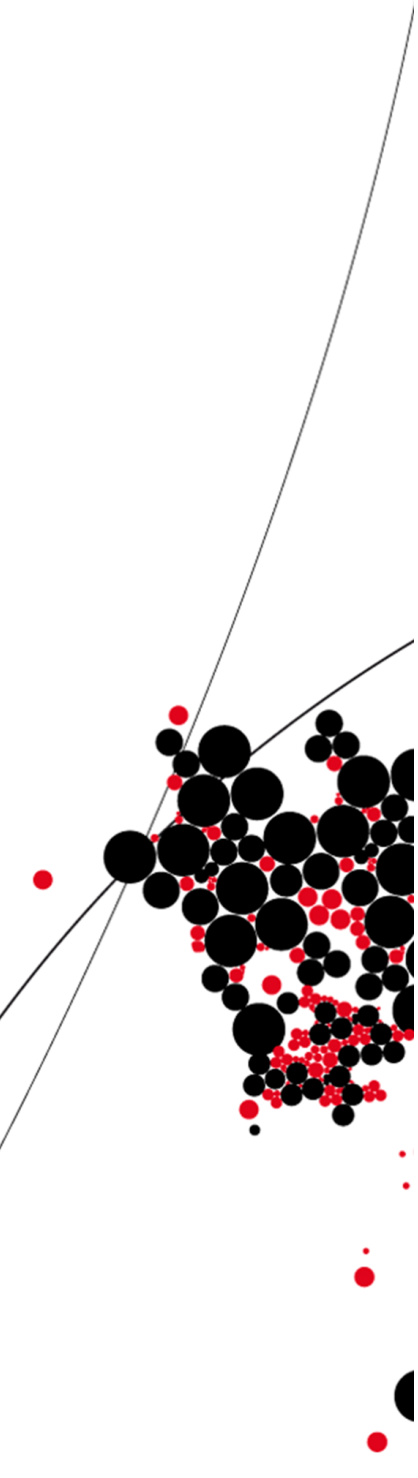
UNIVERSITY OF TWENTE.

**Faculty of Electrical Engineering,
Mathematics & Computer Science**

Automatic Clinical Deterioration Monitoring using Machine Learning Techniques Post Surgery

Pralad Prasad

M.Sc. Thesis Report
August 2022



Commitee:
Chair: prof.dr.ir. H.J. Hermens (BSS)
Daily Supervisor: dr. Y. Wang (BSS)
External Supervisor 1: dr. A. Betken (MOR)
External Supervisor 2: dr. ing. K.H. Chen (CAES)

Biomedical Signals and Systems Group
Faculty of Electrical Engineering,
Mathematics and Computer Science
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

Abstract

The detection time of clinical deterioration (clinical adverse events) has long been viewed as a key factor in indicating the response rate of necessary interventions. However, this detection time is impacted by the frequency of nurse assessment and the number of patients in a given ward. To improve assessment time, nurses employ threshold-based track-and-trigger systems to monitor adverse events, but they suffer from high false alarm rates given the heterogeneity of population. It is desired to improve the performance of clinical deterioration monitoring systems. This study aimed to develop a fully automatic machine learning based system to detect clinical adverse events in 60 postoperative patients with short-termed (two minutes per hour) vital sign data from wearable sensors. Our system focused on extracting and highlighting the most important features from the vital sign data, and using these features, perform a comprehensive test of decision support models from the machine learning sphere. This includes models from the classical statistical machine learning, deep learning and time-series classification domains. Finally, the top three model's performances were compared to existing threshold-based systems. Overall, the best decision support model in the system exhibits a significant boost in performance compared to existing threshold-based systems. It could detect all clinical adverse events ahead of time with an accuracy of 86% and a precision of 42%. The system model also reported an average false positive rate of 15%, almost 67% lesser than existing threshold-based systems.

Keywords

Clinical deterioration, machine learning, deep learning, vital signs, wearable sensors, clinical adverse event, tele-monitoring, eHealth.

Introduction

There are concerns with the growing incidence of complications as the number of patients undergoing surgery every year increases [1]. A complication, or more specifically, a clinical adverse event (CAE) can be defined as a high-risk adverse event that requires immediate medical intervention. Studies have shown the risk of a CAE to be the greatest in the perioperative and postoperative phases of a patient's hospital stay [2]–[4]. Nurses play a major part [5], [6] in a patient's postoperative trajectory, and one of their most important roles involves patient status assessment and monitoring. However, more patients can negatively impact the amount of time nurses spend of patient assessment [7], which could have more serious implications [8]. The most frequent non-operative management errors involve aperiodic vital-sign monitoring and delayed treatment. Hence, there was a requirement for better systems or procedures to address these concerns.

To reduce patient assessment time, track-and-trigger systems were accepted and adopted to assist clinicians in clinical deterioration monitoring. A track-and-trigger system is a monitoring chart that is generally applied to physiological signals that is used to indicate patients of deteriorating conditions. The most adopted track and trigger system used is that of the Early Warning Score (EWS). The EWS works on the principle that the deterioration state of a patient can be scored based on pre-defined thresholds that apply to the primary vital signs – body temperature, heart rate, respiration rate and blood pressure. The Modified Early Warning Score (MEWS), a variation of the EWS, is the most validated system for CAEs and is regarded as the gold standard of CAE monitoring systems. However, its threshold-based nature cannot capture the complexities of the physiological states of patients across multiple cohorts and demographics, and suffer from high false positive rates [9]–[11]. Hence, for clinical deterioration, approaches that do not solely rely on thresholds needed to be explored.

The primary goal for systems looking to improve upon the drawbacks of MEWS is to be able to effectively model the physiological states of patients. But which technique is best suited for CAE detection using just vital signs? Multiple data mining techniques from the machine learning domain have been shown to model physiological states of patients from multiple cohorts using vital sign data. A previous study [12] showed a kernel density estimate model exhibited the best performance in detecting CAEs in its experiments, while another study [13] suggested that a 1-class support vector machine has the best performance for CAE detection. While this study [14] is the most thorough paper which compared multiple machine learning techniques with MEWS, it does so on electronic health record data and not on continuous wearable sensor data, and also does not expand into the deeper, more complex domains of machine learning. Hence, there is still no agreement to the single best algorithm to use for CAE detection, and whether the algorithm actually improves upon the drawbacks of the MEWS.

This study answered these questions by performing an extensive test of 33 models from the machine learning sphere and comparing their performances to that of the MEWS. We proposed a fully automatic system capable of extracting features from vital signs and applying a decision support model to observe its overall performance in comparison to the golden standard of clinical deterioration monitoring systems. The most important indication of an improvement in performance of the final system versus the MEWS is whether the final system had a better false positive rate than that reported by the MEWS. Based on these comparisons, we ensure the final system is able to truly capture the physiological states of patients.

Materials and Methods

Dataset

This retrospective study was conducted on the MoViSign dataset, which was compiled in Almelo, the Netherlands. The data collection was done in MoViSign project funded by Pioneers In Health Care Innovatiefonds, the University of Twente, the Netherlands. The dataset consisted of data from wearable sensors during the ward stay of 60 patients of two surgical-based groups: upper gastrointestinal (GI) tract surgery (n=33) and hip fracture surgery (n=27). In this dataset, 28 patients encountered CAEs comprising 7 pulmonary complications, 2 anastomotic leakages and 12 cardiac complications. The average ward stay of the patients was 5.42 ± 0.25 days. The male-to-female ratio was 7:8 in this dataset.

Wearable Sensors

The wearable sensors used in this study were the Isansys LifeTouch, the Isansys LifeTemp and the Nonin 3150 WristOx2 pulse oximeter. The sensors provided two types of data — the vital sign trace signals and the sensor pre-computed vital signs. The vital sign trace signals are raw physiological signals that allow us to extract additional (and possibly informative) features apart from the standard vital signs, and were chosen as the primary data inputs. The pre-computed vital signs are the heart rate, respiration rate, body temperature and oxygen saturation. The vital sign trace signals are recorded at 99 Hz and the pre-computed vital signs are recorded per minute. Although the trace signals are recorded at a faster rate, they have a maximum per-hour recording time of 2 minutes due to sensor limitations. The target vital sign signals are the electrocardiogram (ECG) and photoplethysmography (PPG) signals, along with the body temperature and oxygen saturation data from the wearable sensors.

Additional Variables

Data based on demographic information like age, height and weight and previous comorbidity history was collected during patient admission. This data had a primary function to be used for model personalization.

Outcome

The CAEs, marked by the onset of therapeutic actions initiated and confirmed by an expert clinician, were the golden standard in this study.

