Long Short-Term Memory Neural Network Equilibria Computation and Analysis

Massinissa Amrouche* **Deka Shankar Anand*** **Aleksandra Lekić†**
Vicenç Rubies Royo‡ **Elaina Teresa Chai§** **Dušan M. Stipanović*** **Boris Murmann§**
Claire J. Tomlin‡

Abstract

This paper presents a comprehensive approach for computing nontrivial equilibria of autonomous Long Short-Term Memory neural networks using a homotopy formulation. Through simulations, it is shown that the eigenvalues of the linearized models around these nontrivial equilibria tend to move closer to the unit circle as the complexity of the training data increases. This provides insights into the dynamical properties of the LSTM neural networks.

1 Introduction

Since Long-Short Term Memory networks (LSTMs) were initially introduced in [1], there has been a wealth of successful examples employing these types of networks for audio, written text and video recognition (see, for example, [2] and references reported therein). While much of the effort has been centered around improving training algorithms for LSTMs, there has been relatively little work in trying to understand their dynamical properties and behaviors. In particular, it has been well recognized that it is not uncommon for the trained neural networks to operate at the edge of chaos [3, 4]. Recently, it has been shown in [5] that the edge of chaos can be characterized by the stability region for the linearized model of the autonomous LSTM model with an equilibrium at zero. Motivated by this work, we use a linear path-following homotopy [6] to compute non-trivial equilibria of the autonomous LSTM and the corresponding linearized models.

Through simulations, as reported for the zero equilibrium in [5] and as performed for non-trivial equilibria in the same way (not provided in this paper due to the space limitation), we observed that the edge of chaos for the nonlinear LSTMs is accurately characterized by the eigenvalues' proximity to the boundary of stability (that is, the unit circle). In order to provide this characterization, we computed and analyzed the eigenvalues of the linearized model of a single layer LSTM neural network, trained on three benchmark tasks: handwritten digit recognition, text generation, and polyphonic music generation. For the image classification, we use the MNIST data set [7] which has only ten classes. Then, the Penn Tree Bank (PTB) data set, which is more rich and has many more classes, is used for the text generation task. The result is that, compared to the MNIST data set, the eigenvalues are pushed even further toward the unit circle. Finally, we use the Nottingham data set [8] for the music generation task, which also corroborates our claim that increasing the number and the complexity of training signals, particularly from different classes, causes the eigenvalues to approach the unit circle, effectively causing the trained network to operate at the edge of stability, that is, chaos.

We hope that these initial results and observations make LSTMs more comprehensible and therefore will open up new research avenues and the possibility for improved network design, training and inference.

The paper is organized as follows. In Section 2, we provide the LSTM discrete-time dynamical model and its basic properties. A homotopy formulation and how it is used to compute the LSTMs' nonzero equilibria, are provided in Section 3. Representative sets of simulations are presented in Section 4. Finally, some concluding remarks are provided in Section 5.

*University of Illinois, Urbana-Champaign, IL, USA.{amrouch2,sadeka2,dusan}@illinois.edu
†University of Belgrade, Belgrade, Serbia. lekic.aleksandra@etf.bg.ac.rs
‡University of California, Berkeley, CA, USA. {vrubies,tomlin}@berkeley.edu
§Stanford University, Stanford, CA, USA. {echai,murmann}@stanford.edu

Workshop on Modeling and Decision-Making in the Spatiotemporal Domain, 32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

2 LSTM Model

In this paper, we are interested in studying the properties of the autonomous LSTM Neural Network (NN), that is the standard LSTM model introduced by [1] without input vectors. Mathematically, the model is written as the following vector difference equation:

$$\begin{aligned} c(k+1) &= \sigma\left(W_f h(k) + b_f\right) \odot c(k) + \sigma\left(W_i h(k) + b_i\right) \odot \tanh\left(W_g h(k) + b_g\right), \\ h(k+1) &= \sigma\left(W_o h(k) + b_o\right) \odot \tanh\left(c(k+1)\right), \end{aligned} \tag{1}$$

where $h(\cdot)^T, c(\cdot)^T \in \mathbb{R}^n$ are, respectively, the output and memory state vectors, $\odot$ is the Hadamard product, and $\sigma(\cdot) := \frac{1}{1+e^{-(\cdot)}}$ is the standard logistic function. $W_{f,i,o,g} \in \mathbb{R}^{n\times n}$ and $b_{f,i,o,g} \in \mathbb{R}^n$ are constant and represent the weights and biases of the neural network.

