. van Harn, A. Larsson, D. McAuley, M. Ranieri, G. Rubenfeld, B. Thompson, H. Wrigge, A. Slutsky, and A. Pesenti, "Epidemiology, patterns of care, and mortality for patients with acute respiratory distress syndrome in intensive care unites in 50 countries," *JAMA: the journal of the American Medical Association*, vol. 315, no. 8, pp. 788–800, 2016.

[6] V. Herasevich, M. Yilmaz, H. Khan, R. D. Hubmayr, and O. Gajic, "Validation of an electronic surveillance system for acute lung injury," *Intensive care medicine*, vol. 35, no. 6, pp. 1018–1023, 2009.

[7] H. C. Koenig, B. B. Finkel, S. S. Khalsa, P. N. Lanken, M. Prasad, R. Urbani, and B. D. Fuchs, "Performance of an automated electronic acute lung injury screening system in intensive care unit patients," *Critical care medicine*, vol. 39, no. 1, pp. 98–104, 2011.

[8] M. W. Sjoding, T. P. Hofer, I. Co, J. I. McSparron, and T. J. Iwashyna, "Differences between patients in whom physicians agree and disagree about the diagnosis of acute respiratory distress syndrome," *Annals of the American Thoracic Society*, vol. 16, no. 2, pp. 258–264, 2019.

[9] T. A. D. T. Force*, "Acute Respiratory Distress Syndrome: The Berlin Definition," *JAMA*, vol. 307, no. 23, pp. 2526–2533, 06 2012. [Online]. Available: https://doi.org/10.1001/jama.2012.5669

[10] N. Reamaroon, M. W. Sjoding, K. Lin, T. J. Iwashyna, and K. Najarian, "Accounting for label uncertainty in machine learning for detection of acute respiratory distress syndrome," *IEEE journal of biomedical and health informatics*, vol. 23, no. 1, pp. 407–415, 2018.

[11] V. Vapnik, *Estimation of dependences based on empirical data*. Springer Science & Business Media, 2006.

[12] V. Vapnik and A. Vashist, "A new learning paradigm: Learning using privileged information," *Neural networks*, vol. 22, no. 5-6, pp. 544–557, 2009.

[13] V. Vapnik and R. Izmailov, "Learning using privileged information: similarity control and knowledge transfer." *Journal of machine learning research*, vol. 16, no. 20232049, p. 55, 2015.

[14] B. Frenay and A. Kaban, "A comprehensive introduction to label noise," *European Symposium on Artificial Neural Networks*, pp. 667–676, 2014.

[15] M. Lapin, M. Hein, and B. Schiele, "Learning using privileged information: Svm+ and weighted svm," *Neural Networks*, vol. 53, pp. 95–108, 2014.

[16] F. Cai and V. Cherkassky, "Generalized smo algorithm for svm-based multitask learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 6, pp. 997–1003, 2012.

[17] J. Chen, X. Liu, and S. Lyu, "Boosting with side information," *Asian Conference on Computer Vision*, pp. 563–577, 2012.

[18] S. Fouad, P. Tino, S. Raychaudhury, and P. Schneider, "Incorporating privileged information through metric learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 7, pp. 1086–1098, 2013.

[19] S. Wang, M. He, Y. Zhu, S. He, Y. Liu, and Q. Ji, "Learning with privileged information using bayesian networks," *Frontiers in Computer Science*, vol. 9, no. 2, pp. 185–199, 2015.

[20] H. Yang and I. Patras, "Privileged information-based conditional regression forest for facial feature detection," in *2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*. IEEE, 2013, pp. 1–6.

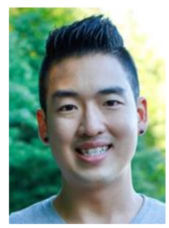
[21] Y. Chen, X. Jin, J. Feng, and S. Yan, "Training group orthogonal neural networks with privileged information," *International Joint Conference on Artificial Intelligence*, 2017.

[22] J. Zhou, X. Xu, S. Pan, I. Tsang, Z. Qin, and R. Goh, "Transfer hashing with privileged information," *International Joint Conference on Artificial Intelligence*, 2016.