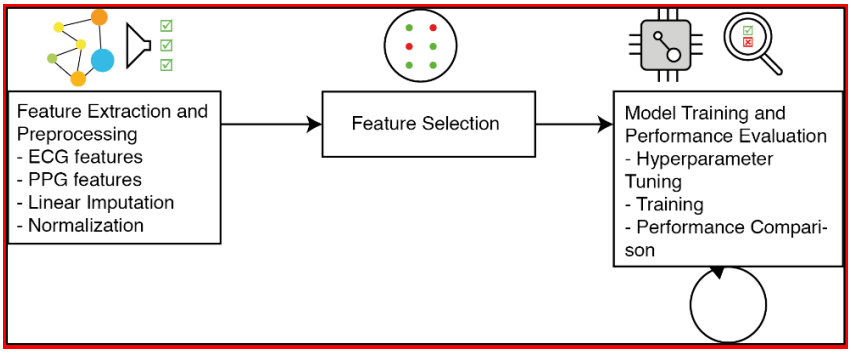


Fig. 2: Flowchart of the various steps in the analysis pipeline

a) Feature Extraction and Preparation

The study design also sets the foundation for setting up the analysis pipeline. Figure 2 provides an overview of the different sections of the analysis pipeline. Initially, we extracted multiple timespan features, with a maximum span of 2 minutes as mentioned earlier, as it was the maximum per-hour recording duration of the sensors and to explore how informative vital sign waves of multiple time periods are to predicting CAEs. The features were grouped into six-second, minute, and patient-specific features. The second-based features were comprised of features extracted from 6-second windows with a 50% overlap, adding up to a theoretical maximum of 39 windows per hour. These included the heart rate, the respiratory rate, the quality of R-peaks, heart rate variability indices extracted from the ECG signal and the morphological features [15] extracted from the PPG signal. The minute-based features included the body temperature and oxygen saturation data. The patient-specific features were quantitative and qualitative variables that include the patient demographic and previous comorbidity data collected during admission. The entire list of features extracted in this study is presented in Appendix A.

A custom automatic extraction process (Fig. 3) was developed for this study to suppress motion artefacts and noise in raw ECG and PPG signals. The process consists of four steps: Firstly, we rejected segments where the ratio of tracepoints were more than one standard deviation from the mean in a 6-second span to eliminate the extremely noisy segments. Next, we needed to check the recognizability of arbitrary ECG waves. An approach based on using the properties of Q,R and S-waves of the ECG signal (or QRS) to measure the ECG quality was chosen due to its capability in distinguishing non-standard waves [16]. This is based on three indices multiplied with the weight vector [0.6, 0.2, 0.2] that was chosen to provide the best accuracy of determining ECG quality from tests conducted in [16]. Based on this final score (*OverScore*) in Eq. 1.4, we rejected segments having an *OverScore* greater than the decision value of 1.25, selected based on the evaluation rating in [16], and in turn the accuracy of ECG quality. As

mentioned previously, the three indices, the power spectrum signal quality index (*pSQI*), the kurtosis SQI (*kSQI*) and the baseline SQI (*baseSQI*), each characterize properties of QRS segments of the ECG signal. The *pSQI* is based on the fact that the QRS wave accumulates ~99% of the energy of an ECG signal. This QRS wave is centered at 10 Hz with a width of 10 Hz. Hence, *pSQI* is the ratio of the QRS energy (integral of the spectral power *P(f)* in the 5-15 Hz band) to the overall energy (integral of the spectral power *P(f)* in the 5-40 Hz band) and is described in Eq. 1.1. This ratio is ideally in the range of 0.5 to 0.8 based on experiments on normal heart rate ranges. The *kSQI* is based on the measure of kurtosis of the signal. The kurtosis is the measure of the number of outliers compared to the normal distribution. It is the fourth standardized moment (the expected value E of a random variable x shifted by its mean μ_x, divided by the signal's standard deviation σ whole raised to the fourth power) of the ECG signal distribution and is computed as per Eq. 1.2. Good signals have an *kSQI* in the range greater than 5, that is based on low skewness in the signal. The *baseSQI* is a measure of the baseline drift of the ECG signal. If there is a significant amount of very low-frequent energy (integral of the spectral power *P(f)* in the 0-1 Hz band), this could be attributed to an abnormal shift from the baseline. Hence, Eq. 1.3 describes the computation of the *baseSQI* where no baseline shift gives the *baseSQI* an ideal value of 1 and higher values of low-frequent energy is linked with low values of baseSQI.

$$pSQI = \frac{\int_{f=5Hz}^{f=15Hz} P(f)df}{\int_{f=5Hz}^{f=40Hz} P(f)df} \tag{1.1}$$

f – Frequency of QRS segment
$P(f)$ – Spectral power density at frequency value f

$$kSQI = \frac{E(x-\mu_x)^4}{\sigma^4} \tag{1.2}$$

x – Random variable of QRS distribution
μ_x – Mean of QRS distribution
$E(a)$ – Expected value of a
σ – Standard deviation of QRS distribution

$$baseSQI = \frac{1-\int_{f=0Hz}^{f=1Hz} P(f)df}{\int_{f=0Hz}^{f=40Hz} P(f)df} \tag{1.3}$$

f – Frequency of QRS segment
$P(f)$ – Spectral power density at frequency value f

$$\text{OverScore} = 0.6\text{pSQI} + 0.2\text{kSQI} + 0.2\text{baseSQI} \tag{1.4}$$

Thirdly, we extracted the R-peaks based on the steepness of the absolute gradient of the ECG signal functions [17]. The fourth step entailed rejecting peaks (rather than entire segments) whose peak-to-peak intervals fall outside the 99% range of intervals. The 99% range was chosen for completeness to reject isolated outlier peaks that might have remained after

removing noisy segments from the previous steps. The complete second and minute-features were then extracted using the library Neurokit2, which is a collection of pre-written ECG and PPG signal processing algorithms in Python [17].

The feature extraction procedure of the PPG signal was conducted according to the above procedure with slight differences. The step dealing with the properties of the QRS segments was not needed for PPG signals due to a lack of QRS segments. Also, instead of rejecting peaks based on their interval range, we instead used an Isolation Forest model [18] to delete outlying peaks as superior accuracy in PPG quality was reported. The final PPG features, were computed from the PPG wave, its first derivative (known as the velocity plethysmogram or VPG), and its second derivative (acceleration plethysmogram or APG). The general principle of the feature extraction implementation was based on the presence of characteristic peaks and troughs in the PPG waveform. Through these peaks and troughs, it was possible to mark the start and end of waves, and the corresponding peaks through a counter as every single (standard) PPG waveform has the same number of peaks. Hence through this technique, it was possible to mark all the interest points in a PPG wave and compute the PPG features. In this way, the complete list of features was available.

Data preparation is also necessary to ensure the models are fed with appropriate inputs. The data preparation consists of two steps. Firstly, from the feature extraction steps, discontinuity present from rejected noisy segments needed to be filled in, since the machine learning models are hard to train with missing values. For this purpose, linear imputation was used. Secondly, models such as the support vector machine and neural networks are sensitive to feature ranges. Feature scaling was done to ensure the techniques employed are invariant to the feature ranges and ensure certain features do not overly affect the model prediction by their magnitude. The feature scaling was applied to ensure standardization of overall feature spaces by first standardizing to remove the mean and scale to unit variance and then restricting the range to (0-1), and hence as a result, each feature is able to contribute to the final outcome equally.

b) Feature Selection

The feature extraction step generated 45 features with a possibility of redundant and "low-importance" features. Here, "low-importance" is used to describe features that have loose correlations with the outcome. Hence, the feature selection step was necessary to eliminate these unnecessary features. The feature selection was comprised of two steps — the first step deals with redundancy and the second step deals with prioritizing "high-importance" features.

The first step was performed by retaining the high variant features from sets of correlated features in an unsupervised manner [19]. The correlated feature sets were generated using pairwise permutations of features and then marked if their Pearson's correlation coefficient is 0.8 or higher, with 0.8 selected to select only highly correlated features. This step marked seven features as redundant, which left us with 38 features.

The second step was split into two approaches to reduce the method-specific bias. The first approach for measuring the "importance" of a feature was based on the change in accuracy after the feature is removed (sequential dropping of features) from an arbitrary estimator's input [31, p. 5], [32]. The second approach is more nuanced. The approach is based on the idea that the global "importance" of features can be approximated from the accumulated local "importance" of features [33] as an alternate method to the first approach. While the first approach focuses on how a model acts with and without certain features, the second approach deals with how the local values of each feature contributes to the outcome. These "importance" values are called Shapley values [34]. The average of the feature "importance" values from the

two approaches for a given feature was computed and features were sorted based on this average.

However, not all features can be selected in this fashion. The patient-specific features only vary per patient and hence had to be treated differently.

