ECE 498/598 Project   
Language Learning using Long Short-Term Memory and a Recurring Neural Network Model

**Elizabeth Willard, Jeffrey Edrington**

Abstract[[1]](#footnote-1)

The goal of this project was to evaluate the effectiveness of a neural network on a database of text to see how well the pattern prediction capabilities would perform. This project used Yelp as database to train the model on and was developed using Keras in Google Colab.

1. Introduction

Language modeling determines the next most likely word to appear in a sequence. Language modeling is important because it gives machines a way to understand qualitative information, rather than quantitative. Day to day usages of language modeling appear everywhere in our lives, like speech recognition, for example.

This project focuses on text generation. Machines can only understand so much of linguistics, and in their current state, they can only begin to breach the depth and complexity of language. A language model will be able to parse a great body of text and use pattern recognition in order to generate a fake body of text as a result. The language model can also operate at multiple different levels, as in, at the character level, sentence level, or paragraph level.

The goal of this project was to create fake reviews using a dataset from Yelp. In order to do this, a recurring neural network (RNN) would be used, as they are a common model used for pattern prediction. The RNN differs from a traditional neural network by having loops, which allows information to stay for longer. In tandem with the RNN, a long short-term memory (LSTM) architecture (Hochreiter et al., 1997) will also be implemented using Keras framework in Google Colab.

This paper is organized as follows. Section 2 will describe related work to this paper. Section 3 will describe the dataset in depth. Section 4 will describe the model construction and implementation. Section 5 will show the qualitative results. Section 6 will conclude this paper.

1. Related Work

The long short-term memory came to be because of issues with storing information over extended periods of time (Hochreiter et al., 1997). Neural networks struggle with decaying error back flow. The LSTM was designed by combining the recurring neural network and a gradient-based learning algorithm. The LSTM also utilizes a unique architecture involving memory cells and gate units. These two components can store information about the current state of the RNN. The gate units are for inputs and outputs. And these gate units, can pull inputs from other memory cells and then decide whether it should be stored.

It has been shown that LSTM models improve the language modeling error rate significantly, compared to your traditional RNN (Sundermeyer, et al., 2012). These models are also not suspect to the issues presented with RNNs when it comes to language modeling. Their model is compiled using categorical cross-entropy as the loss function and softmax. Using these methods, they found that the LSTM had greater reduction in error and could be sped timewise with small sacrifice in performance.

In language learning models, there is normally minimum word error (MWE). A recurring neural network with a long short-term memory architecture implemented with a different type of training in the model can be used to minimize layers (Hori, et al., 2016). This model used speech recognition and listened to multiple recordings to train the model. The model was able to reduce minimum word error by not using the cross-entropy method. By training the model to pay specific attention to MWE, the model was able to reduce errors significantly.

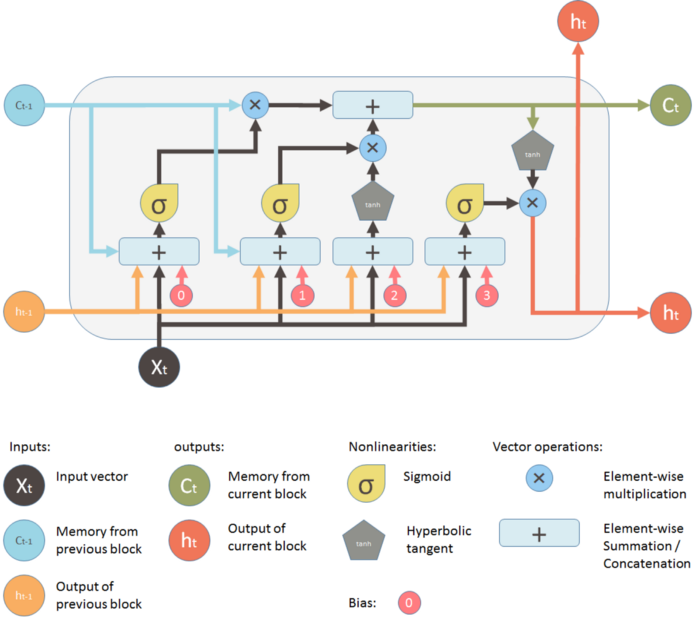
There are also different kinds of architecture for recurrent neural networks, not just the LSTM (Yogatama, et al., 2018). In fact, the LSTM has been shown to struggle with complicated sequences without supervision. The three architectures focused on are the random-access memory using an attention-based LSTM, an LSTM with greater capacity, and a stack augmented RNN. The stack model also performed the best out of these three. The stack model relies push, pop, and stay functions and is reliant on LSTM hidden states in order to determine whether to push, pop, or stay. By adding some sort of attention mechanism to the LSTM, the syntactic errors made by the model are reduced significantly.