[23] R. Marcacini and S. Rezende, "Incremental hierarchical text clustering with privileged information," *Proceedings of the 2013 ACM symposium on document engineering*, pp. 231–232, 2013.
[24] V. Sharmanska, N. Quadrianto, and C. Lampert, "Learning to rank using privileged information," *The IEEE International Conference on Computer Vision*, pp. 825–832, 2013.
[25] S. Wang, D. Tao, and J. Yang, "Relative attribute svm+ learning for age estimation," *IEEE Transactions on Cybernetics*, vol. 46, no. 3, pp. 827–839, 2016.
[26] B. Ribeiro, C. Silva, N. Chen, A. Vieira, and J. Carvalho das Neves, "Enhanced default risk models with svm+," *Expert Systems with Applications*, vol. 39, no. 11, pp. 10 140–10 152, 2012.
[27] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, "Hyperband: A novel bandit-based approach to hyperparameter optimization," *arXiv preprint arXiv:1603.06560*, 2016.
[28] J. Van Hulse and T. Khoshgoftaar, "Knowledge discovery from imbalanced and noisy data," *Data and Knowledge Engineering*, vol. 68, pp. 1513–1542, 2009.
[29] A. Ghosh, N. Manwani, and P. Sastry, "Making risk minimization tolerant to label noise," *Neurocomputing*, vol. 160, pp. 93–107, 2015.
[30] M. Claesen, F. De Smet, J. Suykens, and B. De Moor, "A robust ensemble approach to learn from positive and unlabeled data using svm base models," *Neruocomputing*, vol. 160, pp. 73–84, 2015.
[31] B. Frenay and M. Verleysen, "Classification in the presence of label noise: a survey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 5, pp. 845–869, 2014.
[32] X. Wu, R. He, Z. Sun, and T. Tan, "A light cnn for deep face recognition with noisy labels," *IEEE Transactions on Information Forensices and Security*, vol. 13, no. 11, pp. 2884–2896, 2018.
[33] S. Sukhbaatar, "Training convolutional netwroks with noisy labels," *International Conference on Learning Representations*, 2015.
[34] H. Xiao, B. Biggio, B. Nelson, H. Xiao, C. Eckert, and F. Roli, "Support vector machines under adversarial label contamination," *Neurocomputing*, vol. 160, pp. 53–62, 2015.
[35] A. Bekker and J. Goldberger, "Training deep neural-networks based on unreliable labels," *International Conference on Acoustics, Speech, and Signal Processing*, 2016.
[36] M.-R. Bouguella, S. Nowaczyk, K. Santosh, and A. Verika, "Agreeing to disagree: active learning with noisy labels without crowdsourcing," *International Journal of Machine Learning and Cybernetics*, vol. 9, pp. 1307–1319, 2018.
[37] C. Mantas and J. Abellan, "Credal-c4.5: Decision tree based on imprecise probabilities to classify noisy data," *Expert Systems with Applications*, vol. 41, pp. 4625–4637, 2014.
[38] N. Tomasev and K. Buza, "Hubness-aware knn classification of high-dimensional data in presence of label noise," *Neruocomputing*, vol. 160, pp. 157–172, 2015.
[39] Q. Miao, Y. Cao, G. Xia, M. Gong, J. Liu, and J. Song, "Rboost: Label noise-robust boosting algorithm based on a nonconvez loss function and the numerically stable base learners," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 11, pp. 2216–2228, 2016.
[40] T. Liu and D. Tao, "Classification with noisy labels by importance reweighting," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 38, no. 3, pp. 447–461, 2016.
[41] Z. Wang and Q. Ji, "Classifier learning with hidden information," *Computer Vision and Pattern Recognition*, pp. 4969–4977, 2015.
[42] J. Platt, "Fast training of support vector machines using sequential minimal optimization," *Advances in Kernel Methods*, pp. 41–64, 1998.
[43] D. Pechyony, R. Izmailov, A. Vashist, and V. Vapnik, "Smo-style algorithms for learning using privileged information." in *DMIN*, 2010, pp. 235–241.
[44] D. Pechyony and V. Vapnik, "Fast optimization algorithms for solving svm+," *Stat. Learning and Data Science*, vol. 1, 2011.
[45] T. G. Dietterich, "Approximate statistical tests for comparing supervised classification learning algorithms," *Neural computation*, vol. 10, no. 7, pp. 1895–1923, 1998.
[46] A. Trajman and R. Luiz, "Mcnemar $\chi 2$ test revisited: comparing sensitivity and specificity of diagnostic examinations," *Scandinavian journal of clinical and laboratory investigation*, vol. 68, no. 1, pp. 77–80, 2008.
[47] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[48] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[49] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

Elyas Sabeti is a Postdoctoral Research Fellow at the Michigan Institute for Data Science (MIDAS) at the University of Michigan, Ann Arbor, and a member of Michigan Center for Integrative Research in Critical Care (MCIRCC). He received his Ph.D. of Electrical Engineering from the University of Hawaii at Manoa in 2017. He is currently working on developing clinical decision support systems using machine learning, signal processing and information theory.

Joshua Drews received his MSc in Bioinformatics at the University of Michigan in 2018. He received his B.A. in Mathematics from Northwestern University in 2017. His research interests include pattern recognition and machine learning.



Narathip Reamaroon is a Ph.D. candidate in the Department of Computational Medicine and Bioinformatics at the University of Michigan. He received his B.Sc. from the University of California, San Diego in 2014. He is currently working on developing computer vision methods for analysis of x-ray scans and detection of ARDS. His research interests include applications of machine learning and artificial intelligence for clinical decision support systems.

Elisa Warner is a PhD student in the Department of Computational Medicine and Bioinformatics at the University of Michigan. She also received her B.Sc. in 2011 and MPH in 2015 at the University of Michigan. Her research interests involve the development of novel optimized machine learning techniques, as well as signal and image processing techniques, for use in clinical decision support systems.

Michael Sjoding MD, MSc is Assistant Professor in Internal Medicine, Pulmonary and Critical Care Medicine at the University of Michigan. He also received a masters degree in clinical research from the University of Michigan. As a practicing critical care physician scientist, his research focus is on using routinely collected electronic health data to improve diagnosis and treatment of acute respiratory diseases including the Acute Respiratory Distress Syndrome.

Jonathan Gryak PhD is a Research Manager and Lecturer III of Computational Medicine and Bioinformatics at the University of Michigan. His current research is focused on the design, development, and application of machine learning and computational/mathematical techniques to biomedical applications.

Kayvan Najarian PhD is a Professor of Computational Medicine and Bioinformatics, Professor of Emergency Medicine, and Professor of Electrical Engineering and Computer Science at the University of Michigan. His primary research focus is on computational algorithm using advanced signal/image processing and machine learning to model clinical data. Majority of his works is on designing clinical decision support systems.