The 12th International Conference on Ambient Systems, Networks and Technologies (ANT)
March 23-26, 2021, Warsaw, Poland

Long Short-Term Memory Approach for Routing Optimization in Cloud ACKnowledgement Scheme for Node Network

Siddardha Kaja,[a,]* Elhadi M. Shakshuki,[a] Ansar Yasar[b]

[a]Jodrey School of Computer Science, Acadia Unversity, Wolfville, Nova Scotia, B4P2R6, Canada
[b]Transportation Research Institute, B-3500 Hasselt, Hasselt University, Belgium

Abstract

Routing optimization using machine learning has been receiving a lot of attention recently. Additionally, cloud computing is evolving exponentially in processing power and memory units. This paper proposes a routing optimization approach for a Cloud ACKnowledgement Scheme using machine learning techniques. Our proposed approach is based on synthetic generated data for respective node values in a network. Moreover, it involves a variant of Recurrent Neural Network called Long Short-Term Memory (LSTM). The machine learning model is developed using LSTM through a sliding-window technique. The results achieved are very encouraging. They show that the cloud can mostly predict whether the forthcoming transmission of a certain node in the network will be a success.

Keywords: Machine Learning; Long Short-Term Memory; Acknowledgement Scheme; Acknowledgement Packets; Routing

1. Introduction

Machine Learning (ML) is a trending and widely adopted field of research in Artificial Intelligence. It is utilized in an increasing variety of domains, since ML automatically learns and makes decisions by identifying the data pattern in the collected data without human interference [2]. Novel and powerful large-scale data processing techniques make ML useful within a variety of fields [7]. Furthermore, the advancement of cloud security mechanisms such as trust evaluation fosters additional confidence regarding applying the cloud in different techniques [18][19][20]. In this

---

* Corresponding author. Tel.: 1-902-585-1524.
   Email address: 152576k@acadiau.ca

paper, we take advantage of ML and cloud by incorporating several ML techniques into a Cloud ACKnowledgement Scheme (CACKS) [1][3]. CACKS is a novel scheme that is designed to utilize acknowledgement packets to keep track of the transmissions across a node network. It utilizes the cloud as a monitoring and interacting tool in the network. This scheme also enables the cloud to provide backup to nodes that are affected by a failed transaction. The new optimized route is provided to the node only if the node requests the backup.

Acknowledgement packets (ACK) are generally used by TCP for successful transmission of data [6]. A node recognizes a sequence of TCP packets with the aid of ACK. Every ACK packet is attached with the sequence number of the next data packet in the route. Sufficient data in relation to an ACK packet is also embedded in that ACK packet [4]. In CACKS, we propose a modified ACK packet structure that includes the current node and the forwarding nodes' addresses in both packets of the source and destination. Additionally, CACKS proposes a Negative Acknowledgement packet (NAK) that includes the addresses of all neighboring nodes [4].

Long Short-Term Memory (LSTM) networks are a type of recurrent neural network that is capable of learning the dependency of the order of a single or multiple sequence [14]. Figure 1 shows the LSTM architecture. In this architecture, an RNN cell A takes in a given input sequence of elements X_t value-by-value and calculates the output h_t by combining the input element with a hidden latent state, which is updated for use by the next element. A, h, and X are defined.
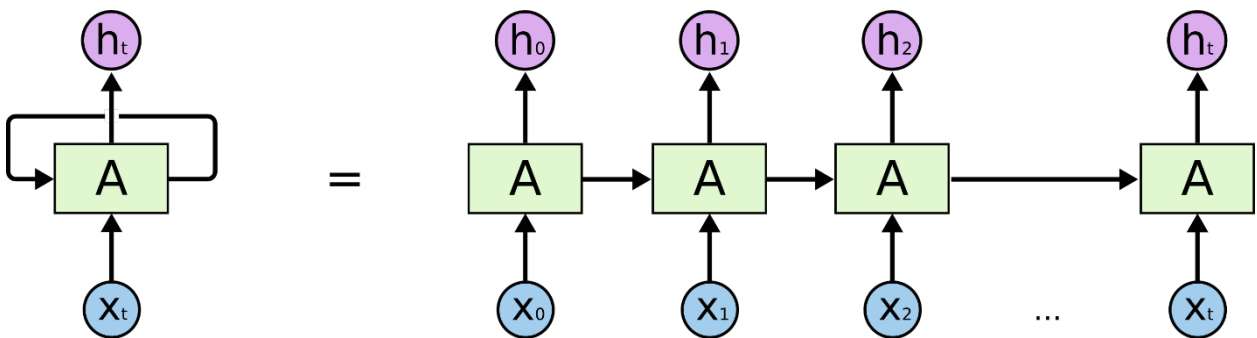

Fig. 1. LSTM Architecture [14].