For the patient-specific features, instead of selecting features, we reduced the dimensionality of the final patient-specific features to reduce information loss. We argue that since these features have the primary function of incorporating model personalization without needing its interpretation, a dimensionality reduction technique would be adequate. The dimension reduction technique used was the Uniform Manifold Approximation and Projection (UMAP) [35], in which features are reduced to lower dimensions based on their topological structures. The reduced features from this step were then added to the final feature list.

c) Statistical Machine Learning

We selected 25 supervised models from 4 model classes (Fig. 6) grouped by their type of learning/function. The decision of choosing diverse models was motivated from the type of algorithms better suited for binary classification (CAE detection) with the final feature set. The performance results from the tests conducted on these models would establish the top decision support model (and model class) from the machine learning sphere. The grouped classes are linear, non-linear, probabilistic, and tree models with implementations provided by Scikit-learn [20], a universal Python repository containing machine learning model implementations. In addition, we also included models like LightGBM [21], XGBoost [22] and CatBoost [23] into the final model list. These models are based on sequential training decision trees on the negative gradients of the data instances with improved training times. Each model that has been tested falls either within or exhibits close resemblance with the mentioned model classes.

d) Deep Learning

We also decided to branch out to the deep learning sphere. While the previous models were easy to interpret, they lacked complexity and had questionable results in generalizing unseen data. Hence, using more complex models could push the possibility of identifying better structures from the feature set. The most familiar models within the deep learning sphere are deep neural networks. Deep neural networks (DNNs) are feed-forward neural networks that incorporate a greater number of layers to extract higher-level representations from the raw inputs and perform gradient-based optimization using backpropagation [40]. This ensured that all abstractions of the combinations of the feature set could be utilized for the final prediction. This study created a custom architecture using a set of application programming interfaces (APIs) defined in the Python deep-learning library, Pytorch, which is a framework used for the design of deep-learning architectures [24]. An illustration of the custom architecture used is presented in Appendix B. The final model properties or hyperparameters were decided through multiple runs of different model versions in a grid-search fashion for the best performance.

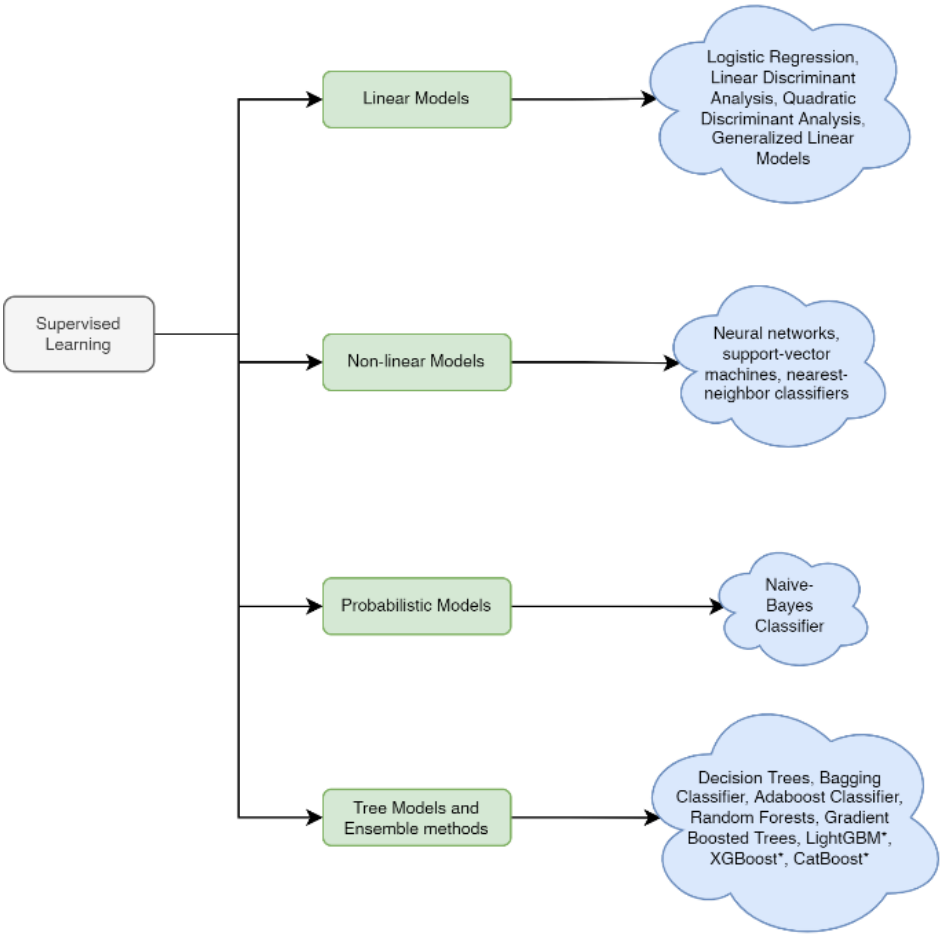


Fig. 6: Complete model list over all model classes for supervised learning classification tasks. The starred models indicate independent models that are not part of the Scikit-learn model list.

e) Time Series Classification

Finally, we decided to use deep learning to explore the temporal structures in the feature set (time series classification). The main purpose for pursuing this direction was to check whether the order of feature values add extra benefit to the CAE detection quality. This study only utilizes the base configuration of the recurrent neural network (RNN) [24] and its other variants, like the Long Short-Term Memory (LSTM) model [25] and the Gated Recurrent Unit (GRU) model [26]. These models are based on the principle that each cell in a recurrent neural network takes the previous time step's output as input in a sequence. One could think of such models as the layers of deep neural networks folded onto each other per time step. The abstractions of these sequences are fit recursively into multiple cells and then provided to a deep neural network as inputs. This was how we modified RNNs for classification purposes. This complete model was termed the LSTM-NN model and was designed using Pytorch as well. Since this is a high-dimensional time series classification problem, the deficiencies of performing backpropagation in the RNN would be compounded. Hence, the testing was only performed on configurations of the LSTM model. An illustration of the custom LSTM-NN architecture used is presented in Appendix B.

f) Model Evaluation

The evaluation of the chosen techniques is also pivotal in assessing the performance of our system model. The evaluation was comprised of two parts– the data split strategy and the evaluation policy.

The dataset was split into a train set of 50 patients and a test set of 10 patients with a CAE-to-non-CAE ratio of roughly 50%. However, the data splitting was done differently (Fig. 5) in the case of hyper-parameter tuning and final model comparison. The hyper-parameter tuning was performed with the help of a 10-patient validation set repeated thrice (general cross-validation strategy with 4:1:1 ratio), where the performance metrics of a particular model configuration were saved in each split. The average of these metrics is the final performance of that particular model configuration, which is repeated with a new configuration, and so on. The final model comparison was done differently than what is standard practice. For the final model comparison, we opted out of using a validation set and instead took three separate splits of the dataset. An untrained version of the model was taken per split and the performance metrics were saved in each split, repeating with untrained versions of the model per split. This was done to not escalate the class imbalance issue further and reduce the possibility of passing lesser CAE data instances to the models. The final performance values of the model were the average of the performance metrics over the three splits.

While many different evaluation policies exist for event detection, the evaluation policy (Fig. 4) used in this paper was based on the number of correctly classified "positive" and "negative" windows across patients. The policy states that positive windows generated by the model within 8-24 hours prior to a registered CAE count as correctly classified CAE instances, while positive windows generated in other time windows are false positives. The policy also states that negative windows generated by the model within 8-24 hours prior to a registered CAE count as false negatives. A "positive"/"negative" window is termed the hour window in which the model under evaluation indicates/does not indicate a CAE moment. Because the exact time at which the CAE registered by an expert clinician is based on the availability of the clinician, a soft margin of 7 hours was added to each CAE. This custom policy was developed to get a bigger picture into how sensitive a model is to a CAE and how selective it is to non-CAE periods.

g) MEWS

The MEWS is a scoring-based algorithm that is computed as an aggregated sum of weighted scores that have vital signs within particular thresholds. The MEWS was only used with continuously measured features so only three vital signs - heart rate, respiration rate and temperature contributed to the MEWS computation. Since the original paper suggests using a MEWS threshold of 4 for the best results [28], we followed the guidelines of the authors.

h) Performance Comparison

To demonstrate the utility of the final system, we performed a comparison of MEWS vs ML techniques. The performance of the top performing models from each ML sub-domain was compared with the MEWS performance as well. For simplification, the MEWS score has been scaled to make it into a familiar probability form. Hence, the maximum score is 1 and its threshold is 0.44. All other models retain a standard threshold of 0.5.

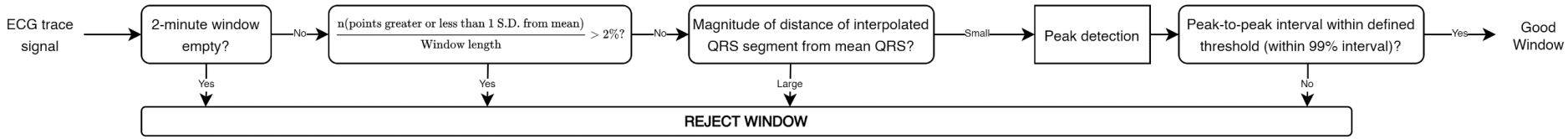


Fig. 3: Flowchart of the (general) feature extraction process for ECG signals. SD – standard deviation, the QRS segment refers to the portion of the ECG containing the Q, R and S waves.

