A New LSTM Model by Introducing Biological Cell State

Lamia Rahman, Nabeel Mohammed, Abul Kalam Al Azad,
Department of Computer Science and Engineering
University of Liberal Arts Bangladesh (ULAB)
Satmasjid Road, Dhaka, Bangladesh
Email:llamiarahaman5@gmail.com, nabeel.mohammed@ulab.edu.bd, abul.azad@ulab.edu.bd

Abstract—**Long Short Term Memory (LSTM) has been a very successful augmented recurrent neural network model employed to learn sequential information with long term dependencies where LSTM can store and compute information for a long period of time. In this study, a biologically inspired variation has been incorporated in LSTM by introducing additive cell state into the functionally computational system. The novel biological variant of LSTM model has been employed to conduct sentiment analysis of textual data. As the learning dataset, fifty thousand movie reviews have been used from IMDB where equal number of review data has been used for training and testing purposes. The comparative performance of the new variant is found to be promisingly better and show more stability than that of the traditional LSTM.**

I. Introduction

In the machine learning computational tool repertoire, Long Short Term Memory (LSTM) is widely employed in learning varied long term sequential dataset. LSTM typically learn a sequence with long term dependencies by virtue of a memory cell infrastructure that uses the interplay of three gates, namely, an input, a forget and a output gate. Since its inception, this model has undergone a handful of variations mostly retaining the recurrent architecture and multiplicative gates in processing the input sequences. Though the interactive structure of multiplicative gates in the model has been very effective, it may be noted that the neurons building blocks of biological nervous system embed an additive interplay of the different channel gates to computational purposes. In this study, a biologically motivated additive inner structure of the interacting gates are considered aligned with the traditional chemically driven different gates or channels elucidated by the seminal works of Hodgkin and Huxley. The Biologically motivated variant of the model proposed here is employed to learn standard IMDB dataset of fifty thousand entries. The performance of the variant model is compared with that of the traditional LSTM with interesting outcome.

The paper is arranged as following: after a brief interlude to the traditional LSTM different variations of the model is discussed. After that the biological motivation of the proposed model is explored followed by elucidation of the proposed biologically motivated model. A brief discussion of the IMDB dataset is presented and the model is employed for sentimental analysis of the dataset. The result and different metrics of performance are presented thereafter followed by concluding remarks.

II. Background

Neural Network based different methods have made great progress in the recent past. Applications like handwriting recognition, speech recognition, language modeling, image captioning and many more have enjoyed renewed success through the use of Neural Networks and deep architectures.

A. Artificial Neural Network (ANN)

Artificial neural network is a computational model which is inspired by natural neurons[3]. Natural neurons have four basic parts which are cell body or soma, axon, dendrites or nerve ending and synapses. In the biological brain dendrites act like input channels; they receive their inputs through the synapses of other neurons. The soma processes the information and turns the processed information as an output value which is sent to the other neurons. In 1943, the first conceptual model of a neural network was developed by Warren S. McCulloch, a neuroscientist, and Walter Pitts, a logician [9]. Like natural neurons, artificial neurons were designed to have input layers, where the input is given, then some hidden layers which process the information and forwards to the output layer. This simple model, known as a Perceptron, is still popular today and used in many classification/learning tasks.

B. Recurrent Neural Network (RNN)

Most of us do not process large amounts of information from start to finish in one go. Instead we process small bits of information, build mental models, and process some more information and so on until we reach target. Traditional neural network models do not have the capacity to process information in steps or sequences. Recurrent Neural Networks were developed specifically to address the sequence learning problem.

RNNs incorporate a loop in its network model for information persistence. The basic feature of a RNN is that - it has at least one feed-back-connection, so the activation can flow round in a loop which enables the network to do temporal processing and learn sequences. It consists of a layer of inputs connected to a set of hidden memory cells, a connected set of

978-1-5090-2906-8/16/$31.00 ©2016 IEEE

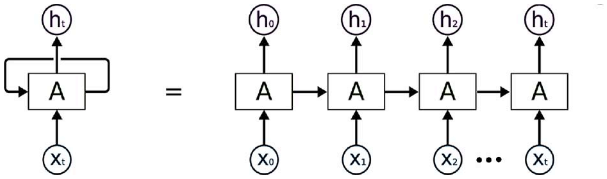

Fig. 1. An unrolled recurrent neural network[12]

recurrent connections amongst the hidden memory cells, and a set of output nodes.

