Recurrent Neural Network-Assisted Adaptive Sampling for Approximate Computing

Yi Feng
Electrical and Computer Engineering
Duke University
Durham, North Carolina
yi.feng@duke.edu

Yi Zhou
Electrical and Computer Engineering
University of Utah
Salt Lake City, Utah
yi.zhou@utah.edu

Vahid Tarokh
Electrical and Computer Engineering
Duke University
Durham, North Carolina
vahid.tarokh@duke.edu

***Abstract*—We propose an adaptive signal sampling approach that dynamically adjusts the sampling rate to approximate the local Nyquist rate of the signal. The proposed adaptive sampling approach consists of a recurrent neural network-based change detector that detects the point of frequency change and a local Nyquist rate estimator based on a multi-rate signal processing scheme. We empirically demonstrate that our adaptive sampling approach significantly reduces the overall sampling rate for various types of signals and therefore improves the computational efficiency of subsequent signal processing.**

***Index Terms*—Adaptive sampling, approximate computing, neural networks, machine learning, signal processing.**

I. Introduction

The amount of raw information has been growing rapidly in recent years. For instance, online social network platforms such as Facebook, Instagram, etc, generate terabytes of daily data in the forms of text, image and video, making it challenging and computationally burdensome to apply traditional signal processing techniques [1], [2]. However, for most users, a majority of the information is not of interest and blind processing can result in a waste of energy/computational power in the information acquisition and subsequent computations. In order to efficiently retrieve information and process large datasets under limited energy/computational constraints (e.g., limited battery power of wireless and mobile devices), efficient sampling and approximate computing schemes have received much attention in the development of next-generation communication and signal-processing units, producing various trade-offs between computational accuracy and efficiency [3], [4].

The general idea of the proposed adaptive sampling is to increase the device's sampling budget when upcoming information requires more attention. Such an attention-based adaptive sampling strategy can reduce energy consumption by sampling only the information of interest at a faster rate. As an example, when the local bandwidth of a continuous-time signal increases or decreases, one would like to adjust the sampling rate accordingly to sufficiently capture the change of information. Ideally, sampling is kept at the minimum required rate to ensure satisfactory signal reconstruction and reduce the computation for processing the sampled information.

A fundamental quantity for sampling a continuous-time signal $x(t)$ is the Nyquist rate [5], which corresponds to the minimum required sampling rate that ensures successful reconstruction of the original signal. However, sampling according to the Nyquist rate of the *entire* signal is inefficient, as the maximum bandwidth of the entire signal is typically far larger than the local bandwidth [6] of different segments of the signal. On the other hand, sampling according to the local Nyquist rate is challenging in the *online* setting where the signal is a streaming data. This is because one does not have access to the local Nyquist rate of the upcoming signal segment. Hence, in order to efficiently sample the objective signal based on its local Nyquist rate [7], two challenges remain: 1) We need a change-detection model to detect the change of the local Nyquist rate of the signal; 2) We need an efficient approach to determine whether the change is triggered by an increase or a decrease of the local bandwidth, and a proper sampling rate needs to be determined for sampling the signal after the change point.

Modern sampling techniques are designed by exploiting different structures of the underlying signal. For example, compressed sensing has been a very popular sampling technique for sparse signal acquisition [9], [10]. The core of compressed sensing is to perform random sampling under incoherent measurements in the representation basis. Recent works also proposed local coherence-based Bernoulli sampling for compressed sensing and matrix completion tasks [11]–[14]. Meanwhile, change detection for time-series data has a long research history [15], [16], [23], [24]. In particular, with the advances of deep learning [8], [28], time-series data prediction methods based on recurrent neural network (RNN) have been proven to be powerful tools in learning the underlying model and dependency of the data [17], [18]. These have been applied to determining inconsistency in the time-series data, e.g., anomaly detection [19], [20], etc.

In this paper, we propose an adaptive sampling strategy for general persistent signals based on the dynamics of the local Nyquist rate. Specifically:

1) We develop a predictive model based on the Long Short-Term Memory (LSTM) recurrent neural network. The predictive model is trained in an online manner using streaming data and can perform online prediction. The LSTM-based predictive model detects the changes in a signal by measuring the prediction error. The benefit of

this proposed method is that, compared to traditional time-series prediction methods, e.g., auto-regressive models, which require pre-processing to identify and remove the periodic, seasonal and trend components of the signal, our LSTM-based online change detector directly identifies the change points based on the prediction error.