Studying the properties of autonomous LSTMs is of much interest because they are representative of the intrinsic dynamical behavior of the general models with input and therefore exploring their behavior may indicate why these systems perform well. In this paper, we study the behavior of autonomous LSTMs with non-trivial equilibria. It is straightforward to see that for $b_g \neq 0$ the equilibria are not trivial and computing them is very hard given that the steady-state equations are transcendental, yet the equilibria's numerical values are needed to compute the linearized models. Thus, in the following section, we introduce a path-following homotopy method to compute the nonzero equilibria.

3 Homotopy Approach to Computing LSTM Nonzero Equilibria

The homotopy continuation method is a well-known technique to solve the problem of finding the roots and fixed points of nonlinear functions. In topology, a homotopy between two continuous functions $F, G : \mathbb{R}^n \to \mathbb{R}^m$ is defined as another continuous function $H : [0,1] \times \mathbb{R}^n \to \mathbb{R}^m$ such that $H(0,\cdot) \equiv G(\cdot)$ and $H(1,\cdot) \equiv F(\cdot)$. Now, If $x_0 \in \mathbb{R}^n$ is a known root of G, and H is continuously differentiable then the homotopy method allows us to track the path $x : [0,1] \to \mathbb{R}^n$ that starts at $x(0) = x_0$ and satisfies $H(t, x(t)) \equiv 0$ for all $t \in [0,1]$. It is clear that, at $t = 1$, $H(1, x(1)) = F(x(1)) = 0$, thus $x(1)$ is a root of F. A more rigorous introduction can be found in [6]. In this section, we explain, briefly, our procedure to find the equilibria of system (1). To this end, we define a new linear homotopy map that allows us to compute the nontrivial equilibria for LSTMs when $b_g \neq 0$ by starting from the trivial solution when $b_g = 0$. To start, consider the LSTM model in equation (1) and let $h(k) = q(k) + q_c$, where $q(\cdot) \in \mathbb{R}^n$ is a new variable and $q_c \in \mathbb{R}^n$ satisfies $W_g q_c + b_g = 0$. The existence of such q_c is guaranteed, if b_g is in the span of columns of W_g, which is a very mild requirement. Therefore, system (1) can be, equivalently, rewritten as:

$$x(k+1) = F(x(k)) + x_c, \tag{2}$$

where $x(k) := [c(k)^T, q(k)^T]^T$ and

$$F\left(\begin{bmatrix} c \\ q \end{bmatrix}\right) = \begin{pmatrix} F_1(q,c) \\ F_2(q,c) \end{pmatrix} = \begin{pmatrix} \sigma(W_f q + \bar{b}_f) \odot c + \sigma(W_i q + \bar{b}_i) \odot \tanh(W_g q) \\ \sigma(W_o q + \bar{b}_o) \odot \tanh(F_1(q,c)) \end{pmatrix}$$

such that, the new biases are defined as $\bar{b}_j = b_j + W_j q_c$, for $j \in \{f, i, o\}$. Note that the equilibria of (2) are the roots of the map $x \to x - F(x) - x_c$. Therefore, we define our homotopy map as:

$$\begin{aligned} H(t,x) &:= (1-t)(x - F(x)) + t(x - F(x) - x_c) \\ &= (x - F(x)) - t x_c. \end{aligned} \tag{3}$$

such that the equilibria of (2) are the solutions, at $t = 1$, to $H(t,x) = 0$. Now, define

$$\mathcal{H}_\epsilon^{-1} := \{(t,x) \in [0,1] \times \mathbb{R}^n |\ H(t,x) = \epsilon\} \tag{4}$$

