An LSTM-based Recurrent Neural Network for Neonatal Sepsis Detection in Preterm Infants

A. Honoré[1,2], H. Siren[2,3], R. Vinuesa[3], S. Chatterjee[3] and E. Herlenius[2]

1. Div. Information Science and Engineering, KTH Royal Institute of Technology, Stockholm, Sweden
2. Department of Women's and Children's Health, Karolinska Institutet, Stockholm, Sweden
3. FLOW, Engineering Mechanics, KTH Royal Institute of Technology, Stockholm, Sweden
{honore, hsiren, rvinuesa, sach}@kth.se, eric.herlenius@ki.se

***Abstract*— Early and accurate neonatal sepsis detection (NSD) can help reduce mortality, morbidity and antibiotic consumption in premature infants. NSD models are often designed and evaluated in case control setups and using data derived from patient electrocardiogram (ECG) only. In this article, we evaluate our models in a more realistic retrospective cohort study setup. We use data from different modalities, including ECG, chest impedance, pulse oximetry, demographics factors and repetitive measurements of body weights. We study both the vanilla and long-short-term-memory (LSTM) Recurrent Neural Networks (RNN) architectures in a sequence to sequence mapping framework for NSD. We compare the performances of the models with logistic regression (LR) on a variety of classification metrics in a leave-one-out cross validation framework. The population we used contains 118 very low birth weight infants, among which 10 experienced sepsis. We showed that LSTM-based RNNs are both (1) more conservative and (2) more precise than LR or vanilla RNN, with a true negative rate at least +26% higher and a precision score of 0.16 compared to 0.06 for LR. This indicates that LSTM-based RNNs have the potential to reduce the false alarm rate of existing linear models, while providing a reliable diagnostic aid for neonatal sepsis.**

Keywords*— Neonatal sepsis detection, Recurrent neural network, LSTM.

I. Introduction

Infants born preterm, before 37 weeks of gestation, are at risk of developing a sepsis after birth. Of all live births, 10% are preterm worldwide, of which 10-25% are affected by sepsis [1]. Sepsis increases mortality and morbidity on this population. Although early antibiotic treatment can help avoid adverse consequences of sepsis, the over-usage of antibiotics has harmful side effects and should be avoided [2]. Traditional methods to diagnose sepsis involve blood cultures which are invasive, slow and often inaccurate. Clinical decision support systems (CDSS), based on patient monitoring were shown promising for early neonatal sepsis detection (NSD) [3, 4]. These methods rely on features extracted from the routinely monitored electrocardiogram-derived inter-beat-interval (IBI) signal. It was shown that sepsis could be detected up to 24 hours early, using window-based features and linear predictors [4]. Features computed on the pulse oximetry derived blood oxygen saturation level (SpO_2) were also shown useful to predict infection related conditions in preterm infants [5]. The respiratory behavior of patients is heavily impacted by sepsis, and thus adding chest impedance derived respiratory frequency should help increase NSD [6]. Linear predictive models are limited in their ability to use interdependencies between input features calculated from windows of monitoring signals. Moreover, features extracted from sliding windows of signals are correlated, and this is often not taken into account in linear predictive models.

To address these issues, we explore the use of recurrent neural networks (RNN) for NSD. RNNs are non-linear and dynamical models that are capable of leveraging both correlation among features extracted from individual time frames as well as dependencies in time, thus making them particularly suited in our context. These models have also proven useful in a variety of timeseries classification tasks, including NSD in a case control study setup [7]. We thereby examine whether RNNs can improve over linear classification models in a more realistic scenario by performing a cohort study with 118 patients, among which 10 experienced sepsis. The study was conducted by first comparing a vanilla RNN architecture with a linear predictor. We then evaluated long short-term memory (LSTM) architecture with the two baseline models. All the performance metrics we used for comparison are computed on a cross-validation scheme. Finally, we display prediction plots of example patient cases.

II. Methods

II-A. Population

We performed a retrospective cohort study on a population of 118 very low birth weight (VLBW) infants hospitalized at Karolinska University Hospital. Among these, 10 patients were diagnosed with at least one sepsis while under full monitoring. The birth weight of the infants was $927 \pm 282g$ and the gender distribution was 44% male (52) and 56% female (66). On average 523 ± 471 hours of monitoring data was available per patient. The study was approved by the Swedish Ethical Review Authority (2020-02487).

