IEEE Open Journal of Control Systems

Received 10 December 2025; revised 13 March 2026; accepted 1 April 2026. Date of publication 17 April 2026; date of current version 21 May 2026. Recommended by Guest Editor Insoon Yang.

Digital Object Identifier 10.1109/OJCSYS.2026.3685094

Reducing Annotation Effort in Patient-Ventilator Asynchrony Detection With Distance-Based Clustering

LARS VAN DE KAMP [1,2], DOLF WELLER [3], RICK THIJSSEN [2], BRAM HUNNEKENS [1], TOM BAKKES [4], SIMONA TURCO [5], CORSTIAAN DEN UIL [3], TOM OOMEN [6,7] (Senior Member, IEEE), AND NATHAN VAN DE WOUW [2] (Fellow, IEEE)

(Intersection of Machine Learning with Control)

[1]Demcon Life Sciences and Health, 5683CR Eindhoven, The Netherlands
[2]Dynamics and Control Section, Department of Mechanical Engineering, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands
[3]Maasstad Hospital, 3079 DZ Rotterdam, The Netherlands
[4]Biomedical Diagnostics Lab, Department of Electrical Engineering, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands
[5]Signal Processing Systems Group, Department of Electrical Engineering, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands
[6]Control Systems Technology Section, Department of Mechanical Engineering, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands
[7]Delft Center for Systems and Control, Delft University of Technology, 2628 CD Delft, The Netherlands

CORRESPONDING AUTHOR: LARS VAN DE KAMP (e-mail: l.v.d.kamp97@gmail.com).

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by ondazione I.R.C.C.S. Policlinico San Matteo under Reference No. 41223.

ABSTRACT Objective: This study aims to reduce expert annotation effort in detecting patient–ventilator asynchrony (PVA) by introducing a semi-supervised learning framework for time series classification. Methods and procedures: We propose a model-independent framework that integrates hierarchical clustering and dynamic time-warping (DTW) for efficient data selection and label projection. The framework includes five steps: data collection, selection, annotation, projection, and model training. It is validated using a fully labeled dataset from Fondazione I.R.C.C.S. Policlinico San Matteo and applied to an unlabeled dataset from Maasstad Hospital, where annotation consistency and label quality are analyzed. Results: The framework reduces annotation effort by over 75% while closely resembling classification performance. On the San Matteo dataset, the model trained with projected labels achieved performance close to that of a fully supervised model. The method effectively captured rare PVA types and improved macro-averaged F1 scores compared to random sampling. On the Maasstad dataset, despite annotation inconsistencies, the framework demonstrated moderate detection performance (75% micro-averaged F_1 score) using labels from a single clinical expert. Conclusion: Our semi-supervised framework enables scalable and efficient annotation of clinical time series data, maintaining model accuracy with minimal expert input. It is robust across datasets and adaptable to varying signal quality and annotation consistency.

INDEX TERMS Data selection, label projection, patient-ventilator asynchrony, semi-supervised learning.

I. INTRODUCTION

Interpretation of time-series data plays a critical role in decision making in fields ranging from industrial automation to healthcare monitoring. This data captures the dynamics of the underlying processes and is essential for tasks such as prediction, anomaly detection, and classification. In recent years, supervised learning has emerged as a powerful tool for such tasks, but its success hinges on the availability of high-quality, large, annotated datasets. In practice, obtaining large, accurately labeled datasets is a cost- and time-intensive process and is often limited by the scarce availability of domain experts. As a result, only a limited amount of available

data can be annotated. This requires a data selection procedure that selects an informative subset of the unlabeled dataset to maximize model classification performance.

The problem of selecting data for annotation under the constraint of limited expert availability, with the objective of maximizing model performance, is the main focus of this paper. The limited annotation budget imposed by expert resources is treated as a key constraint, and subsequently, data must be selected strategically to ensure that the resulting subset after annotation yields the greatest possible benefit to model accuracy. Furthermore, the data selection procedure must be independent of the classification model, such that it is applicable to different classification models.

In the literature, data selection methods are predominantly model-dependent. This includes well-established approaches such as active learning and core-set selection. In active learning [1], a model selects data based on the uncertainty of its prediction, and subsequently provides the unlabeled data to an annotator. This process is performed iteratively, but requires a pre-trained model as a starting point. In core-set selection [2], a set of the most informative training samples is selected from the complete set. Both active learning and core-set selection are model-dependent, which makes them computationally expensive and more difficult to generalize. In [3], a more computationally efficient method for data selection is presented by using proxy models instead of the real models. However, we aim to develop a method that selects data directly from raw time-series inputs, without relying on any pre-trained models. This allows it to function as an add-on to existing and deployed classification algorithms.

Another significant challenge lies in the potential degradation of model performance due to the limited size of the annotated dataset. To mitigate this, information derived from the labeled data can be strategically utilized to assist in the annotation of unlabeled samples. This process is referred to as label projection, which involves projecting information from the annotated set to the unlabeled set to generate pseudo-labels [4]. By incorporating these pseudo-labeled samples into the training process, we aim to enhance the performance of the model while mitigating the need for extra annotation. By doing this, we end up with a semi-supervised learning framework.

Other semi-supervised learning approaches also exist in literature, such as MixMatch [5] and Ladder Networks [6]. These methods primarily focus on improving neural network training through consistency regularization or representation learning using unlabeled data, respectively. In [7], both methods are compared for the use-case of patient-ventilator asynchrony detection. In contrast, the current work addresses annotation reduction at the data selection level and proposes a *model-independent* framework that can be integrated with a wide range of classification architectures. To the best of our knowledge, this model-independent method is different from existing semi-supervised learning frameworks.

Although the above data selection techniques show viable methods to select a limited dataset while maintaining classification performance, they do not satisfy the requirement that the selection procedure must be model independent. Furthermore, a method for training a time series classification model with semi-supervised learning that includes a full workflow for data collection, data selection based on raw time series data, data annotation, data projection, and semi-supervised model training is not addressed in the literature.

In this paper, we introduce a semi-supervised learning framework for time series data based on a distance-based cluster method that enables efficient data selection, annotation, and projection, while maintaining classification model performance. To the best of our knowledge, this model-independent method is different from existing semi-supervised learning frameworks. The clustering process facilitates informed data selection and enables the application of label projection to unlabeled instances within each cluster. This method is applied to patient-ventilator asynchrony (PVA) detection in mechanical ventilation, a crucial machine for patient breathing support in intensive care units (ICUs).

The first contribution of this paper is a semi-supervised learning framework that includes a novel method for efficiently selecting time series data and projecting labels on time series data. The second contribution is the application of the presented framework to patient-ventilator asynchrony detection in mechanical ventilation, which includes the following:

a) A method for collecting and annotating data for patient-ventilator asynchrony detection.
b) Clinical validation of the semi-supervised learning framework that includes the data selection, label projection, and performance of the trained PVA detection algorithm using a fully labeled clinical dataset from the Fondazione I.R.C.C.S. Policlinico San Matteo.
c) Application of the entire semi-supervised learning framework to a new unlabeled dataset from the Maasstad Hospital, Rotterdam, which includes a data collection procedure, data selection, data annotation (and an additional consistency analysis), data projection, and model training.

Patient–ventilator asynchrony detection in mechanical ventilation constitutes a suitable benchmark for evaluating the proposed semi-supervised learning framework, due to the availability of large unlabeled datasets, constrained expert annotation capacity, and the existence of mature classification models. Mechanical ventilators are complex mechatronic devices that are essential for patients unable to breathe independently. The goal of mechanical ventilation is to ensure adequate oxygenation and carbon dioxide elimination for a large variety of patients [8]. Synchronization of the ventilatory support with the patient's demand is one of the greatest challenges in supportive ventilation. A timing mismatch between the delivery of support and the patient demand is called Patient-Ventilator Asynchrony (a quantitative definition of PVA is given in Section III-C). Severe levels of PVA are observed in many ventilated patients, ranging from 24% (of the

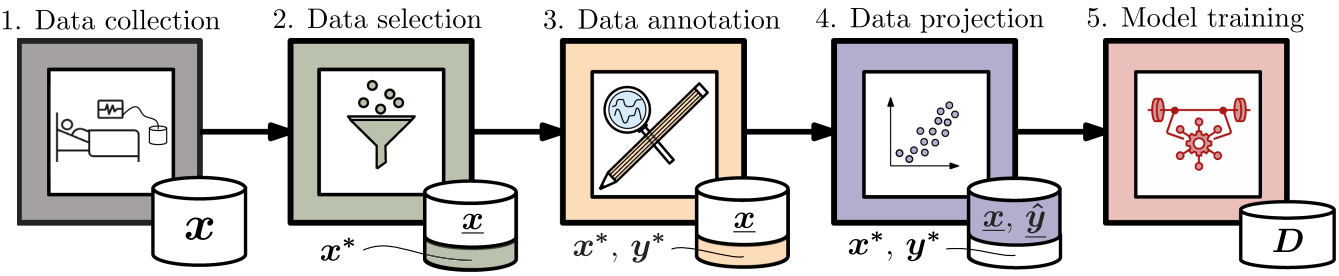

FIGURE 1. Schematic representation of the semi-supervised learning framework, which contains the following steps: data collection, data selection, data annotation, data projection, and model training. The collected multi-variable time-series are represented by x, the selected data by x^* and the remaining input data by $\underline{x}$. The annotated labels are represented by y^*, the projected labels by $\hat{\underline{y}}$, and the training set by D.

total ventilated population) in [9] to 43% in [10]. According to [9], [11], [12], [13], PVA is associated with a prolonged stay in the ICU and even increased mortality, but is very challenging to detect by the human eye [14].

