Deep Modeling of Longitudinal Medical Data

Baoyu Jing [1] **Huiting Liu** [1] **Mingxing Liu** [1]

Abstract

Robust continuous detection of heart beats from bedside monitors are very important in patient monitoring. The most exiting methods are QSR detectors which are based on electrocardiography (ECG) data. However, ECG data sometime might be very noisy, which will lead to the unreliability of QRS detectors. Recurrent neural networks (RNNs) are designed to model sequential data, and recent studies on RNNs make it possible to use deep learning methods to solve the above problem. Phased-LSTM successfully learns the hidden pattens of the event-based data. RNNs have shown their strength in the problem of sequence labeling and sequence translation. In this paper, we view the heart beat detection problem from two different perspectives: sequence labeling and sequence translation, and design two different deep learning architectures. Since heart beat sequence is event-based data, phased-LSTM is used as basic neuron of our deep learning architectures. As far as we known, no one has explored deep learning methods for the problem.

1. Introduction

Robust continuous detection of heart beats from bedside monitors plays a critical role in patient monitoring (Moody et al., 2014). The most existing heat beat detectors used in hospitals are QRS detectors (Pan & Tompkins, 1985), and usually only operate on electrocardiography (ECG) data which records the electrical activity of the heart over a period of time using electrodes placed on the skin (Bonow et al., 2011). Even though ECG data is reliable in most of the cases, sometimes it might be very noisy and thus not reliable. One way to solve this problem is to build a model which take into account not only the the local information, but also the historical information of ECG data. An alternative way is to leverage the monitoring data from other modalities, such as blood pressure (BP) and pulmonary arterial pressure (PAP). The intuition is that it is rare that monitoring data of all modalities are noisy at the same time.

Few researchers have explored the methods in this area until the 15th PhysioNet challenge. In the challenge, participants were given the task of writing an algorithm to examine an arbitrary multi-channel recording, and produce a series of annotations indicating the likely locations of heartbeats in the recording (Moody et al., 2014). The data used in the challenge contains the signal record for each patient, ranging from four to eight signals. Apart from the ECG signal, other physiologic signals like BP and ART, are provided to assist for robust beat detection.

After investigating the methods proposed by the top teams of the challenge, we divide their work into two types. The first line of work apply different kinds of signal filters to locate the potential heart beat positions for different signal sources, and then develop algorithms to merge the result of different signals (Yang et al., 2014; De Cooman et al., 2014; Vollmer, 2014; Johnson et al., 2014). The other line of work focuses on applying sequential graphical models to find the patterns in the data. Techniques like Hidden Markov Model (HMM) and pattern mining are used (Ghosh et al., 2014; Pimentel et al., 2014).

As far as we have known, no one has explored the deep learning methods for the robust heart beat detection problem. In this paper, we explore the practical possibility of applying deep learning methods to solve this problem. In fact, we can view the heart beat detection problem either as a sequence labeling problem (Graves et al., 2006) or sequence translation (sequence to sequence) (Sutskever et al., 2014) problem. From the perspective of sequence labeling, we can view the heart beat detection problem as a frame-wise classification problem. To be more specific, for a given sequence, each point in the sequence is treated "independently", and the task is to classify each point in the sequence as positive instance (heart beat) or negative instance (not heart beat). From the perspective of sequence translation problem, we can first encode the information of the data sequence into a vector, and then decode the vector into heart beat sequence.

In addition, a key challenge in modeling electronic health record (EHR) data is that the data is usually collected dur-

*Equal contribution [1]School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA. Correspondence to: Baoyu Jing <byjing@cs.cmu.edu>.

10708 Course Project

ing clinical visits and thus the data is sparse and irregularly sampled. Currently, complicated latent variable graphical models, such as CRF (Lafferty et al., 2001), are the most common approaches to accurately model the longitudinal EHR data. However, these graphical model based approaches might over-fit on the massive amount of historical data (Lafferty et al., 2001).

Recurrent neural network(RNN) models are designed to model sequential data(Mikolov et al., 2010; Hochreiter & Schmidhuber, 1997). However, most of the RNNs are unable to model the sparse and irregularly sampled sequential data such as heart beat sequence. Recently, (Neil et al., 2016) has proposed a Phased-LSTM which is able to learn from the these data, and thus suitable for the hear beat detection task. Therefore, Phased-LSTM is used as basic neuron in both of our deep learning architectures.

Based on the above thoughts, firstly, only ECG data is used to detect heart beat. Then multi-modal data, such as BP and ART are used to detect heart beat.