as the set of all the solution curves to $H(t,x) = \epsilon$, for some fixed $\epsilon \in \mathbb{R}^n$. Let these curves be parameterized using a new independent variable $\theta \in \mathbb{R}$ such that $(t(\theta), x(\theta)) \in \mathcal{H}_\epsilon^{-1}$ for all $\theta \in \mathbb{R}$. Since $H \in C^\infty(\mathbb{R}^{n\times n} \times [0,1])$, and moreover if we assume that the Jacobian matrix $H'(t,x) := [\partial H/\partial x \ \ \partial H/\partial t]$ is of full rank, that is $\text{rank}(H'(t,x)) \geq n$ for all $(t,x) \in \mathcal{H}_\epsilon^{-1}$, then the implicit function theorem ensures, that $\mathcal{H}_\epsilon^{-1}$ is composed solely of continuously differentiable curves that are solutions to the following differential equation:

$$\frac{\partial H(x(\theta), t(\theta))}{\partial x} \cdot \frac{dx(\theta)}{d\theta} + \frac{\partial H(x(\theta), t(\theta))}{\partial t} \cdot \frac{dt(\theta)}{d\theta} = 0 \tag{5}$$

Therefore, the equilibrium point of system (2) is simply the solution, at $t = 1$ and for $\epsilon = 0$, of the differential equation (5) along with the initial condition $(t(\theta), x(\theta)) = (0, 0)$. A singular case may occur when the Jacobian matrix $[\partial H/\partial x \ \ \partial H/\partial t]$ is not of full rank yet Sard's theorem ensures that the set of singular values are of measure zero. Thus, there exists always an $\epsilon \in \mathbb{R}^n$, arbitrarily close to the origin, such that $\mathcal{H}_\epsilon^{-1}$ is composed only by continuously differentiable curves that are solutions to (5). Therefore, the homotopy method allows us to get arbitrarily close to any equilibria of (2).

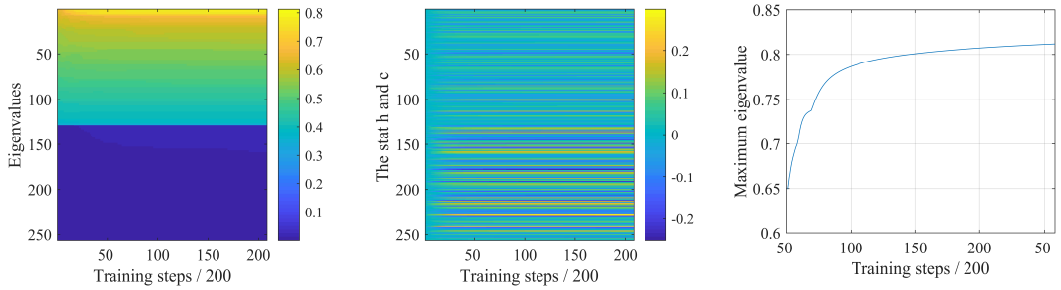
4 Simulations

In this section, we provide a number of simulations corresponding to nonzero equilibria analysis to show how the behavior of the LSTMs are correlated with the location of the eigenvalues and how they propagate during training depending on the number of training samples, classes, and tasks. The first simulation is related to the training of the LSTM neural network to recognize handwritten digits from MNIST database [7] and we observe that the eigenvalues, generally, move towards the unit circle during training. However, they do not reach it, which we believe is a consequence of the training set consisting of only 10 different labels/classes (that is, digits from 0 to 9). In the second experiment, we use a more rich class of training signals; the Penn Tree Bank (PTB) data set [9] to train an LSTM neural network, in the context of language modeling and word prediction. Results show that the eigenvalues are pushed more closely toward the unit circle which confirms our proposition that the more rich and complex the input data is, the more eigenvalues are pushed toward the unit circle. To confirm this trend, we train an LSTM network, using the Nottingham data set [8], to predict melodies and harmonics in the context of polyphonic music modeling.