In recent years, substantial research has been done to develop algorithms to automatically and continuously detect PVA [15], [16]. These predictive models are validated on small clinical datasets due to the absence of large, accurately labeled clinical datasets. To implement these algorithms in clinical practice, however, larger and more diverse labeled datasets are required. In the clinical field, a proliferation of unlabeled data is available that human experts must first annotate to make it useful. This annotation process of time series is time-consuming and error-prone; hence, this use case is a perfect example for the efficient time series data selection and annotation process introduced in this paper.

The paper follows the following structure. In Section II, the semi-supervised learning framework that includes time series data selection and projection is presented, which relates to the first contribution. Thereafter, in Section III, the data collection and annotation for the patient-ventilator asynchrony detection use-case is presented (sub-contribution a. of the second contribution). Subsequently, in Section IV, the model for automatic detection of patient-ventilator asynchrony is introduced. Next, in Section V, the semi-supervised learning framework is clinically validated on an existing labeled dataset from the Fondazione I.R.C.C.S. Policlinico San Matteo (sub-contribution b. of the second contribution). Subsequently, in Section VI, the data collection, selection, and annotation steps of the framework are discussed for an unlabeled clinical dataset from the Maasstad hospital, and an inter-expert consistency analysis is conducted (sub-contribution c. of the second contribution). Thereafter, in Section VII, a semi-supervised PVA detection model is trained on annotated and projected labels from the Maasstad dataset (sub-contribution c. of the second contribution). Finally, conclusions and recommendations are given in Section VIII.

II. METHODS AND PROCEDURES

In this section, we discuss the proposed semi-supervised learning framework for training a classifier when starting from an unlabeled dataset. First, in Section II-A, the semi-supervised learning framework, from data collection to the model training is discussed. Next, in Section II-B, a detailed description of the novel distance-based cluster approach for the data selection is presented. Finally, in Section II-C, the label projection method based on the distance-based clustering is presented.

A. SEMI-SUPERVISED LEARNING FRAMEWORK

The proposed semi-supervised model framework presented in this paper requires the following steps: Data collection, data selection, data annotation, data projection, and model training. Each of these steps is shown in Fig. 1 and explained below.

In the *data collection*, informative signals for the problem at hand must be measured (and thus collected) so that the classifier can be trained and, at a later stage, can make accurate predictions. All realizations of the problem must occur in the data collected, such that a model generalizes well for all possible input data. The result from the data collection is a set of unlabeled data $\boldsymbol{x}$, see Fig. 1.1.

In the *data selection*, the unlabeled dataset $\boldsymbol{x}$ is partitioned into an informative subset $\boldsymbol{x}^*$ of the collected unlabeled data and the remaining set $\underline{\boldsymbol{x}}$, see Fig. 1.2. During this step, the most informative data are selected from the unlabeled set to reduce the set size that requires annotation, while maintaining as much variety as possible from the collected dataset.

During *data annotation*, the set $\boldsymbol{x}^*$ is annotated, resulting in the labels $\boldsymbol{y}^*$, see Fig. 1.3. The annotation process is more expensive (time and cost-wise) than the data collection process. Especially in healthcare applications, data can often only be annotated by human expert inspection. As a result, only the partitioned informative subset $\boldsymbol{x}^*$ of all the data collected is annotated. After the annotation process, an informative subset of the data is annotated, resulting in $(\boldsymbol{x}^*, \boldsymbol{y}^*)$, while the remaining part of the data $\underline{\boldsymbol{x}}$ remains unlabeled.

In the *data projection*, the labels of the remaining unlabeled set $\underline{\boldsymbol{x}}$ are predicted and denoted by $\hat{\underline{\boldsymbol{y}}}$, referred to as pseudo-labels, using the annotated data $\boldsymbol{y}^*$ from the selected set $\boldsymbol{x}^*$. As a result, we obtain a dataset $(\boldsymbol{x}^*, \underline{\boldsymbol{x}})$ that is completely labeled $(\boldsymbol{y}^*, \hat{\underline{\boldsymbol{y}}})$, but consists of data that is annotated by experts $\boldsymbol{y}^*$ and annotated via label projection $\hat{\underline{\boldsymbol{y}}}$, see Fig. 1.4.

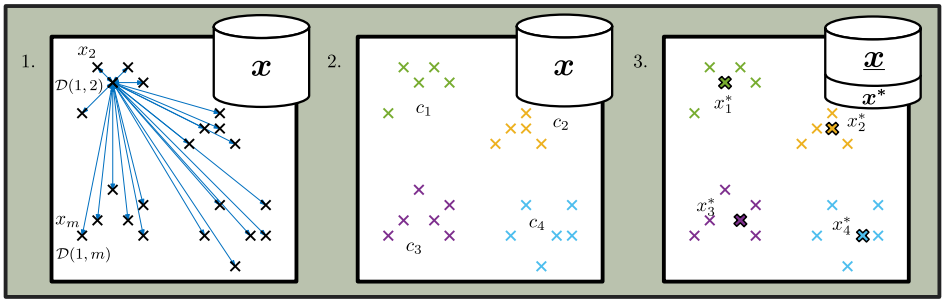

FIGURE 2. Schematic representation of the data selection procedure. During data selection, the steps in Section II-B are followed to partition the dataset $\boldsymbol{x}$ into two sets: $\underline{\boldsymbol{x}}$ and $\boldsymbol{x}^*$.

In the *model training*, a classifier is trained on training data D, which is a combination of data annotated by experts and data with pseudo-labels obtained from the data projection, i.e., a semi-supervised learning approach is obtained.

In the next two subsections, the data selection procedure and the data projection procedure are explained in more detail. The annotation of the data and training of the classification model are discussed in Section IV for the use case of patient-ventilator asynchrony detection in mechanical ventilation.

B. DATA SELECTION

Consider an unlabeled set $\boldsymbol{x} := \{x_1, x_2, \ldots, x_N\}$, where each instance x_i, for $i \in \{1, 2, \ldots, N\}$, is a discrete multivariate time series $x_i \in \mathbb{R}^{n \times q_i}$, with (variable) signal length q_i and n the number of variables. From this unlabeled dataset $\boldsymbol{x}$, a smaller subset $\boldsymbol{x}^* \subseteq \boldsymbol{x}$ must be selected, such that in an ideal case the model trained on only $\boldsymbol{x}^*$ converges to the same model as the model trained on the full set $\boldsymbol{x}$ (if the labels would be available). Note that this typically does not occur in practice because the size of $\boldsymbol{x}^*$ (and thus the information) is limited by the available annotation effort. If $|\boldsymbol{x}^*| \ll |\boldsymbol{x}|$, then the annotation effort for human experts is significantly reduced.

We propose a distance-based clustering method to reduce the unlabeled dataset by partitioning the set into a small informative subset $\boldsymbol{x}^*$ and a remaining set $\underline{\boldsymbol{x}}$, see Fig. 2. Clustering enables us to select only the most representative samples $\boldsymbol{x}^* = \{x_1^*, \ldots, x_K^*\}$ from each cluster of the unlabeled set, where K is the number of clusters. More specifically, the clustering method used is Agglomerative Hierarchical Clustering (AHC) [17]. The distance measure used for clustering time series data is the dynamic-time warping (DTW) distance [18]. In DTW, two time series signals are (nonlinearly) warped such that the shape of the two time series is as similar as possible, and subsequently, their distance is measured using the Euclidean distance. In this way, time series that have similar shapes are clustered together. The computational complexity of this method is $\mathcal{O}(IJ)$, where I and J are the length of the time series signal.

The following steps are conducted (and shown in Fig. 2) to select a subset of the data:

1) Compute the DTW distance between all instances from the unlabeled data $\boldsymbol{x} = \{x_1, \ldots, x_N\}$, i.e., $\mathcal{D}(i, j) = d_{\text{DTW}}(x_i, x_j) \;\; \forall i, j \in \{1, 2, \ldots, N\}$, where N is the number of instances in the unlabeled set, see also Fig. 2.1.
2) Cluster the unlabeled set $\boldsymbol{x}$ into K clusters using the DTW distances $\mathcal{D}$ and AHC with complete linkage. This leads to the clusters $\boldsymbol{c} = \{c_1, \ldots, c_K\}$, where each cluster c_i contains one or multiple instances from $\boldsymbol{x}$, see also Fig. 2.2, for an example with $K = 4$ clusters.
3) Compute the most representative instance in each cluster, which is the instance with the smallest maximum distance to the other instances, i.e.,

$$x_a^* = \underset{u \in c_a}{\operatorname{argmin}} \max_{v \in c_a} (d_{\text{DTW}}(u, v)), \tag{1}$$

where (u, v) are instances within cluster c_a with $a \in \{1, 2, \ldots, K\}$. This results in the set of most representative instances $\boldsymbol{x}^* = \{x_1^*, \ldots, x_K^*\}$, where $K < N$, see the bold crosses in Fig. 2.3. Partition the training set $\boldsymbol{x}$ into two sets $\boldsymbol{x}^*$ and $\underline{\boldsymbol{x}}$, where $\underline{\boldsymbol{x}} := \boldsymbol{x} \backslash \boldsymbol{x}^*$ with $(\backslash)$ the set difference operator.