The architecture of an LSTM cell is shown in Figure 2. h_{t-1} is the straight line at the top of the cell. It is coming from the earlier cell and carrying its state from the previous cell. While h_t is the new state of the cell, x_t is the new input value for the cell. The LSTM continuously synchronises the containing elements using three gates: r_t, z_t, h-bar$_t$. These gates regulate how they behave as long-term or short-term memory cells. An LSTM cell contains an input gate, an output gate, and a forget gate through which the information flows. However, the modern new LSTM cell architecture combines the input gate and the forget gate into a single update gate [14]. This formation allows the cell to remember all of the values across unpredictable time steps, making the LSTM well-suited to tackle the challenge of analyzing and predicting the time series data.
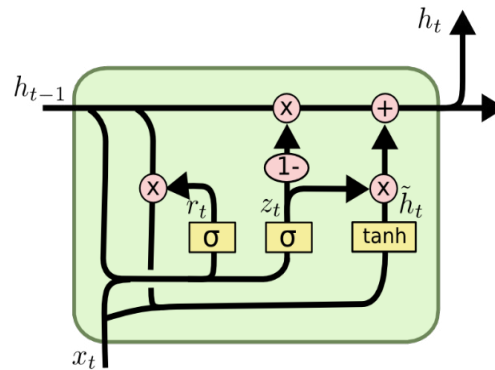
Fig. 2. LSTM Cell Architecture [14].

Our current approach gives an understanding of the ML technique employed by the cloud in this scheme. CACKS is applied to obtain a machine learning model that can predict the next best choice for the affected node by a transaction failure. The best choice among the neighbours of the affected node gives the next forwarding node. The ML technique used in this approach aims to analyze the patterns in failures of the nodes in the network. It also predicts the neighbour that is most likely to transmit a data packet successfully. Generally, all of the data from acknowledgement packets is overlooked after a data transmission [9]. However, in our research, all of the data from acknowledgement packets is processed to obtain information about a particular network.

2. Cloud ACKnowledgement Scheme Approach

CACKS proposes that acknowledgment packets are collected and utilized by the cloud. Figure 3 describes our proposed Cloud ACKnowledgement Scheme (CACKS). This sample network contains a cloud that is connected to each of the nodes in the network. Also, there is a sender node S, a destination node D, and three intermediate nodes A, B, and X. An overview of the communication process between the cloud and the nodes in the network is given below.

A secure connection is established between the cloud and node S when the cloud receives $P_{ack(Initial)}$ from node S. After the cloud responds with a *ping()*, data packet transmission takes place. All of the acknowledgement packets are sent to the cloud after every hop in the route. Once the final acknowledgement packet $P_{ack(D)}$ is collected, the cloud forwards $P_{ack(D)}$ to the sender node. This scheme requires a special acknowledgement structure to carry some additional data that is required to detect the location of the data packet in the network.
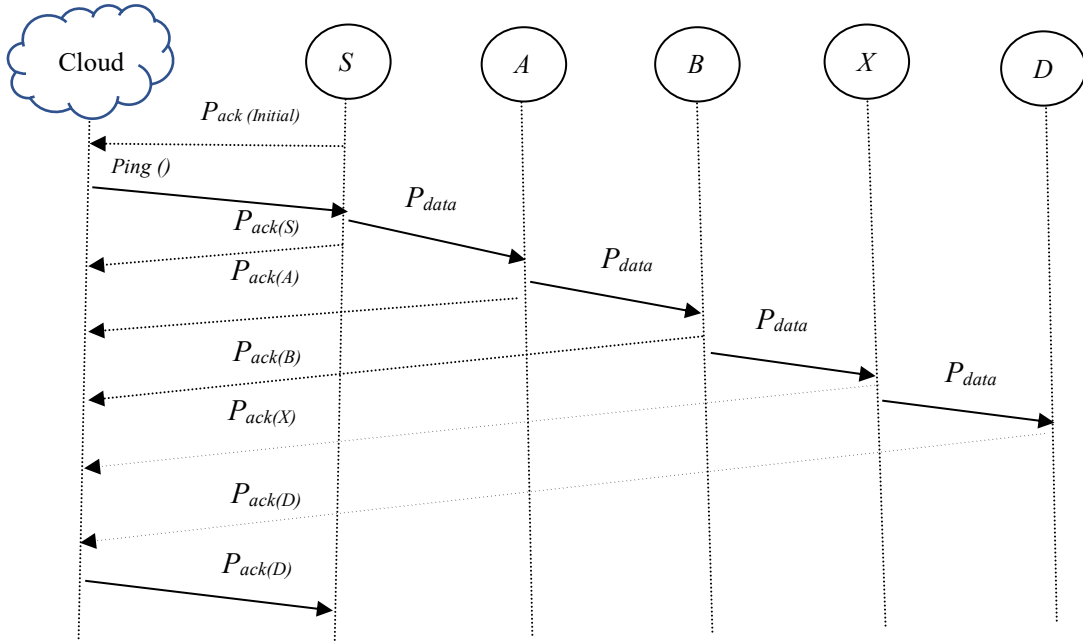
Fig. 3. Cloud ACKnowledgement Scheme.