II-B. Dataset Preparation

In this section we describe how our datasets are created. We first describe how the signals were collected, how the features were extracted from said signals and how the data was labeled.

II-B1. Monitoring Data

We collected high frequency data from Philips IntelliVue MX800 Patient Monitor (Philips Healthcare, Amsterdam, Netherlands). In parallel, we built a detailed clinical event timeline from the Electronic Health Records (EHR) in place at Karolinska University hospital, Stockholm, Sweden. From these timelines, we used the identified time of the collection of a blood culture as the time of sepsis suspicion. The pre-processed monitoring data, resampled at 1 Hz were used as basis signals. We used the electrocardiogram-derived IBI signal, the level of blood oxygen saturation measured by pulse oximetry (SpO2) and the respiratory frequency derived from chest impedance. The IBI signal was further filtered to allow the use of the sample entropy feature [8]. Ectopic beats and the strong non-physiological frequency content of the signals were removed using the composition of a moving median filter of width 3 samples and a Butterworth filter of order 6 with cut frequencies of 0.0021 and 0.43 Hz [9, 10].

II-B2. Feature Extraction

We extracted features on sliding windows of length 55 minutes with 50% overlap. We have shown that this window length leads to good performances in a preliminary experimental study [11]. A sliding window starting up to 24 hours from sepsis diagnosis was labeled 1, and we labeled 0 the windows starting earlier or after the time of sepsis suspicion. When several sepsis where diagnosed on the same patients at most 14 days apart, the frames contained between the two diagnosis were labeled 1. We calculated the range of the signals and the statistics up to order 4 to characterize the distribution of the signal samples in a time frame. For this, we extracted the minimum, maximum, mean, standard deviation, skewness and kurtosis of all the signals. The skewness characterizes the deviation of the signal distribution from a Gaussian distribution while the kurtosis captures the dispersion of the variance around it's expectation [3]. We additionally compute the sample asymmetry and the sample entropy of the IBI signal. These two features, together with the standard deviation, are part of the commercialized HeRO system [4]. We added two static demographics features: the sex and the birth weight, as well as the postnatal age and the repetitive body weight measurements.

Overall, we get a dataset of $N = 118$ patients, $\mathscr{D} = (\mathbf{X}^{(k)}, \mathbf{Y}^{(k)})\}_{k=1}^{N}$, where $\mathbf{X}^{(k)} = [\mathbf{x}_1^{(k)}, \ldots, \mathbf{x}_{T_k}^{(k)}]$ is the series of length T_k of $d = 24$ dimensional samples for patient k and $\mathbf{Y}^{(k)} = [y_1^{(k)}, \ldots, y_{T_k}^{(k)}]$ is the series of length T_k of binary labels for patient k.

II-C. Dataset Description

We denote the collection of the timeseries from patients with at least one frame labeled 1 as the positive population. On the contrary, the negative population is comprised of patients with all frames labeled 0. We report our results on these populations separately as well as on the overall population. This allow us to determine whether our prediction algorithms tend to output more false alarms on the sickest patients than on the healthiest patients.

A summary of these populations is presented in Table 1.

	# of patients	**Average # of samples per patient**	**# of samples**	
Patients			**Total**	**Prevalence**
Positive	10	2 099 (1 280)	20 992	2.86 %
Negative	108	1 053 (952)	113 676	0 %
Overall	118	1 141 (1 027)	134 668	0.48 %

Table 1. Description of the datasets. The prevalence corresponds to the ratio: $\frac{\text{\# class 1 samples}}{\text{\# class 0 samples}}$.

II-D. Recurrent Neural Networks (RNN)

Here the construction of the examined RNNs is described, the format of the input data discussed as well as the aspects related to model training presented.

II-D1. Architecture

RNNs are a family of deep learning models suited to tasks containing sequential data. These models are, in theory, able to capture non-linearity and long term dependencies in input time sequences. The model architecture can be formulated as a state space model:

$$\mathbf{h}_t = \mathbf{g}(\mathbf{x}_t, \mathbf{h}_{t-1}), \quad \hat{y}_t = f(\mathbf{h}_t) \tag{1}$$

where $\mathbf{g}$ is a trainable function returning the hidden state at time t, $\mathbf{h}_t$, from (1) the data at time t, $\mathbf{x}_t$, and (2) the hidden state at the previous time step, $\mathbf{h}_{t-1}$. f is a trainable linear function of the hidden state at time t with a sigmoid activation returning the estimated probability of sepsis at time t, $\hat{y}_t$. This process allows RNN models to leverage information from previous time steps and to use them to the predict the output sequence of labels. A key advantage of these models is that they are suited to capture dependencies in long sequences.