After the data selection, the set $\boldsymbol{x}^*$ is annotated by experts which gives the labels $\boldsymbol{y}^* = \{y_1^*, y_2^*, \ldots, y_K^*\}$ with $y_k(t) \in \mathbb{R}$ for $k \in \{1, 2, \ldots, K\}$. Note that annotation effort by domain experts is significantly reduced when $\boldsymbol{x}^*$ is much smaller than $\boldsymbol{x}$ ($K \ll N$). In contrast the labels for $\underline{\boldsymbol{x}}$ are not obtained by expert labeling but rather estimated using a data projection procedure, which gives $\underline{\hat{\boldsymbol{y}}} = \{\underline{\hat{y}}_1, \underline{\hat{y}}_2, \ldots, \underline{\hat{y}}_{N-K}\}$ and is explained in Section II-C. The data selection step is performed offline and only needs to be executed once, thereby maintaining the applicability of the method to large datasets.

Remark: The selection of a cluster representative may be influenced by outlier signals within a cluster. The hierarchical clustering step based on DTW similarity mitigates this effect by separating dissimilar signals into smaller clusters. Nevertheless, small outliers within clusters may still affect the representative choice.

C. DATA PROJECTION

Consider the set $\boldsymbol{x} := \{x_1, x_2, \ldots, x_N\}$ which is partitioned into the label set $\boldsymbol{x}^*$ with labels $\boldsymbol{y}^*$ and the unlabeled set $\underline{\boldsymbol{x}}$ during the *data selection* procedure. During data projection, we generate the pseudo-labels $\hat{\boldsymbol{y}}$ for the unlabeled dataset $\underline{\boldsymbol{x}}$ using the information from $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ as follows (see also Fig. 3:

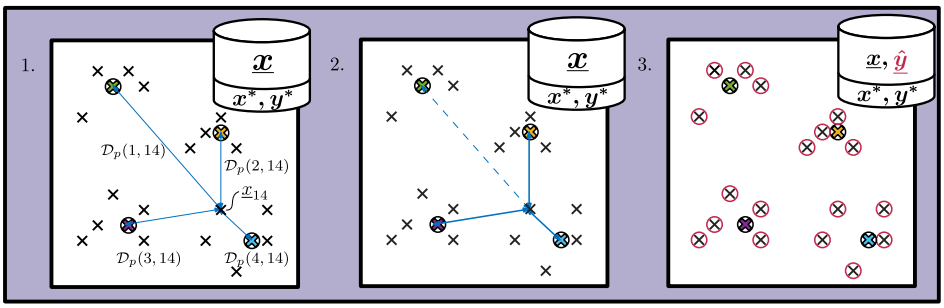

FIGURE 3. Schematic representation of the data projection procedure. During data projection, the steps in Section II-C, where steps 3 and 4 are visualized in the last figure of the data projection visualization, are followed to achieve a completely labeled dataset. The labels y^* of the selected data x^* are projected onto the remaining data $\underline{x}$, which gives the projected labels $\underline{\hat{y}}$.

1) Compute the DTW distance between all instances from the labeled dataset $\boldsymbol{x}^*$ and the unlabeled set $\underline{\boldsymbol{x}}$, that is, $\mathcal{D}_p(i, j) = d_{\text{DTW}}(x_i^*, \underline{x}_j)$, for which $i \in \{1, 2, \ldots, K\}$ and $j \in \{1, 2, \ldots, N-K\}$, where K is the number of instances in the labeled dataset, see Fig. 3.1, where $K = 4$. Furthermore, every $\mathcal{D}_p(i, j)$ is accompanied by an alignment policy π_{ij}^* (see [19] for details) showing how every data point within $x_i^*(w)$, with $w \in \{1, 2, \ldots, W\}$, is mapped by time warping onto a data point in $\underline{x}_j(q)$, with $q \in \{1, 2, \ldots, Q\}$, where W, Q are the length of x_i^* and $\underline{x}_j$, respectively. Let $g(q, \pi_{ij}^*)$ be a function of the time series mapping where
$$g(q, \pi_{ij}^*) = \min \left\{ w \in \{1, 2, \ldots, W\} | (w, q) \in \pi_{ij}^* \right\} \tag{2}$$
with $w \in \{1, 2, \ldots, W\}$. This function shows that $\underline{x}_j(q)$ is aligned with $x_i^*(w)$.
2) Find the M closest instances of $\boldsymbol{x}^*$ with respect to $\underline{x}_j$ with $j \in \{1, 2, \ldots, N-K\}$ using $\mathcal{D}_p(\cdot, j)$ and denote them as x_c^j, see Fig. 3.2 (where $M = 3$ and $x_c^{14} = \{x_2^*, x_3^*, x_4^*\}$).
3) Estimate the label $\underline{y}_j$ of $\underline{x}_j$ based on the label y_i^* of x_i^* using the alignment path π_{ij}^* and the mapping function $g(q, \pi_{ij}^*)$ between x_i^* and $\underline{x}_j$, i.e.,
$$\underline{\hat{y}}_j(q) = y_i^*(g(q, \pi_{ij}^*)) \quad \forall q \in \{1, 2, \ldots, Q\}. \tag{3}$$
Here x_i^* must be a member of the set x_c^j as defined in step 2). The alignment path is known from the data selection step, in Section II-B, making this step computationally efficient.
4) Repeat step 3) for all members in the set x_c^j and round the average of the results to obtain the pseudo-label $\hat{y}_j$ of $\underline{x}_j$.

Thus, after the *data projection*, we obtain a training set that contains both the annotated labels as well as the projected labels, i.e., $\boldsymbol{D} = \{(\boldsymbol{x}^*, \boldsymbol{y}^*), (\underline{\boldsymbol{x}}, \underline{\hat{\boldsymbol{y}}})\}$. Note that the data projection step can adversely affect the performance of the trained classification model when the projection method results in incorrect labels. In particular, if a cluster representative is incorrectly annotated by the expert, this error may propagate to all samples within that cluster, which is a general limitation of semi-supervised learning approaches that rely on label propagation. In practice, this risk can be reduced by keeping cluster sizes relatively small, thereby limiting the potential impact of individual annotation errors and improving the robustness of the label projection step.

III. METHOD: DATA COLLECTION AND ANNOTATION FOR PATIENT-VENTILATOR ASYNCHRONY

In this section, the collection (Step 1 of Fig. 1) and annotation (Step 3 of Fig. 1) of time series data for the detection of patient-ventilator asynchrony is discussed. First, in Section III-A, PVA is elaborated upon. Thereafter, in Section III-B, the data collection procedure is presented. Lastly, in Section III-C, the annotation process is presented, where the annotation rules for the PVA use case are introduced.

A. DEFINITION OF PATIENT-VENTILATOR ASYNCHRONY

Patient-ventilator asynchrony is a mismatch between patient demands and ventilator delivery in mechanical ventilation. In pressure support ventilation, severe and timing asynchronies are the most occurring PVA types [16]. Severe asynchronies are called severe because there is a very large mismatch between the patient and the ventilator, e.g., the ventilator misses a patient breath completely (ineffective effort) or the ventilator triggers without a patient breath (auto trigger). A timing asynchrony occurs if there is a (small) timing mismatch in either the inspiration and/or expiration, e.g., the ventilator triggers too late (delayed triggering) or too early (premature triggering); or the ventilator starts its expiration too early (early cycling) or too late (delayed cycling). A list of asynchrony types that occur during PVA can be found in [16], where objective rules are constructed to distinguish the different types of asynchrony. These objective rules are based on the inspiration and expiration start of the patient and the ventilator. Detecting the specific type of PVA is crucial because each type of PVA is accompanied by a different treatment plan. For the remainder of this paper, the following abbreviations are used for certain (a)synchrony types: normal inspiration (NI), premature triggering (PT), delayed triggering (DT), normal expiration

(NE), premature cycling (PC), delayed cycling (DC), expiratory ineffective effort (IEe), inspiratory ineffective effort (IEi), double triggering (DbT), and auto triggering (AT).

B. DATA COLLECTION

In the data collection process, data from a variety of patients is collected such that the dataset is diverse and includes most PVA types. Data collected in ventilated patients are raw ventilatory signals sampled at 50 Hz, which are airway pressure $p_{\text{aw}}(t)$, patient flow $Q_{\text{pat}}(t)$, patient volume $V_{\text{pat}}(t)$, together with ventilator inspiration start time T_{vi} and ventilator expiration start time T_{ve}. Furthermore, an additional measurement is performed to measure the esophageal pressure $p_{\text{es}}(t)$.