The main contributions of this paper are:

- We explore the application of deep learning methods (sequence labeling and sequence translation) to robust heart beat detection problem.
- Phased-LSTM is used to model electronic data.
- We conduct experiments on not only ECG data, but also on multi-modal data.

2. Related Work

In this section, we first review some previous work focusing on extracting heart beat from multi-modal signals. Then we review two models that serve as the building block of our proposed method.

2.1. Heart Beat Detection

The methods used by the participants in the challenge are two-folds. The first line of work is straightforward. The idea is to use some heuristic method to extract candidate heart beat positions for each "reliable" signal sources, and then try to merge them. Most teams participating the challenge use this kind of method.

Vollmer et al. (Vollmer, 2014) pre-process the data with high pass filtering and range filtering, and then get the threshold in different time windows to identify heart beat. To combine the information from different signals, they first validate the confidence (e.g., to see if it has too much noise) of that signal channel, and then merge the extracted heart beats if there are a sequence of points within 150 milliseconds. De Cooman et al. (De Cooman et al., 2014) first identify the noiseless segment of the ECG and apply existing ECG peak detection algorithm to find the heart beat there. Then they use the heart beats extracted here as golden standard to select other reliable signal sources. After applying peak detection on all confident signal sources, they combine the candidate heart beat positions together with an adaptive time window size.

Johnson et al. (Johnson et al., 2014) develop two efficient merging algorithms to combine the predictions among different signals. They are based on the signal quality indices (SQI) metric and the regularity of the RR interval time-series (REG). The algorithms are both used to detect the noise and artifacts in the wave for one signal. In this way, predictions made inside the noisy part will not be merged. Yang et al. (Yang et al., 2014) only look at the ECG data and the blood pressure data. They first use existing peak detecting tools on the two types of data. Then they try to locate heart beats with clear patterns, e.g., a "sandwich" model where one ECG peak is sandwiched by two blood pressure peaks. In the process, they also learn the average delay between the heart beat and the blood pressure peak so as to predict heart beat when the signal is noisy.

The other line of work uses machine learning and data mining methods that are suitable for fitting sequential data. Pimentel et al. (Pimentel et al., 2014) apply a hidden semi-Markov model (HSMM) on the data. They calculate the slope function value as features from the ECG and BP signal graph. There are two states defined in the model: one point can either be on the QRS complex (a wave that contains a heart beat) or off the segment. They extend the standard HMM by integrating the probability of remaining in a state in the model. This probability is modeled as Gaussian distributions.

2.2. Phased LSTM

Recurrent neural networks (RNNs), such as LSTM, are specifically designed neural networks for modeling sequential data. RNNs have been proved to be powerful for learning the patterns of sequences and generating the sequential data (Hochreiter & Schmidhuber, 1997; Mikolov et al., 2010; Graves, 2013), due to memories they are equipped with. However, traditional RNNs implicitly assume a fixed sampling rate for the input sequential data, which violates the fact that many sequential data is event-based such as heart beat sequence. To deal with this problem, (Neil et al., 2016) recently proposes a Phase-LSTM which extends the LSTM by adding a new time gate. Different from the traditional LSTM, Phased-LSTM adds a new time gate (Figure 1). The opening and closing of the gate k_t is controlled by an independent rhythmic oscillation specified by three parameters; updates to the cell state c_t and h_t are permitted only when the gate is open. Due to the fact that the gates of different neurons are open at irregularly sampled time points. This allows the RNNs to work with event-driven, asynchronously sampled input data. The experiment results show that Phased-LSTM preforms significantly better than the traditional RNNs, such as LSTM and batch-normalized LSTM (Laurent et al., 2016), on event-based data.

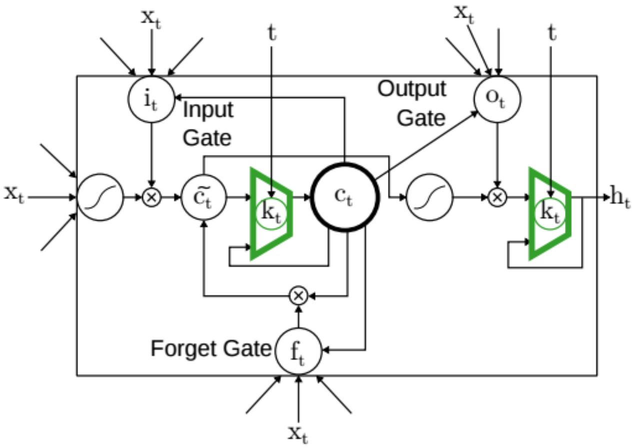

Figure 1. Phased-LSTM model, with time gate k_t controlled by time stamp t.