2) Based on the LSTM-based change detector, we further propose a multi-stream sampling method to identify the lowest sufficient sampling rate for the signal after the change point in order to improve the sampling and computational efficiency.

The outline of the paper is given next. In Section II, we establish the notation, review relevant background and mathematically formulate the problem to be considered in this paper. Section III presents the proposed LSTM-based change detector. Section IV describes our multi-stream sampling mechanism for identifying the appropriate sampling rate for the signal following the detection of a change point and quantifies the performance of the proposed adaptive sampling strategy. In Section V, we make our conclusions and discuss potential future directions of research.

II. Preliminaries

A. Nyquist Sampling Rate and Signal Reconstruction

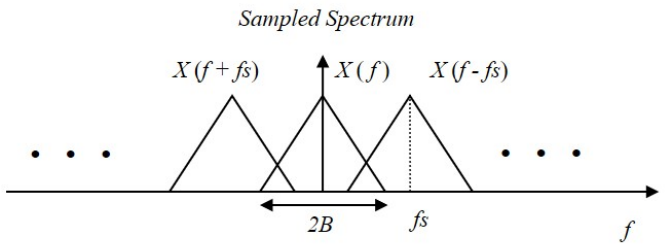
As illustrated in Figure 1, to recover a bandlimited signal $x(t)$ with one-sided bandwidth B Hz using discrete samples, the sampling rate needs to be at least $2B$ samples per second (in order to avoid aliasing). This is the well-known Nyquist sampling rate and is widely used in the design of modern digital signal processing devices for analog-to-digital conversion. Under the Nyquist sampling rate, replicas of the original signal due to sampling, e.g., $X(f)$, $X(f-f_s)$, $X(f+f_s)$, ... , do not overlap in the frequency domain, and therefore the original signal can be perfectly reconstructed (interpolated) using the low-pass filtering

$$x_{int}(t) = \sum_{n=-\infty}^{\infty} x_n \text{sinc}\Big(\frac{t-nT}{T}\Big). \tag{1}$$

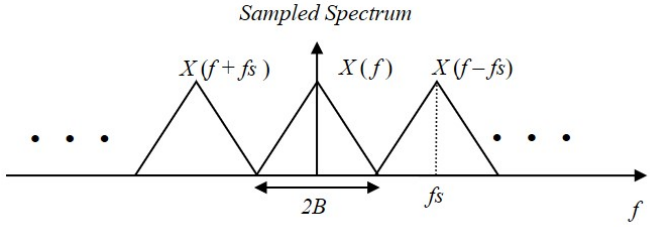
Since in practice we only have finite amount of samples, spline interpolations [30] are often used to construct an approximation of the signal. Spline interpolation uses piece-wise polynomials to approximate the objective function. In particular, for cubic interpolation splines, the interpolated function is given by piece-wise cubic functions between adjacent knots as

$$\widehat{x}_{int}(t) = \begin{cases} a_1 + b_1 t + c_1 t^2 + d_1 t^3, & \text{if } t \in [t_0, t_1] \\ a_2 + b_2 t + c_2 t^2 + d_2 t^3, & \text{if } t \in [t_1, t_2] \\ \quad \vdots & \\ a_n + b_n t + c_n t^2 + d_n t^3, & \text{if } t \in [t_{n-1}, t_n] \end{cases} \tag{2}$$

where the coefficients of the interpolated functions produce a signal $\widehat{x}_{int}(t)$ that is continuous and has continuous first and second derivatives for all the points in the interval.

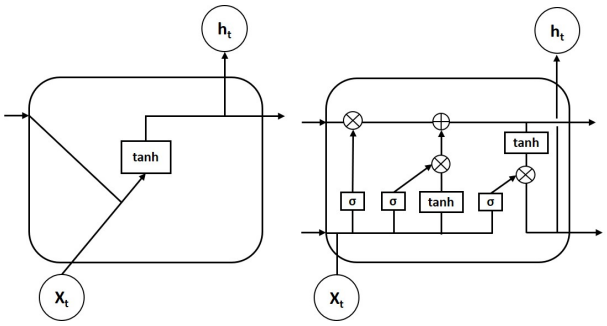


(a) $f_s < 2B$



(b) $f_s = 2B$ (Nyquist rate sampling)

Fig. 1. Illustrations of the spectra for a sampled continuous-time signal at different sampling rate f_s.



(a) RNN cell (b) LSTM cell

Fig. 2. Comparison between an RNN cell and an LSTM cell.