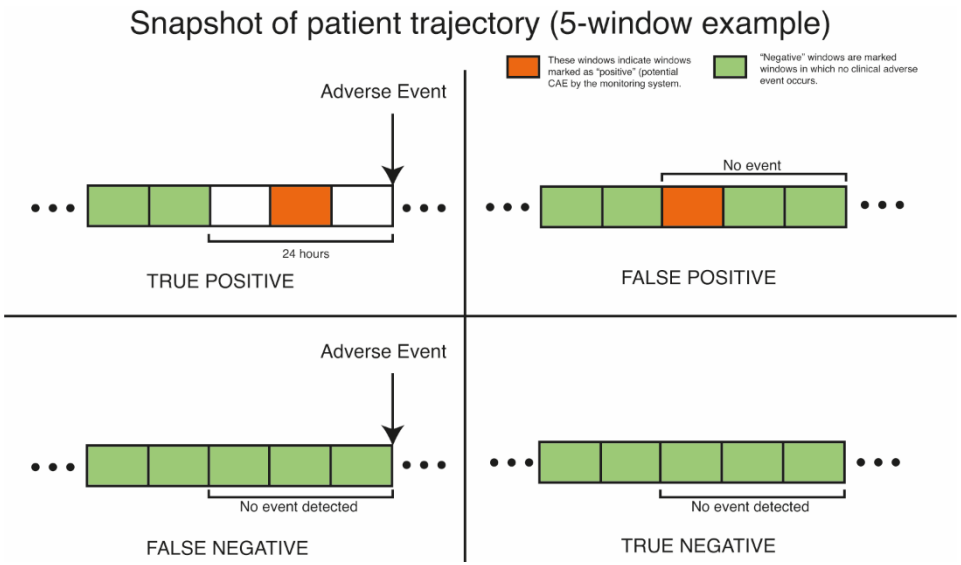


Fig. 4: Custom evaluation policy used in this study for CAE detection. Each block signifies a 8-hour period which contains the combined predictions from the model. Under the "strict" criterion, all predictions within a block must be negative to render a "negative" window, but even one positive prediction within a block can convert the entire block into a "positive" window.

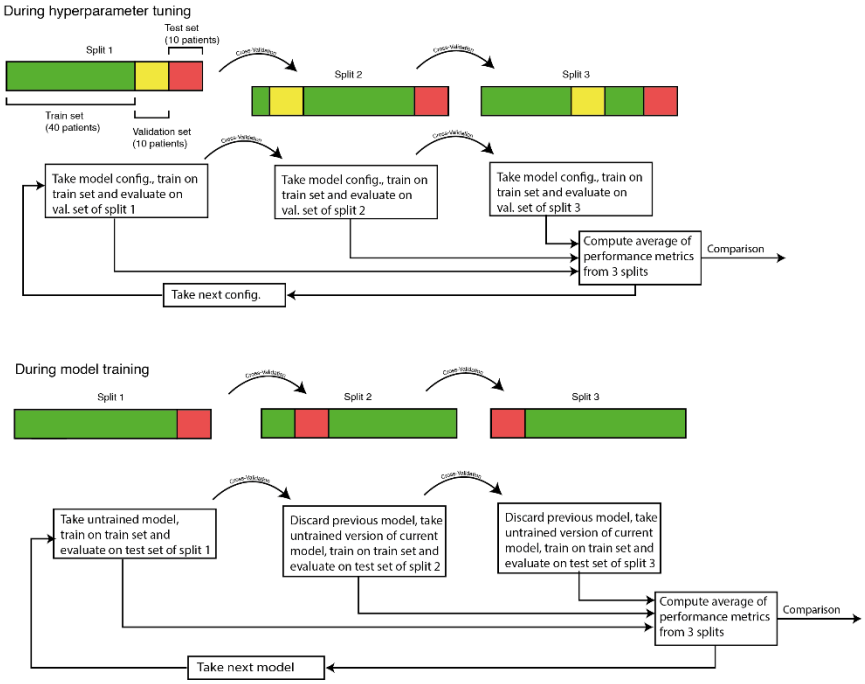


Fig. 5: Data splitting strategy used during hyperparameter tuning and model training.

Results

e) Feature Selection

From the feature selection tests, the relative importance of the top features is presented in Fig. 7. The top 10 features are as follows. The top ranked feature was the body temperature. The 2^{nd}-8^{th} most important features all originate from the velocity and acceleration photoplethysmogram (VPG+APG) waves. The 9^{th} most important feature is the high frequent heart rate or spectral power density of the heart rate in [0.15-0.4] Hz frequency range. Finally, the 10^{th} most important feature is the oxygen saturation. Since no improvement in accuracy over a decision tree (with all other conditions frozen) was observed beyond a relative importance of 25%, we set the selection threshold at this percentage. Hence, the ideal number of features that retain enough information to detect adverse events without risking overfitting was 15. For the patient specific features, UMAP reduced 17 patient features to just 3 features - able to explain 90% of total variance.

f) Statistical ML models

The results of the machine learning model tests are summarized in Table 1. From these tests, the best performing model was the Gaussian Naïve-Bayes classifier. Over the three train-test splits (i.e. training and evaluation runs of the model), the Gaussian Naïve-Bayes classifier detected 4 out of 6 cardiovascular CAEs and 7 out of 12 CAEs in total. The classifier detected 2 cases of atrial fibrillation, 2 (out of 3) cases of pneumonia, 1 case of urinary retention, 1 case of urinary tract infection, 1 case of wound blistering and 1 case of delirium.

Model	Accuracy	Precision	Sensitivity	Specificity	F1 Score
Gaussian Naïve-Bayes	**65.3%** ± 10%	**30%** ± 13%	**62.3%** ± 18%	**66.3%** ± 9%	**40%** ± 15%
Quadratic Discriminant Analysis Classifier	62.3% ± 10%	28.3% ± 12%	61% ± 10%	62.6% ± 10%	37.5% ± 13%
Decision Tree	60% ± 3%	20.3% ± 5%	60.1% ± 16%	60% ± 2%	30% ± 7%

Table 2: Comparison of test-set performance of top 3 ML models over 3 splits

g) Deep Learning models

The results of the deep learning model test are summarized in Figure 8. The best hyperparameters used to train the model over the three splits are shown in Appendix B. Over the three train-test splits, the deep neural network classifier was able to classify 11 out of 12 CAEs, and all cardiovascular CAEs. The only CAE it could not detect was that of wound blistering.

e) Time Series Classification models

The results of the LSTM-NN time-series model are summarized in Figure 8. The best hyperparameters used to train the model over the three splits are shown in Appendix B. Over the three train-test splits, the LSTM-NN model was able to classify 12 out of 12 CAEs. In

the case of the LSTM-NN model, there is a shift in resolution as the multivariate time series yield a single prediction for an arbitrary hour window. This resolution shift contrasts with the deep neural network model, where the resolution of a prediction is essentially 6-seconds. To maintain a similar resolution as the time-series model, we establish the strict criterion that states that even a single 6-second epoch in an hour that renders a positive CAE classification yields a positive CAE classification for that hour and vice versa.

e) MEWS

The results of MEWS was also compared to the other models in Figure 8. With the MEWS, we were able to classify 11 out of 12 CAEs, and all cardiovascular CAEs. The only CAE that was not detectable by MEWS was a myocardial ischemia CAE. However, despite the impressive sensitivity, the results of the MEWS score came with a downfall. It reported classifications with a false positive rate of 46.6%. That means approximately half of non-CAE hour windows were incorrectly classified as CAEs.

f) Performance Comparison

In terms of the AI models, we observed that as the complexity of model increases (MEWS < Gaussian NB < DNN < LSTM-NN), then so does the overall performance of the model. The functioning of the models could be demonstrated for two patients against MEWS with the help of 2 random patients, A and B from the test set.

Discussion

This study has developed a robust system capable of accurately predicting CAEs. The analysis pipeline extracted and synchronized raw short-term vital sign features from wearable sensor data in a deterministic fashion robust to motion artefacts. We then tested several candidate models that utilize these features and compared their performances. Although each candidate model scaled up and improved performance compared to existing threshold-based systems, the LSTM-NN decision support model stood out as the best support model. It improved the false positive rate that existing threshold-based track-and-trigger systems suffer from while being used as decision support models.

We also shortlisted the most important research insights from our tests.

Firstly, a 6 second window span is ideal for retaining granular data from our extracted features. Selecting 6 seconds stems from picking a resolution that would not smooth the high frequent variance present in the features. The overlap also provides a smooth window-to-window progression of feature values. This window length choice is not universal and is different for specialized features like respiratory rate and the low-frequency power of the heart rate (two 6-second windows).

Secondly, linear imputation excels over other imputation techniques when dealing with missing values. The reason linear imputation was confirmed was through a small test. Two copies of the feature table were filled using linear and k-nearest neighbor imputation and fitted against a decision tree model. With no other changes in either the feature table or the model parameters, the model fitted on the linear-imputed feature table had better accuracy, hence why linear interpolation was chosen as the final imputation method.