These models can be visualized as multiple copies of the same network, each passing a message to a successor (Figure 1). One of the appeals of RNNs is that they might be able to connect previous information to the present task. Traditional RNNs were indeed able to achieve this, but for short sequences. Unfortunately, as that gap grows, RNNs are unable to successfully model long term dependencies. Long Short Term Memory (LSTM) was developed to address this short coming of traditional RNNs.

C. Long Short Term Memory (LSTM)

Long Short Term Memory networks are a special kind of RNN, capable of learning long-term dependencies. It is widely used because of its feature of remembering information for long periods of time. This is done by creating special modules that is designed to allow information to be gated-in and gated-out when needed. A basic difference between LSTM and RNN is that LSTM uses a memory cell with linear activation function to store information. LSTM was first proposed almost 20 years ago [4]. The design incorporates a memory cell using logistic and linear units with multiple interactions as shown in Equations 1 – 5. Where is the logistic sigmoid function and i, f, o and c are respectively the input gate, forget gate, output gate, and memory cell activation vectors and have exactly the same equations with different weights *Ws*.

$$i_t = \sigma(W^{(xi)}x_t + W^{(hi)}h_{t-1} + W^{(ci)}c_{t-1} + b^{(i)}) \tag{1}$$

$$f_t = \sigma(W^{(xf)}x_t + W^{(hf)}h_{t-1} + W^{(cf)}c_{t-1} + b^{(f)}) \tag{2}$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tanh(W^{(xc)}x_t + W^{(hc)}h_{t-1} + b^{(c)}) \tag{3}$$

$$o_t = \sigma(W^{(xo)}x_t + W^{(ho)}h_{t-1} + W^{(co)}c_{t-1} + b^{(o)}) \tag{4}$$

$$h_t = o_t \cdot \tanh(c_t) \tag{5}$$

Each memory cell c_t has its net input modulated by the activity of an input gate, and has its output modulated by the activity of an output gate. These input and output gates provide a context-sensitive way to update the contents of a memory cell. The forget gate modulates amount of activation of memory cell kept from the previous time step, providing a method to quickly erase the contents of memory cells.

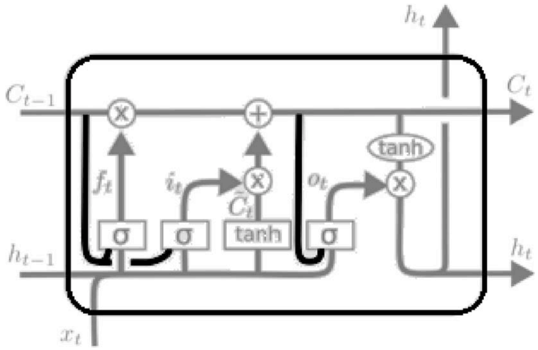

Fig. 2. LSTM variant: With peephole connection

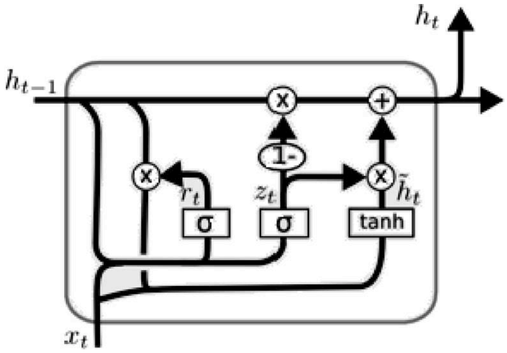

Fig. 3. LSTM variant: Gated Recurrent Unit

Alternate versions of LSTM have been proposed. Some of these are discussed below.