B. Long Short-Term Memory Recurrent Neural Network

Recurrent neural networks (RNNs) are well known to have the capability to learn and predict from historical data. The structure of RNN contains recurrent loops that iterate over the input data sequentially using the same set of parameters. This structure is useful in capturing the dependencies among the historical data and is particularly useful in building prediction models that predict from the past. In practice, however, the simplest form of RNNs cannot handle long-term dependencies of the data, and the LSTM-networks are developed for this purpose. We compare the structures of an RNN cell and an LSTM cell in Figure 2.

Instead of having a single neural network layer, as in the vanilla RNN, LSTM networks consists of cells that include four neural network layers that interact in a specialized way in order to have long-term memories as described by the following set of equations:

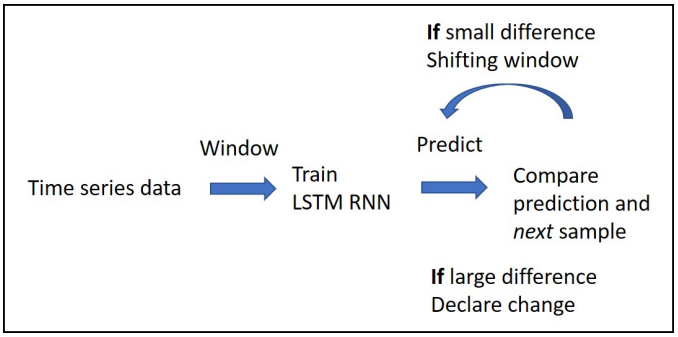


Fig. 3. LSTM-based online change detection flow chart.

$$
\begin{aligned}
f_t &= \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right) \\
i_t &= \sigma\left(W_i x_t + U_i h_{t-1} + b_i\right) \\
o_t &= \sigma\left(W_o x_t + U_o h_{t-1} + b_o\right) \\
c_t &= f_t * c_{t-1} + i_t * \tanh\left(W_c x_t + U_c h_{t-1} + b_c\right) \\
h_t &= o_t * \tanh\left(c_t\right)
\end{aligned} \tag{3}
$$

where * denotes the Hadamard product. In particular, the design of LSTM cell avoids the vanishing gradient problem, which is a bottleneck in training simple RNNs. Besides LSTM networks, there are many other variants of LSTM networks that adopt cells with different network structures, e.g., GRU networks [27].

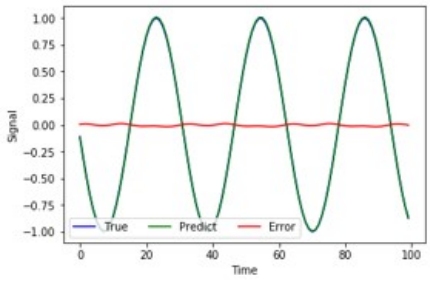
III. LSTM-Assisted Change Point Detector

Consider a continuous-time signal with a change point at time t_c where the local bandwidth changes from B_1 to B_2. Our objective is to perform fast online detection in order to find the change points based on discrete samples.

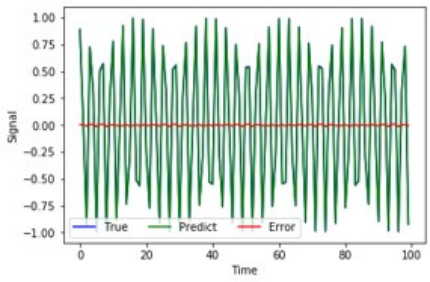
We consider an LSTM-based online change detector for dynamically detecting the change point of the local bandwidth of general bandlimited signals, as illustrated in Figure 3. Specifically, we use an stacked LSTM with two layers and both layers have M features in the hidden states. Our LSTM-based online change detector can be presented using the following steps:

- **Step 1:** Fix a sampling rate f_s and obtain W samples of the signal. Train the LSTM with an input size much smaller than W using the obtained samples. The loss function is the mean-square error loss.
- **Step 2:** Apply the trained LSTM network to predict the next sample of the signal.
- **Step 3:** Observe the next sample of the signal and compare it with the predicted value from the LSTM network. Declare a change if the accumulated L_2-norm of the values of differences between the predicted samples and the corresponding observed values in a selected window is larger than a certain threshold.