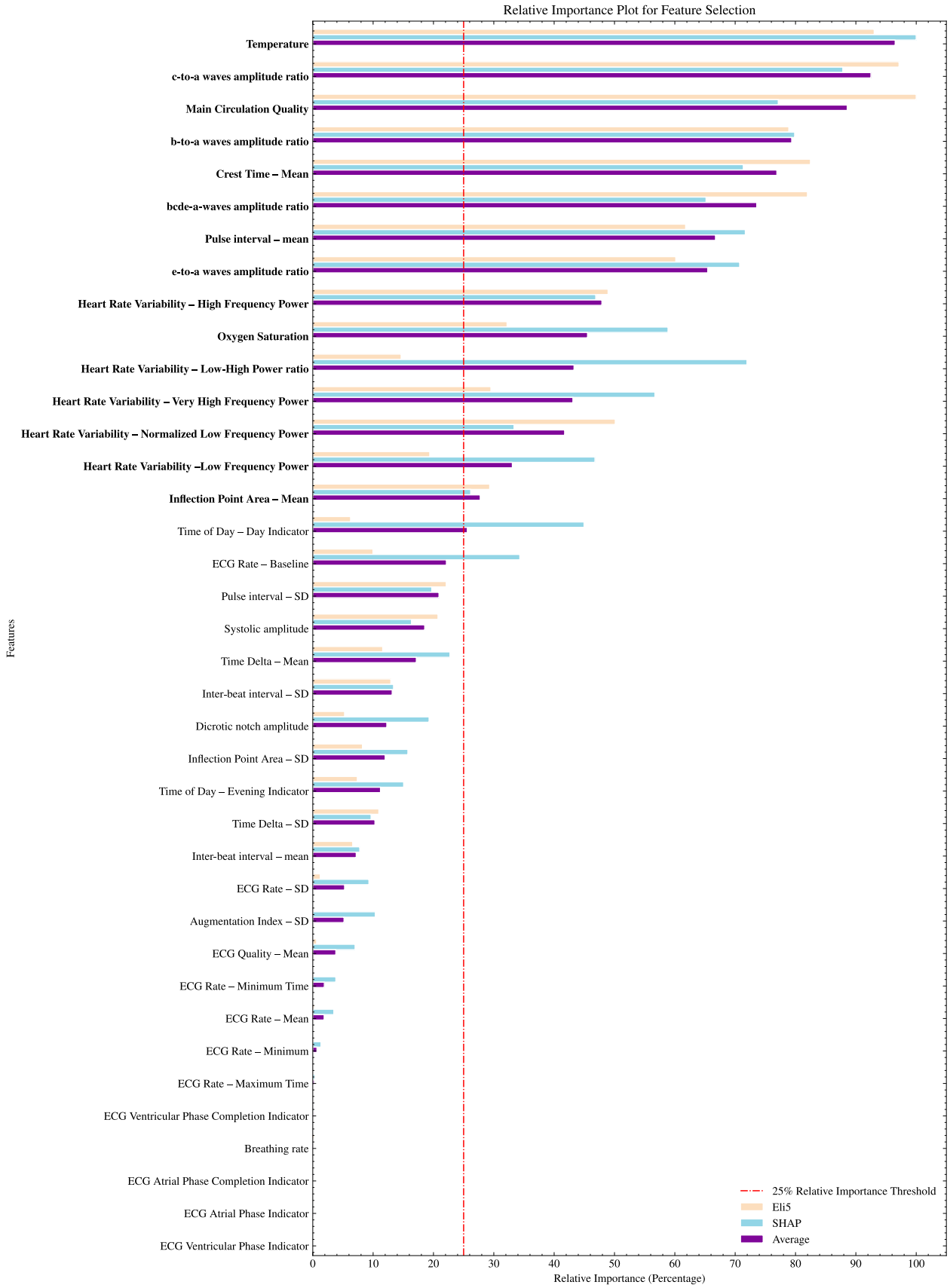


Fig. 7: Relative Importance plot for feature selection. The boldened features represent the final feature list whose mean importance is greater than 25% of the maximum feature importance derived from both methods.

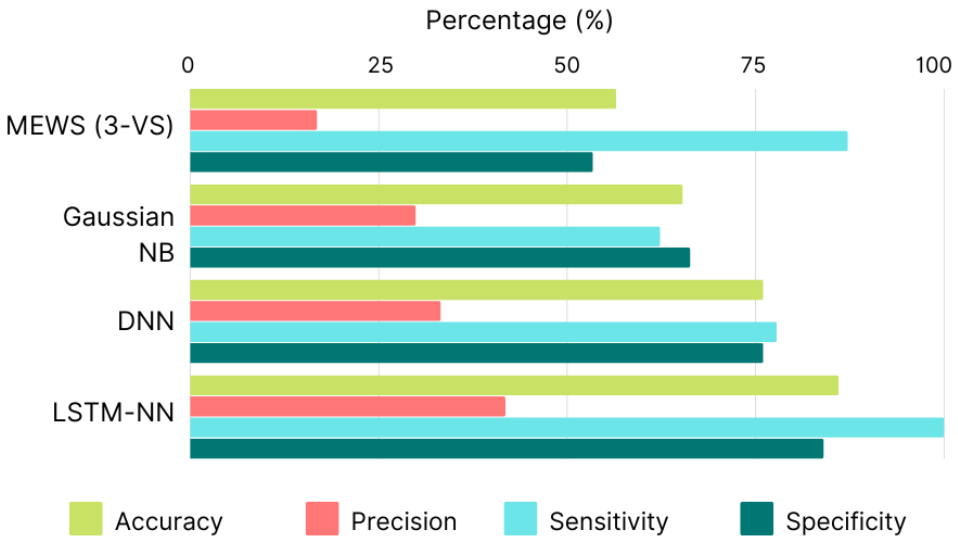


Fig. 8: Overall comparison of the performance of different decision support models

Thirdly, the morphological features of the PPG wave are crucial for CAE detection. While the body temperature being the highest-ranked feature is not a surprise, the following seven features are derived from the morphological properties of the PPG wave and its derivative forms. The role of these features is in line with other studies using these features for disease classifications. For example, Alty et al. [46] proved how the crest time, which is number 5 on the list, could be used to detect cardiovascular diseases. This study [47] also shows how the APG waveform features are good indicators of artery stiffness and possible atherosclerosis.

Fourthly, there is a tradeoff between extracting high-frequent respiration rate values and the amount of noise rejection that could be applied. It has been accepted that an unstable breathing rate is an excellent indicator of clinical deterioration. However, the results put the breathing rate as one of the bottom-3 features. One probable source of error could lie in the method of extracting the breathing rate in this study. In this study, since we extract the instantaneous breathing rate within 2 minutes, we count the number of peaks from the ECG-derived respiration (EDR) signal. While this method is ideal for extracting low-and-medium values of the breathing rate, there may be a mismatch with the higher breathing rate values. Since the automatic noisy window rejection procedure also depends on whether peak intervals fall in the outlying range, it is likely that a portion of good ECG beats may also be rejected. Hence, there exists a tradeoff between the quality of the breathing rate value and the amount of high-frequent noise to mix in the feature table. In this study, we prioritized the latter.

Fifthly, the impressive performance of the LSTM-NN affirms that utilizing temporal context in AI models do improve the prediction quality. When it comes to the LSTM-NN model, one might question whether machine learning based models primed for time series classification would also show respectable results. Other recommended ML models customized for time-series classification tasks like tree-ensemble models like the Time Series Forest [48] and dictionary-based models like the WEASEL+MUSE model [49] were also tested. Unfortunately, despite these models being less complex than the LSTM-NN model, they were not sensitive enough to render positive classifications and were discarded.

We can now discuss the functioning of the models for the two patients A and B.

Patient A has had previous cardiovascular comorbidities. We can gain some interesting insights from observing the model's predictions which is presented in Figure 9. The red and yellow zones represent 0-8 hours and 8-24 hours prior to a CAE respectively. In the case of MEWS, it seems to perform fairly well for Patient A where an alarm is raised well right before the start of the "in-time" window and 10 hours before the CAE. When we look at the Gaussian NB model, it also has a similar alarming point and has a frequent alarm rate to the point of complication. The deep neural network exhibits a more surged alarm progression. However, it is still consistent with the level change of the physiological state of patient A as is seen from the previous two models. The best performing model, the LSTM-NN provides its first positive classification later that the other three models. However, although it is comparatively later, we still see 3 alarms being generated from the model.

Patient B had no history of previous CAEs. For Patient B, the progression of the four models is presented in Figure 10. Here, the focus is on the negative window reporting capabilities or the false alarm rate of the four models. The MEWS score might look like it is doing a good job, but in the span of 5 days, it reported 15 false alarms. That is significant when compared to the performance of the other models on Patient B's trajectory. The Gaussian NB model shows perfect selectivity on his trajectory as it doesn't report a single alarm in his entire ward stay. The deep neural network does report three false alarms, but that is still $1/5^{th}$ of the total alarms reported by MEWS. The LSTM-NN model reports 4 false alarms, but that is still less than half of the MEWS reported false alarms. In this way, we get a clearer picture of the capabilities and the prediction styles of each of the main candidate models in comparison with the golden standard.

The advantages of the candidate models lie in their specificity compared to the results of the MEWS score. The Gaussian Naïve-Bayes, deep learning model and LSTM-NN classifiers was also able to predict CAEs at a false positive rate of 33.7%, 24% and 15.6%. The LSTM- NN reported a 66.5% reduction in the false-positive rate reported by MEWS. Another advantage is the data coverage needed for making a prediction. For this study, only 2 minutes of vital sign signal data was available per hour. This means the system is flexible with recorded values any time during the hour, and can be beneficial in periods of heavy motion, since any 2 minutes within an hour can be used for feature extraction and inference. In essence, such types of flexible recording schemes in wearable sensors could significantly reduce the noise rejection algorithms used.