2.3. RNN based Sequence Labeling

There are two different types of RNN based sequence labeling methods. First one, called frame-wise classification(Graves et al., 2006), assumes each frame in a sequence is "independent" (in the context of RNN, current frame is dependent to the previous frames), and then trains a frame-wise classifier to label the frames in the given sequence. While the other one, called connectionist temporal classification (CTC), interprets the network outputs as a probability distribution over all possible label sequences, conditioned on a given input sequence(Graves et al., 2006). Even though the CTC performs much better than frame-wise classification, CTC takes the entire sequence as input. Since heart beat detection needs a real-time output, thus it is not suitable for hear beat detection task. Therefore, we apply the frame-wise classification approach for the heart beat detection task with phased-LSTM based RNN.

2.4. RNN based Sequence Translation

Sequence translation is also called sequence-to-sequence learning, which takes one sequence as input, then generates a different output sequence. This method is extensively studied in the context of machine translation. (Sutskever et al., 2014)(2) proposes to use multi-layer LSTM to map the input sequence into a vector with a fixed dimensionality, and then use another deep LSTM to translate the information of the vector into the target sequence. (Bahdanau et al., 2014; Graves, 2013) introduces novel differentiable attention mechanism that allows neural networks to focus on different part of the sequence. However, the attention mechanism will significantly increase the training time, thus we only explore the first approach in this paper.

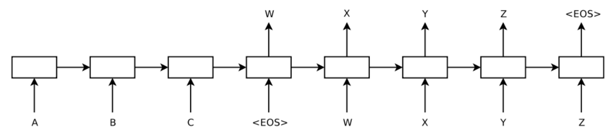

Figure 2. Sequence to sequence model reads an input sentence ABC and produces WXYZ as the output sentence. The model stops making predictions after generating end-of-sentence token.

The original work of (Sutskever et al., 2014) takes the entire sequence as input, and thus not suitable for real-time heart beat detection. In this paper, we propose a new dynamic mechanism for training the model and generating target sequences.

3. Proposed Methods

We view the problem of heart beat detection from two different perspectives: sequence labeling (frame-wise classification) and sequence translation. In either case, we use Phased-LSTM as the basic neuron, since it has been proven to be good at modeling event-based data such as heart beat.

3.1. Sequence Labeling

For a monitoring sequence S, assume that the noisy frames are in the middle or at the end of S. If a model could learn patterns from the non-noisy frames in the front section of S, it should be able to robustly predict the heart beat point from the noisy frames. Fortunately, RNN based sequence labeling models are able to do so.

In this paper, we explore frame-wise classification model. The framework of our model contains only one hidden layer, which is comprised of Phased-LSTM neurons. The output layer is a fully connected layer with sigmoid activation function. The output is treated as probability of each frame to be a heart beat point. The loss of the model is the cross entropy between predicted probability with the ground truth. Figure 3 illustrate the framework of model.

One drawback of this approach is that when predicting the heart beat point, the model might predict many frames around the reference point (human labeled heart beat point) as the heart beat points. This is because the model assumes each frame is "independent" to its following frames. To solve this problem, we cluster the consecutive predicted heart beat points together, and treat the centroids of clusters as the predicted points.

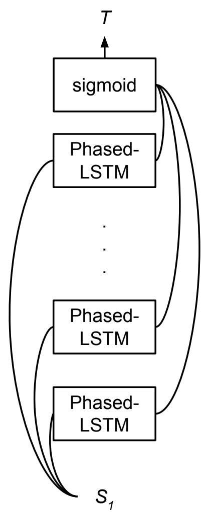

Figure 3. The framework of frame-wise classification model. S and T are input and output respectively.

3.2. Sequence Translation

An alternative approach for heart beat detection problem is sequence translation. We can view a monitoring data sequence (e.g. ECG data) as source sequence, while heart beat sequence as target sequence. The task is to translate the source sequence into target sequence. The framework of sequence-to-sequence model in the context of machine translation first encodes the *entire* source sequence into a vector of fixed number of dimensions, and then decodes the vector into target sequence (Figure 2). However, for heart beat detection problem, the proposed models are required to detect heart beat in real-time.