Normally, language modeling regards bodies of texts are merely just sets of words, with little regard for grammar or word order (Mouse, et al., 2017). Sentiment analyzing (also known as opinion mining) identifies and analyzes opinions in the body of a work and determines the attitude a writer has. Sentiment analysis has normally involved in the past tokenization and parsing of texts, where the document is viewed as sequences rather than words. There are two types of LSTMs, unidirectional and bidirectional. The bidirectional model performs much better in terms of identifying context around a word since it reads to the right and the left of input text, rather than just in one direction like a unidirectional model.

1. Dataset and Features

The dataset was sourced from Kaggle. This dataset is officially provided by Yelp. The Yelp data base includes roughly 9GB of data, including check-ins, business information, reviews, and user profiles. For the purposes of this project, only the reviews were used from this dataset, which comprised about 3GB worth of data. Some of the reviews are shown in the table below.

Table 1: Reviews from the Yelp Database

****This is definitely my favorite fast food sub shop. Ingredients are everything, and everything I see and taste here tells me that they're using top-grade fresh ingredients. The brisket sandwich is probably my favorite... and it's the one my wife ALWAYS gets. Unlike her, I often bounce around the menu to try different things. Definitely a step up from Subway, Quiznos, Jimmy Johns, etc in my opinion. As with all of my reviews, I grade each place relative to what I perceive to be its peers - so five star compared to them.

I love Deagan's. I do. I really do. The atmosphere is cozy and festive. The shrimp tacos and house fries are my standbys. The fries are sometimes good and sometimes great, and the spicy dipping sauce they come with is to die for. The beer list is amazing and the cocktails are great. The prices are mid-level, so it's not a cheap dive you can go to every week, but rather a treat when you do. Try it out. You won't be disappointed!

This data needed preprocessing, however, before the model could train off it. Firstly, the data was mounted to a drive in Google Drive, so that Google Colab could run it. From here, the data was cleaned by removing all punctuation and lower casing the words, to make tokenization easier.

Raw text cannot be fed directly to the language model, so it must be tokenized. Keras has a framework Tokenizer API that must be built in order to prepare documents for the model. Tokenization is used in order to encode the text data as numbers (also known as a sequence) so they can be used as an input for the language model. These numbers are N-gram tokens, which is a special term for sequences collected from text data.

The next issue with preparing the data for the language model is that each sequence could have different lengths. By padding the data and making their lengths equal, the language model can finally read in the data.

1. Methods

The LSTM is an improved version of the RNN and is the crux of language learning, as aforementioned in the previous sections.

The LSTM architecture is used to avoid dependency issues that occur over long periods of time, this happens quite naturally as the RNN is designed to learn and retain information. The LSTM is capable of being selective about what information is accepted, which helps circumvent the issues with dependency.

In the context of language learning, in a traditional RNN, the RNN would not have an issue with predicting the last word based off of the rest of the words in the sentence, however, when it came to context of a specific word, the model would want to be able to identify cases where that word was used so it could analyze the context, with the traditional RNN, this is nigh impossible due to its architecture. The LSTM navigates this issue by using cell states and gates. The gates regulate what information goes in and out of the cell states.