D. Variations of LSTM

A popular variant of LSTM, proposed by [2], adds peephole connections (figure 2), which enable the gates to incorporate the value of c_{t-1} in their updates.

$$f_t = \sigma(W_f \cdot [C_{t-1}, h_{t-1}, x_t] + b_f) \tag{6}$$

$$i_t = \sigma(W_i \cdot [C_{t-1}, h_{t-1}, x_t] + b_i) \tag{7}$$

$$o_t = \sigma(W_o \cdot [C_{t-1}, h_{t-1}, x_t] + b_o) \tag{8}$$

A slightly more interesting variation on the LSTM is the Gated Recurrent Unit (GRU) which was introduced by Cho, et al. [1]. It combines forget and input gates into a single gate which is known as an update gate (z_t in Equations 9 – 12). It also merges the cell state and hidden state. The resulting model, shown in figure 3, is simpler than standard LSTM models, and has been growing increasingly popular.

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \tag{9}$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \tag{10}$$

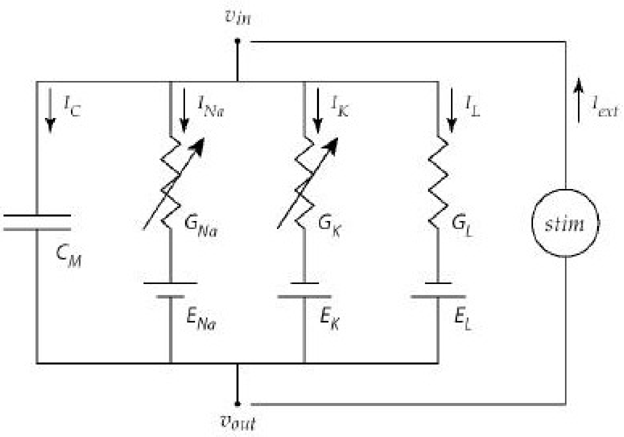

Fig. 4. Electrical equivalent circuit for a short segment of squid giant axon[6]

$$\tilde{h_t} = \tanh(W_r \cdot [r_t * h_{t-1}, x_t]) \tag{11}$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h_t} \tag{12}$$

III. Biological motivation for additive model

At the heart of the neuro-biological computation lies the physiological building block known as neuron. Neuron, anatomically, is primarily a cellular structure consists of two anatomical processes known as dendrite and axon which function as the input and output apparatus to and from neuron, respectively. The bio-chemical properties of the neuron are well elucidated by the works of Alan Hodgkin and Andrew Huxley [5]. In explain the firing or the action potential of neuron, sodium and potassium ion gates or channels with complex voltage-gated properties were posited into an elegant mathematical model.

Hodgkin and Huxley neuron model uses a series of voltage clamp experiments and varying extracellular sodium and potassium concentrations. In this celebrated model the properties of an excitable cell are described by a set of four ordinary differential equations which are given in Equations 13-16. Where I is the current per unit area, and α_i and β_i are rate constants for the i-th ion channel, which depend on voltage but not time. $\bar{g}_n$ is the maximal value of the conductance. n, m and h are dimensionless quantities between 0 and 1 that are associated with potassium gate activation, sodium gate activation, and sodium gate inactivation. It is interesting to note that the neuro-computations are governed by the interplay of the stochastic behavior relating the opening and closing rates of ion gates which appear mathematically additive as seen in Equation 14. The motivation of developing a biologically inspired LSTM came from the mathematical model of Hodgkin and Huxley where neurons, the neuro-computational unit in the living organism, process information controlling the traffic of different ions through gates. The relation between Hodgkin Huxley model and LSTM is that the latter model adopts a very similar mathematical structure of the input, output and forget gates as the related biological counterpart.

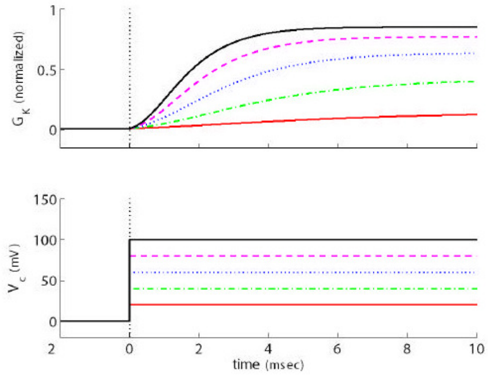

Fig. 5. Mathematical model of the voltage dependent and time dependent properties of the Na+ and K+[6]

Hodgkin and Huxley created a model by using a series of voltage clamp experiments and varying extracellular sodium and potassium concentrations. In that model the properties of an excitable cell are described by a set of four ordinary differential equations which are given below in Equations 13 – 16.