Thus, we propose to dynamically train the model and translate the sequence. Before we present our idea, it is worth noticing that the source $S = \{S_1, ..., S_N\}$ and target sequence $T = \{T_1, ..., T_N\}$ in the problem of heart beat detection have the same length, where N is the length of entire sequence. Now, our idea is that we can first segment the source and target sequences into sub-sequences of length n:

$$S^{(i)} = \{S_{i \cdot n+1}, ..., S_{(i+1) \cdot n}\}$$

$$T^{(i)} = \{T_{i \cdot n+1}, ..., T_{(i+1) \cdot n}\}$$

Then instead of encoding the information of entire source sequence, we only encode a sub-sequence $S^{(i)}$ of it. Notice that the initial hidden states and cell states of encoder when encoding $S^{(i)}$ are the last output and last cell state of encoder after encoding $S^{(i-1)}$. After encoding the information of $S^{(i)}$ which has the length of n, we decode the output of encoder into $h_j^{(i)}$ (the sequence of outputs of decoder) which also has the length of n. Finally, sequence $h_j^{(i)}$ is transformed into $T_j^{(i)}$ through a sigmoid unit. The loss of the model is also the cross entropy between the predicted probability with the ground truth. The framework of our model is illustrated in the Figure 4.

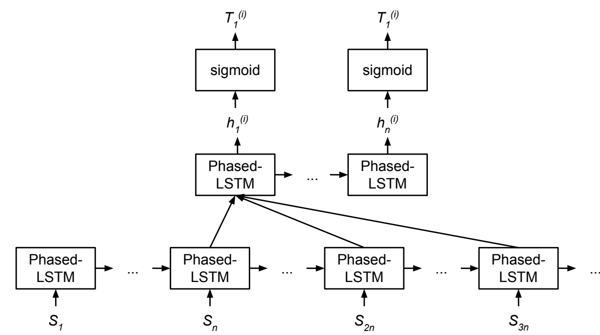

Figure 4. Dynamic sequence to sequence model. S refers to the source sequence, T refers to the target sequence, and h refers to the output sequence of decoder.

This modification allows the model to generate the corresponding heart beat sequence $T^{(i)}$ for sub-sequence $S^{(i)}$ in nearly real time when n is sufficiently small. If we choose $n = 1$, then the model is very similar to frame-wise classification model. If we choose $n = N$, then our model is same as the sequence-to-sequence model used in machine translation. It will be perfect if n is close to the real heart beat rate. We will leave this problem as a future work and in this paper we fix n as the sampling frequency.

This model also suffers from predicting many points around real heart beat points as hear beat. We use the same clustering method mentioned in previous section to tackle with this problem.

4. Experiments

4.1. Experiment Settings

4.1.1. Different Modalities

We evaluate proposed model from two perspectives: single modality and multi-modality. For single modality, we choose ECG data sequence, since it is the most common and informative modality. We conduct experiments on two of our frameworks with Phased-LSTM neurons and LSTM neurons, LSTM based models are baselines for Phased-LSTM models. For multi-modality experiments, we use all of the available modalities in the dataset. Again, we use both Phased-LSTM and LSTM as basic neurons. In addition, we compare our results with original sequence-to-sequence model (without dynamic training and translation) to show the feasibility of sequence-to-sequence model. Besides, we also compare our results with top teams in 15th PyhsioNet challenge.

4.1.2. Details of Models

Firstly, in the experiments of single modality (ECG data): for frame-wise classification model, the hidden layer contains 100 neurons and the dimension for hidden vector is 10; for sequence-to-sequence model, no matter with or without dynamic mechanism, we also use 100 neurons for hidden layer, and the dimension of hidden vector is also 10. In multi-modal experiments, we use same models (as we used for ECG data) for each single modality. Then a fully connected layer is used to combine the outputs of different modalities together.

4.1.3. Evaluation Metrics

In the following experiments, we adopt two metrics defined by the challenge to evaluate the performance. These two metrics are **sensitivity** (commonly known as recall) and **predictivity** (commonly known as precision).

4.2. Dataset Description

We use the dataset provided by 15th PhysioNet challenge (Moody et al., 2014). Currently, only training and augmented training sets are publicly available.[1] The training set contains 100 records, and we randomly choose 20 of them as development dataset and leave the rest as the training set. The augmented training set contains 100 records from the original test dataset. Both training and test datasets contain signals at most 10 minutes in length (or occasionally shorter). The signals are multi-parameter recordings of human adults, including patients with a wide range of problems as well as healthy volunteers. Each signal record contains four to eight signals, the first of which is always an ECG signal. The remaining signals could be any of a variety of simultaneously recorded physiologic signals that might be useful for robust beat detection, such as blood pressure (BP), arterial line (ART), pulmonary arterial pressure (PAP), and respiration (Resp). The signals are digitized at rates between 120 and 1000 Hz; in any given record, however, all signals are sampled at the same, fixed frequency. Figure 5 shows an illustration of the signals in the training dataset:

4.3. Challenges and Preprocessing of the Dataset

4.3.1. ECG data sequence

We observe that there are significant differences between the ECG sequence in the training set and test set. Most