II-D2. Input Data

For each patient, the sequence of estimated sepsis probabilities were obtained by (1) segmenting the timeseries into sub-sequences of 50 samples (i.e. 22.9 hours) with a step size of 1 sample, and (2) using the RNN in a many-to-one setup [12], i.e. where only the last predicted label is used to compute the loss.

II-D3. Loss Function

Our dataset is imbalanced, with 11 times as many negative patients as positive patients. To account for this, the functions $\mathbf{g}$ and f are trained with back-propagation on a weighted cross-entropy loss (Eq. 2). Let us denote $\hat{\mathbf{Y}}$ a sequence of estimated sepsis probabilities, and $\mathbf{Y}$ the corresponding true sequence of labels for a patient k. The weighted cross-entropy loss, between $\hat{\mathbf{Y}}$ and $\mathbf{Y}$,

can be written:

$$\mathscr{L}(\mathbf{Y},\hat{\mathbf{Y}}) = -\sum_{t=1}^{T_k} w_{y_t}\left(y_t \log(\hat{y}_t) + (1-y_t)\log(1-\hat{y}_t)\right), \quad (2)$$

where we dropped the index k for readability, and the weights w_0 and w_1 are inversely proportional to the class frequencies of samples in the training data sets.

II-E. Performance Assessment

In this section, the methods for validating the performance of our models are detailed and the relevant performance metrics discussed.

II-E1. Cross-validation

In order to leverage as much information as possible from our dataset, we used a leave one out cross-validation scheme to evaluate our models. The process is depicted as a flow chart in Figure 1. The training sets are composed of all but one positive patient and 10% of the negative patients selected randomly. The validation sets are composed of the left out positive patient and the remaining 90% of the negative patients. With this split method, the test set is representative of the prevalance of positive patients in the cohort. This scheme was repeated 10 times. At each run, a new positive patient was assigned to the validation set such that all positive patients were used once for validation. Our results are reported as the sample mean and standard deviation of performance metrics computed on the validation sets. We compare the performances of the models on (1) the positive patients only, (2) the negative patients only, (3) the overall patient population. Comparing scores by separating patients in the validation sets allow us to compare the specificity, and whether our model tends to output more false alarms on the positive patients, expected to be the sickest.

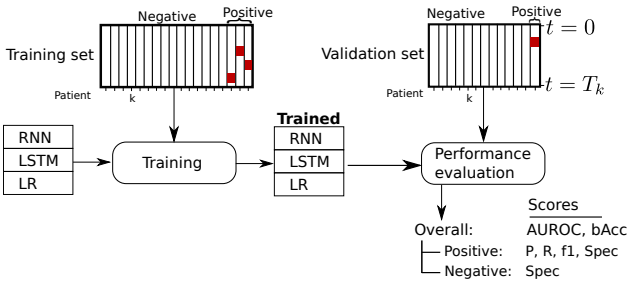

Figure 1. Training and validation procedure flowchart.

II-E2. Scores

We computed the number of true positives (TP), the number of true negatives (TN), the number of false positive (FP) and the number of false negatives (FN). From these we reported the precision: $\frac{TP}{TP+FP}$, recall/sensitivity (sen): $\frac{TP}{TP+FN}$, specificity (spec): $\frac{TN}{TN+FP}$, F_1-score: $2\frac{p\times r}{p+r}$ and balanced accuracy (bAcc): $\frac{1}{2}(\text{sen} + \text{spec})$. For these metrics, the individual samples were classified "positive" when the output probability $\hat{y}_t > 0.5$. We also report the threshold independent area-under-the-receiver-operating-characteristic (AUROC) on the overall population.

III. Experimental Design

III-A. Baseline Models

We use a logistic regression (LR) model as a baseline. In LR models, the sepsis probability is computed as a linear combination of the input vector:

$$\hat{y}_t = \text{sigmoid}(\mathbf{w}^T\mathbf{x}_t + b), \quad (3)$$

where $\mathbf{w} \in \mathbb{R}^d$ and $b \in \mathbb{R}$ are trainable weights and a bias term. The LR model considers consecutive window-based feature samples as independent. LR was optimized with the LBFGS solver on the binary cross-entropy loss, using the sklearn library [13].