Figure 1. Diagram of the LSTM Building Block.

Note. S. Y. 2016. *Understanding LSTM and its Diagrams* (<https://medium.com/mlreview/understanding-lstm-and-its-diagrams-37e2f46f1714>). Reprinted with permission.

The figure above should provide a more in-depth diagram of how the LSTM model works. This model shows how the input is considered throughout the memory cell and whether the information from the previous cell is stored or not. If it is store, then as you can see, the output is changed.

From here, it’s easy to understand how the model is built. Due to computational power constraints, only three layers were used in this model, an input layer, an LSTM layer, and a dropout layer. The model is configured using ‘sequential’ since we are reading in sequences and wants to read these sequences in order. The model is also compiled with the loss function categorical cross-entropy. Cross-entropy is intended for multi-class problems and will calculate a score that summarizes the probability distributions for all classes.

The LSTM layer was configured using ‘softmax’, since we want the model to read in the sequences and give a normalized probability distribution at the end. The softmax activation is typical with the categorial cross-entropy loss since the softmax function predicts the probability for each class. The resulting model is shown in the figure below.

Table

Description automatically generatedFigure 2. The Resulting Layers in the LSTM.

1. Experiments/Results/Discussion

The model was run for 100 epochs, which took roughly six hours to load with one team member’s computer that has the following specifications in the table below.

Table 2: The Specifications of Testing Machine

Processor: Intel i7-8750H @ 2.20Ghz (12 CPUs)  
----------------------------------------------------------------------  
RAM: 8.00GB  
----------------------------------------------------------------------  
GPU: NVIDIA GeForce GTX 1660 Ti

If the epochs were any lower, however, than 100, then the results were nonsensical completely and phrases of words would just repeat themselves, so we concluded that 100 epochs were the bare minimum amount needed in order to run the project.

In order to create reviews to verify that the model was properly trained and working as intended, a function needed to be written that would take one to two words and a desired length of the review and subsequently write a review.

In order to do this, the user inputted text would be tokenized, padded, and then passed into the trained model. The model would then string along predicted words to get a predicted sentence. Ten review results are shown in the table below.

Table 3: The Model’s Fake Reviews

**This** is a very cozy and cute australian cafe where the owner serves toasts with vegemite yum and flat whites and a whole assortment of other deliciousness the atmosphere is very relaxed and quiet the owner is ready to tell you about his experience and that send about 10 minutes of watching the last year was dead and it was like

**At this** is one of the best tasting ive ever had and every time we have gone in they have given us a free table about the front desk was not a bit of a drive and to one for it and didnt also have to come back to this location nothing worse than having your food

**My husband** and i started eating here a few months ago its close to home and convenient for a quick bite food is very good weve only tried the mahi mahi and fish n chips but recommend both prices are pretty resonable for the area and not much different than any other restaurant service is a hit and miss the young gals working there are mostly very friendly and attentive with an occasional employee not wanting to put

**The restaurant** is overcrowded with kitschy things on the walls and in every corner we also noticed all the flowers outside were fake and pretty worn out which made the first tacos experience i have been to a little cautious an hour experience has a change songs is a little white with a little cheese meals if you arent in the area

**Here** was a chicago style deep dish homemade type crust good sauce i ordered extra cheese italian sausage canadian bacon and jalepeno was good and definitely filling again to be to my reputable for some people now around salted caramel was the best thing i have to say that the problem i have been in the

**Notice** the most favorite and spot for a menu where only downside is no large guy ever so 15 good pretty friendly and fresh salmon mediocre they had a nice curry job my year and i both told that they would drop started extra restaurants to work to be singled out to a ymca looking in going to our table where