[1] https://physionet.org/challenge/2014/

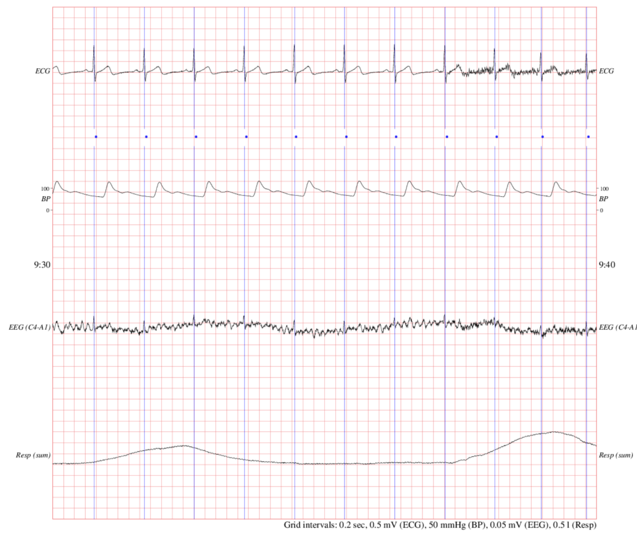

Figure 5. An illustration of electronic signals in the dataset (part of #199 sample in the training set). ECG, BP, EEG and Resp are four different signals it has. The blue dots are human labeled heart beat points (reference points).

of ECG signals in the training set are of very high quality (Figure 6). The differences are reflected on two perspectives. On the one hand, some records in the test set has no ECG signal, as shown in Figure 7. On the other hand, some ECG signals in the test set are extremely noisy (Figure 8). The curves of these signals are very difficult for human to tell which points are the heart beats. Hence, when conducting experiments on single modality (ECG data), we remove the records that don't have ECG signals.

Figure 6. The ECG data is in good shape.

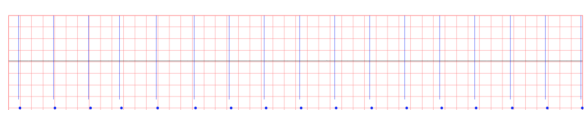

Figure 7. There is no signal in this ECG data.

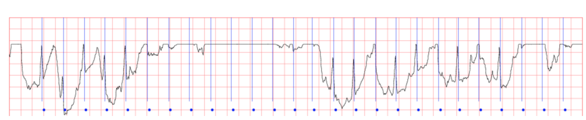

Figure 8. Highly noise ECG data.

4.3.2. Differences of Modalities

The modalities in the test set only have a little overlap with the modalities in the training set. The most commonly shared modality of the two dataset is ECG, as shown in Table 1. In some extreme cases of test dataset, all modalities of them never occur in the training set. This is a very tricky problem for model based machine learning. Generally, it is impossible to train a model on one modality and then test it on an unseen modality. When test models on the test dataset, we only use the modalities that appear in the training set.

Training Set	Test Set
'ECG', 'EMG', 'EOG', 'EOG(right)', 'EEG(C4-A1)', 'EEG(O2-A1)', 'EEG(C3-O1)', 'Resp(abdomen)', 'Resp(abdominal)', 'Resp(chest)', 'Resp(nasal)', 'Resp(sum)', 'SO2', 'SV', 'BP'.	'Nosignal', 'ART', 'LeadAVF', 'LeadII', 'Resp(chest)', 'EMG', 'Pressure2', 'CO2', 'Pressure1', 'ECG2', 'EEG(C3-O1)', 'ECGII', 'ECG', 'EOG(right)', 'EEG(C4-A1)', 'Nothing', 'LeadIII', 'CO2off', 'SV', 'II', 'CVP', 'leadAVL', 'leadII', 'Pressure', 'EOG', 'PAP2', 'PAP', 'BP', 'Pressure4', 'Resp(nasal)', 'CVPoff', 'CVP3', 'Resp.Imp.', 'Resp(abdominal)', 'SO2', 'PAPoff', 'signal', 'ART1','Pressure3', 'NoPAP', 'lead2', 'ECGIII'

Table 1. Different modalities in training set and test set

4.3.3. Different Sampling Frequencies

All the signals in the training set are sampled at the same, fixed frequency, which is 250Hz. However, signals in the test dataset have been sampled at rates between 120 and 1000 Hz. Phased-LSTM is able to deal with this differences, while LSTM is unable to deal with it. It has been shown by (Neil et al., 2016), that Phased-LSTM performs significantly better than LSTM on these kind of data. For fairness, we re-sample the data in the test dataset. The details of re-sampling process is as follows: if the sampling frequency f is less than 250Hz, this means there are less sample data points in one second. We then duplicate each original data point for about $(\frac{250}{f})$ (round) times to make up missing data. If the sampling frequency f is larger than 250Hz, this means there are more sample data points in one second. We then have to re-sample from the original data points. Concretely, We only select the data point with index=$i * (\frac{f}{250})$. One thing to note here is that when we are scaling the data points, we also need to move the labeled heart beat position correspondingly.