Figures 4 and 5 illustrate the modified acknowledge packet and negative acknowledgement packet, respectively. The usual TCP acknowledgement packets carry some critical information about the data packet transmission, including:

- Source Port: Data packet originating address.
- Destination Port: Final target address of the data packet.
- SEQ Number: Sequence number of the data packet. This is used to identify the order of the payload.
- ACK Number: Acknowledgement number of the data packet.
- Flags: State of connection indicator.
- Window Size: Size of the node's buffer.
- Checksum: Detects if the data packet is corrupted.

TCP acknowledgement packets are not much different from TCP packets, except that TCP packets carry a payload, and their length is not null. TCP acknowledgement packets do not carry any payload. The modified CACKS acknowledgement packet introduces two new pieces of information: Current Node Address and Forwarded Node Address. The modified negative acknowledgement packets embed critical information like Current Node Address, Attempted Forwarding Node Address, and Neighboring Nodes Addresses.
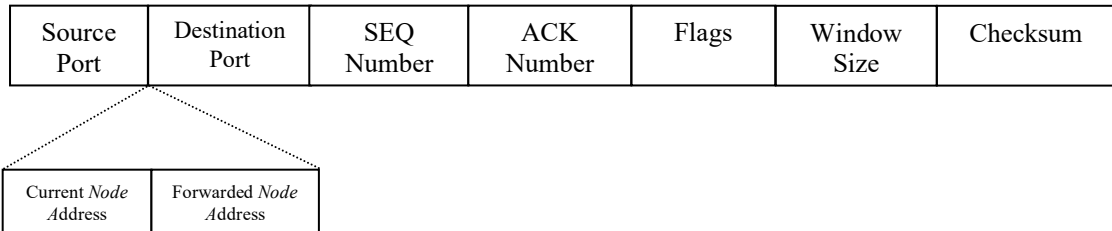

Fig. 4. Modified ACK Packet for CACKS.

Source Port	Destination Port	SEQ Number	ACK Number	Flags	Window Size	Checksum

Current *Node Address*	Attempted Forwarding *Node Address*	Neighboring *Node Addresses*

Fig. 5. Modified NAK Packet for CACKS.

Using the modified ACK packets and NAK packets, the CACKS scheme holds information about the data transactions as well as node activity. The CACKS scheme proposes that, whenever a transaction failure occurs due to interference, node mobility, and queue overflow [8], a node has a choice to ask the cloud to suggest the next best node to transmit data. The cloud ensures that there is always a backup for the nodes over a network.

Every data transmission generates acknowledgement packets. All of the embedded information in the acknowledgement packets is collected and condensed as records for each node in the cloud. Every node has a numerical value, and these values are initialized at the beginning to zero. For every successful transmission, the node value is incremented by 1. When an aborted transmission occurs, the node value is decremented by 0.6. Using this strategy of +1 and -0.6 gives the simple linear regression that is commonly used in forecasting [15]. Synthetic datasets are generated with the desired pattern for each node. The generated synthetic data consists of three datasets of different sizes with dimensions of 100000×100, 100000×500, and 100000×1000, respectively.