We first present the predictive performance of the LSTM-based predictor with $M = 100$ for a bandlimited signal without any change points. Consider a sinusoidal signal with

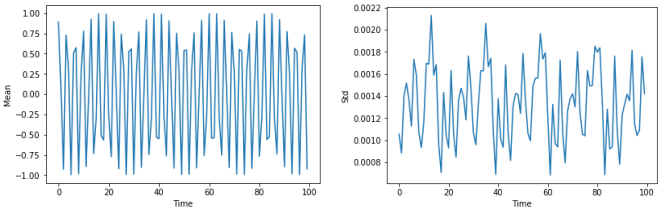


(a) Three periods, 30 samples per period.



(b) Thirty-two periods, 3 samples per period.

Fig. 4. Performance of the LSTM-based predictor for sine functions.



(a) Mean value of predictions. (b) Standard deviation of predictions.

Fig. 5. Prediction performance of LSTM under different random initializations.

frequency f_1. The goal is to achieve a good prediction performance based on the samples of the signal. Figure 4 (a) and (b) show the predictive performance of the LSTM network trained on dense samples (about 30 samples per period) and sparse samples (about 3 samples per period), respectively. The error signal shown in red is the difference between the predicted samples and the corresponding observations of the signal. It can be seen from Figure 4 that the error signals have close-to-zero mean and standard deviation (std), and the predicted and true signals appear indistinguishable. This demonstrates that the proposed LSTM-based predictor has a good performance in predicting stable and periodic signals.

In addition, we present the performance of the LSTM-based predictor under multiple initialization points that are generated according to the same Gaussian distribution. Figure 5 (a) shows the mean value curve of the predicted signals obtained from the LSTM based on 16 distinct initialization points, and (b) further presents the standard deviation of the prediction curves. From the standard deviation curve, it can be seen that the performances of the LSTM predictor under

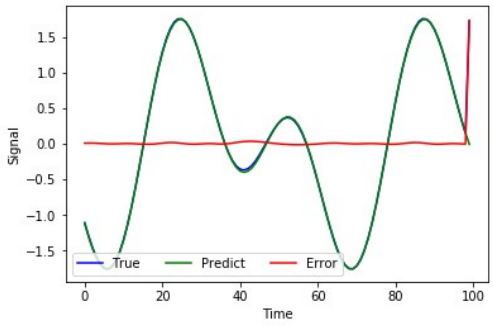


Fig. 6. Change detection via LSTM-based predictor. Last 100 time instances before signal changes where the error signal (in red) shoots up.

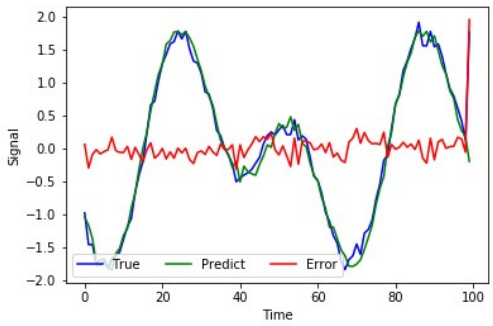


Fig. 7. Robustness of the LSTM-based change detector under AWGN.

different realizations of the initialization are all superior and indistinguishable, suggesting the same statistical trend among different initializations.

In Figure 6, we apply the LSTM to predict a band-limited signal that has multiple frequency components and has a change in signal bandwidth. Specifically, the signal changes from $sin(2\pi f_1 t)+sin(4\pi f_1 t)$ to $2sin(8\pi f_1 t)$ at time instance $t_c = 100$. Before the change point t_c, LSTM can accurately predict the signal waveform and the error signal is very small, i.e., mean value of -0.004, standard deviation of 0.010. Right after the change point, the newly obtained sample deviates from the predicted value, producing an error signal of sufficient amplitude to confidently detect the change point.

As noise perturbations degrade the performance of many machine learning models, we test the robustness of this LSTM-based predictor to the injection of additive white Gaussian noise (AWGN) into the signal shown in Figure 6. Figure 7 shows the result where the observed samples and new samples are corrupted by an AWGN with zero mean and standard deviation of 0.1. The result demonstrates the robustness of the LSTM-based change detection to Gaussian noise, and the change point at $t_c = 100$ is still successfully detected. In particular, the error signal before the change point has empirical mean -0.002 and standard deviation 0.125, which is comparable to the added AWGN samples that have zero mean and standard deviation 0.1, implying the robustness of the proposed LSTM-based model to Gaussian noise.