$$I = C_m \frac{dV_m}{dt} + \bar{g}_k n^4 (V_m - V_k) + \bar{g}_{Na} m^3 h (V_m - V_{Na}) + \bar{g}_t (V_m - V_t) \tag{13}$$

$$\frac{dn}{dt} = \alpha_n(V_m)(1 - n) - \beta_n(V_m)n \tag{14}$$

$$\frac{dm}{dt} = \alpha_m(V_m)(1 - m) - \beta_m(V_m)m \tag{15}$$

$$\frac{dh}{dt} = \alpha_h(V_m)(1 - h) - \beta_h(V_m)h \tag{16}$$

Where I is the current per unit area, and and are rate constants for the i-th ion channel, which depend on voltage but not time. is the maximal value of the conductance. n, m, and h are dimensionless quantities between 0 and 1 that are associated with potassium channel activation, sodium channel activation, and sodium channel inactivation. The main motivation behind creating a biological LSTM came from the biological model of Hodgkin and Huxley where they created an additive model of giant squids axons. The gating functions in an LSTM model resemble the functionality of the three ion channels (leakage, Sodium, Potassium channels). However, a study of the traditional LSTM model, in Equations 1-5, shows that it is multiplicative model. In this work we modify the LSTM model to have an additive cell state, thus rendering it closer to the biological model of Hodgkin and Huxley.

IV. Proposed model: Biologically variant LSTM

We propose a biologically inspired LSTM model as outlined in the previous section. Our proposed model is shown in Equations 17 – 21, with the particular changes to the cell state shown in Equation 19.

Here the value is determined for is 0.001 for ensuring computational precision. Our only change was to the way in which we update c, which in this new model is an incremental update from the older cell state, c_{t-1}, where the increment depends on a fraction of the summation of i and f. This new model is much simpler and biologically plausible.

$$i_t = \sigma(W^{(xi)}x_t + W^{(hi)}h_{t-1} + W^{(ci)}c_{t-1} + b^{(i)}) \quad (17)$$

$$f_t = \sigma(W^{(xf)}x_t + W^{(hf)}h_{t-1} + W^{(cf)}c_{t-1} + b^{(f)}) \quad (18)$$

$$c_t = c_{t-1} + \delta * (i + f + b_c) \quad (19)$$

$$o_t = \sigma(W^{(xo)}x_t + W^{(ho)}h_{t-1} + W^{(co)}c_{t-1} + b^{(o)}) \quad (20)$$

$$h_t = o_t \cdot \tanh(c_t) \quad (21)$$

V. Method and Dataset

In our proposed model of LSTM, we are introducing a change to how the model updates/calculates the cell state c_t, to more closely resemble a biological process (more on this in Section IV). The performance of proposed model will be compared with that of LSTM on a sentiment analysis task on a the popular IMDB movie review dataset.

A. Sentiment Analysis

Sentiment analysis is a computational process to automatically estimate the sentiment expressed by people, (most commonly) in written material. The sentiment could have been expressed over anything worth having an opinion on - a product we want to use; book we want to read or can be a movie we want to watch. Sentiments have been categorized into two basic categories: positive or negative and also an n-point scale eg. good, very good, satisfactory[10]. In this age of globalization, e-commerce sites have achieved great popularity and success. This has also meant that it is becoming increasing difficult to manually sift through all the product reviews, as the volume of comments and reviews generated by platforms such as Twitter, Facebook, blogs is simply too high for manual inspection [11]. Sentiment analysis tools can play an important role here, as they can automatically gauge user/review sentiment. Since early 2000, sentiment analysis has become one of the most popular areas in natural language processing [7]. Plainly, the challenge for a sentiment analysis model is to understand the sentiment, to detect which words are using as a positive review or which words are for negative. Normally “good”, “amazing”, “wonderful” are thought as a positive words and “bad”, “terrible”, “poor” are considered as a negative sentiment words, and also there are other phrases which express sentiment. A list of such words or phrases are known as a sentiment lexicon[7]. However, a sentence containing such earmarked words may not actually express the assumed sentiment. This is especially true for very human actions such as sarcasm. Essentially systems which are more robust than just word-based solutions are required. Neural Networks, in particular Recurrent Neural Networks and its variants have enjoyed resurgence in this area over the last few years.