We also use a 1-layer vanilla RNN architecture. In this model, the time dependency is taken into account in a state space model, as described in Eq. 1. The output sepsis probability at time t is computed as a sigmoid function of the latent vector, rather than the input vector at time t. This architecture can be written:

$$\begin{aligned} \mathbf{h}_t &= \mathbf{g}(\mathbf{x}_t, \mathbf{h}_{t-1}) = \tanh(W_i\mathbf{x}_t + \mathbf{b}_i + W_s\mathbf{h}_{t-1} + \mathbf{b}_s) \\ \hat{y}_t &= f(\mathbf{h}_t) = \text{sigmoid}(\mathbf{w}_o^T\mathbf{h}_t + \mathbf{b}_o) \end{aligned} \quad (4)$$

where $W_i \in \mathbb{R}^{h\times d}$, $\mathbf{b}_i \in \mathbb{R}^h$, $W_s \in \mathbb{R}^{h\times h}$ and $\mathbf{b}_s \in \mathbb{R}^h$ are trainable weight matrices and bias vectors for the latent space vector computation. $\mathbf{w}_o \in \mathbb{R}^h$, $\mathbf{b}_o \in \mathbb{R}$ are trainable vectors for the sepsis probability computation. The vanilla RNN was trained with back-propagation using the Adam optimizer [14], a learning rate of 0.001 and 100 epochs. The results of both LR and vanilla RNN are shown in Table 2.

III-B. LSTM-based RNN Architecture and Training

The RNN model we used in this study was the long short-term memory (LSTM) unit [15]. LSTMs are popular architecture that rely on a cell state to mitigate the vanishing gradient problem, often encountered when training RNNs. The readers can find the details of the architecture in e.g. [16]. We report the results for models with recurrent units of hidden size $h = 50, 100, 150$ and 200, and with a number of recurrent units composing the model of 3 and 4. The weights of the linear transforms composing the models were initialized from a uniform distribution with support $\mathscr{U}(k) = [-\frac{1}{\sqrt{k}}; \frac{1}{\sqrt{k}}]$, where k is the number of input features to the transform [17]. The back-propagation algorithm we used was the Adam [14] optimizer, with a learning rate and a number of training epochs fixed at 0.001 and 200 respectively. All the sub-sequences from the same patients were fed to the RNN algorithms in a single batch. Thus the batch size varies from 102 to 4065 sub-sequences, depending on the input patient. The models were implemented in Python using the Pytorch [18] library. All

the experiments were performed on a server computer with 2× Intel(R) Xeon(R)Gold/6230/CPU@2.10GHz and 4× Tesla V100/SXM2/16GB.

IV. Results

IV-A. Baseline Models

We present the results for our baseline models in Table 2.

Positive patients				Negative patients	Overall	
Prec.	**Recall**	**F1-score**	**Spec.**	**Spec.**	**AUROC**	**bAcc**
Logistic regression						
0.06 (0.04)	**0.60** (0.39)	**0.11** (0.07)	**0.73** (0.28)	**0.77** (0.04)	**0.81** (0.15)	0.60 (0.23)
Vanilla RNN						
0.04 (0.04)	0.59 (0.41)	0.07 (0.07)	0.58 (0.24)	0.70 (0.11)	0.71 (0.18)	**0.62** (0.18)

Table 2. Mean (standard deviation) performance metrics for the baseline LR and vanilla RNN. Spec.: Specificity, Prec.: Precision, bAcc: Balanced accuracy.

The performances in terms of AUROC are 14% higher for the LR model (0.81) than for the vanilla RNN (0.71) when considering the overall population. On this population, the balanced accuracy is the only score that we reported higher for the vanilla RNN than the LR model, although we note that the standard deviation is very large. The specificity is also higher for the LR model (0.77) compared to the vanilla RNN (0.70) when only the negative patients are considered. For the positive patients, the specificity of LR is 26% higher (0.73) than that of vanilla RNN (0.58) although again the standard deviation is large. The other metrics: precision, recall and f1-score are similar for the two models.

These results indicate that LR performs marginally better than the vanilla RNN. This RNN architecture is thus not adequate to capture the time dependencies in consecutive window-based features.

IV-B. LSTM Architectures

The results for the LSTM-based RNNs are reported in Table 3.