IV. Multi-Stream Signal Sampling and Processing

A. Multi-Stream Processing for Sampling Rate Identification

After change point detection, we use an attention-based mechanism to sample at the highest allowed rate. Newly obtained data is used to train the LSTM network to predict the signal after the change point. We then determine the lowest sufficient sampling rate that can recover the original continuous-time signal.

We assume that the signal is bandlimited and the maximum sampling rate available to our adaptive sampler is f_{max}. Our proposed algorithm involves multi-stream data processing, which consists of the following key steps:

- **Step 1:** Sample at rate f_{max} for a "short while" in order to fully train the LSTM neural network. These samples are referred to as the observed values.
- **Step 2:** Sub-sample the obtained data sample stream in Step 1 at $K-1$ reduced sampling rates, e.g., $f_{max}\cdot 2^{-1}$, $f_{max}\cdot 2^{-2}$, $f_{max}\cdot 2^{-3}$, ..., $f_{max}\cdot 2^{-K+1}$. This results in total K data streams that are sampled at different rates.
- **Step 3:** Interpolate the $K-1$ data streams with reduced sampling rates obtained in Step 2 to the highest sampling rate f_{max} using the cubic spine interpolation method.
- **Step 4:** Compute the point-wise difference e_{jn} between the interpolated data sample of the j-th stream at time instance n and that of the corresponding observed value.
- **Step 5:** Among the stream indices $j = 1, ..., K$, output the stream index j that minimizes the objective function:

$$\sum_{n=N}^{N+d-1} e_{jn}^2 + \lambda f_{max}\cdot 2^{-j+1} \tag{4}$$

where d is the number of points which we compare to observed values, and λ is a tuning parameter. Finally, adopt the new sampling frequency as $f_{max}\cdot 2^{-j+1}$ until the next change point is detected.

The tuning parameter λ provides a trade-off between minimizing the cumulative squared error over time and minimizing the selected sampling rate. For smaller λ values, the sampler tends to over-sample in order to minimize the error of predictions, whereas larger λ values give higher penalties for selecting higher sampling rates than actually needed. Figure 8 illustrates this procedure for identifying a suitable sampling rate.

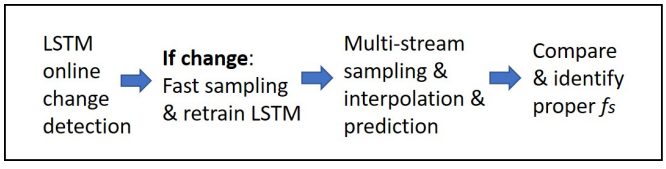


Fig. 8. LSTM-assisted multi-stream signal processing flow chart for adaptively identifying the proper sampling rate.

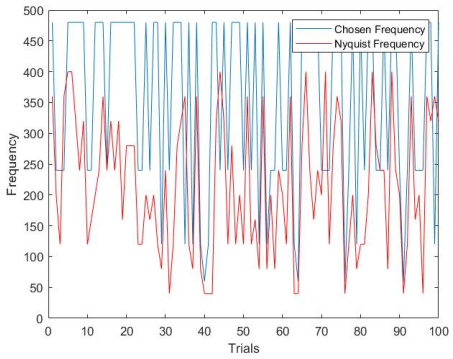


Fig. 9. Performance of the sampling frequency selection in the new segment after a change point over many trials.

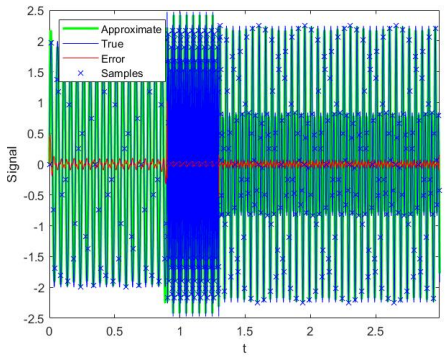


Fig. 10. LSTM-assisted adaptive sampling and signal reconstruction for a signal with large variations in local bandwidths. The proposed adaptive sampling scheme provides significant saving in number of sampling operations.

B. Performance Evaluation and Discussion

We now present various experiments to evaluate the performance of our adaptive sampling approach. We first generate 100 random signals with two signal segments and change time $t_c = 0.49$ s. The different local bandwidths are drawn uniformly from the set {20, 40, 60, ..., 200} Hz at random, and

$$x(t) = \begin{cases} A_1 sin(2\pi f_1 t) + A_2 sin(2\pi f_2 t), & \text{if } t \in [0, t_c] \\ A_3 sin(2\pi f_3 t) + A_4 sin(4\pi f_4 t), & \text{if } t \in [t_c, 1] \end{cases} \quad (5)$$

where A_1, A_2, A_3, A_4 are drawn uniformly from the interval [-2,2]. We let the adaptive sampler choose rate from the set {30, 60, 120, 240, 480} Hz, and we set the regularization parameter λ in eq. (4) to 0.002.

