Comparative Study on the Performance of LSTM Networks for STT Conversion Using Variations in Attention Mechanism Approaches and Loss Functions

Kruthi N Raj

Pre-Final Year(B.E.) Student, Department of Computer Science and Engineering, Bangalore Institute of Technology, Bangalore

---***---

Abstract - *Speech-to-text (STT) is the process of transforming verbal speech into text. This technique is also known as speech recognition. Although the words are almost similar, speech recognition is sometimes used to refer to the broader process of extracting meaning from speech, or interpreting speech. The Continuous Automatic Speech Recognition (ASR) system captures the sound wave of speech and generates the appropriate text. The subject of Natural Language Processing (NLP) research, such as image and speech recognition, is rapidly changing from statistical approaches to neural networks. These systems have historically used Hidden Markov Models (HMMs), which use a stochastic process to describe the sound of speech. The popularity of ASR deep learning systems has grown with increasing computer power and the amount of training data. In this paper, I compare the changes in attention mechanisms and loss functions made to a Speech-To-Text (STT) transformation system built using Long Short Term Memory (LSTM). Attention techniques used include Bahdanau's attention and Luong's attention. The loss functions used are the cross entropy loss and the Connectionism Time Classification (CTC) loss.*

***Key Words*: Speech-To-Text; Speech recognition; Long short term memory; attention mechanism; Bahdanau Attention; Luong Attention; loss function; Cross-entropy loss; Connectionist temporal classification**

1. INTRODUCTION

The capacity of a computer or software to recognise and convert spoken words into legible writing is referred to as speech-to-text, also known as speech recognition. Computer science, linguistics, and computer engineering are all involved in speech recognition. Many current technology and devices, as well as text-focused apps, may include speech recognition functions to make device use easier or hands-free. This programme may be especially useful for people who have hearing loss. Acoustic and linguistic modelling methods are used to recognise speech. Language modelling associates sounds with word sequences to aid in distinguishing between words that sound similar; acoustic modelling depicts the relationship between linguistic units of speech and audio signals. In order to improve system accuracy, Hidden Markov models were commonly employed to recognise temporal patterns in speech. This method will randomly change systems, presuming that future states are independent of past states. Other methods of speech recognition include natural language processing (NLP) and N-grams.

The use of natural language processing (NLP) accelerates and simplifies the process of speech recognition. N-Grams, on the other hand, are a simple way of modelling language. They contribute to the development of a sequence's probability distribution. AI and Machine Learning are being utilised to develop more sophisticated speech recognition software. These systems use grammar, structure, syntax, and audio and voice signal synthesis to process speech. The more machine learning software is used, the more it learns, making topics like accents easier to understand. Deep Learning (DL) has shown remarkable success in a wide range of AI applications. One such application is automatic speech recognition (ASR). Deep learning for voice recognition using sequence-to-sequence data has recently emerged as a new paradigm. Sequence-to-sequence (seq2seq) models are gaining popularity in the field of automated speech recognition (ASR). A number of sequence-to-sequence models have been studied in the literature. In this paper, I describe the use of one such seq2seq model, Long-Short Term Memory (LSTM) layers, for STT translation. The proposed study examines the attention mechanisms Bahdanau Attention and Luong Attention, as well as the loss functions Cross-entropy loss and Connectionist Temporal Classification (CTC) loss. The STT conversion performance of the LSTM trained on the TensorFlow Speech Command Dataset was evaluated using multi-class accuracy score. In addition, the performance of four LSTM based models for STT conversion with combinations of attention mechanisms and loss functions mentioned above, are compared using accuracy score.

2. HISTORY OF SPEECH RECOGNITION

Investigation of speech recognition systems dates back to the 1950s. The timeline in Figure 1 summaries the evolution of speech recognition systems over the past 70 years.

1. Audrey developed in 1952 by three bell lab researchers recognised digits.

2. 10 years later, IBM introduced IBM Shoebox capable of regaining 16 words and digits. It was capable of identifying basic mathematical operation commands and computing the solution.

3. In the 1970s, harpy was developed, being able to recognise 1011 words.

4. In 1980s, speech recognition systems were implemented using Hidden Markov Models (HMM)

5. In 2001, Google introduced Voice Search, the first voice-enable application that helped users to solve their queries by speaking to a machine.