Figure 6 depicts the synthetic dataset and the data for a network of N_x nodes. This dataset has 100,000 rows and N_x columns. A filter is applied to the neighbours that separate a particular node. N_p and N_q are the neighbours of our node that requested the backup from the cloud. This measure improves the processing time, considering the transitory nature of the network.

Fig. 6. Filtering Nodes.

During our preliminarily research for this work, the aim was to build a machine learning model that recognizes the pattern of node values and potential node values. Predicting the forthcoming value with the help of the prior transactions is a challenge. Because our synthetic data is similar to time series data, the time series machine learning model provides an answer to our problem [13]. A time series model is able to predict the future. It collects data at a regular interval, which is then called time series data. It analyzes the data and the patterns of time increments in order to predict forthcoming events. This technique allows largely accurate assumptions about the future trends to be made based on the collected data [10]. Keeping time series capabilities in mind, we introduced LSTMs to this machine

learning model. LSTMs provide the leverage to understand the context that is needed to make predictions regarding the future [11].

3. Preliminarily Results

In this paper, we considered a dataset of 10 nodes with 10,000 transmission records. For every successful transaction, a node was either awarded with 1 or penalized with 0.6 to its value. This dataset was fed to a Long Short-Term Memory (LSTM) architected Recurrent Neural Network model (RNN) [5]. LSTM networks have the ability to remember and learn from long sequences in time series data. LSTM networks were the optimum choice for this research because our data is similar to time series data. Considering that this dataset contains 10,000 values for each node, we use LSTM through a sliding-window approach to predict the next best node for the communication. This technique produced assumptions about the future trends with great accuracy [12].

Due to the stochastic nature of the LSTM algorithm, every time we ran our experiments, the results varied. Figure 7 shows how our model is trained using a two-layered stacked LSTM architecture coupled with a dense output layer to make a prediction. We used RMSprop optimization algorithm and Mean Squared Error (MSE) as our loss function [16][17]. This configuration produces a model with the most accurate predictions. The randomly selected results seem promising from our initial tests, as is shown in Figure 5. The MSE is limited to ~2.516%. From Figure 8, it can be observed that our time series LSTM model produced good results. The graph depicts the actual and predicted values for 10 nodes. The actual and the predicted value trends appear consistent.
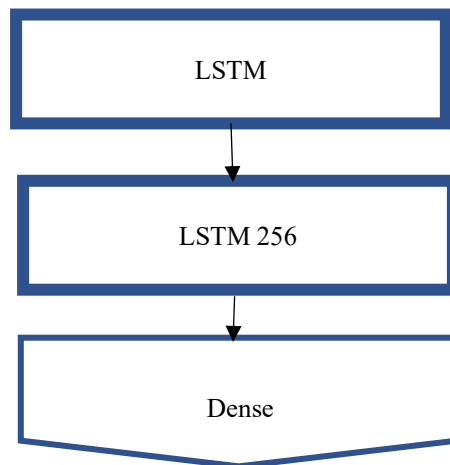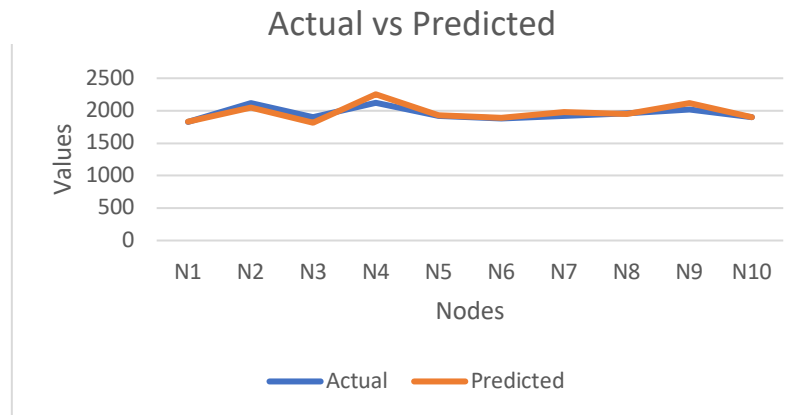

Fig. 7. Our Network Architecture.

Fig. 8. Predicted vs. Actual Values of Nodes.

4. Conclusion and Future Work

In this paper, we proposed a machine learning-based acknowledgement scheme to optimize backup routing using the cloud as a monitoring and backup service provider to the nodes in the network. The initial results demonstrate that machine learning techniques are capable of capturing the dynamics of node value trends and are able to make accurate predictions. In the future, we will work with a large synthetic dataset. We are also planning to experiment with several LSTM models and techniques to determine the best approach for node value forecasting.