We trained the LSTM network with multiple different initialization models. Similar to the observations made in Figure 5 for training the LSTM-based predictor, we observe same statistical trend and the rate selection appears identical for the same signal after training using different initializations.

In Figure 9, we show the sampling rate chosen by our adaptive sampler for the second segment of the considered signals correspondingly. One can see from the figure that our adaptive sampler chooses a sufficient sampling frequency for 95% of the total number of signals.

In Figure 10, we apply our adaptive sampler to sample a signal with multiple change points and test its performance. Here we consider a signal that consists of three segments as follows

$$x(t) = \begin{cases} sin(2\pi f_1 t) + sin(2\pi f_2 t), & \text{if } t \in [0, t_{c1}] \\ 2sin(2\pi f_3 t) + 0.5sin(4\pi f_4 t), & \text{if } t \in [t_{c1}, t_{c2}] \\ 1.5sin(2\pi f_5 t) + sin(4\pi f_6 t), & \text{if } t \in [t_{c2}, 3] \end{cases} \quad (6)$$

where $f_1 = 20$ Hz, $f_2 = 20$ Hz, $f_3 = 120$ Hz, $f_4 = 100$ Hz, $f_5 = 40$ Hz, $f_6 = 20$ Hz, and the change points are at $t_{c1} = 0.9$ s, $t_{c2} = 1.3$ s.

The signal bandwidth significantly increases at the first change point and decreases at the second change point. The maximum allowed sampling rate is $f_{max} = 480$ Hz. The results are shown in Figure 10, where the markers correspond to the samples, the green curve corresponds to the reconstructed signal via interpolation, and the blue curve denotes the true signal with time increment $\Delta t = 0.001$ s. It can be seen that the error signal is very small except for around the change points. In particular, when excluding the error near the change points, the error has zero mean and standard deviation about 0.032. Furthermore, we compare the proposed adaptive sampling method with the traditional Nyquist sampling. In this experiment, our adaptive sampler requires only about 40% of the total samples, compared to the traditional Nyquist sampling. In particular, instead of sampling the entire signal at the highest rate corresponding to the signal segment $t \in$ [0.9, 1.3] as would be performed by the traditional Nyquist sampler, our adaptive sampler exploits the proposed LSTM prediction model to dynamically adjust the local sampling rate in each segment and leads to significant reduction in sampling complexity.

It is also important to explore the robustness of the proposed method to small frequency perturbations. In Figure 11, we consider a signal whose local bandwidths of its two segments change from 38 Hz to 61 Hz at the change point t_c = 1.5, e.g., we perturb the considered local bandwidths of the two signal segments 40 Hz and 60 Hz, by 2 Hz and 1 Hz, respectively. It can be seen from the figure that the LSTM-based change detector in our adaptive sampler can detect the change point of the signal under the existence of the local bandwidth perturbations. In particular, the number of samples sampled by our adaptive sampler is about 80% of those sampled by the traditional Nyquist sampling for this experiment.

We further explore the effectiveness of our adaptive sampler in sampling the signals that have very similar signal segments. In Figure 12, the local bandwidth of the sinusoidal signal changes from 21 Hz to 22 Hz at time t_c = 0.95 s. Since the change of the signal is small, it is reasonable to use a smaller threshold value for the purpose of successful change point detection. Otherwise, the adaptive sampler could miss the change point and the corresponding performance is shown in (a). In this case, the adaptive sampler does not save samples. As the allowed sampling rate in our design is discretized,