6. By 2011, voice command based virtual assistants like Apple's Siri, Amazons's Alexa and Google's Home were released.

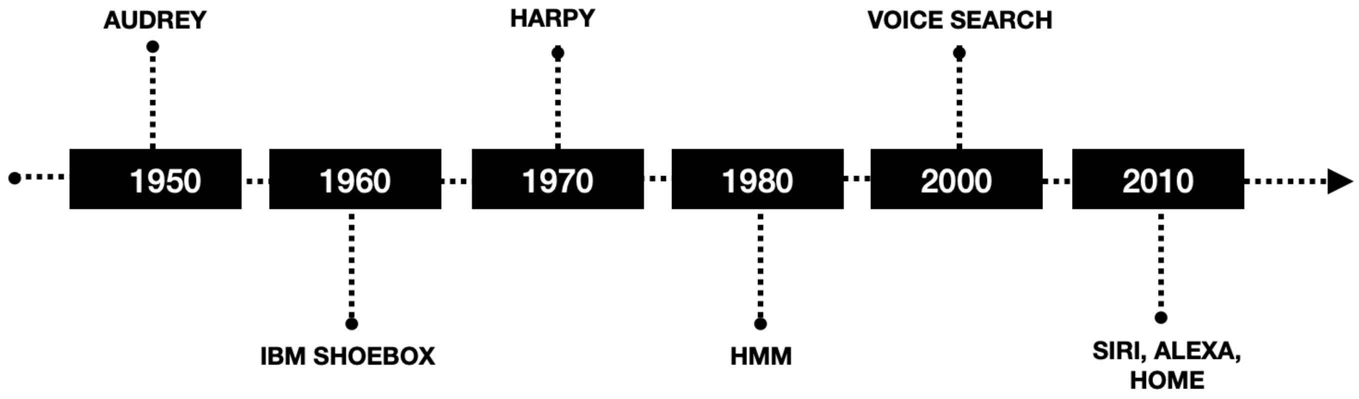

Figure 1 : Speech recognition system evolution timeline

3. SIGNAL PROCESSING

Brief overview of Audio Signal and processing

Audio signals are analog or digital representations of sounds, and audio signal processing is the application of algorithms and techniques to these signals. An audio signal's essential characteristics include amplitude, crest and trough, wavelength, cycle, and frequency. Audio signals, which are often in the form in analog form, use a lot of memory and are computationally expensive. As a result, these analogue signals must be transformed into digital form, which is known as sampling the signal.

Techniques for Extracting Audio Signal Features

The initial stage in STT conversion is to extract features from the audio source that will be fed into the model as input. Extraction methods are classified into three types:

1. Time-domain - The features in this case are audio signal amplitudes captured at various time intervals. The drawback is that it ignores the frequency component entirely.

2. Frequency-domain - The features in this category are audio signal amplitudes captured at various frequencies. The drawback is that it ignores the time component entirely.

3. Spectrogram - A spectrogram is a 2D representation of time and frequency in which each point indicates the amplitude of a certain frequency at a specific time in terms of colour intensity.

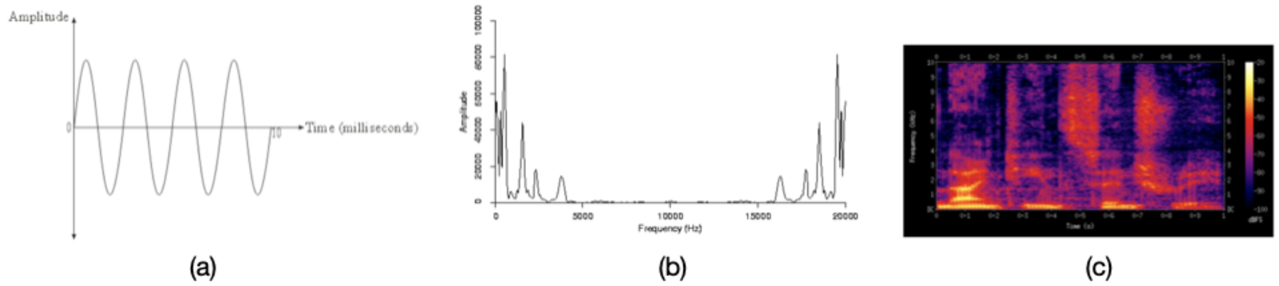

Figure 2 : Audio feature extraction techniques. (a) Tine-domain; (b) Frequency-domain; (c) Spectrogram

4. LONG SHORT TERM MEMORY

Recurrent Neural Network (RNN) is a neural network with internal memory that can handle sequential data, that is, data whose parts are connected to each other. Each input given to an RNN is reliant on the previous ones, and they memorise the information they process. But these RNN networks have drawbacks. They fail to store information for a longer length of time, lack finer control over which bits of context are processed further and which are disregarded, and exhibit exploding and disappearing gradients throughout the network training process due to backtracking. Long Short Term Memory (LSTM), a kind of recurrent neural network (RNN), were created largely to solve situations in which RNNs fail. As they have a deliberately implanted memory unit called a cell, these LSTMs are naturally capable of storing information for a lengthy period of time. A detailed discussion of their architecture and operation is beyond the scope of this paper.