4.4. Baseline Results

We list the baseline results (the results of top teams in 15th PhysioNet challenge) here. Their results are based on the multi-modality data. The first two approaches (Johannesen et al., 2014) and (Vollmer, 2014) apply sliding windows and filtering methods to deal with the problem. (Pimentel et al., 2014) uses semi-HMM to deal with the problem.

	Train (S/P)	Test (S/P)
(Johannesen et al., 2014)	99.90/99.90	85.5/88.0
(Vollmer, 2014)	99.90/99.70	91.51/83.43
(Pimentel et al., 2014)	N/A	89.7/83

Table 2. The performance of top teams in PhysioNet challenge. S and P refers to Sensitivity and Predictivity respectively.

4.5. Experiments on Sequence Labeling

First, we run the Phased-LSTM and LSTM model on development dataset. The results are in table 3. Figure 9 offers a intuitive observation: the predicted heart beats match the label data with a high precision and recall. Phased LSTM and LSTM has similar sensitivity but Phased LSTM outperforms LSTM on Predicitivity. It is also in line with our expectation that Phased LSTM performs better than traditional RNNs on event-based data.

Then, we run the Phased-LSTM and LSTM model only with ECG modality on the whole test set except those records without ECG signal. As shown in the 1 & 2 rows in table 3, both Phased-LSTM and LSTM don't perform well in the test dataset because of the reality of the data. The predictivity is caused by a large amount of False positive. The first idea we come up with is to use clustering to reduce the number of positive predictions. But after investigating the results, we find that it is not that case. An typical type is Figure 8. From the figure, we can't find a clear match between the heat beats and the peaks in the signal. The prediction for part of the ECG data for this patient is shown in Figure 10. We can see that the pattern of the predicted signals quite resemble the pattern of the original signals. However, from timetick around 600 to timetick around 1400, we predicted all of them as heart beats, which leads to many false positives.

We also explore the possible improvement with multi-modality. We adopt the full set of modalities in the training set to train the model and force the test data to align with the training modalities by filling the mismatch with a mask value. However, as we mentioned above, the modalities in the test set don't match well with the modalities in the training set. So, we don't expect multi-modal data will help much. The experiment results show that multi-modal data help to improve the sensitivity, while doesn't help for pre-

Dataset Description	Cell Type	Sensitivity (%)	Predictivity (%)
Dev set with 1 modality	Phased-LSTM	97.57	94.96
Dev set with 1 modality	LSTM	97.59	88.68
Test set with 1 modality	Phased-LSTM	93.29	25.83
Test set with 1 modality	LSTM	92.85	31.72
Dev set with multiple modalities	Phased-LSTM	99.69	21.46
Test set with multiple modalities	Phased-LSTM	99.08	21.67

Table 3. Comparison Between Different Models on Different Datasets for Frame-Wise Classification

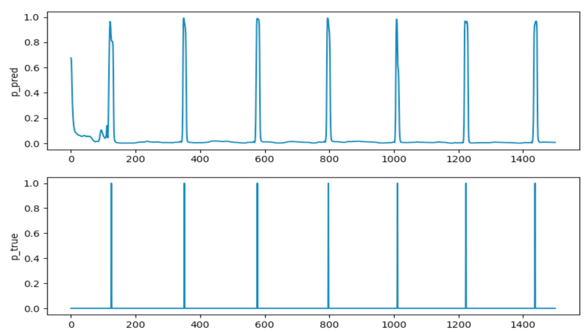

Figure 9. A good prediction matching the labeled data well. The upper plot is our predicted result, and the bottom plot is the labeled heart beat position.

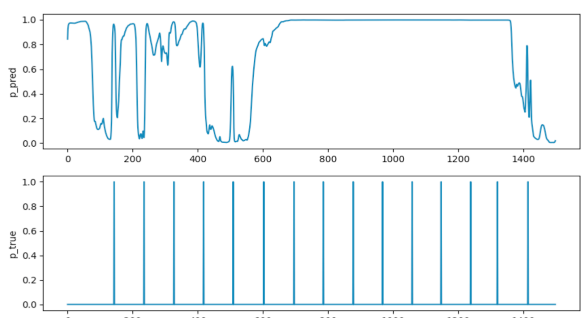

Figure 10. A bad prediction which yields high recall but low precision. The upper plot is our predicted result, and the bottom plot is the labeled heart beat position.

dictivity.

