Using Convolutional Neural Networks And Long-Short Term Memory Cells To Analyze Syntactical Structure Within Complex Communication Systems.

Peter Ma

Purpose: is to help advance the search for Extraterrestrial Intelligent (ETI) signals by using deep learning algorithms to understand intelligent complex communication systems on Earth. By understanding the communication systems on Earth we can develop computationally sensitive search methods.

Information Theory 1 | Deep Learning 2 | CNN-LSTM 3 | ...

Prior to this proposal, many scientists and researchers had made significant strides in the search for ETI signals. One such notable framework, developed at the SETI Institute, applys a technological and an informational filter on potential ETI signals. The new methods discussed in this paper will heavily base its proposal on said frameworks.

Problems and Methods

There exists 2 main **problems** within the status quo:

1. Radio telescopes, such as the Arecibo telescope, generate nearly 100 million samples of data per second from scanning the sky. This immense amount of data requires an analysis that can detect signs of an intelligent communication system. The SETI@HOME project cannot accomplish such a task because their current method involves detecting narrow-band radio signals, which can only detect signs of technology rather than inate intelligence.

2. The information filter (framework used to search for intelligent communication systems) requires manual operation that cannot be easily accelerated by computer automation. For example the process of signals labeling is done by humans because computer algorithms preform the same task equally as fast and as accurate. This framework is simply cannot scale to such a huge data set.

Method: To help narrow the initial search for ETI signals, the use of deep learning can help increase the sensitivity of the search for both signs of technology and intelligence. This method will introduce the use of a CNN-LSTM network to extract features from spatial and sequential information from a given candidate signal. This deep learning algorithm would then be cross-validated by an information filter driven by statistical analysis.

1. Introduction - The Information Filter

This method of search was first developed by the Dr.Laurence Doyle @ the SETI Institute. Firstly, the information filter seeks to understand the informational capacity and the syntactical structure within an unknown communication system. By understanding the informational capacity and the structure of the communication system we can gauge the intelligence of the sender because intelligence is directly dependent on the sophistication of a sender's communication system. If a language is too primitive to hold complex ideas, we can assume the sender's intelligence is limited by its communication system.

Informational Capacity

If the informational capacity of a communication system fails to either hold knowledge, it is ruled as a non-intelligent system of communication. For example, it is impossible to translate Shakespeare into a squirrel monkey's language because it's communication system is too primitive to hold the complex concepts within Shakespeare's text.

The carrying capacity of a communication system can be determined by measuring the entropy within each system. Here we borrow Claude Shannon's measurement of entropy, where n denotes the rank/level of entropy. The larger the n the greater the informational capacity of the system.

$$H_n = -\sum_{i,j...n}^{N} p(i,j,...n)log_2(p_{i,j...n-1}(i,j,...n)) \qquad [1]$$

Syntactical Structure

Furthermore, the syntactical structure of a communication system is also a necessary requirement for an intelligent signal. This is because, with any language system, there must exist rules that allows a receiver to decode a sender's message. Thus you must have grammar, syntax and structure that constrains the variation of messages. Without structure, a receiver will never comprehend a sender's signal, which defies the purpose communication. Thus in order to defiferntiate an intelligent signal the signal must also contain structure.

That being said, we can detect the existence of structure within an unknown communication system by looking at Zipf's law. Zipf's law shows that for some dataset, the graph of a sorted set of frequencies of appearance of distinct elements within the set, will approximate a linear function with a negative 45° slope (when the axis are in intervals of powers of ten).

Zipf's law is simply an observed behavior across every known intelligent language on Earth. This includes all recorded human languages, dolphin whistles and whale songs. We postulate that Zipf's law is an indication of structure that limits randomness within a language. This function is defined by Zipf as the following.

$$f(k; s, N) = \frac{1/k^s}{\sum_{n=1}^{N} (1/n^s)} \qquad [2]$$

Information Filter In Practice

The information filter requires us the understand the signals make up of a message. This is similar to understanding the alphabet of an alien language. This can be done by arranging a K-means clustering algorithm on a dataset and clustering the messages into 64 similar categories.