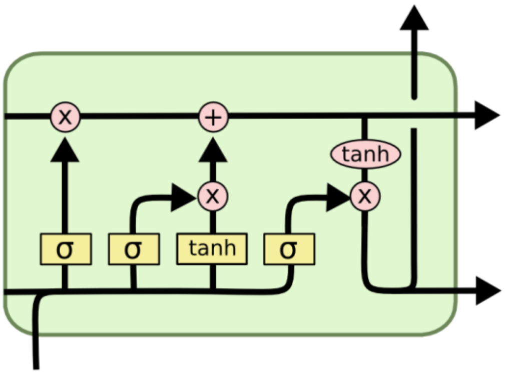

Figure 3: LSTM network

5. ATTETION MECHANISM

One of the most important concepts introduced in Deep Learning research is the attention mechanism. The most fundamental definition of attention is the ability to choose focus on a few things while disregarding others. The attention mechanism seeks to apply the same selective focus action into deep neural networks. This technique improves on encoder-decoder RNNs and LSTMs, which have significant disadvantages such as not being able to perform effectively on long sentences for RNNs, being forgetful in particular regions for LSTMs, and being unable to prioritise certain input words over others. The attention mechanism implemented not only considers all incoming words, but each of them is assigned relative importance. Rather than a single vector representation for each sentence, the mechanism prioritises individual input vectors of the input sequence depending on their attention weights.

The 2 types of attention mechanism implemented in this study are Bahdanau's attention and Luong's attention

Bahdanau's Attention

Bahdanau et al. presented an attention mechanism that learns to align and translate at the same time. It is an additive mechanism that combines encoder and decoder states in a linear fashion. In contrast to the seq2seq paradigm without attention, all hidden states of the encoder (forward and backward) and decoder create the context vector. Input and output sequences are aligned using an alignment score parameterised by a feed-forward network, which aids in directing attention to the most important information. The model predicts the target word based on the context vector associated with the source position and previously produced target words.

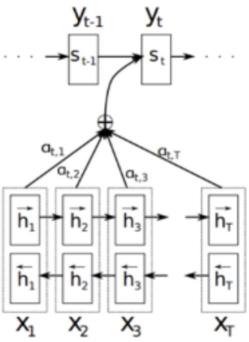

Figure 4: Bahdanau attention mechanism

Luong's attention

In contrast to Bahdanau attention, the Luong mechanism is multiplicative. It employs metric multiplication to convert encoder and decoder states into attention scores, making it both quicker and more space-efficient. Luong proposed two forms of attention based on the place of attention in the source sequence: 1) Global and 2) Local. Local attention is focused on a limited fraction of the source locations per target word, whereas global attention is focused on all source positions. Both forms of attention generate a context vector in order to gather relevant source information in order to identify the current target word. Attention vectors given as inputs to successive time steps provide information to the model about previous alignment choices. However, both forms of attention originate context vectors in distinct ways. The global context vector is the weighted average of all the source hidden states based on the alignment vector. It uses all of the source sequence words to forecast target words, which is computationally costly and limits its ability to analyse lengthier sentences. This disadvantage is overcome by utilising Local attention, in which the aligned location is chosen monotonically or predictively.

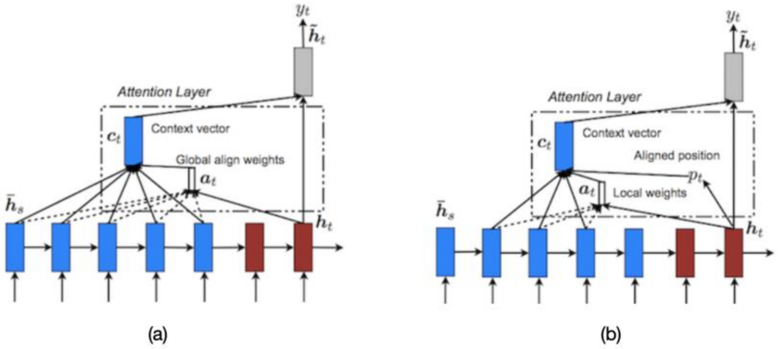

Figure 5: (a) Luong global attention mechanism ; (b) Luong local attention mechanism

The two primary differences between the Bahdanau and Luong attention methods are as follows:

1. Bahdanau calculates the alignment vector using the output of the previous time step, whereas Luong utilises the current decoder's hidden state.