We reported results for varying number of layers and varying hidden size. The best performing LSTM-based RNN in terms of both AUROC and bAcc computed on the overall population is the model with 3 hidden layers and a hidden size $h = 200$. We note that given the large standard deviation, this is only marginally better than a similar model with 4 hidden layers. All the models perform similarly on the negative population with a specificity ≥ 0.97 for all the models, and reaching 0.99 for the smallest model with 3 layers and a hidden size $h = 50$. Across all the LSTM-based RNNs, the specificity on the positive population has consistently lower mean and higher standard deviation than on the negative population. The best model in terms of mean and standard deviation for this metric is a LSTM-based RNN with 4 layers and hidden size 50. This result indicates that the models have a tendency to produce more false alarms on the positive patients than on the negative patients.

The model with the best recall and f1-score, and the second best precision is the LSTM-based RNN with 3 layers and a hidden size of 200. The model with also 3 layers but a hidden size $h = 100$ has similar performances across all metrics. The models with 4 hidden layers all have lower precision, recall and f1-score. Given that the models with 3 layers have less trainable parameters, we further compare the results to the baseline models using LSTM-based RNN with hidden size $h = 100$ and $h = 200$.

In comparison with the baseline models, LSTM-based RNNs have a specificity that is $+26\%$ higher than the LR model on the negative population, and $+19\%$ on the positive population. The standard deviation is also lower for the two LSTM-based RNNs. It however comes at the cost of the recall that is 40% lower for the LSTM-based RNNs, while their precision remains higher than the LR 0.16 vs 0.06. This indicates that the LSTM-based RNNs are more conservative that the LR models, thus triggering fewer false alarms while missing some positive frames.

IV-C. Example Cases

In Figure 2 we show the predicted sequences of sepsis probabilities $\hat{\mathbf{Y}}$ versus the post natal age for two example patients, along with the corresponding confusion matrices. The sequences of estimated sepsis probabilities are obtained from LSTM-based RNNs with 3 layers and a hidden size $h = 100$ and 200.

For patient 1, we see that the two peaks in probability of sepsis occur slightly after the labeled segments. This is confirmed in the confusion matrices where very few samples labeled 1 are accurately predicted "1" by the LSTM models. We also note that the two LSTM-based models output a large sepsis probability between 200 and 350 hours after birth. This time segment is located between two diagnosis of sepsis. Our models might detect truly adverse patterns in the vital signs that our labeling method does not characterize as "septic".

For patient 2, the models succeed in detecting the frame within the segment labeled 1. We also note that the model with a hidden size of 200 outputs high sepsis probabilities even before the segment labeled 1. As is shown on the confusion matrices, the models fail to correctly classify frame within the segment labeled 1. Both models falsely output high probabilities 13 to 14 days after the positive label segment. Our detailed clinical timeline show that the patient was unstable

		Positive patients				Negative patients	Overall	
Hidden size	**# Layers**	**Prec.**	**Recall**	**F1-score**	**Spec.**	**Spec.**	**AUROC**	**bAcc**
50	3	0.06 (0.08)	0.23 (0.36)	0.08 (0.12)	0.92 (0.14)	**0.99** (0.02)	0.72 (0.22)	0.6 (0.18)
100	3	**0.17** (0.24)	0.31 (0.39)	**0.18** (0.26)	0.91 (0.21)	0.97 (0.03)	0.74 (0.26)	0.64 (0.2)
150	3	0.05 (0.06)	0.19 (0.32)	0.07 (0.09)	0.93 (0.17)	0.97 (0.03)	0.77 (0.23)	0.59 (0.16)
200	3	0.16 (0.31)	**0.36** (0.4)	**0.18** (0.31)	0.87 (0.17)	0.98 (0.03)	**0.81** (0.18)	**0.66** (0.2)
50	4	0.04 (0.12)	0.03 (0.07)	0.03 (0.09)	**0.94** (0.13)	0.97 (0.03)	0.61 (0.23)	0.5 (0.04)
100	4	0.12 (0.24)	0.18 (0.33)	0.14 (0.27)	0.91 (0.13)	0.98 (0.02)	0.73 (0.25)	0.58 (0.17)
150	4	0.07 (0.09)	0.29 (0.41)	0.08 (0.12)	0.9 (0.17)	0.98 (0.04)	0.78 (0.21)	0.63 (0.19)
200	4	0.11 (0.16)	0.32 (0.43)	0.14 (0.2)	0.93 (0.17)	0.98 (0.02)	0.79 (0.22)	0.64 (0.21)

Table 3. Mean (standard deviation) performance metrics over the validation folds for various architectures of the LSTM-based RNNs.