4.6. Experiments on Sequence Translation

In this section, we show the experiment results using the sequence translation methods proposed in Section 3.2. We implemented two models here: the standard Seq2Seq model and the modified Seq2Seq model. The standard Seq2Seq encodes the whole ECG data and decodes to the heart beat positions. As discussed above, this model is not practical for real-time predictions, but can help to get an idea on the performance of the Seq2Seq model. We then show the results of the modified Seq2Seq model, which is suitable for real-time heart beat detection.

The experiment results for the standard Seq2Seq model is shown in Table 4. Similar to the experiments for sequence labeling models, we measured the sensitivity and predictivity on the dev set and test set, using the LSTM Cell and Phased-LSTM Cell respectively. We can see that the metrics on the dev set is again quite good. Also, there is not much difference between the LSTM and Phased-LSTM, with Phased-LSTM has a bit higher predictivity and LSTM has a bit higher sensitivity. However, when the model is applied to the test set, it's a bit surprised to see that both the sensitivity and the predictivity drop dramatically.

We plotted the predicted heart beats and the labeled heart beat for some instances to analyze the reason behind the result. We found that the output of the Seq2Seq model is generally in good patterns, which shows its ability to learn the internal patterns of the ECG data. In good cases, the predicted results match the labeled data quite well (Figure 11). However, in some cases, we found a time lag of the predicted results to the labeled results, as is shown in Figure 12. The prediction is in the same pattern with the real hear beats, but always some distance away. One potential reason is that in the standard Seq2Seq model, we encode the entire ECG data so that we learn the major pattern of the data (e.g., a heart beat every 500ms). In this way, the model will also predict the heart beat using the same major pattern, ignoring any minor deviations. For example, it's possible that in a short period of time, the patient's heart beats every 300ms. This pattern won't be captured by the model, but will influence the distribution of the real heart beat data. If the model still predicts according to the major pattern, there will be time lag in the prediction, and leads to both bad recall and precision.

The modified model can relieve the problem mentioned above, since it encodes the data of only a short period of time, learn the patterns and decodes to output. It will then capture all kinds of the patterns in the whole ECG data, and predict the heart beat accordingly. We post the results for this model in Table 5.

The results for the modified Seq2Seq model is shown in Table 5.

Dataset Description	Cell Type	Sensitivity (%)	Predictivity (%)
Dev set	Phased-LSTM	89.46	90.79
Dev set	LSTM	90.44	87.31
Test set	Phased-LSTM	34.22	38.75
Test set	LSTM	32.17	40.73
Dev set with multiple modalities	Phased-LSTM	92.81	89.33
Test set with multiple modalities	Phased-LSTM	43.32	50.21

Table 4. Comparison Between Different Models on Different Datasets for Standard Seq2Seq Model

Dataset Description	Cell Type	Sensitivity (%)	Predictivity (%)
Dev set	Phased-LSTM	99.71	97.13
Dev set	LSTM	99.62	96.55
Test set	Phased-LSTM	45.21	46.13
Test set	LSTM	45.12	46.6
Dev set with multiple modalities	Phased-LSTM	99.81	99.01
Test set with multiple modalities	Phased-LSTM	50.33	56.23

Table 5. Comparison Between Different Models on Different Datasets for Modified Seq2Seq Model

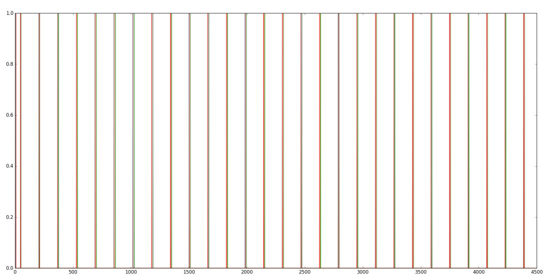

Figure 11. A good prediction matching the labeled data well with the standard Seq2Seq model. The green spikes mark the heart beats predicted, while the red spikes mark the labeled heart beat position.

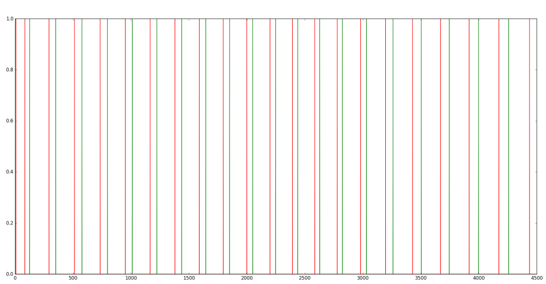

Figure 12. A bad prediction with the standard Seq2Seq model. The green spikes mark the heart beats predicted, while the red spikes mark the labeled heart beat position. The predicted heart beats have a time lag to the ground truth.