After calculating the individual signal elements we can then classify the elements within the set and sort the set based on its frequency of appearance. This requires a human to label the symbols within a message. After sorting the data we can then calculate the Zipfian Distribution of f(k;s,n) on the set and if the values approximate to the values given by the function we can say it obeys Zipf's law.

Problems

Since the informational filter is very slow and requires humans operators to test and look for each signal within the data set. This not an efficient way of scaling to larger data sets as it cannot actively increase SETI@home's computational sensitivity. This approach is however very effective in confirming a message and is a highly effective tool at cross-validating a sample.

2. Deep Learning Proposal

Since the informational filter cannot actively increase the sensitivity of our search method, a rapid computational filter trained on thousands of intelligent signals may help expedite and advance our search for ETI signals. This way we can develop an all encompassing framework that takes into account a mathematical analysis, a technological filter and a data-driven analysis on ETI signals.

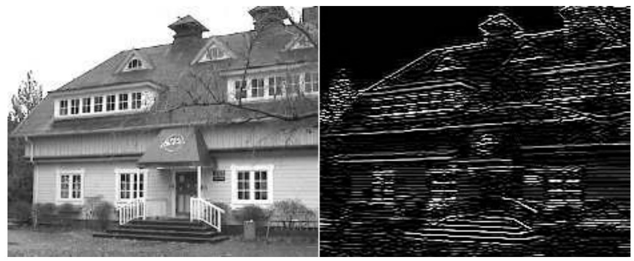
Currently, deep learning has evolved to a point where models can effectively analyze sequential and spatial data in great detail. Here we will explore a deep learning model that can approximate the understanding of syntax within a communication system, the Convolutional-LSTM model.

Convolution

Firstly, the proposed model contains two main parts: a convolutional layer and a LSTM layer. Both layers will work together to help analyze the data.

Convolutional Neural Networks (CNN) are models built to extract features from spatial data. This model sweeps through a sample and applys a convolution operation between the dataset and a kernel. The output of this layer will only contain the features of original data. For example, if a filter $\begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix}$ is convolved with an image, the result will only show horizontal edges within the image.

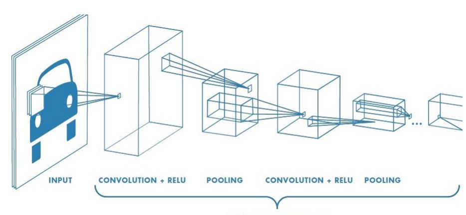
Fig. 1. Feature extraction of edges



The convolution operation is, seen in eq [3], where $Input \in^{m.n}, kernel \in^{t}, feature \in^{F,n_c}$ a method of feature extraction.The values of the filters are learn-able parameters, meaning depending on the dataset, the model can learn how to extract features of the dataset.

$$C(I, k) = (I \circledast k)[F, n_c] = \sum_j \sum_l k[l] \cdot I[F - l, n_c - j] = f[F, n_c] \qquad [3]$$

Fig. 2. Structure of a traditional Convolutional Neural Network



Long Short Term Memory Cell

Secondly, the purpose of the LSTM cell is to analyze the temporal continuity of the dataset. When analyzing human text, the first part of any sentence must conform with the last part of the sentence in order to transfer knowledge. This is important since this model specializes in understanding structural continuity with the message. If there is no continuity it can easily detect its anomalies. This makes the model highly suited for this type of operation.

Furthermore, this model contains gates that help the model remember and forget certain features within each part of the message. This feature allows the model to train on huge datasets and operate on a far larger scale than a traditional Recurrent Neural Network.