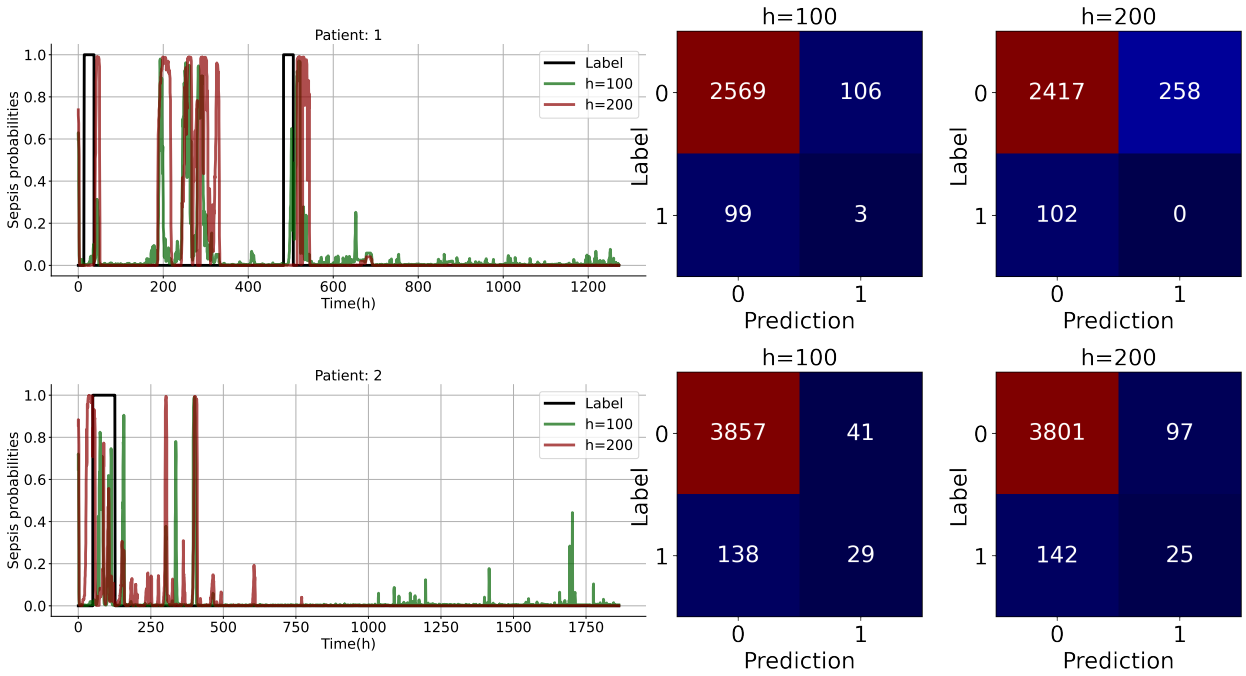

Figure 2. Timeseries of sepsis probabilities and associated confusion matrices for two example patients. We depict the results from two LSTM-based RNN with 3 layers and with hidden size $h = 100, 200$.

until $t = 400$ hours, with multiple reports of apnea and bradycardia. This could explain the high false alarm rate.

The confusion matrices confirm that in both cases, the labeled 0 segments after the last sepsis diagnosis are well modeled with a large majority of the frames correctly labeled 0 by both LSTM-based RNNs.

V. Discussion

The 1-layer vanilla RNN model was not able to leverage the time dependencies in the data. This could be attributed to insufficient model complexity, or vanishing gradient during training. A limitation in the detection task is our labeling method which might not accurately capture septic time periods. One sign of this could be that the LSTM seldom managed to correctly classify the negative samples occurring right after the positive samples. Another limitation of our work is the lack of experiments on an external testing cohorts. This would enable the study the study of the generalization capabilities of the models when predictions are performed on patients from other NICUs.

VI. Conclusion

We showed using an internal cross-validation scheme that LSTM-based RNN had similar performances than LR in terms of overall population scores for neonatal sepsis detection. The increased specificity for non-septic patients however, makes these models less prone to false alarms, in turn less likely to provoke alarm fatigue in clinical wards. This is an important characteristic enabling the deployment of predictive models in clinical practice.