Acknowledgements

References

[1] S. Kaja, E. M. Shakshuki and A. Yasar, "Cloud Acknowledgement Scheme for a Node Network," in *The 14th International Conference of Future Network and Communications, FNC 2019*, 2019.

[2] "Machine Learning: What it is and Why it Matters," 24 October 2020. [Online]. Available: https://www.sas.com/en_ca/insights/analytics/machine-learning.html.

[3] K. Martineau, "MIT News on Campus and Around the World," 16 September 2019`. [Online]. Available: https://news.mit.edu/2019/what-extra-computing-power-can-do-0916. [Accessed 24 October 2020].

[4] S. Kaja, E. M. Shakshuki and A. Yasar, "Enhanced Cloud Acnowledgement Scheme for a Node Network," in *The 17th International Conference on Mobile Systems and Pervasive Computing (MobiSPC)*, 2020.

[5] J. Brownlee, "LSTMs for Human Activity Recognition Time Series Classification," 28 August 2020. [Online]. Available: https://machinelearningmastery.com/how-to-develop-rnn-models-for-human-activity-recognition-time-series-classification/. [Accessed 15 October 2020].

[6] "What is ACK (networking)?," Network Encyclopedia, [Online]. Available: https://networkencyclopedia.com/ack/. [Accessed 18 October 2020].

[7] H.-N. Dai, R. C.-W. Wong, H. Wang, Z. Zheng and A. V. Vasilakos, "Big Data Analytics for Large-scale Wireless Networks: Challenges and Opportunities.," in *ACM Computing Surveys*, 2019.

[8] M. S. Khan, D. Midi, M. I. Khan and E. Bertino, "Fine-Grained Analysis of Packet Loss in MANETs," in

IEEE Access , 2017.

[9] G. H. Olson, "Technique for acknowledging packets". United States Patent US5245616A, 1992.

[10] J. Lin, S. Williamson, K. DBorne and D. DeBarr, "Pattern Recognition in Time Series," in *Advances in Machine Learning and Data Mining for Astronomy*, CRC Press, Taylor & Francis Group, 2012, pp. 617-645.

[11] R. Kompella, "Using LSTMs to forecast time-series," towards data science, 17 January 2018. [Online]. Available: https://towardsdatascience.com/using-lstms-to-forecast-time-series-4ab688386b1f. [Accessed 31 October 2020].

[12] L. Li, S. Dai and Z. Cao, "Deep Long Short-term Memory (LSTM) Network with Sliding-window Approach in Urban Thermal Analysis," in *IEEE/CIC International Conference on Communications Workshops in China (ICCC Workshops)*, 2019.

[13] A. Siswanto, "LSTMs for Financial Time Series Forecasting," in *MIT 6.864 Spring 2020*, 2020.

[14] C. Olah, "colah's blog," 27 August 2015. [Online]. Available: http://colah.github.io/posts/2015-08-Understanding-LSTMs/. [Accessed 27 October 2020].

[15] R. J. Hyndman and G. Athanasopoulos, Forecasting: Principles and Practice, OTexts; 2nd edition, 2018.

[16] C. Kathuria, "Regression — Why Mean Square Error?," towards datascience, 5 December 2019. [Online]. Available: https://towardsdatascience.com/https-medium-com-chayankathuria-regression-why-mean-square-error-a8cad2a1c96f. [Accessed 20 October 2020].

[17] RMSprop, *Keras Optimizers.*

[18] E. G. Petrakis and X. Koundourakis, "iXen: Secure Service Oriented Architecture and Context Information Management in the Cloud," *Journalof Ubiquitous Systems and Pervasive Networks (JUSPN),* vol. 14, no. 2, pp. 01-10, 2021.

[19] E. Badidi, "QoS-Aware Placement of Tasks on a Fog Cluster in an Edge Computing Environment," *International Journal of Ubiquitous Systems and Pervasive Networks (JUSPN) ,* vol. 13, no. 1, pp. 01-09, 2020.

[20] F. M. M.Erritali, "Student orientation using machine learning under MapReduce with Hadoop," *International Journal of Ubiquitous Systems and Pervasive Networks (JUSPN),* vol. 13, no. 1, pp. 21-26, 2020.