Our general procedure is to train single-layer LSTMs on the data sets mentioned previously, where the inputs are vectors of fixed size representing the pixel intensities for a given row in the context of image recognition, words in the context of language modelling, and melodies/harmonics in the context of music modeling. We perform a softmax operation on the output of the LSTM in order to get a probability distribution. Then, the loss is defined to be the cross-entropy between the distribution and the corresponding label/prediction in "one-hot" encoding format. Training is finally accomplished via backpropagation through time (BPTT) with various batch sizes. Throughout the training, the weight matrices and bias vectors are stored at constant intervals. Then, equilibria of the autonomous LSTMs, constructed using each of these weights and biases, are computed using the homotopy method.

4.1 Equilibria of LSTM neural networks used for MNIST digit recognition

This first simulation set is used to demonstrate the dependence of the absolute value of eigenvalues during training. A neural network configuration with the input layer consisting of 10 neurons, hidden layer with 128 neurons and 10 output neurons has been chosen. The particular training task, as mentioned earlier, is for the LSTM network to recognize handwritten digits from the MNIST data set [7]. The inputs were vectors with fixed size representing the pixel intensities for a given row in an image. To demonstrate how the eigenvalues propagate during training itself, we trained the LSTM with a data set of 100 samples and stored the weights and biases every 200 gradient steps. The corresponding eigenvalues are plotted in Fig. 1. The magnitude of the largest eigenvalues are shown as a function of the training step in Fig. 1, where an increasing trend can be clearly observed. Fig. 1c indicates that their maximum is ~ 0.81, which can be explained by the small number of classes in the data set. This result corresponds to the previous analysis.



(a) The eigenvalues at the equilibrium

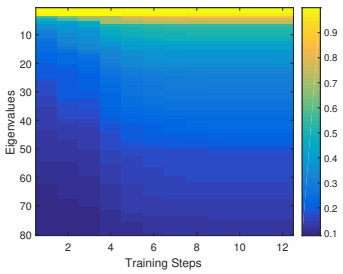
(b) The value of the static h and c of the equilibrium

(c) Dependence of the magnitude of the maximum eigenvalue

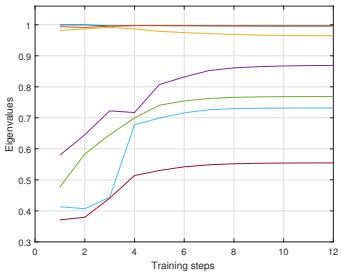
Figure 1: Dependence of the (a) eigenvalues' magnitudes, (b) equilibria and (c) maximum eigenvalues' magnitude during the training with 100 handwritten images.

4.2 Equilibria of LSTM neural network used for language modeling

To show that the eigenvalues can be pushed further toward the unit circle, we use the Penn Tree Bank (PTB) data set to train the LSTM to predict the next word given some previous words. This data set has $\sim$ 10k words in its vocabulary, which is much larger than the number of classes in the previous example. The configuration of the network and its training is done as explained above. The size of the input, output, and cell vectors is 200. Fig. 2a shows how the largest eigenvalues propagate during



(a) The 80 largest eigenvalues at equilibrium



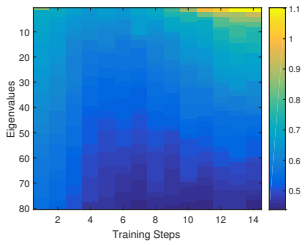
(b) The 7 largest eigenvalues at the equilibrium

Figure 2: Dependence of (a) the eigenvalues' magnitude, (b) the maximum eigenvalues' magnitude during the training on PTB

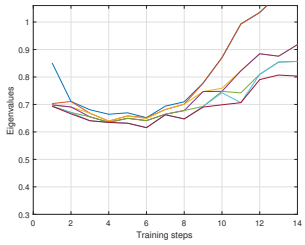
training. It is clear that they are pushed toward the unit circle. Fig. 2b indicates, in fact, that two eigenvalues have a magnitude of ~ 0.99 (very close to the unit circle). Note that, because the real part of these two eigenvalues was also ~ 0.99, the convergence of the homotopy solver was very slow. This is due to the fact that, in that case, $\dot{t}(\theta)$ in equation (5) was very close to zero which makes t increase very slowly toward one.