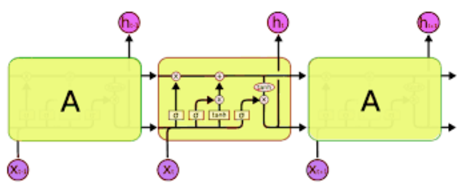
For the LSTM model itself we have a few main gates that help preform the forward passage of data. This is described as Forget Gate (f) eq[4], Input Gate(I) eq[5], Output Gate (O) eq[6] and Cndate(C').

$$C_t = C_{t-1}f_t + (sigmoid(X_t + H_{t-1}) * C_t') \quad [4]$$

$$I_t = tanh(X_t + H_{t-1})sigmoid(X_t + H_{t-1}) \quad [5]$$

$$O = sigmoid(X_t + H_{t-1})tanh(C_t) \quad [6]$$

Fig. 3. Structure of LSTM cell



The structure of the cell can be seen in the figure [3]. This cell propagates forward continuously until it reaches the end of the sequence. Each element within the dataset is placed into the model and the next input includes the output from the last. This recurrent nature allows the model to selectively remember and forget the key features within the model. This information is then passed into an Artifical Neural Network where the final classification can be made.

3. Implementation

To implement the CNN-LSTM we first need to test its effectiveness against pre-exisitng models in order to prove its suitability for the task at hand. The goal is to test 4 types of neural networks and observe the best performing model. In this paper I will implement a traditional CNN, LSTM, and CNN-LSTM and ConvLSTM. All models will be trained with the same SGD optimizer, trained on the same hardware and the same dataset in order to give each model a fair chance.

Traditional Convolution Neural Network

This model contains 1.60×10^7 trainable weights with 5 convolutional layers. The performance, shown in figure 4, is very poor in terms of generalization. The model cannot learn the behaviours of the dataset, instead it memorized the dataset. It's failure is shown by its low validation accuracy and high validation loss. This is to be expected as the CNN model is not sensitive to sequential variation as opposed to an LSTM model. You can see the implemented architecture in figure 5.

Fig. 4. CNN preformance

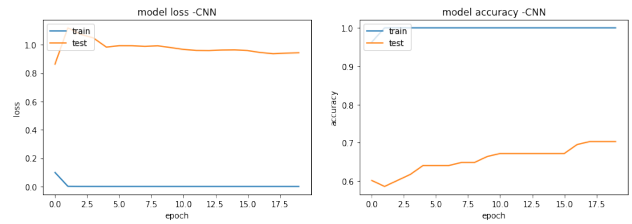
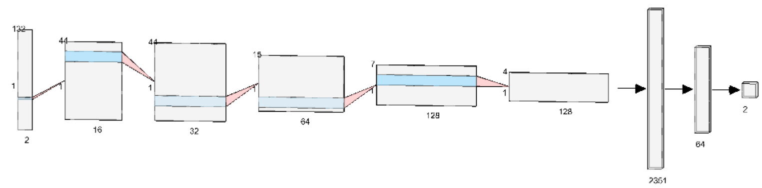


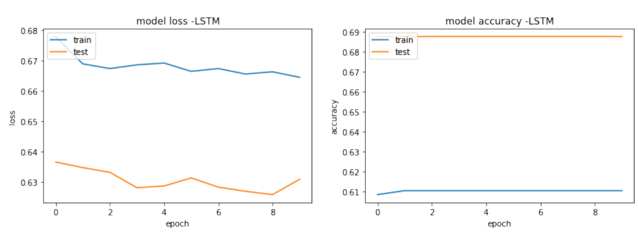
Fig. 5. CNN Architecture (Dimensions reduced by factor 10^2)



Traditional LSTM Cell

The traditional LSTM cell contains only 2.00×10^5 trainable parameters. The LSTM cell slides along the data and takes in input sequentially. Despite preforming better than the CNN model, this model fails to train. The model cannot achieve a performance level greater than 65% accuracy. This maybe caused by the high variability of the data. Despite the input data being compressed to a lower frequency (4.41kHz instead of 44.1kHz), the LSTM gates cannot remember information from such a long sequence of 132,300 values. This is the result of vanishing gradients. Thus the model fails to train. This is shown its training performance.