Despite the success demonstrated, there are some limitations to consider. The biggest concerns are that of overgeneralization and interpretability. The question still arises of how the candidate models will behave with patients of other cohorts. It has to be seen in the face of unseen data whether the model tends to be more selective or more sensitive. The deep neural network and LSTM-NN models also lack interpretability. While the MEWS score has its drawbacks, its simple-to-use algorithm means its end-user can effectively pinpoint the sources of the largest score-affecting variables for diagnosis purposes. Meanwhile, in the deep neural network and the LSTM-NN architectures, it is significantly difficult to pinpoint the root predictors. Hence, an interesting direction to look into could involve studies to explore more interpretable time-series models.

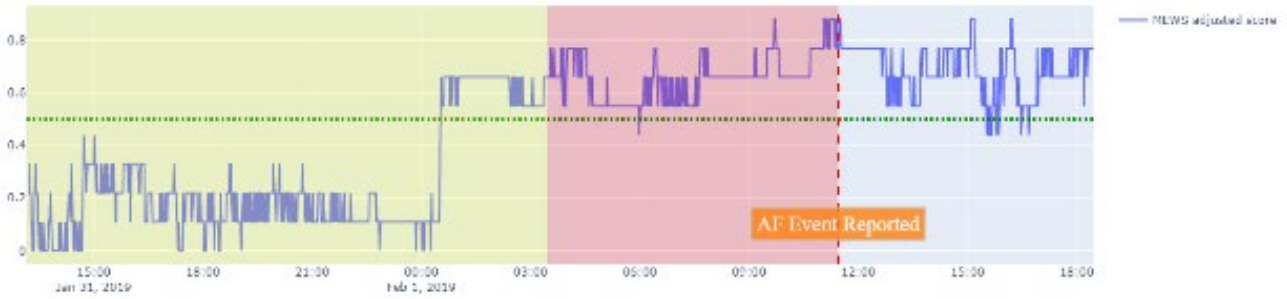


Fig. 9.(a)

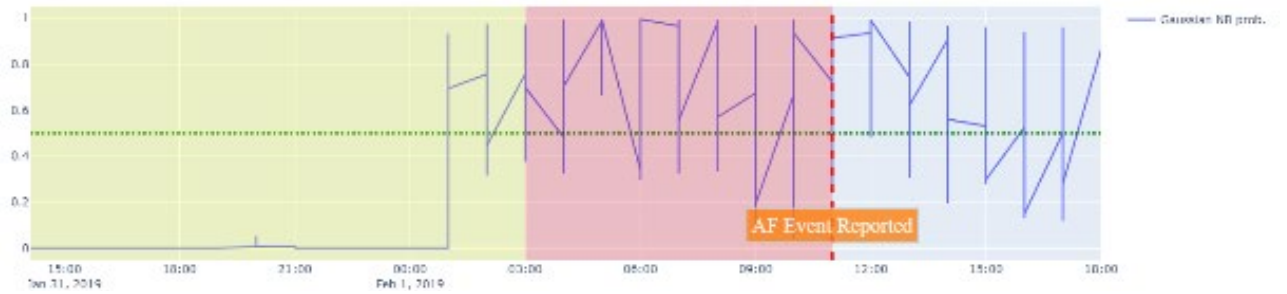


Fig. 9.(b)

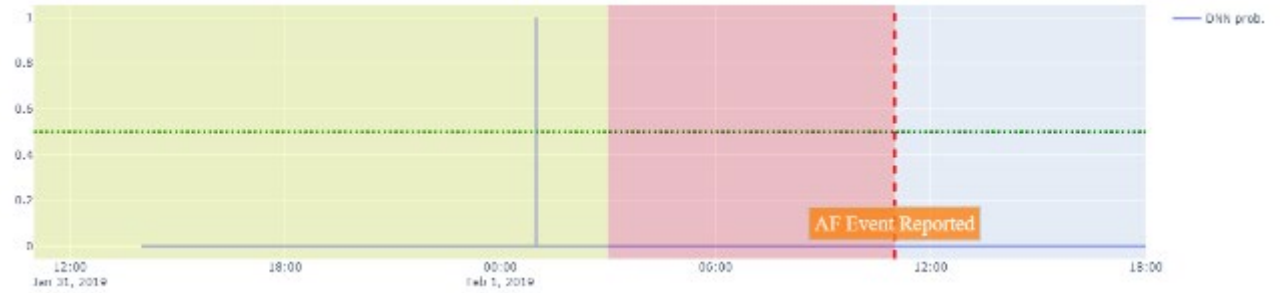


Fig. 9.(c)

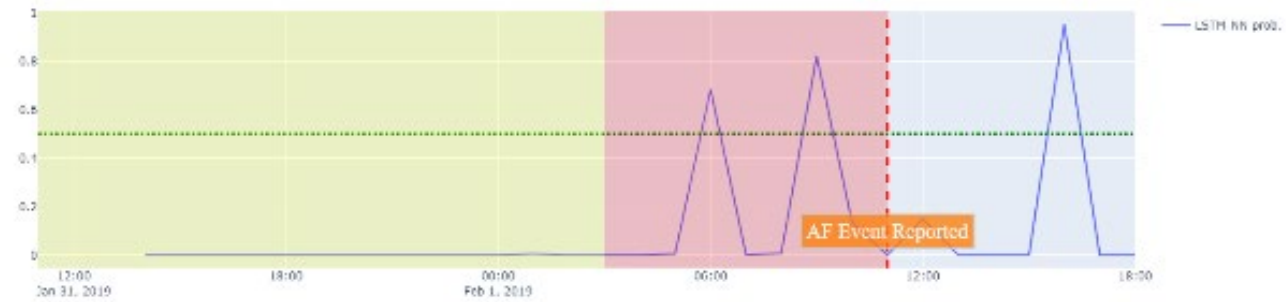


Fig. 9.(d)

Fig. 9: 9.(a) presents the progression of the MEWS score for the trajectory of Patient A. 9.(b) presents the progression of the Gaussian NB probability for the trajectory of Patient A. 9.(c) presents the progression of the DNN probability for the trajectory of Patient A. 9.(d) presents the progression of the LSTM-NN probability for the trajectory of Patient A. The green bar is the threshold for the respective model.

Fig. 10.(a)

Fig. 10.(b)

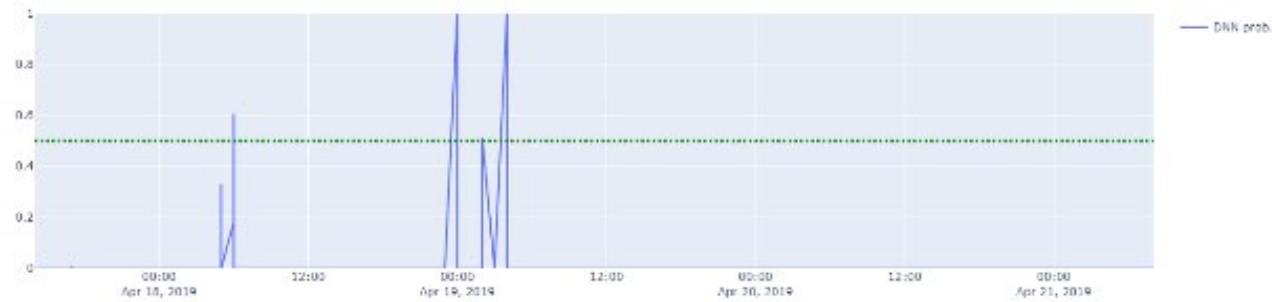


Fig. 10.(c)

Fig. 10.(d)

Fig. 10: 10.(a) presents the progression of the MEWS score for the trajectory of Patient B. 10.(b) presents the progression of the Gaussian NB probability for the trajectory of Patient B. 10.(c) presents the progression of the DNN probability for the trajectory of Patient B. 10.(d) presents the progression of the LSTM-NN probability for the trajectory of Patient B. The green bar is the threshold for the respective model.

In order to dismiss concerns about the generalizability of the system, external validation is crucial. Additionally, although the current deep learning models are black-box solutions, converting the problem from binary classification to multiclass classification could help redefine the problem space and make smaller and more specialized models for detecting certain CAEs. Furthermore, for longer time periods, it might be useful to use additional features that measure points of deviations in trends of the original features (change point analysis) to

improve the interpretation of time-series based predictors. For the case of the LSTM-NN model, current research in interpreting recurrent network time-series classifications is limited. However, work on interpreting convolution-based deep networks or transformers is thriving. This would involve converting the time series streams into images that could be trained with such image networks. As a continuation to this study, testing with multiple populations, more focused time-series analysis techniques or better time series representations are possible directions that could be explored further.

Conclusion

This study focused on the development of a remote monitoring system that utilizes features from wearable sensor data and a decision support model capable of predicting CAEs. We developed an analysis pipeline to extract features from short-term vital signs from wearable sensors and tested various candidate models to establish their performance. Although all the tested models provide good sensitivity to CAEs with good false-positive rates, the LSTM-NN model has superior performance in every metric compared to the golden standard of track-and-trigger systems. The results also provide clarity to certain research choices undertaken in the study. While this study presents preliminary results that posit the proposed model positively, additional multi-hospital testing and experimentation with other techniques are necessary to confirm its generalization and interpretability capabilities, respectively.