5. Conclusion

In conclusion, we explore to look at the problem of heart beat detection from the view of sequence labeling and sequence translation. We also explored the performance of using a new RNN cell, Phased-LSTM to help capture the patterns in ECG data. We deliver some experiment results on the models we tried. We think it will be better to make use of different modalities to improve the performance, which is left as our future work.

References

Bahdanau, Dzmitry, Cho, Kyunghyun, and Bengio, Yoshua. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

Bonow, Robert O, Mann, Douglas L, Zipes, Douglas P, and Libby, Peter. *Braunwald's heart disease: a textbook of cardiovascular medicine*. Elsevier Health Sciences, 2011.

De Cooman, Thomas, Goovaerts, Griet, Varon, Carolina, Widjaja, Devy, and Van Huffel, Sabine. Heart beat detection in multimodal data using signal recognition and beat location estimation. In *Computing in Cardiology Conference (CinC), 2014*, pp. 257–260. IEEE, 2014.

Ghosh, Shameek, Feng, Mengling, Nguyen, Hung, and Li, Jinyan. Predicting heart beats using co-occurring constrained sequential patterns. In *Computing in Cardiology Conference (CinC), 2014*, pp. 265–268. IEEE, 2014.

Graves, Alex. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013.

Graves, Alex, Fernández, Santiago, Gomez, Faustino, and Schmidhuber, Jürgen. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd inter-*

national conference on Machine learning, pp. 369–376. ACM, 2006.

Hochreiter, Sepp and Schmidhuber, Jürgen. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Johannesen, Lars, Vicente, Jose, Scully, Christopher G, Galeotti, Loriano, and Strauss, David G. Robust algorithm to locate heart beats from multiple physiological waveforms. In *Computing in Cardiology Conference (CinC), 2014*, pp. 277–280. IEEE, 2014.

Johnson, Alistair EW, Behar, Joachim, Andreotti, Fernando, Clifford, Gari D, and Oster, Julien. R-peak estimation using multimodal lead switching. In *Computing in Cardiology Conference (CinC), 2014*, pp. 281–284. IEEE, 2014.

Lafferty, John, McCallum, Andrew, Pereira, Fernando, et al. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the eighteenth international conference on machine learning, ICML*, volume 1, pp. 282–289, 2001.

Laurent, César, Pereyra, Gabriel, Brakel, Philémon, Zhang, Ying, and Bengio, Yoshua. Batch normalized recurrent neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pp. 2657–2661. IEEE, 2016.

Mikolov, Tomas, Karafiát, Martin, Burget, Lukas, Cernockỳ, Jan, and Khudanpur, Sanjeev. Recurrent neural network based language model. In *Interspeech*, volume 2, pp. 3, 2010.

Moody, George, Moody, Benjamin, and Silva, Ikaro. Robust detection of heart beats in multimodal data: the physionet/computing in cardiology challenge 2014. In *Computing in Cardiology Conference (CinC), 2014*, pp. 549–552. IEEE, 2014.

Neil, Daniel, Pfeiffer, Michael, and Liu, Shih-Chii. Phased lstm: Accelerating recurrent network training for long or event-based sequences. In *Advances in Neural Information Processing Systems*, pp. 3882–3890, 2016.

Pan, Jiapu and Tompkins, Willis J. A real-time qrs detection algorithm. *IEEE transactions on biomedical engineering*, (3):230–236, 1985.

Pimentel, Marco AF, Santos, Mauro D, Springer, David B, and Clifford, Gari D. Hidden semi-markov model-based heartbeat detection using multimodal data and signal quality indices. In *Computing in Cardiology Conference (CinC), 2014*, pp. 553–556. IEEE, 2014.

Sutskever, Ilya, Vinyals, Oriol, and Le, Quoc V. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pp. 3104–3112, 2014.

Vollmer, Marcus. Robust detection of heart beats using dynamic thresholds and moving windows. In *Computing in Cardiology Conference (CinC), 2014*, pp. 569–572. IEEE, 2014.

Yang, Bo, Teo, Soo Kng, Hoeben, Bart, Monterola, Christopher, and Su, Yi. Robust identification of heartbeats with blood pressure signals and noise detection. In *Computing in Cardiology Conference (CinC), 2014*, pp. 565–568. IEEE, 2014.