In recent years, esophageal pressure p_{es} measurements using balloon-tipped esophageal catheters have become increasingly common in critical care. The integration of this technology into modern ventilators has made it more accessible in clinical practice, facilitating its use for real-time assessment of respiratory mechanics. However, despite these advances, the technique remains underutilized in many clinical settings. This underuse can be attributed to ongoing debates about technical challenges, the complexity of data interpretation, and uncertainties regarding its integration into routine clinical decision-making [20]. Nevertheless, esophageal pressure monitoring provides critical insights into respiratory mechanics, supporting personalized ventilation strategies and potentially improving outcomes for critically ill patients, [21].

The pressure measured in the esophagus (p_{es}) serves as a surrogate for pleural pressure (p_{pl}) [21]. This measurement provides valuable insight into lung mechanics and the work of breathing (WOB) in spontaneously breathing patients [22], [23]. The shape of the p_{es}-curve thus provides valuable information regarding the timing and magnitude of patient effort. By analyzing the timing relationships between p_{aw}, Q_{pat}, and p_{es} curves, various patient-ventilator asynchronies can be detected, including premature (PT) and delayed triggering (DT), ineffective efforts (IEe and IEi), double triggering (DbT), and both delayed (DC) and premature cycling (PC) [23].

C. ANNOTATION FOR PATIENT-VENTILATOR ASYNCHRONY

Annotation of PVA in ventilator-supported breaths is challenging and requires clinical expertise. We conjecture that annotating the patient's inspiration (T_{pi}) and expiration (T_{pe}) start times and, subsequently, using the objective rules in [16] to find the corresponding asynchrony type is the most consistent way to build a training set for automatic PVA detection. The PVA types are derived from the patient timings; therefore, annotating these timings provides more insight into the underlying mechanisms of PVA. Furthermore, definitions (based on the measured signals) for the patient's inspiration and expiration exist [23], making it more user-friendly to label consistently.

From a clinical perspective, the annotation of the inspiratory and expiratory timing for patient-ventilator asynchrony in this study is based on the detailed analysis of three primary waveforms: airway pressure (p_{aw}), patient flow (Q_{pat}), and esophageal pressure (p_{es}), see Fig. 4.

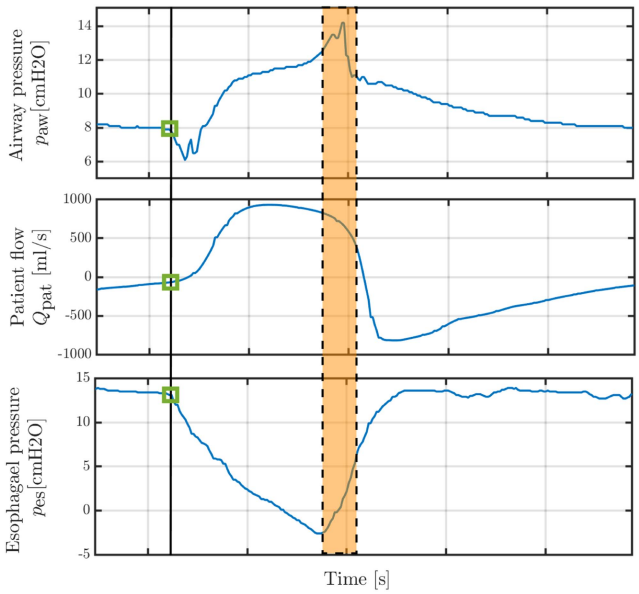

FIGURE 4. Schematic overview of the definition of the patient's inspiration start (□) and the area where the expiration start is located (▭) based on the esophageal pressure.

The start of the patient's inspiration and expiration is defined based on the esophageal pressure. A negative p_{es} deflection reflects the start of the inspiratory effort due to a drop in pleural pressure (see the green rectangle in Fig. 4). The most negative point in the p_{es} curve marks the transition from inspiration to expiration. An exact definition for the start of the expiratory effort solely based on the esophageal pressure remains disputed. In [23], the halfway point in the upward p_{es} curve is arbitrarily chosen. In this study, we define an area in which the expiration start is located, which is between the most negative point and the halfway point of the upward p_{es} curve, see orange area in Fig. 4. The clinical expert must decide where to place the start of the expiration exactly by using information from the airway pressure, patient flow, and esophageal pressure signals.

IV. METHOD: AUTOMATIC DETECTION OF PVA

In this section, a classification model for automatic detection of PVA in mechanical ventilation is presented. The definition of this model is necessary for the model training in step 5 of Fig. 1. The data used to train the model is discussed in Section IV-A. Subsequently, in Section IV-B, the structure of the detection model (based on neural networks) is introduced. Lastly, the model fit criterion and the optimization algorithm are introduced in Section IV-C.

A. MODEL DATA

The input data for the automatic detection network are the airway pressure and patient flow, $x(t) = [p_{\text{aw}}(t), Q_{\text{pat}}(t)]$ sampled at 50 Hz. The choice of using these signals is based

on practical implications as they are readily available on all mechanical ventilators. The esophagus pressure $p_{es}(t)$ is only used for annotation because this is not a standard-of-care measured signal as described in Section III-A. The time-series data is segmented into single ventilator breaths based on the ventilator timings.

After the collection of the unlabeled dataset $\boldsymbol{x}$, the data selection, annotation, and projection workflow is followed as discussed in Section II. The amount of data selected heavily depends on the time available for labeling. Note that if only little time is available for labeling, the dataset needs to be reduced more and some informative data may not make it to the selected set. Furthermore, the larger the set (i.e., more clusters), the more accurate the label projection is (because the data within a cluster is more similar).

During annotation, the clinical experts label the start of the patient's inspiration(s) T_{pi} and expiration(s) T_{pe} in each mechanical breath. Multiple patient breaths could be located within a single mechanical breath (if an asynchrony occurs), i.e., $T_{\text{pi}}, T_{\text{pe}} \in \mathbb{R}^m$ where m is the number of patient breaths within a single input signal. These labels are converted to a time series label, the respiratory state

$$y(t) := \begin{cases} 1, & \text{if } T_{pi}(k) < t \leq T_{pe}(k), \\ 0, & \text{elsewhere}, \end{cases} \tag{4}$$

for breath index $k \in \{1, 2, \ldots, m\}$. Let (4) indicate whether a patient's inspiration is active. Formulating the respiratory state $y(t)$ as a time series allows convenient architecture choices and loss functions when training a model for automatic PVA detection. The output of the model $\hat{y}(t)$ should predict the ground-truth labels $y(t)$ as closely as possible. Note that the patient timings are reconstructed by detecting the jumps in the binary signal ($\hat{T}_{pi}$ is found when $\hat{y}(t)$ goes from 0 to 1 and $\hat{T}_{pe}$ is found when $\hat{y}(t)$ goes from 1 to 0).

B. MODEL STRUCTURE

The overall model structure is defined by the inputs, outputs, and the model structure itself, which is schematically visualized in Fig. 5.

To obtain a mapping from the inputs x to the desired outputs y, a bi-directional Recurrent Neural Network (RNN) is considered. In this paper, a bi-directional RNN with Long Short-Term Memory (LSTM) cells is used, because these models are effective sequence models used in practical applications [24] and proven to perform well for PVA detection [16]. This model structure allows a mapping that can handle varying sequence lengths (and thus varying breath lengths). Asynchronies are detected breath-by-breath; hence, the asynchrony type is determined after a mechanical breath is finished.

A bidirectional network is employed because the detection of patient timing events benefits from information both preceding and following the patient timings. In particular, certain timings become more distinguishable when the subsequent signal evolution is taken into account. By processing the sequence in both forward and backward directions, the network can leverage information from future samples within the analyzed window, thereby achieving more accurate and robust event detection. Standard RNNs have problems with exploding/vanishing gradients when classifying long time-series [25]. The problem of exploding/vanishing gradients is tackled by using LSTM cells in the RNN. These cells contain multiplicative gates that are capable of extracting and storing information over longer periods [26].

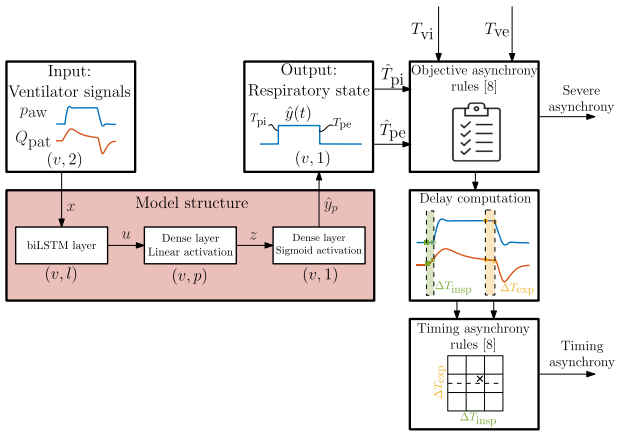

FIGURE 5. Definition of the model structure that is used for patient-ventilator asynchrony detection. Showing the inputs, the overall model structure, and the desired output, and the three process steps to obtain the asynchrony type label.