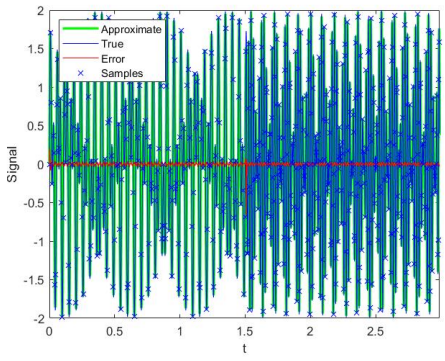
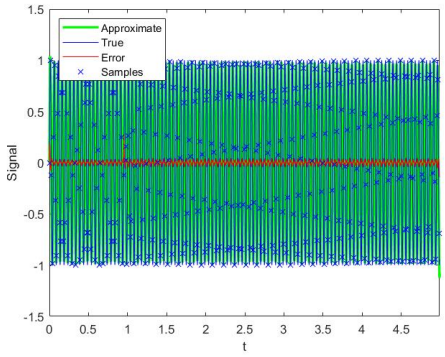
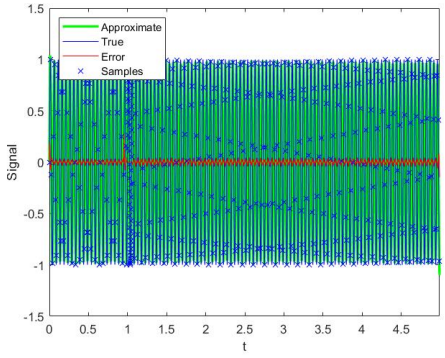


Fig. 11. LSTM-assisted adaptive sampling and signal reconstruction for a signal having two segments with a moderate local bandwidth change.



(a)



(b)

Fig. 12. LSTM-assisted adaptive sampling and signal reconstruction for a signal having two segments with very small change. The adaptive sampling scheme does not provide saving in sampling operations.

the reconstruction error will still remain relatively small. For a smaller threshold value, the proposed scheme may not immediately detect the change and hence requires additional samples. In fact, due to the fast sampling performed after the change point detection, the adaptive sampler samples more as shown in (b), compared to the samples obtained by using the traditional Nyquist rate sampling.

From the above experiments, we can see that the proposed LSTM-assisted adaptive sampling scheme will in general provide great saving in sampling operations when the change of the local bandwidth is dramatic, e.g., we can expect a huge amount of sample savings in sampling signals that have narrow bandwidth in most of the signal segments and wide bandwidth in certain signal segments. The total sample complexity is determined by the variations of local bandwidths and the duration of the segments.

V. Conclusion

In this paper, we propose an LSTM-assisted adaptive sampling strategy that can improve the sampling efficiency by introducing certain error tolerance as a type of approximate computations. This method dynamically adjusts the sampling rate when a change point is encountered in the signal. It operates on a need-to-sample basis and samples more only when needed. Our future work includes investigating the computation efficiency and trade-offs of this proposed adaptive sampling scheme for various operation-intensive applications in order to amortize the computation cost induced by the LSTM network and multi-stream processing, as well as deriving theoretical bonds and guarantees for the approximate computation error.

Acknowledgment

This project is supported by National Science Foundation Grant ECCS-1848810. Yi Feng would like to acknowledge the support of the postdoctoral research fellowship from Fonds de recherche du Quebec - Nature et technologies.

References

[1] G. B. Giannakis, F. Bach, R. Cendrillon, M. Mahoney, J. Neville, "Signal Processing for Big Data," *IEEE Signal Processing Magazine*, pp. 15-16, January 2014.

[2] S. Bi, R. Zhang, Z. Ding, S. Cui, "Wireless Communications in the Era of Big Data," *IEEE Signal Processing Magazine*, pp. 190-199, October 2015.

[3] S. Mittal, "A Survey Of Techniques for Approximate Computing," *ACM Computing Surveys*, Article No. 62, pp. 1-34, May 2016.

[4] A. Agrawal, J. Choi, K. Gopalakrishnan, S. Gupta, R. Nair, J. Oh, D. A. Prener, S. Shukla, V. Srinivasan, Z. Sura, "Approximate Computing: Challenges And Opportunities," in Proc. *IEEE International Conference on Rebooting Computing*, pp. 1-8, San Diego, CA, October 2016.

[5] H. Nyquist, "Certain Topics in Telegraph Transmission Theory," *Transactions of the American Institute of Electrical Engineers*, pp. 617-644, April 1928.

[6] B. Boashash, "Estimating and Interpreting the Instantaneous Frequency of a Signal-Part 2: Algorithms and Applications," *Proceedings of the IEEE*, pp. 520-538, April 1992.

[7] S. Feizi, V. K. Goyal, M. Medard, "Locally Adaptive Sampling," in Proc. *IEEE Annual Allerton Conference on Communication, Control, and Computing*, pp. 152-159, Urbana, IL, September-October 2010.

[8] X. -W. Chen, X. Lin, "Big Data Deep Learning: Challenges and Perspectives," *IEEE Access*, pp. 514-525, May 2014.