4.3 Equilibria of LSTM neural network applied for polyphonic music generation

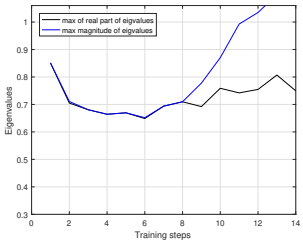
In this experiment, we use the Nottingham data set [8] to train a single layer LSTM neural network in the same manner explained above. The input vectors are of dimension 68; the first half of the input vector represents the basic musical notes (or the melodies) and the second part the chords (or harmonics). The dimension of the output vector is 200. The results confirm that the eigenvalues are pushed toward the unit circle during training. Fig 3a-b exhibits clearly this tendency. Fig 3c



(a) The 80 largest eigenvalues at the equilibrium



(b) The seven largest eigenvalues at the equilibrium



(c) Maximum magnitude and real-part of eigenvalues at equilibrium

Figure 3: Dependence of (a) the eigenvalues' magnitude, (b) the largest eigenvalues' magnitude during the training on Nottingham database (c) spectral radius vs maximum real part of eigenvalues during the training

depicts that, although the spectral radius gets closer and crosses the unit circle, all real parts of the corresponding eigenvalues are well smaller than 1, which implies that $\dot{t}(\theta)$ in equation (5) is non-zero, and therefore, t can monotonically increase towards 1 without causing any numerical issues for the homotopy. This has a consequence that the convergence of the homotopy solver was faster in this case than in the previous simulation. Note that, it is the maximal eigenvalue magnitude that characterizes the boundary of stability, which we stipulate does coincide with the edge of chaos, as shown for the trivial equilibrium in [5].

5 Conclusion

A homotopy-based approach to compute equilibria of autonomous LSTM neural networks, is provided in this paper. Through simulations, it has been shown how the locations of the eigenvalues of the linearized models around the computed equilibria depend on the training sample sizes and tasks. Some of our current research activities are focused on providing connections among the computation of the weights and biases, complexity of the training data, and the location of the equilibria.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. 1528036 titled ASPIRE: Automation Supporting Prolonged Independent Residence for the Elderly.

References

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

[2] "Transitioning entirely to neural machine translation," https://code.facebook.com/posts/289921871474277/transitioning-entirely-to-neural-machine-translation/, 2017.

[3] N. E. Barabanov and D. V. Prokhorov, "Stability analysis of discrete-time recurrent neural networks," *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 292–303, 2002.

[4] T. Laurent and J. V. Brecht, "A recurrent neural network without chaos," in *Proceedings of the 2017 International Conference on Learning Representations (ICLR)*. arXiv preprint arXiv:1612.06212, 2017.

[5] D. M. Stipanović, B. Murmann, M. Causo, A. Lekić, V. Rubies Royo, C. J. Tomlin, E. Beigne, Thuries, Z. S., M., and S. Lesecq, "Some local stability properties of an autonomous long short-term memory neural network model," in *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2018, available at https://ieeexplore.ieee.org/document/8350958/.

[6] W. Zangwill and C. Garcia, *Pathways to Solutions, Fixed Points, and Equilibria*. Englewood Cliffs, NJ: Prentice-Hall, 1981.

[7] Y. LeCun, "The MNIST database of handwritten digits," *http://yann. lecun. com/exdb/mnist/*, 1998.

[8] N. Boulanger-Lewandowski, Y. Bengio, and P. Vincent, "Modeling temporal dependencies in high-dimensional sequences: Application to polyphonic music generation and transcription," *arXiv preprint arXiv:1206.6392*, 2012.

[9] M. P. Marcus, M. A. Marcinkiewicz, and B. Santorini, "Building a large annotated corpus of english: The penn treebank," *Computational linguistics*, vol. 19, no. 2, pp. 313–330, 1993.