CRediT authorship contribution statement

Author contribution (later – during submission)

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Bibliography

[1] T. G. Weiser *et al.*, "Estimate of the global volume of surgery in 2012: an assessment supporting improved health outcomes," *The Lancet*, vol. 385, p. S11, Apr. 2015, doi: 10.1016/S0140-6736(15)60806-6.

[2] M. Zegers *et al.*, "The incidence, root-causes, and outcomes of adverse events in surgical units: implication for potential prevention strategies," *Patient Safety in Surgery*, vol. 5, no. 1, p. 13, May 2011, doi: 10.1186/1754-9493-5-13.

[3] "Global patient outcomes after elective surgery: prospective cohort study in 27 low-, middle- and high-income countries," *British Journal of Anaesthesia*, vol. 117, no. 5, pp. 601–609, Nov. 2016, doi: 10.1093/bja/aew316.

[4] S. Mohammed Iddrisu, J. Considine, and A. Hutchinson, "Frequency, nature and timing of clinical deterioration in the early postoperative period," *J Clin Nurs*, vol. 27, no. 19–20, pp. 3544–3553, Oct. 2018, doi: 10.1111/jocn.14611.

[5] U. Nilsson, R. Gruen, and P. S. Myles, “Postoperative recovery: the importance of the team,” *Anaesthesia*, vol. 75, no. S1, pp. e158–e164, 2020, doi: 10.1111/anae.14869.
[6] D. Massey, W. Chaboyer, and V. Anderson, “What factors influence ward nurses’ recognition of and response to patient deterioration? An integrative review of the literature,” *Nurs Open*, vol. 4, no. 1, pp. 6–23, Jan. 2017, doi: 10.1002/nop2.53.
[7] L. H. Aiken, “Hospital Nurse Staffing and Patient Mortality, Nurse Burnout, and Job Dissatisfaction,” *JAMA*, vol. 288, no. 16, p. 1987, Oct. 2002, doi: 10.1001/jama.288.16.1987.
[8] L. G. Glance, A. W. Dick, J. W. Meredith, and D. B. Mukamel, “Variation in Hospital Complication Rates and Failure-to-Rescue for Trauma Patients,” *Annals of Surgery*, vol. 253, no. 4, pp. 811–816, Apr. 2011, doi: 10.1097/SLA.0b013e318211d872.
[9] H. Gao *et al.*, “Systematic review and evaluation of physiological track and trigger warning systems for identifying at-risk patients on the ward,” *Intensive Care Med*, vol. 33, no. 4, pp. 667–679, Mar. 2007, doi: 10.1007/s00134-007-0532-3.
[10] G. B. Smith, D. R. Prytherch, P. E. Schmidt, P. I. Featherstone, and B. Higgins, “A review, and performance evaluation, of single-parameter ‘track and trigger’ systems,” *Resuscitation*, vol. 79, no. 1, pp. 11–21, Oct. 2008, doi: 10.1016/j.resuscitation.2008.05.004.
[11] S. Gerry *et al.*, “Early warning scores for detecting deterioration in adult hospital patients: systematic review and critical appraisal of methodology,” *BMJ*, p. m1501, May 2020, doi: 10.1136/bmj.m1501.
[12] M. A. F. Pimentel, D. A. Clifton, L. Clifton, P. J. Watkinson, and L. Tarassenko, “Modelling physiological deterioration in post-operative patient vital-sign data,” *Med Biol Eng Comput*, vol. 51, no. 8, pp. 869–877, Aug. 2013, doi: 10.1007/s11517-013-1059-0.
[13] L. Clifton, D. A. Clifton, P. J. Watkinson, and L. Tarassenko, “Identification of patient deterioration in vital-sign data using one-class support vector machines,” p. 7, 2011.
[14] M. M. Churpek, T. C. Yuen, C. Winslow, D. O. Meltzer, M. W. Kattan, and D. P. Edelson, “Multicenter Comparison of Machine Learning Methods and Conventional Regression for Predicting Clinical Deterioration on the Wards:,” *Critical Care Medicine*, vol. 44, no. 2, pp. 368–374, Feb. 2016, doi: 10.1097/CCM.0000000000001571.
[15] M. Elgendi, “On the Analysis of Fingertip Photoplethysmogram Signals,” *Curr Cardiol Rev*, vol. 8, no. 1, pp. 14–25, Feb. 2012, doi: 10.2174/157340312801215782.
[16] Z. Zhao and Y. Zhang, “SQI Quality Evaluation Mechanism of Single-Lead ECG Signal Based on Simple Heuristic Fusion and Fuzzy Comprehensive Evaluation,” *Front Physiol*, vol. 9, p. 727, 2018, doi: 10.3389/fphys.2018.00727.
[17] D. Makowski *et al.*, “NeuroKit2: A Python toolbox for neurophysiological signal processing,” *Behav Res*, vol. 53, no. 4, pp. 1689–1696, Aug. 2021, doi: 10.3758/s13428-020-01516-y.
[18] F. T. Liu, K. M. Ting, and Z.-H. Zhou, “Isolation Forest,” in *2008 Eighth IEEE International Conference on Data Mining*, Dec. 2008, pp. 413–422. doi: 10.1109/ICDM.2008.17.
[19] S. Galli, “Feature-engine: A Python package for feature engineering for machine learning,” *JOSS*, vol. 6, no. 65, p. 3642, Sep. 2021, doi: 10.21105/joss.03642.
[20] F. Pedregosa *et al.*, “Scikit-learn: Machine Learning in Python,” *MACHINE LEARNING IN PYTHON*, p. 6.
[21] G. Ke *et al.*, “LightGBM: A Highly Efficient Gradient Boosting Decision Tree,” in *Advances in Neural Information Processing Systems*, 2017, vol. 30. Accessed: May 15, 2022. [Online]. Available:

https://proceedings.neurips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html
[22] T. Chen and T. He, “xgboost: eXtreme Gradient Boosting,” p. 4.
[23] A. V. Dorogush, V. Ershov, and A. Gulin, “CatBoost: gradient boosting with categorical features support.” arXiv, Oct. 24, 2018. Accessed: May 15, 2022. [Online]. Available: http://arxiv.org/abs/1810.11363
[24] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors,” *Nature*, vol. 323, no. 6088, Art. no. 6088, Oct. 1986, doi: 10.1038/323533a0.
[25] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
[26] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio, “On the Properties of Neural Machine Translation: Encoder-Decoder Approaches.” arXiv, Oct. 07, 2014. Accessed: May 15, 2022. [Online]. Available: http://arxiv.org/abs/1409.1259

Appendix A: List of Features

The complete list of features extracted from the dataset are provided below as a key to the final features used, their purposes and the category they belong to. The feature category refers to which time span the feature was extracted from: 6S denotes the 6-second time span, M denotes the minute-based time span, and PS denotes the patient specific features.

Feature no.	Feature Category	Feature	Description of feature
1	6S	ECG Rate – Baseline	The baseline heart rate within a 6-second span, from which other ECG rate features are relatively computed.
2	6S	ECG Rate – Maximum **(R)**	The maximum heart rate recorded within a 6-second span.
3	6S	ECG Rate – Minimum	The minimum heart rate recorded within a 6-second span.
4	6S	ECG Rate – Mean	The mean heart rate recorded within a 6-second span.
5	6S	ECG Rate – SD	The standard deviation of the heart rate recorded within a 6-second span.
6	6S	ECG Rate – Maximum Time	The time at which maximum ECG rate occurs.
7	6S	ECG Rate – Minimum Time	The time at which minimum ECG rate occurs.
8	6S	ECG Atrial Phase Indicator	Indication of whether the onset of the event concurs with respiratory systole (1) or diastole (0).
9	6S	ECG Atrial Phase Completion Indicator	Indication of the stage of the current cardiac (atrial) phase (0 to 1) at the onset of the event.
10	6S	ECG Ventricular Phase Indicator	Indication of whether the onset of the event concurs with respiratory systole (1) or diastole (0).
11	6S	ECG Ventricular Phase Completion Indicator	Indication of the stage of the current cardiac (ventricular) phase (0 to 1) at the onset of the event.
12	6S	ECG Quality – Mean	Index denoting the relative quality of ECG signal.
13	6S	Breathing Rate	Respiratory rate from the ECG-derived respiratory (EDR) signal in 12-second span. (2-6S stitched)
14	M	Heart Rate Variability – High Frequency Power	The spectral power density pertaining to high frequency band i.e., 0.15 to 0.4 Hz.
15	M	Heart Rate Variability – Very High Frequency Power	The variability, or signal power, in very high frequency i.e., 0.4 to 0.5 Hz.
16	M	Heart Rate Variability – Low-High Power ratio	The ratio of low frequency power to high frequency power.
17	M	Heart Rate Variability – Normalized Low Frequency Power	The normalized low frequency, obtained by dividing the low frequency power by the total power.
18	M	Heart Rate Variability – Normalized High Frequency Power **(R)**	The normalized high frequency, obtained by dividing the high frequency power by the total power.
19	M	Heart Rate Variability – Log transformed HF **(R)**	The log transformed HRV_HF.