B. Dataset

To test our model on a sentiment analysis task we decided to test against a standard dataset. We are using the IMDB movie review dataset [8]to do the sentimental analysis on our model. The dataset was constructed by collecting 5000 reviews from movies from IMDB, which doesnt allow more than 30 reviews per movie. The dataset has an even number of positive and negative reviews, so randomly guessing yields 50% accuracy. Based on previous work on polarity classification, only highly polarized classification are considered. The score is given for negative review is ¡=4 out of 10, and ¿=7 out of 10 for positive reviews. Neutral reviews were not included in the dataset. The dataset was evenly divided for the training and test sets.

C. Implementation Detail

Our models were implemented using the open source, Python-based, Keras deep learning library. We utilized Kerass sequential model, which allowed us to create a multi-layer, deep networks. The first layer of the sequential model is an embedded layer, which is used to implement the word to vector representation [8]. The second layer is an LSTM/ layer which has 128 dimensional internal states. Max-feature is set to 2000 in the code which means it will take 2000 unique word from the dataset at the training time. The maximum length considered was set to 100 characters. Both models were trained on this data set for 15 epochs.

For comparison, we wrote a new Keras layer which implements our proposed version. A new model was created which mimics the one described above, except that the second layer (LSTM) layer was replaced with the one we developed.

VI. Results

As with other neural network-based studies, we present our results in terms of the loss and accuracy during training and validation. The loss function used is categorical cross entropy and accuracy is expressed as a fraction with 1.0 being completely accurate. Before discussing the results, it is worthwhile to review that fact that the loss values are in fact used to train the network (through back propagation) and are ideally meant to be reflective of the generalised performance. Practically however we find that reduction in loss values are not always reflected through improvement in accuracy, however a level of mirroring is expected.

Figures 6 and 7 shows how the loss and accuracy measures for the training and validation sets. Two things are immediately apparent - first, the behaviour of the loss function is inconsistent between the training and validation sets. The inconsistency is highlighted even more when we take into account the accuracy measures, as the loss values do not accurately reflect the effectiveness of the model.

In comparison, Figures 8 and 9 shows how the loss and accuracy measurements change over the 15 epochs for the proposed model. The accuracy measures are comparable to the LSTM model. In both cases, the accuracy measures plateau out around the 80% mark. However, the divergence between the validation loss and training loss values is not as large