The overall model structure is schematically depicted in Fig. 5. The model consists of an LSTM layer with l cells, a dense layer with p cells and linear activation functions, and a dense layer with one cell and a sigmoid activation function. The LSTM layer maps the inputs into l different LSTM outputs u. Those LSTM outputs u are combined by a dense layer with linear activation into a pre-output vector $z \in \mathbb{R}^p$. Next, a dense layer with a single cell and sigmoid activation function transforms z into the output $\hat{y}_p(t, \theta)$, which is a value between 0 and 1, that is interpretable as the probability that a particular respiratory state (inspiration/expiration) occurs at the time step t. Let θ represent the model parameters, e.g., the weights and biases. To determine which respiratory state is predicted by the model, $\hat{y}_p(t, \theta)$ is transformed to a one-hot output vector as follows:

$$\hat{y}(t) := \begin{cases} 1 & \text{if } \hat{y}_p(t) \geq 0.5, \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

In words, $\hat{y}(t)$ is the estimated label at time instance t that represents the respiratory state, which can be either 1 (during inspiration) or 0 (during expiration). In the post-processing, the asynchrony type can be derived based on $\hat{y}(t)$ by using the objective asynchrony definitions from [16].

C. MODEL FIT CRITERION AND OPTIMIZATION ALGORITHM

The optimal values of the weights and biases $\theta^\star$ are determined to obtain the optimal mapping from the inputs x

to the ground-truth labels y with the proposed model structure. This is achieved by minimizing a loss function. The loss function, or fit criterion, considered in this paper is the binary focal cross-entropy loss function. This loss function is a sample-wise weighting scheme that counteracts class imbalance, and learns sharper boundaries because errors at jumps (at the inspiration and expiration start) are penalized more [27]. The binary focal cross-entropy loss function is defined as

$$L_{CE}(\theta) = -\sum_{i=1}^{N}\sum_{t=1}^{v}(1-y_i(t))\hat{y}_{p,i}(t)^{\phi}\log(1-\hat{y}_{p,i}(t)) + y_i(t)(1-\hat{y}_{p,i}(t))^{\phi}\log(\hat{y}_{p,i}(t)), \quad (6)$$

where ϕ is the focal factor, N is the number of respiratory cycles within the dataset, v the sequence length of a single breath of use case i in samples, $y_i(t) \in \mathbb{B}$ the ground truth label vector of case i at time t, and $\hat{y}_{p,i}(t,\theta) \in \mathbb{R}_{[0,1]}$ the predicted label vector of case i at time sample t.

The loss landscape, as created by (6), is navigated by the optimizer. The goal of the optimizer is to find the set of parameters θ that leads to the lowest loss $L_{CE}(\theta)$, given the set of training data. For this, the AdamW optimizer is employed, which is a variation of the Adam optimizer, as proposed by [28].

V. RESULTS: CLINICAL VALIDATION OF METHOD [SAN MATTEO DATASET]

In this section, the steps 2,4, and 5 of the framework in Fig. 1 and as proposed in Section II, are evaluated on a clinical dataset. This dataset consists of 15 patients recorded in the Fondazione I.R.C.C.S. Policlinico San Matteo (Pavia, Italy) [23]. Patients are subjected to different levels of pressure support ventilation and were connected to either a GE Healthcare Engstrom (Madison (WI), USA) or a Hamilton G5 ventilator (Bonaduz, Switzerland). The airway pressure, patient flow, volume and esophageal pressure are recorded over time. Start times of the inspirations and expirations are labeled by a single expert in the field, according to the method explained in [23].

This dataset is already fully labeled, effectively rendering the data selection and label projection obsolete. However, in order to evaluate the proposed methodology, the dataset is treated as if it were unlabeled. As such, the following procedure is applied for each individual patient:

1) Hide all labels of the dataset.
2) Segment time-series into individual breaths and perform data selection.
3) Recover only the labels of the most representative breaths and keep the remaining labels hidden (simulating a true annotation step).
4) Distribute annotated representative breaths over training (70%), validation (20%) and test (10%) datasets, while ensuring an even distribution of PVA-types per dataset.

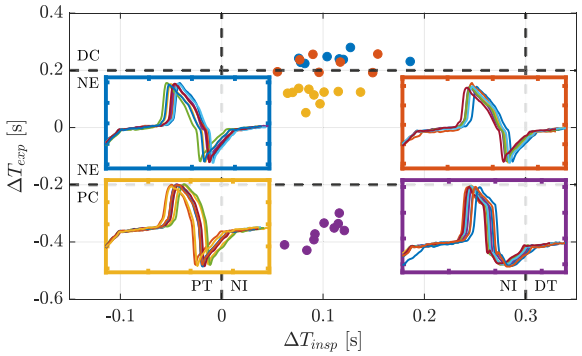

FIGURE 6. The respiratory delay plane divided by the PVA-type thresholds - - -, together with the corresponding PVA-types. In this plane, the breaths of four typical clusters of a single patient are displayed, which are: cluster 1 (——), 2 (——), 3 (——), and 4 (——). The inset figures display the time-series data (patient flow Q_{pat} against time) of the corresponding clusters.

5) Perform label projection to obtain pseudo-labels. Beware not to use the validation and test datasets in order to prevent information leakage via label projection.
6) Add pseudo-labeled data to the training dataset.

Then, for all patients simultaneously:

7) Use the training and validation dataset to train a neural network to predict the respiratory state $\hat{y}$.
8) Evaluate the performance based on the independent test dataset.

The remaining hidden labels are used, *a posteriori*, to assess the performance of the data selection (Section V-A) and the quality of the projected labels (Section V-B). Thereafter, the performance of the neural network is evaluated (Section V-C).

A. DATA SELECTION

The goal of the data selection is to obtain a reduced order subset of the complete dataset, such that the information lost by the reduction is limited. From each cluster, a single breath is attained and labeled. Therefore, the number of clusters (per patient) determines the reduction factor.

In consultation with the clinical experts and by careful investigation of the unlabeled data, 50 clusters per patient has been selected as a good balance between labeling effort and (expected) information loss. This amounts to 750 labeled breaths, making up 23% of the complete dataset, i.e. saving 77% of labeling effort.

For timing PVA-types, the PVA-type is defined by the inspiratory delay ($\Delta T_{\text{insp}} := T_{\text{vi}} - T_{\text{pi}}$) and expiratory delay ($\Delta T_{\text{exp}} := T_{\text{ve}} - T_{\text{pe}}$). In Fig. 6, four typical clusters with only timing PVA-types are plotted in the respiratory delay plane, making use of the aforementioned hidden labels. It shows that the data selection is not only able to cluster similarly looking breaths, but is also able to compactly cluster the breaths in terms of their location in the respiratory delay plane, i.e., in terms of their timing PVA-type. Here, we emphasize that the clustering was unaware of this underlying pattern, since it is performed merely on the time-series waveform data.

TABLE 1. Absolute number (and relative portion) of PVA-types within three datasets.

PVA-type	Reduced		Reduced & projected		Annotated	
NI	562	(29%)	2566	(31%)	2580	(31%)
PT	1	(<1%)	1	(<1%)	1	(<1%)
DT	187	(10%)	688	(8%)	673	(8%)
NE	238	(12%)	1321	(16%)	1290	(15%)
PC	95	(5%)	486	(6%)	490	(6%)
DC	417	(22%)	1448	(17%)	1474	(17%)
IEe	419	(22%)	1864	(22%)	1916	(23%)
IEi	7	(<1%)	8	(<1%)	12	(<1%)
DbT	0	(0%)	0	(0%)	0	(0%)
AT	0	(0%)	0	(0%)	1	(<1%)

Prior to DTW computation, each signal is normalized by L_2 normalization (vector-norm scaling) to reduce amplitude variability across patients and devices. Furthermore, to incorporate local shape information, each time sample was embedded in a short window of length five, resulting in a shape-based representation prior to alignment. Standard DTW is then applied without a warping window constraint. The local distance was computed using the Euclidean norm.

In general, the data selection is successful if the unlabeled breaths in a cluster are of the same PVA-type as the representative (i.e., annotated) breath in that cluster, because that is an indication that the cluster contains similar breaths. This is again analyzed by leveraging the hidden labels. Out of all breaths, 73% are in a cluster with identical *timing* PVA-type and 85% are in a cluster with identical *severe* PVA-type. This difference between timing and severe PVA-types can be explained by the definition of timing PVA itself. One can see in Fig. 6 that cluster 2 disregards the sudden threshold that defines discrete timing PVA-types, despite all breaths in that cluster looking similar. This underlines why our approach of learning respiratory timings is more nuanced than learning PVA-types directly.

The goal of the proposed data selection is to create a reduced order training dataset that maximizes diversity. Consequently, it must be able to capture uncommon PVA-types in the training dataset, as this enhances its diversity. Table 1 shows that is able to capture the uncommon PT and IEi PVA-types. However, it was not able to capture the uncommon AT PVA-type. In the remainder of this analysis, we do not consider the asynchrony types PT, DbT, and AT because their frequency of occurrence is too low.