Fig. 6. LSTM performance

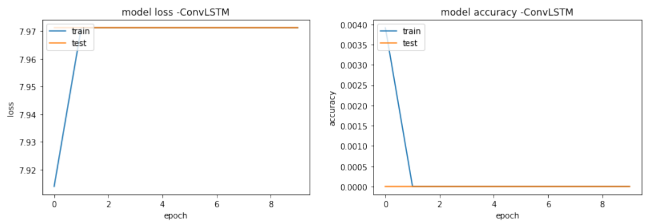


Convolutional LSTM Cell

The Convolutional LSTM is still an LSTM model that, instead of reading each input, reads a group of inputs and extracts the features from each group and then it passes it onto an LSTM cell. Each cell contains a convolution layer. This is different from a CNN-LSTM where entire input is convolved and fitted into an LSTM.

That being said, this model appears to preform the worst, as it suffers from vanishing gradients even when the model contains LeakyRelu activations, dropout and a reduced number of parameters. The reasons for its poor performance is caused by the model's size. This can be see in training.

Fig. 7. ConvLSTM performance



CNN - LSTM Cell

The CNN-LSTM model uses an initial convolution layer which extracts the features from the sample before passing it onto the LSTM network to process the sequential data. This way, the convolutional layer learns to condense the dataset to features which increases training speed. The LSTM layer helps the convolution layer to detect sequential variations and thus these two work together to help make an accurate prediction. This performance boost can be seen in the training and in the validation test figure [8].

Fig. 8. CNN-LSTM performance

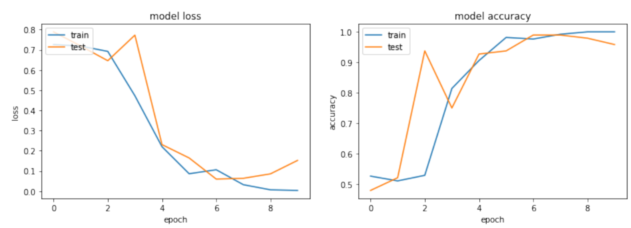
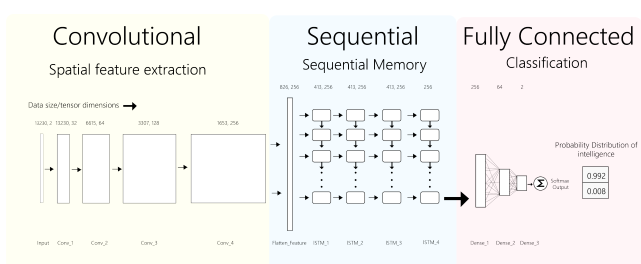


Fig. 9. CNN-LSTM Architecture (see full image on last page)



Dataset Composition

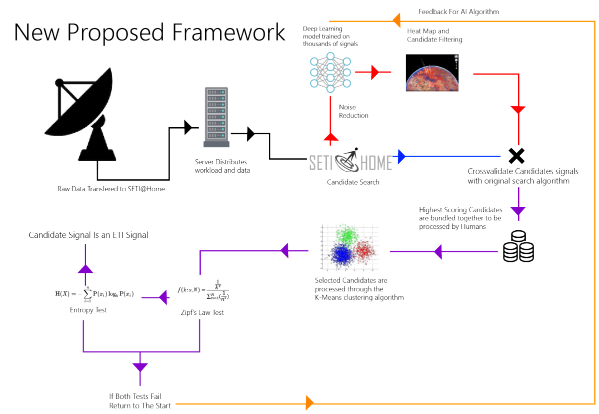
The data set used contains a variety of sound samples from various elements of human conversations, music, movies, dialogues and non-human samples which contains wind, waves, crackling of the fire and of static TV background sounds. This created a total data set of around 10 hours of continuous audio. Each sound file is converted into an Numpy array which contains a 839 samples of 30 second sound clip with 2 channels of audio. This audio clip converted to a sampling at a rate of 4.41kHz which creates an input shape that looks like: $[839, 132300, 2]$

4. Integration

The result of this tests shows that the CNN-LSTM model can, in fact, detect changes in sequential data and generalize the informational filter. That being said, it is not as precise as the original framework. Instead this computational tool is used to increase the sensitivity of the existing technology used to search for ETI signals.