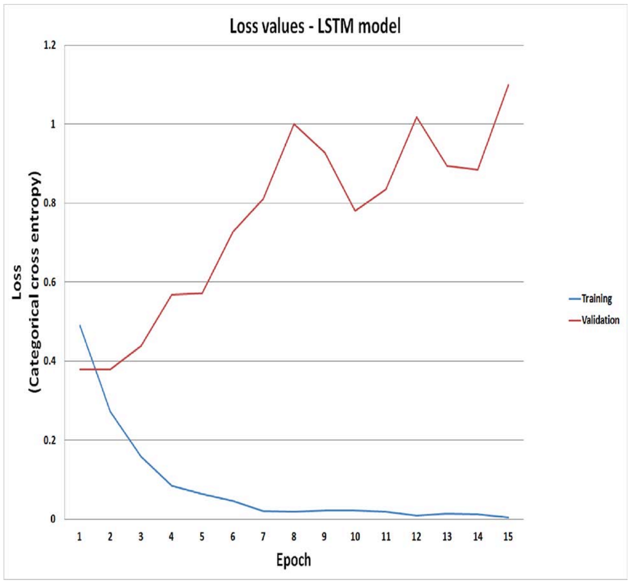

Fig. 6. Training and validation loss of LSTM

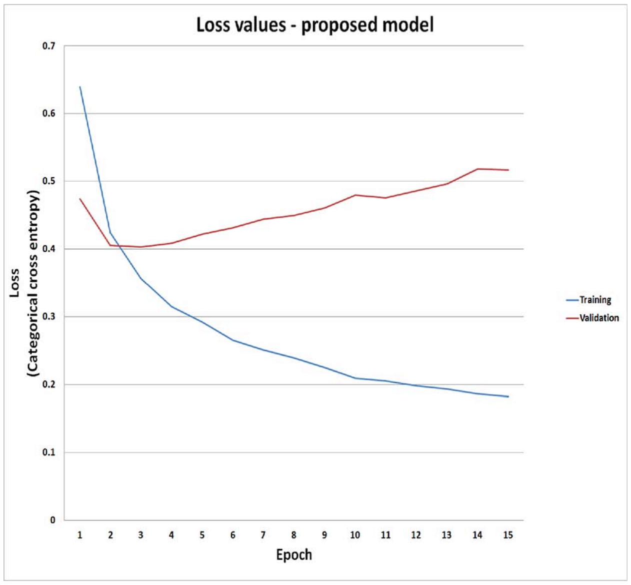

Fig. 8. Training and validation loss of the proposed model

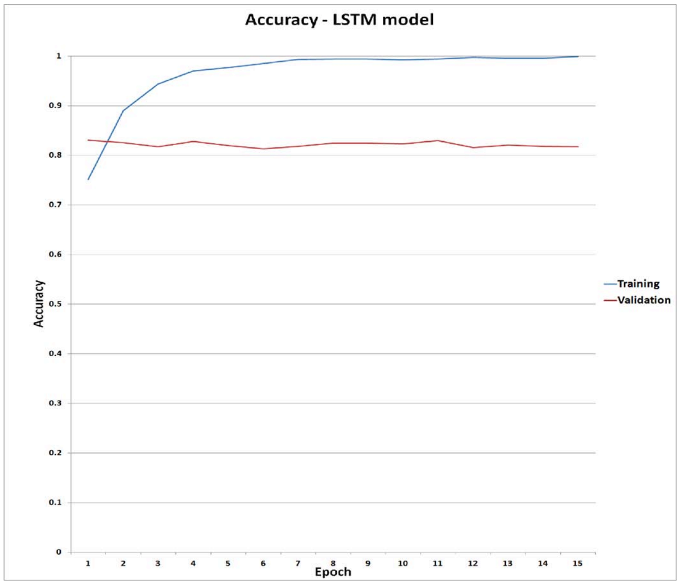

Fig. 7. Training and validation accuracy of LSTM

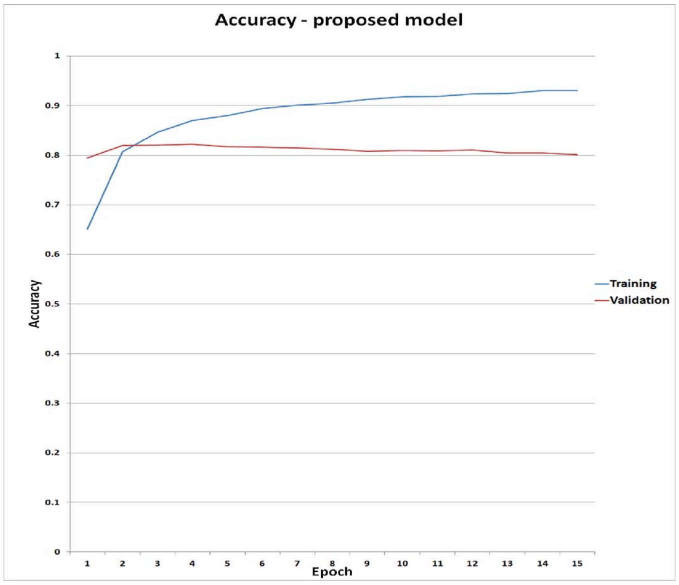

Fig. 9. Training and validation accuracy of the proposed model

as the LSTM model. Also, the proposed model is not as prone to over-training as the LSTM model, which can be seen by comparing the training accuracy for the LSTM and proposed models. It should be noted, that these differences arise purely through the introduction of an additive cell state model with no other changes done in the training pipeline. These two things, lower divergence of the loss values and better resistance to over training leads us to conclude the the proposed additive model displays more stability compared the LSTM's multiplicative model, while maintaining comparable accuracy.

Finally for a direct comparison, figure 10 shows the validation accuracy of the LSTM and proposed models. Both models stagnate at around 80% accuracy, so the proposed model does not result in an improvement in accuracy. However in terms of consistency and stability, the proposed additive model shows superior behaviour over the training and validation sets while being less susceptible to over-training.