B. DATA PROJECTION

For label projection to work well, the projected timings must accurately resemble the underlying annotated timings. Hence, the quality of the projected labels is assessed by the projection error of inspiration timing ($e_{\text{insp}} := \hat{T}_{\text{pi}} - T_{\text{pi}}$) and expiration timing ($e_{\text{exp}} := \hat{T}_{\text{pe}} - T_{\text{pe}}$), where $\hat{T}_{\text{pe}}$ and $\hat{T}_{\text{pe}}$ are the projected timings for the inspirations and expirations, respectively. The pairs $(\hat{T}_{\text{pi}}, T_{\text{pi}})$ and $(\hat{T}_{\text{pe}}, T_{\text{pe}})$ consist of a projected timing and the closest (in time) annotated timing. A projected timing is matched to an annotated timing if it occurs within 0.5 seconds of the latter, which also defines the maximum possible temporal error. Projected timings without a match are counted as false positives, and annotated timings without a match are counted as false negatives. Once matched, the exact temporal difference is computed, i.e., e_{exp} and e_{insp}.

Out of all projected timings, 93% are paired to annotated timings. Note that, this is an a posteriori analysis, therefore, the erroneous projections are used for training, which can result in reduced classification performance. Furthermore, in Table 1 it can be seen that the label projection accurately recovers the annotated PVA-type distribution from the altered PVA-type distribution of the reduced-order dataset. This indicates that, overall, the projected timings closely resemble the annotated timings.

C. SEMI-SUPERVISED LEARNING

In order to evaluate the performance of the proposed methods for model training, four methods are compared:

- *Random sampling + training* (rs+tr) trains a model on annotated data obtained by random data sampling. It does not use any of the proposed methods, but has an equal amount of annotated data samples per patient as the proposed methods.
- *Data selection+training* (ds+tr) trains a model on annotated data obtained by the proposed data selection step.
- *Data selection + label projection + training* (ds+lp+tr) trains a model on a combination of annotated and pseudo-labeled data obtained by the data selection and label projection steps.
- *Full labeling + training* (fl+tr) trains a model on annotated data obtained by fully labeling the dataset. It does not use any of the proposed methods and solely serves as a performance reference for other methods.

To ensure a fair comparison, identical validation and test sets are used for all methods. These sets are generated by randomly sampling from each asynchrony type to guarantee representation of all classes. The training set is constructed using the corresponding data selection strategy and thus varies across the four methods. Consequently, the optimal hyper parameter settings determined through grid search on the validation dataset are: a learning rate of 0.01, batch size of 64, gradient clipping at 0.01, hidden state size of 32, weight decay of 0.01, and a focal factor of 2.

The performance of a trained neural network is mainly affected by three sources of disturbance: the initialization of the trainable parameters, the processing order of samples during training and the particular distribution of samples between training/validation/test datasets. To account for these disturbances, we adopt the approach of [29]. Specifically, we train each method 15 times. Within a run, the disturbances are fixed and equal for all methods, whereas between runs, the disturbances are varied. This ensures that the average performance over all runs is expected to be unbiased in terms of the disturbances, which allows for meaningful comparison between individual tests.

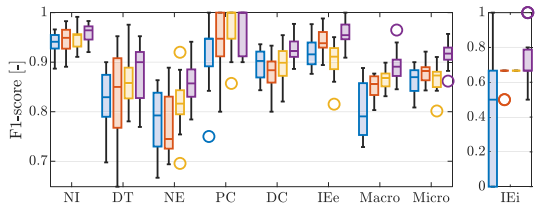

FIGURE 7. Box plot of the performance on the independent test dataset of rs+tr (——), ds+tr (——), ds+lp+tr (——) and fl+tr (——) for all different PVA-types. The F1-score is shown for all PVA-types separately, as well as the macro-average and micro-average, which is plotted alongside. The PVA-type IEi is displayed separately for visibility.

For this analysis, each predicted instance (i.e., PVA) must be paired to an annotated instance. Similar to Section V-V-B, the predicted instance is paired to the closest (in time) annotated instance. If no annotated instance exists within 0.5 s the predicted instance is paired with an instance of a null-class, which indicates no PVA got detected. Any unpaired annotated instances are then paired with a null-class instance as well.

As a performance measure, the F1-score F_c is computed for a class $c \in \{\text{NI, DT, NE, PC, DC, IEe, IEi}\}$ in the test dataset, as

$$F_c = \frac{2p_c r_c}{p_c + r_c} \in [0, 1], \tag{7}$$

where p_c and r_c are the precision and recall for class c, respectively. In addition, the macro-averaged and micro-averaged F1-scores, F_{macro} and F_{micro} respectively, are evaluated in order to provide insight into the overall performance [29]. Macro-averaging gives equal weight to each class regardless of their size, whereas micro-averaging gives equal weight to each sample.

In Fig. 7, it can be seen that fl+tr outperforms all other methods in terms of median performance for all PVA-types. This is due to the model being trained on the fully labeled dataset, which contains more diverse data and/or more accurate labels. Both ds+tr and ds+lp+tr outperform rs+tr for PVA-types NI and DT, whereas this is not the case for NE and DC, which are hard to differentiate in general. A great performance increase of ds+tr and ds+lp+tr with respect to rs+tr is noted for the rarest PVA-types in the dataset: PC and IEi. This is an indication that using the proposed methods, as opposed to not using those methods, allows for better capturing of uncommon classes in the training set.

This hypothesis is further substantiated by performing Wilcoxon signed-rank tests for F_{macro} and F_{micro} [29]. This is a non-parametric test for the null hypothesis that the median of the differences in matched pairs is equal to 0, where probability values smaller than 0.05 are considered statistically significant. Fig. 7 and Table 2 show that the ds+tr significantly increases performance with respect to rs+tr, for uncommon classes as well as for samples overall, as indicated by both a significantly higher F_{macro} and F_{micro}, respectively.

Furthermore, implementing ds+lp+tr significantly increases performance for uncommon classes with respect to both rs+tr and ds+tr, while not significantly changing performance overall. This is indicated by significantly higher F_{macro} and no significantly different F_{micro}, respectively. The lower F_{micro} of ds+lp+tr is mainly caused by the limited performance on the large class of IEe, which has been attributed to inaccurate projections for that particular class.

Thus, our methods are able to moderate class-imbalance issues by significantly increasing performance of rare classes while gaining (or without losing) overall performance with respect to not using the proposed methods. Also, we emphasize that the models trained using the proposed methods thus approach the predictability of a model trained on fully labeled datasets, while reducing the labeling effort from 100% to 23%.

TABLE 2. Probability values obtained by the wilcoxon signed-rank test. Values that pass the significance threshold of 0.05 are marked in boldface.

Experiments		F_{macro}	F_{micro}
rs+tr	ds+tr	**0.0181**	**0.0302**
rs+tr	ds+lp+tr	**0.0034**	0.6387
rs+tr	fl+tr	**6.10e-5**	**6.10e-5**
ds+tr	ds+lp+tr	**0.0125**	0.2524
ds+tr	fl+tr	**6.10e-5**	**1.83e-4**
ds+lp+tr	fl+tr	**0.0054**	**1.83e-4**

VI. RESULTS: DATA QUALITY ANALYSIS [MAASSTAD DATASET]

In this section, the data collection (Step 1 in Fig. 1) and annotation (Step 3 in Fig. 1) of the semi-supervised framework are discussed in more detail for a new clinical study at Maasstad hospital, the Netherlands. First, in Section VI-A, the data collection procedure for the Maasstad Hospital use case is presented, and the data selection method is applied. Thereafter, in Section VI-B, different perspectives on the annotation process are presented, where the annotation differences between clinical experts and researchers are highlighted. In Section VI-C, results of an inter-expert analysis are presented to investigate the consistency of the labeling.

A. DATA COLLECTION AND SELECTION FOR THE MAASSTAD USE CASE

A total of 15 patients were recorded for 90 minutes each, resulting in a large unlabeled dataset. This was an observational, anonymous, ventilator waveform sampling study. The study was reviewed and approved as a non-WMO study by the local Ethics Committee (Maasstad Hospital and Medical Research Ethics Committees United (MEC-U)). Patients connected to a Getinge Servo U (Solna, Sweden) were subjected to different levels of pressure support ventilation. The raw (unfiltered) airway pressure, patient flow, volume, esophageal pressure, and ventilatory timings were recorded with a sample frequency of 100 Hz. Subsequently, the raw dataset was segmented into over 26000 single respiratory cycles and down-sampled to 50 Hz.

In consultation with the clinical experts, it was decided to select 300 breaths per patient, summing to 4500 breaths being presented to the clinical experts. This results in a relative labeling effort of roughly 17%. A total of 5 clinical experts annotated between 2500 and 4500 individual respiratory cycles, and two researchers annotated 2500 breaths as well. All respiratory cycles are annotated multiple times to analyse the annotation consistency between experts. For the clinical experts, every cycle is annotated thrice, and for the researchers, every breath is annotated twice.

The quality of measured data is important for annotation consistency and quality. The stored data for annotation (the airway pressure p_{aw}, patient flow Q_{pat}, and the esophageal pressure p_{es}) contains only raw signals, meaning noise and other artifacts are not filtered from the signals. Furthermore, proper placement of the esophageal catheter is essential for accurate pressure measurement. After nasal insertion and careful positioning, confirmation can be achieved through curve analysis and the Baydur occlusion test during an expiratory no-flow period. Ideally, the $\Delta p_{\text{es}}/\Delta p_{\text{aw}}$ ratio should be close to 1, but a margin of 10-20% is generally accepted, resulting in an acceptable range of 0.8 to 1.2 [22].