ACKNOWLEDGMENTS

This work was supported in part by grants from KTH Digital Futures, the Stockholm County Council and the Swedish Research Council. This study was approved by the Swedish Ethical Review Authority (2020-02487). The authors thank Daniel Uvehag for his assistance in the administration of our compute environment, as well as Ronny Grover and Lars Särnå for assisting us in the collection of monitoring data.

REFERENCES

[1] S. Chawanpaiboon, J. P. Vogel, A.-B. Moller, P. Lumbiganon, M. Petzold, D. Hogan, S. Landoulsi, N. Jampathong, K. Kongwattanakul, M. Laopaiboon, C. Lewis, S. Rattanakanokchai, D. N. Teng, J. Thinkhamrop, K. Watananirun, J. Zhang, W. Zhou, and A. M. Gülmezoglu, "Global, regional, and national estimates of levels of preterm birth in 2014: A systematic review and modelling analysis," *The Lancet Global Health*, vol. 7, no. 1, pp. e37–e46, Jan. 2019.

[2] J. C. Alverdy and M. A. Krezalek, "Collapse of the Microbiome, Emergence of the Pathobiome, and the Immunopathology of Sepsis," *Crit Care Med*, vol. 45, no. 2, pp. 337–347, Feb. 2017.

[3] R. Joshi, D. Kommers, L. Oosterwijk, L. Feijs, C. Van Pul, and P. Andriessen, "Predicting Neonatal Sepsis Using Features of Heart Rate Variability, Respiratory Characteristics and ECG-Derived Estimates of Infant Motion," *IEEE Journal of Biomedical and Health Informatics*, pp. 1–1, 2019.

[4] J. F. Hicks and K. Fairchild, "Heart rate observation (HeRO) monitoring was developed for detection of sepsis in preterm infants.[...] The HeRO monitor is now in use in many NICUs in the USA and was approved in 2012 for use in Europe." p. 5, 2013.

[5] K. D. Fairchild, D. E. Lake, J. Kattwinkel, J. R. Moorman, D. A. Bateman, P. G. Grieve, J. R. Isler, and R. Sahni, "Vital signs and their cross-correlation in sepsis and NEC: A study of 1,065 very-low-birth-weight infants in two NICUs," *Pediatr Res*, vol. 81, no. 2, pp. 315–321, Feb. 2017.

[6] B. A. Sullivan and K. D. Fairchild, "Vital signs as physiomarkers of neonatal sepsis," *Pediatr Res*, pp. 1–10, Sep. 2021.

[7] C. León, P. Pladys, A. Beuchée, and G. Carrault, "Recurrent Neural Networks for Early Detection of Late Onset Sepsis in Premature Infants Using Heart Rate Variability," *2021 Computing in Cardiology (CinC)*, vol. 48, Sep. 2021, pp. 1–4.

[8] D. E. Lake, J. S. Richman, M. P. Griffin, and J. R. Moorman, "Sample entropy analysis of neonatal heart rate variability," *Am. J. Physiol. Regul. Integr. Comp. Physiol.*, vol. 283, no. 3, pp. R789–797, Sep. 2002.

[9] D. Nabil and F. Bereksi Reguig, "Ectopic beats detection and correction methods: A review," *Biomedical Signal Processing and Control*, vol. 18, pp. 228–244, Apr. 2015.

[10] S. Akselrod, D. Gordon, F. A. Ubel, D. C. Shannon, A. C. Berger, and R. J. Cohen, "Power spectrum analysis of heart rate fluctuation: A quantitative probe of beat-to-beat cardiovascular control," *Science*, vol. 213, no. 4504, pp. 220–222, Jul. 1981.

[11] A. Honoré, D. Forsberg, K. Jost, K. Adolphson, A. Stålhammar, E. Herlenius, and S. Chatterjee, "Classification and feature extraction for neonatal sepsis detection," Mar. 2022.

[12] N. K. Manaswi, "RNN and LSTM," *Deep Learning with Applications Using Python : Chatbots and Face, Object, and Speech Recognition With TensorFlow and Keras*, N. K. Manaswi, Ed. Berkeley, CA: Apress, 2018, pp. 115–126.

[13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, and D. Cournapeau, "Scikit-learn: Machine Learning in Python," *MACHINE LEARNING IN PYTHON*, p. 6.

[14] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *arXiv:1412.6980 [cs]*, Jan. 2017.

[15] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[16] H. Sak, A. Senior, and F. Beaufays, "Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition," Feb. 2014.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," Feb. 2015.

[18] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," p. 12.