VII. Conclusion

In this study, the proposed model incurs the motivation from the traditional neuronal cell model structure where the interactive gates are additive rather than multiplicative as in the traditional LSTM. This rather simple modification to the model when applied in a sentimental analysis task yielded a better comparative stability in training and testing. The comparative analysis demonstrates the training loss and validation

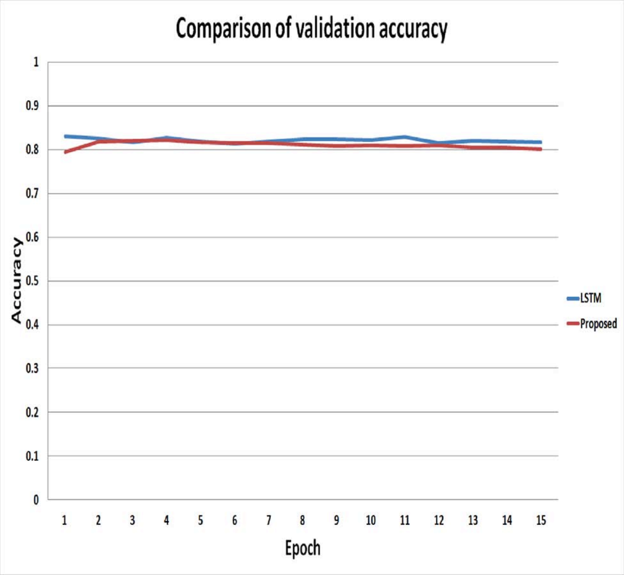

Fig. 10. Comparison of validation accuracy

loss of traditional LSTM, showing excellent results at the time of training period as the loss function is almost null after some time from the start. But when it comes to testing, the loss increases continuously, where the loss and validation loss of the proposed variant shows more stable result. The loss of the proposed model is not as good as LSTM at the time of training but when it comes to testing, the growth of the the loss function is not nearly as acute as that of the LSTM model.

The discrepancy between loss and validation loss of the biologically motivated variant is found to be significantly lower compared to that of the LSTM. Furthermore, the LSTM model achieves almost 100% training accuracy within the 15 epochs, but the validation accuracy showed no significant improvement; which leads us to conclude that the LSTM model was over training.

We have demonstrated that the proposed additive model is not as prone to over training, at least for the sentiment analysis task shown in this paper, while achieving a comparable validation accuracy, in the same number of epochs. As the current model shows promise, the further study is planned to accommodate noise as per the biological reality and a wider scope of the learning dataset is deemed for implementation.

References

[1] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[2] F. A. Gers and J. Schmidhuber, "Recurrent nets that time and count," in *Neural Networks, 2000. IJCNN 2000, Proceedings of the IEEE-INNS-ENNS International Joint Conference on*, vol. 3. IEEE, 2000, pp. 189–194.

[3] C. Gershenson, "Artificial neural networks for beginners," *arXiv preprint cs/0308031*, 2003.

[4] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[5] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *The Journal of physiology*, vol. 117, no. 4, p. 500, 1952.

[6] S. Koslow and S. Subramaniam, "Electrophysiological models in: Databasing the brain: From data to knowledge," 2004.

[7] B. Liu, "Sentiment analysis and opinion mining," *Synthesis lectures on human language technologies*, vol. 5, no. 1, pp. 1–167, 2012.

[8] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011, pp. 142–150.

[9] D. Shiffman, S. Fry, and Z. Marsh, *The nature of code*. D. Shiffman, 2012.

[10] G. Vinodhini and R. Chandrasekaran, "Sentiment analysis and opinion mining: a survey," *International Journal*, vol. 2, no. 6, 2012.

[11] X. Wang and Z. Min, "Parallel algorithm for hill cipher on mapreduce," in *Progress in Informatics and Computing (PIC), 2014 International Conference on*. IEEE, 2014, pp. 493–497.

[12] K. Yao, T. Cohn, K. Vylomova, K. Duh, and C. Dyer, "Depth-gated recurrent neural networks," *arXiv preprint arXiv:1508.03790*, 2015.