Several factors can complicate p_{es} curve interpretation. Dominant cardiac oscillations can obscure the baseline, while active use of expiratory muscles can elevate the expiratory limb above baseline, complicating assessment. Additionally, patient position and esophageal spasms can significantly alter signal quality, occasionally resulting in false positive pressures exceeding 100 cmH_2O, which makes accurate analysis challenging. In spontaneously breathing patients, catheter migration over time can further impact the reliability of these measurements. It should also be noted that proper training and sufficient clinical exposure are essential for building the expertise needed to accurately interpret p_{es} curves.

In the upcoming section, the different perspectives on annotation are highlighted and, in the section thereafter, the inter-expert consistency analysis is conducted.

B. DATA ANNOTATION PERSPECTIVES FOR THE MAASSTAD USE CASE

In this section, we highlight the differences in perspective on the annotation rules for the clinical experts and the research team. It is important to note that both the clinical expert team and the engineering research team adhered to the PVA annotation rules as defined in Section III-C. However, differences in their annotations arose because the rules allow for some degree of interpretation.

The *clinical expert team* has extensive experience and specialized expertise in interpreting p_{es} signals. This waveform was used as the primary reference for determining the onset and termination of the respiratory cycles of the patient. In cases where the p_{es} signal was ambiguous or less clearly defined, clinicians referred to the pressure and flow curves as secondary sources to refine the timing assessments. Besides that, the clinical team is used to analyzing more heavily filtered signals typically displayed on ventilator screens during bedside monitoring. The use of raw, less-filtered signals may have influenced the interpretation of key respiratory events, potentially affecting the precision of timing assessments. Finally, bedside clinicians also rely on direct observation of the patient's respiratory effort, including visual inspection of the breathing pattern and, in some cases, tactile feedback from the contraction of expiratory muscles, providing a more comprehensive and contextually informed approach than only inspecting timings using ventilatory signals.

The *engineering research team* became familiar with patient-ventilator asynchrony through the literature by inspecting mainly airway pressure and patient flow curves. These two signals are informative for classifying different PVA types [30]; however, the esophageal pressure gives additional information, which should make the annotation process more convenient. Although the researchers did not receive proper training in interpreting the p_{es} signal, it was still useful information to annotate the data, as the definitions of patient inspiration and expiration start, see Section III-C. The researchers always focused on all three signals together during annotation. Hereby, the researchers are less subject to noise in one of the three signals, e.g., a disturbance in the p_{es} signal looks like a patient breath, but because the other signals do not change, it is noise and not a patient breath.

C. LABEL QUALITY FOR THE MAASSTAD USE CASE

The accuracy and consistency of these annotations were influenced by the substantial volume of data reviewed, with each clinician evaluating between 2,500 and 5,000 individual respiratory cycles. This volume provides a robust foundation for clinical interpretation, but also introduces variability associated with human assessment.

The inter-expert consistency analysis is conducted on the patient's inspiration and expiration timing level and the asynchrony level. On the timing level, we compute the absolute timing difference between the clinical experts for every individual respiratory cycle. Additionally, we compute the absolute timing differences between the researchers. The results are shown in Fig. 8. In this paper, we define agreement between experts as the average absolute timing error being smaller than 0.05 seconds. In Fig. 8, it is shown that, for the clinical team, agreement on inspiration timings is reached for 60.6% (1171/1931) of the respiratory cycles while agreement on expiration timings is reached for 35.3% (681/1931) of the respiratory cycles. For the research team, the annotators agree on 76.9% (1463/1903) of the respiratory cycles for the inspiration timings and 63.3% (1204/1903) of the respiratory cycles for the expiration timings. Although the research team showed higher consistency, this does not guarantee accuracy. A significant portion of annotations lacks expert agreement, making them unsuitable for training an automatic detection network.

The cause for the annotation consistency difference between the clinical team and the research team might be related to the difference in the perspective taken during the annotation as described in Section VI-B. If the esophageal pressure

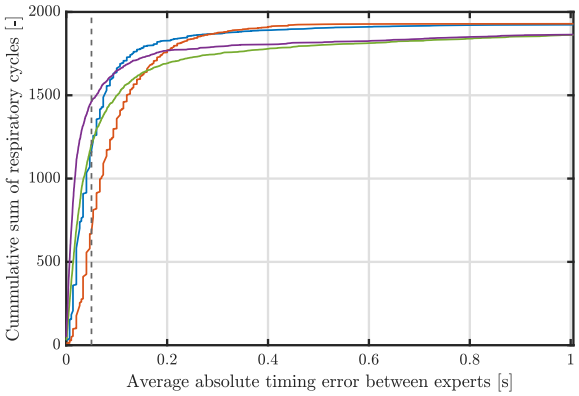

FIGURE 8. Comparison of the cumulative average absolute difference between the clinical team and the engineering researcher team in the timings. The absolute difference between inspiration and expiration timings for the clinical team are given by (——) and (——), respectively. The absolute difference between inspiration and expiration timings for the research team are depicted by (——) and (——), respectively. The vertical dotted line at 0.05 s is the user-defined maximum absolute difference between timings. The research team is more consistent than the clinical team.

TABLE 3. Absolute number and relative portion of PVA-types within two datasets. ∗double triggers are removed from the dataset

PVA type	Reduced		Reduced & projected	
NI	3662	(46%)	23414	(47%)
PT	77	(<1%)	220	(<1%)
DT	121	(2%)	667	(1%)
NE	2393	(30%)	16262	(33%)
PC	185	(2%)	916	(2%)
DC	1282	(16%)	7123	(14%)
IEe	88	(1%)	130	(<1%)
IEi	10	(<1%)	13	(<1%)
DbT	0*	(0*%)	0*	(0*%)
AT	166	(2%)	903	(2%)

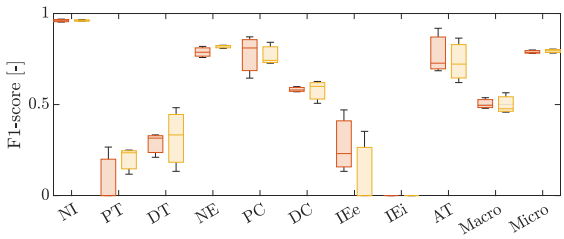

FIGURE 9. Box plot of the performance on the independent test dataset of ds+tr (——) and ds+lp+tr (——). The F1-score is shown for all PVA-types separately, as well as the macro-average and micro-average.

contains a lot of noise or other artifacts, it is expected that the annotation consistency decreases for the clinical team because they rely on this signal the most. In general, the research team shows better consistency, but the labels should be reviewed to make sure that they are not only consistent but also accurate.

The quality of annotated data must be further improved to develop an effective PVA detection algorithm. Based on the findings of this study, we strongly recommend the establishment of a standardized definition of patient expiration, developed and endorsed by a broad consensus of clinical experts. This should lead to the biggest improvement in annotation consistency across experts and institutions. Furthermore, the quality of annotation can be improved by improving the signal quality of measurements associated with the patient's breathing effort. In particular, esophageal measurements are affected by noise and artifacts, which complicate annotation. Improving the quality of these signals would further support accurate and consistent annotation practices.

In general, the combined labels from the clinical team should not be used to train an automatic detection network because there is too much disagreement between the experts. Using all expert labels would result in too much label noise and decreased detection performance. Therefore, we continue with the labels of a single clinical expert to apply it to, and demonstrate, the semi-supervised learning framework as introduced in Section II. This implies that the detection model automates the expertise of this single expert.

VII. RESULTS: SEMI-SUPERVISED LEARNING USING PROJECTED LABELS [MAASSTAD DATASET]

In this section, the performance of the PVA model trained with our semi-supervised learning framework is evaluated in a practical clinical setting, through close collaboration with the Maasstad Hospital (Rotterdam, the Netherlands). The model is trained on Maasstad data and tested on a separate test set, also from the Maasstad Hospital. In order to prevent the challenges in label consistency between experts, as discussed in Section VI, from interfering with the evaluation of our methods, only the labels of a single clinical expert are considered in the remainder of this section. Note that double triggers (DbT) are excluded from the dataset. Also, the experts were allowed to dismiss particular breaths if they were unable to confidently provide accurate annotation. Table 3 shows the distribution of PVA-types according to the selected clinical expert.

For this new dataset, the model is trained similarly to the model described in Section V. In this case, only the methods ds+tr and ds+lp+tr are compared by performing 3 runs each. Also, the hyper parameters from Section V-C are reused.

The performance of each training method for all runs can be seen in Fig. 9. For both methods, it shows that the high performance on these large classes is what pulls the micro-averaged F1-score up. However, the poor performance on the other, less common, classes keeps the macro-averaged F1-score down.

Although no convincing statistical conclusion can be drawn about the difference between the two methods based on merely 3 runs, it is evident that both models underperform to the previously trained models from Section V. Although they would likely benefit slightly from further tailoring of the training process to this particular dataset, we argue that the problem originates from the data itself.