The goal here is to develop a tool that can work along side the pre-existing framework as a method of cross-validation. This way we can both increase the algorithm's sensitivity. Figure [10] shows the basic framework.

Fig. 10. Proposed Framework (see full image on last page)



The integration process would look like this: first the raw data would be passed into the servers which gets distributed to the SETI@HOME network. This network then processes samples which denoises the data of unwanted artifacts. The cleaned data is then passed into the Deep Learning model which generates a probability distribution for signs of intelligence. This would create a heat map of the entire dataset. While the model is executing, the computer would then process the original candidate signals and at the end, the two datasets are cross-validated validate each other. From that point the cross-validated data is put through a more precise validation process of information filtering. This will run the K-means clustering algorithm on the candidate signals where the sliced signals are put to a zipf law test and a entropy test. If the signal passes through all checkpoints with no failures, the signal is considered an ETI signal.

5. Conclusion

In conclusion, the development of a deep learning model can help increase the sensitivity of our computational search. By combining the technology filter, the informational filter, and the deep learning model we can further push the capabilities of our SETI search.

Fig. 11. CNN-LSTM Model Architecture

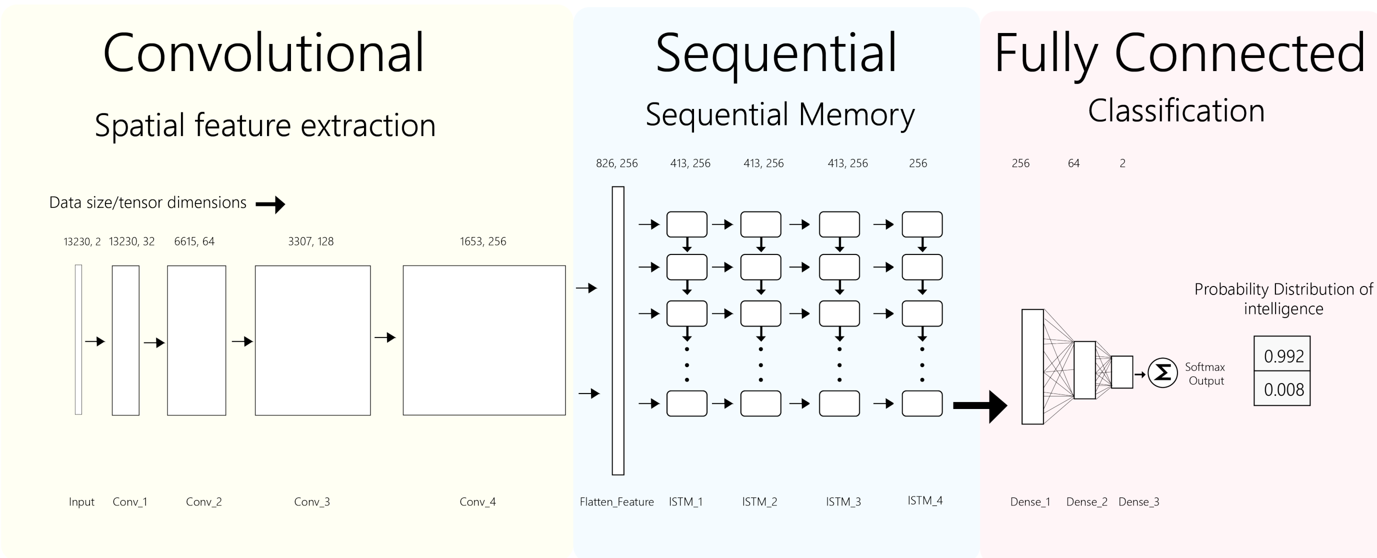


Fig. 12. Proposed Framework