20	M	Heart Rate Variability –Low Frequency Power	The spectral power density pertaining to low frequency band i.e., 0.04 to 0.15 Hz
21	6S	Systolic amplitude	The mean of the amplitude of the systolic peaks of the PPG wave in a 6-second span.
22	6S	Dicrotic notch amplitude	The mean of the amplitudes of the dicrotic notches of the PPG wave in a 6-second span.
23	6S	Inter-beat interval – mean	The mean time interval between successive systolic peaks of the PPG wave in 6-second span.
24	6S	Inter-beat interval – SD	The standard deviation of the time intervals between successive systolic peaks of the PPG wave in 6-second span.
25	6S	Pulse interval – mean	The mean of the difference of the start and end of a PPG waveform in a 6-second span.
26	6S	Pulse interval – SD	The standard deviation of the difference of the start and end of a PPG waveform in a 6-second span.
27	6S	Inflection Point Area – Mean	The mean inflection point area ratio i.e. the ratio of the two sub-areas separated by the dicrotic notches in a 6-second span.
28	6S	Inflection Point Area – SD	The standard deviation of the inflection point area ratio i.e. the ratio of the two sub-areas separated by the dicrotic notches in a 6-second span.
29	6S	Augmentation Index – Mean **(R)**	The mean ratio of the dicrotic amplitude to that of the systolic peak in a 6-second span.
30	6S	Augmentation Index – SD	The standard deviation of the ratio of the dicrotic amplitude to that of the systolic peak in a 6-second span.
31	6S	Time Delta – Mean	The mean peak-to-peak time interval in the velocity photoplthysmogram (VPG) waveform in a 6-second span.
32	6S	Time Delta – SD	The standard deviation of the peak-to-peak time interval in the velocity photoplthysmogram (VPG) waveform in a 6-second span.
33	6S	Crest Time – Mean	The mean of the time from the foot of VPG waveform in a 6-second span.
34	6S	Crest Time – SD **(R)**	The standard deviation of the time from the foot of VPG waveform in a 6-second span.
35	6S	Main Circulation Quality	Indicator variable used to show how many waves are recognizable from the APG in a 6-second span.
36	6S	b-to-a waves amplitude ratio	The mean of the ratios of the early systolic negative wave to the early systolic positive wave in a 6-second APG waveform span.
37	6S	c-to-a waves amplitude ratio	The mean of the ratios of the late systolic reincreasing wave to the early systolic positive wave in a 6-second APG waveform span.
38	6S	d-to-a waves amplitude ratio **(R)**	The mean of the ratios of the late systolic decreasing wave to the early systolic positive wave in a 6-second APG waveform span.
39	6S	e-to-a waves amplitude ratio	The mean of the ratios of the early diastolic positive wave to the early systolic positive wave in a 6-second APG waveform span.

40	6S	bcde-a-waves amplitude ratio	The mean of the $(b-c-d-e)/a$ ratios in a 6-second APG waveform span.
41	6S	cdb-a-waves amplitude ratio **(R)**	The mean of the $(c+d-b)/a$ ratios in a 6-second APG waveform span.
42	M	Temperature	The per-minute measured body temperature.
43	M	Oxygen Saturation	The per-minute measured oxygen saturation.
44	M	Time of Day – Day Indicator	Categorical variable used to indicate the measurement recorded from 08-16 hours.
45	M	Time of Day – Evening Indicator	Categorical variable used to indicate the measurement recorded from 16-24 hours.
46	PS	Age	The age of a patient.
47	PS	Length	The height of a patient.
48	PS	Comorbidity – Infective indicator	Categorical variable (0/1) used to indicate the presence of infective diseases.
49	PS	Gender	The gender of a patient.
50	PS	Weight	The weight of a patient.
51	PS	Comorbidity – Pulmonal indicator	Categorical variable (0/1) used to indicate the presence of pulmonal comorbidities.
52-54	PS	Additional features	The representation of patient data in 3 dimensions using dimensionality reduction.

Table A1: Complete list of all features extracted for this study. Only relevant features shown in literature to have links with CAEs were chosen. The label **(R)** stands for redundant features that were omitted.

Appendix B: Deep Learning

The best hyperparameters used in this study for the deep learning networks is presented in Table B1. A visualization of the DNN and the LSTM-NN networks are presented in Fig. B2.

Hyperparameter	Value
Hidden layer size	256
Number of layers	3
Batch size	494
Dropout threshold	0.1
Optimizer	Adam
Learning rate	0.001

Table B1.(a)

Hyperparameter	Value
Hidden layer size (NN)	64
Number of LSTM layers	3
Number of NN layers	1
Embedding size	47
Batch size	14
Optimizer	Adam
Learning rate	0.001

Table B1.(b)

Table B1: B1.(a) is the final hyperparameters of the deep neural network tested. B1.(b) is the final hyperparameters of the complete LSTM neural network tested.

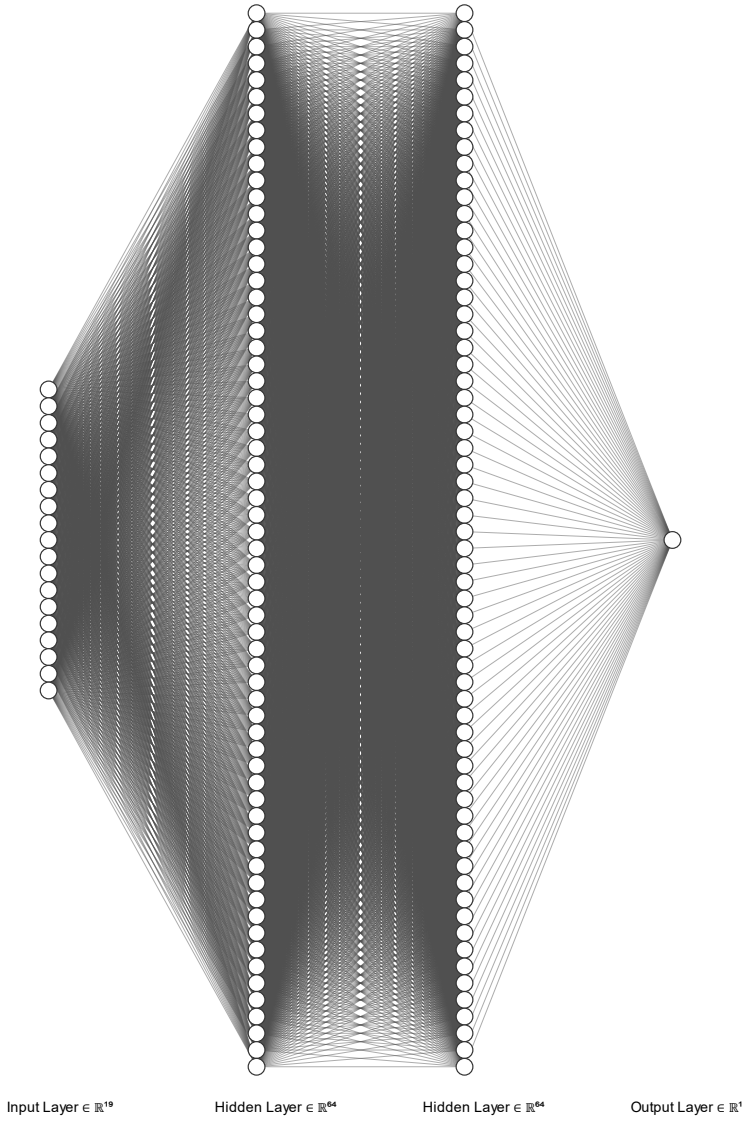


Fig. B2.(a)

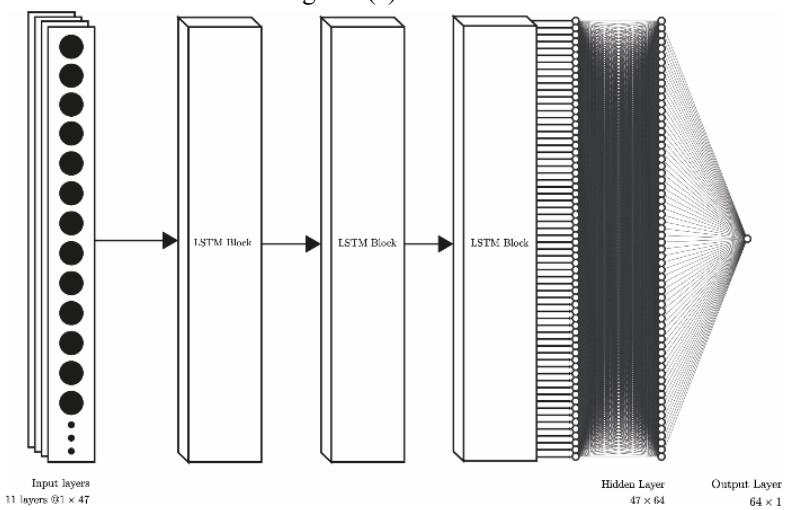


Fig. B2.(b)

Fig. B2: B2.(a) is the base architecture of the deep neural network. B2.(b) is the base architecture of the complete LSTM neural network.