By visual inspection of the data, it is clear that the Maasstad dataset contains noticeably more irregularities than the San

Matteo dataset. Therefore, the new dataset may be substantially more difficult to annotate accurately. This would introduce noise to the labels, which can severely reduce learning performance. Discussion with the clinical experts reveals that it can be particularly difficult to accurately annotate breaths that contain asynchronies. This is in line with the results presented in this section. Moreover, the potential challenges during annotation and the resulting inter-expert inconsistency, as elaborated in Section VI, strengthen the conclusion that further investigation into the intra-expert consistency of this dataset is required.

VIII. CONCLUSION

In this paper, a semi-supervised learning framework for time series detection is presented that significantly reduces annotation effort for experts while maintaining performance. This framework consists of five steps structural steps: data collection, selection, annotation, projection, and model training. For the data selection and projection, Agglomerative hierarchical clustering (AHC) and dynamic time-warping (DTW) are used to select a diverse subset from a large unlabeled dataset. Subsequently, this subset is annotated by experts and projected on the remaining set. As a result, a detection model is trained using a fully labeled dataset with only a small fraction of the annotation effort.

The framework is validated through patient-ventilator asynchrony detection in mechanical ventilation. Using a fully labeled clinical dataset, it demonstrates a reduction in labeling effort by over a factor of four while only slightly reducing classification accuracy. Additionally, the framework is applied to another clinical data set, which is unlabeled. After data selection, the clinical experts annotated a small but diverse subset. From an inter-expert consistency analysis, it was concluded that the agreement on timing annotation between clinicians is low. As an experiment, labels from a single expert were used to train a detection model, which still resulted in moderate detection performance. These results show that the primary bottleneck in time-series-based detection is not necessarily model capacity, but efficient and consistent labeling.

Beyond patient-ventilator asynchrony detection, the proposed methodology is broadly applicable to other time-series problems where annotation is costly, time-consuming, or requires domain expertise. The framework is for example suitable for biomedical signal analysis, industrial monitoring, wearable sensing, and anomaly detection tasks involving large volumes of unlabeled time-series data. For scenarios involving multiple relevant signals, univariate DTW can be extended to multivariate DTW. In cases where time-dependency is critical, similarity assessments should incorporate time-warping magnitude using advanced DTW techniques [18], [31], [32]. Future work regarding PVA detection should improve annotation consistency and define patient expiration timing more clearly. In general, the framework can benefit from exploring self-training and uncertainty-aware networks may enhance robustness and reduce inconsistencies in training data. This is crucial as label projection can increase the label noise in the training set. Furthermore, a quantitative comparison with methods such as active learning would help assess the relative efficiency gain of the proposed framework in reducing annotation effort. Lastly, it is interesting to investigate transforming the raw time-series signals in a more discriminative feature space prior to clustering to reduce variability within clusters and thereby increase the achievable data reduction.

ACKNOWLEDGMENT

The authors wish to thank the Fondazione I.R.C.C.S. Policlinico San Matteo (reference number 41223) for giving access to the dataset. Furthermore, authors also thanks the clinical experts from the Maasstad Hospital for their annotation efforts. The data from the Maasstad hospital is considered to be part of a non-WMO study (reference number W23.177) and transferred according a data transfer agreement.

REFERENCES

[1] Z. Li, B. B. Gupta, X. Chen, and X. I. N. Wang, "A survey of deep active learning," *ACM Comput. Surv.*, vol. 54, no. 9, pp. 1–40, 2021.
[2] B. Mirzasoleiman, J. Bilmes, and J. Leskovec, "Coresets for data-efficient training of machine learning models," in *Proc. 37th Int. Conf. Mach. Learn.*, 2020, pp. 6950–6960.
[3] C. Coleman et al., "Selection via PROXY: Efficient data selection for deep learning," in *Proc. Int. Conf. Learn. Representations*, 2020.
[4] J. E. van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Mach. Learn.*, vol. 109, no. 2, pp. 373–440, 2020.
[5] D. Berthelot, N. Carlini, I. Goodfellow, A. Oliver, N. Papernot, and C. Raffel, "Mixmatch: A holistic approach to semi-supervised learning," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, 2019, pp. 5049–5059.
[6] A. Rasmus, H. Valpola, M. Honkala, M. Berglund, and T. Raiko, "Semi-supervised learning with ladder networks," in *Proc. 29th Int. Conf. Neural Inf. Process. Syst.*, 2015, pp. 3546–3554.
[7] A. Sabbah, "Semi-supervised learning for the detection of patient- ventilator asynchrony," Master thesis, Eindhoven Univ. Technol., 2024.
[8] M. A. Warner and B. Patel, "Mechanical Ventilation," in *Benumof and Hagberg's Airway Management*, 3rd ed. Amsterdam, The Netherlands: Elsevier, 2013, pp. 981–997.
[9] A. W. Thille, P. Rodriguez, B. Cabello, F. Lellouche, and L. Brochard, "Patient-ventilator asynchrony during assisted mechanical ventilation," *Intensive Care Med.*, vol. 32, no. 10, pp. 1515–1522, 2006.
[10] L. Vignaux et al., "Patient-ventilator asynchrony during non-invasive ventilation for acute respiratory failure: A multicenter study," *Intensive Care Med.*, vol. 35, no. 5, pp. 840–846, 2009.
[11] L. Blanch et al., "Asynchronies during mechanical ventilation are associated with mortality," *Intensive Care Med.*, vol. 41, no. 4, pp. 633–641, 2015.
[12] S. K. Epstein, "How often does patient-ventilator asynchrony occur and what are the consequences?," *Respir. Care*, vol. 56, no. 1, pp. 25–35, 2011.
[13] T. Pham, L. J. Brochard, and A. S. Slutsky, "Mechanical Ventilation: State of the Art," *Mayo Clinic Proc.*, vol. 92, no. 9, pp. 1382–1400, 2017.
[14] D. Colombo et al., "Efficacy of ventilator waveforms observation in detecting patient-ventilator asynchrony," *Crit. Care Med.*, vol. 39, no. 11, pp. 2452–2457, 2011.
[15] T. Bakkes et al., "Automated detection and classification of patient–ventilator asynchrony by means of machine learning and simulated data," *Comput. Methods Programs Biomed.*, vol. 230, 2023, Art. no. 107333.
[16] L. van de Kamp, J. Reinders, B. Hunnekens, T. Oomen, and N. van de Wouw, "Automatic patient-ventilator asynchrony detection framework using objective asynchrony definitions," *IFAC J. Syst. Control*, vol. 27, 2024, Art. no. 100236.
[17] X. Ran, Y. Xi, Y. Lu, X. Wang, and Z. Lu, "Comprehensive survey on hierarchical clustering algorithms and the recent developments," *Artif. Intell. Rev.*, vol. 56, no. 8, pp. 8219–8264, 2023.

[18] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-26, no. 1, pp. 43–49, Feb. 1978.
[19] L. van de Kamp, B. Hunnekens, T. Oomen, and N. van de Wouw, "Time-series out-of-distribution data detection in mechanical ventilation," *IEEE Open J. Control Syst.*, vol. 4, pp. 236–249, 2025.
[20] J. J. Wisse et al., "Clinical implementation of advanced respiratory monitoring with esophageal pressure and electrical impedance tomography: Results from an international survey and focus group discussion," *Intensive Care Med. Exp.*, vol. 12, no. 1, 2024, Art. no. 93.
[21] L. Ball, D. Talmor, and P. Pelosi, "Transpulmonary pressure monitoring in critically ill patients: Pros and cons," *Crit. Care*, vol. 28, 2024, Art. no. 177.
[22] A. H. Jonkman, I. Telias, E. Spinelli, E. Akoumianaki, and L. Piquilloud, "The oesophageal balloon for respiratory monitoring in ventilated patients: Updated clinical review and practical aspects," *Eur. Respir. Rev.*, vol. 32, 2023, Art. no. 220186.
[23] F. Mojoli et al., "Timing of inspiratory muscle activity detected from airway pressure and flow during pressure support ventilation: The waveform method," *Crit. Care*, vol. 26, no. 32, 2022, Art. no. 32.
[24] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
[25] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Proc. 30th Int. Conf. Mach. Learn.*, 2013, pp. 1310–1318.
[26] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM networks," in *Proc. Int. Joint Conf. Neural Netw.*, 2005, vol. 4, pp. 2047–2052.
[27] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.
[28] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. 7th Int. Conf. Learn. Representations*, 2019.
[29] O. Rainio, J. Teuho, and R. Klén, "Evaluation metrics and statistical tests for machine learning," *Sci. Rep.*, vol. 14, no. 1, pp. 1–14, 2024.
[30] F. Mojoli et al., "Waveforms-guided cycling-off during pressure support ventilation improves both inspiratory and expiratory patient-ventilator synchronisation," *Anaesth. Crit. Care Pain Med.*, vol. 41, 2022, Art. no. 101153.
[31] Y. S. Jeong, M. K. Jeong, and O. A. Omitaomu, "Weighted dynamic time warping for time series classification," *Pattern Recognit.*, vol. 44, no. 9, pp. 2231–2240, 2011.
[32] M. Herrmann and G. I. Webb, "Amercing: An intuitive and effective constraint for dynamic time warping," *Pattern Recognit.*, vol. 137, 2023, Art. no. 109333.