[9] D. Donoho, "Compressed Sensing," *IEEE Transactions on Information Theory*, pp. 1289-1306, April 2006.

[10] E. J. Candès, T. Tao, "Near-Optimal Signal Recovery From Random Projections: Universal Encoding Strategies?" *IEEE Transactions on Information Theory*, pp. 5406-5425, December 2006.

[11] Y. Zhou, H. Zhang, Y. Liang, "On Compressive Orthonormal Sensing," in Proc. *IEEE Annual Allerton Conference on Communication, Control, and Computing*, pp. 299-305, Urbana, IL, 2016.

[12] Y. Zhou, Y. Liang, "Demixing Sparse Signals via Convex Optimization," in Proc. *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4202-4206, New Orleans, LA, March 2017.

[13] H. Zhang, Y. Zhou, Y. Liang, "Analysis of Robust PCA via Local Incoherence," in Proc. *International Conference on Neural Information Processing Systems*, pp. 1819-2827, Montreal, QC, October 2015.

[14] Y. Chen, "Incoherence-Optimal Matrix Completion," *IEEE Transactions on Information Theory*, pp. 2909-2923, May 2015.

[15] M. Basseville, I. V. Nikiforov, *Detection of Abrupt Changes: Theory and Application*, Prentice Hall, 1993.

[16] S. Aminikhanghahi, D. J. Cook, "A Survey of Methods for Time Series Change Point Detection," *Knowledge and Information Systems*, pp. 339-367, May 2017.

[17] J. T. Connor, R. D. Martin, L. E. Atlas, "Recurrent Neural Networks and Robust Time Series Prediction," *IEEE Transactions on Neural Networks*, pp. 240-254, March 1994.

[18] T. Guo, Z. Xu, X. Yao, H. Chen, K. Aberer, K. Funaya, "Robust Online Time Series Prediction with Recurrent Neural Networks," in Proc. *IEEE International Conference on Data Science and Advanced Analytics*, pp. 816-825, Montreal, QC, October 2016.

[19] S. Chauhan, L. Vig, "Anomaly Detection in ECG Time Signals via Deep Long Short-Term Memory Networks," in Proc. *IEEE International Conference on Data Science and Advanced Analytics*, pp. 1-7, Paris, France, October 2015.

[20] P. Malhotra, L. Vig, G. Shroff, P. Agarwal, "Long Short Term Memory Networks for Anomaly Detection in Time Series," in Proc. *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pp. 89-94, Bruges, Belgium, April 2015.

[21] S. Hochreiter, J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, pp. 1735-1780, November 1997.

[22] Q. Lyu, J. Zhu, "Revisit Long Short-Term Memory: An Optimization Perspective," in Proc. *Advances in Neural Information Processing Systems Workshop on Deep Learning and Representation Learning*, pp. 1-9, December 2014.

[23] R. P. Adams, D. J. MacKay, "Bayesian Online Changepoint Detection," *arXiv preprint arXiv:0710.3742*, pp. 1-7, October 2007.

[24] J. Ding, Y. Xiang, L. Shen, V. Tarokh, "Multiple Change Point Analysis: Fast Implementation and Strong Consistency," *IEEE Transactions on Signal Processing*, pp. 4495-4510, June 2017.

[25] J. Ding, J. Zhou, V. Tarokh, "Asymptotically Optimal Prediction for Time-Varying Data Generating Processes," *IEEE Transactions on Information Theory*, pp. 3034-3067, November 2018.

[26] L. F. S. M. Qaisar, M. Renaudin, "Computationally Efficient Adaptive Rate Sampling and Filtering," in Proc. *European Signal Processing Conference*, pp. 2139-2143, Poznan, Poland, September 2007.

[27] K. Cho, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," in Proc. *Conference on Empirical Methods in Natural Language Processing*, pp. 1724-1734, Doha, Qatar, October 2014.

[28] I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*, MIT Press, 2016.

[29] J. G. Proakis, D. K. Manolakis, *Digital Signal Processing: Principles, Algorithm, Applicaitons*, Pearson Press, 2007.

[30] J. H. Ahlberg, E. N. Nilson, J. H. Walsh, *Theory of Splines and Their Applications*, Academic Press, 1967.

[31] F. Marvasti, *Nonuniform Sampling: Theory and Practice*, Springer, 2001.

[32] J. Durbin, S. J. Koopman, *Time Series Analysis by State Space Methods*, Oxford University Press, 2012.