**Today** the locations are in for a treat this place was superior to the aveda spa i had last visited true confession i had a little fall in old scottsdale the day before my complaints is extremely minor not put in and the next day we were at the way to win the non way and quite a good way to wind down from an energizing show without losing the dazzle

**Dr. Seuss** is so overthetop ridiculous the prices are outlandish for the french side of the back of my beloved warp core breach booze bowl well see but its going to have to be pretty good to replace the hole in my heart from a heart and much value fyi i would have to return around shop while i was so impressed

**Tacos** is good but not a great place to be very confused on my saturday we also eat to the bar and the eggs was amazing and the grilled tomato service were also very good the staff was nice and a hard of badger alumni here for the game the alumni association runs raffles and has

**These** guys always get the job done right the store sizes were to a great room and a little chewy in the inside and a little chewy on the camera good sauce i have about 10 of a small with cups the nail potatoes rock it are all much but i might have to return when they open tomorrow and was

As you can see from the table above, the results of the model are somewhat sensical. In order to probe the model, we picked five words that were commonly used at the beginning of reviews such as “My husband,” “At this,” “This,” “The restaurant,” “Today.” The reviews that start with these words have a little more context around them, assuming that the model is more familiar with their repetitive usage throughout the database, whereas phrases like “Dr. Seuss” the model just assumes it’s the name of some weird restaurant, with more data from multiple different sources, it’s theoretically assumed that the model could learn context. As for now, the model does not grasp grammatical concepts or even context of words. This explains why some of the word choices of the model are somewhat strange to the human eye.

1. Conclusion and Future Work

While the model does generate unique reviews, they have no understanding of context with the model unable to really comprehend grammatical structure. If given more time, it would have also been theoretically beneficial to develop the model to be bidirectional, rather than unidirectional, since the model would be able to better understand the context surrounding the words and possibly even punctuation.

For the future, it would be imperative to implement more grammatical structure to the dataset, by keeping punctuation and capitalization somehow, and circumventing the restrictions of tokenization for the model.

Some issues with the model also had to do with computational power consumption. The model took six hours to process 100 epochs, and if given a better machine or more time, the model could possibly process more epochs and create an even more accurate resulting review.

This model proves the power of language learning models and their ability to harness predicative power to generate bodies of text.

1. Contribution

Elizabeth Willard worked on the report and part of the code. Jeffrey Edrington worked on the code and the presentation.

References

Contributors, T. (2020, March 02). What is Language Modeling? Retrieved December 17, 2020, from https://searchenterpriseai.techtarget.com/definition/language-modeling.

Semeniuta, Stanislau and Severyn, Aliaksei and Barth, Erhardt. *A Hybrid Convolutional Variational Autoencoder for Text Generation.* Cornell University, Ithaca, NY, 2017.

Hochreiter, Sepp and Schmidhuber, Jurgen. Long Short-Term Memory. In Langley, P. (ed.), *Neural Computation,* pp. 1735–1780, 1997.

Hori, Takaaki and Hori, Chiori and Watanabe, Shinji and Hershey, Josh R., Minimum Word Error Training of Long Short-Term Memory Recurrent Neural Network Language Models for Speech Recognition*.* IEEE, 2016*.*

Yogatama, Dani and Miao, Yishu and Melis, Gabor and Ling, Wang and Kuncoro, Adhiguna and Dyer, Chris and Blunson, Phil. Memory Architectures in Recurrent Neural Network Language Models*.* ICLR, 2018.

Mouse, Amr El-Desoky and Schuller, Bjorn. *Contextual* Bidirectional Long Short-Term Memory Recurrent Neural Network language Models: A Generative Approach To Sentiment Analysis. *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics:* Volume 1, Long Papers, pp. 1023-1032.Imperial College, London, UK,

Sundermeyer, Martin and Schluter, Ralf and Ney, Hermann. LSTM Neural Networks for Language Modeling, *Interspeech,* pp. 194-197, 2012.

1. [↑](#footnote-ref-1)