2. Bahdanau solely employs the concat score alignment model, whereas Luong employs the dot, general, and concat alignment score models.

6. LOSS FUNCTION

The loss function is used to evaluate the model's performance, and it is designed to measure as the gap between the expected and predicted outputs. The greater the value of the loss function, the farther the predicted output deviates from the expected one. The loss function's output is sent back to the network, allowing it to adapt itself. This is referred to as learning. The key goal is to optimise our algorithm such that the loss function is as little as possible.

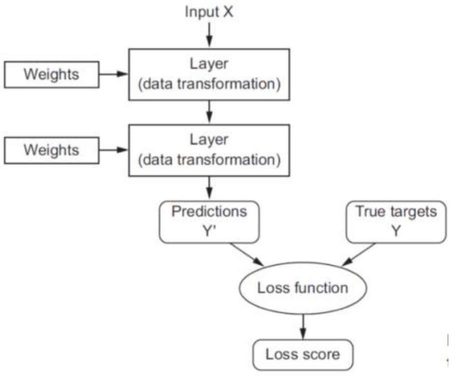

Figure 6: Outline of a network with loss function used to measure the performance of and optimise the model

The 2 loss functions utilised in this study are cross-entropy loss function and CTC loss.

Cross-Entropy Loss Function

Cross-entropy is determined by the difference between two probability distributions for a given random variable or series of occurrences. This loss function, also known as log loss, compares the predicted class probability to the actual anticipated output and generates a log score, with which it penalises the probability depending on how distant it is from the real value. The penalty is log in nature, with a high value near to 1 for a significant difference in probability and a small value close to 0 for projected probability close to real value probability. Cross-entropy is used to update the model's weights during training in order to optimise the method, with the goal of minimising the loss score as much as possible, with 0 being the optimum result.

$$L_{\mathrm{CE}} = -\sum_{i=1}^{n} t_i \log(p_i), \quad \text{for n classes,}$$

where t_i is the truth label and p_i is the Softmax probability for the i^{th} class.

Figure 7: Cross-entropy mathematical formula where log is computed to base 2

Connectionist temporal classification (CTC) loss

CTC is essentially a loss function, similar to cross-entropy, that is frequently used in applications such as voice recognition to solve sequence issues with time as a variable. Without this, an aligned dataset is necessary, which in the context of speech recognition means that every character in a transcription must be aligned in its exact place in the audio file. Because of its flexibility to assign a probability to any label given an input, CTC does not require aligned data. It calculates a loss between a continuous (unsegmented) time series and a target sequence by summing the likelihood of all potential alignments between the input and the label, resulting in a loss value that is differentiable with regard to each input node. The alignment of input to target is expected to be “many-to-one,” limiting the length of the target sequence to the length of the input. This makes voice recognition much easy to implement in CTC.

the ground truth of the word sequence

acoustic frames

$$\mathcal{L}_{CTC} = -\log P(\boldsymbol{S}|\boldsymbol{X})$$

$$P(\boldsymbol{S}|\boldsymbol{X}) = \sum_{c \in A(\boldsymbol{S})} P(\boldsymbol{C}|\boldsymbol{X})$$

sum over all possible paths
(e.g. ccεaaεtt, ccccaεtt, cεaεtttt, ...)

$$P(\boldsymbol{C}|\boldsymbol{X}) = \prod_{t=1}^{T} y(c_t, t)$$

joint probability of a path
(e.g. ccεaaεtt)

Figure 8: CTC loss function

7. EXPERIMENTAL WORK

The NVIDIA Tesla T4 GPU, which is accessible in Google Colab, was used to train the models.

7.1. DATASET

TensorFlow Speech Commands Datasets contain 65,000 one-second audio recordings of thirty-second word utterances from thousands of users. In the background noise folder, there are additional lengthier audio recordings of silence. The dataset includes train and test folders containing audio files that are provided with labels. The test dataset includes audio commands labelled as yes, no, up, down, left, right, on, off, stop, and go. Figure 10 depicts the number of recordings for each label in the train dataset.

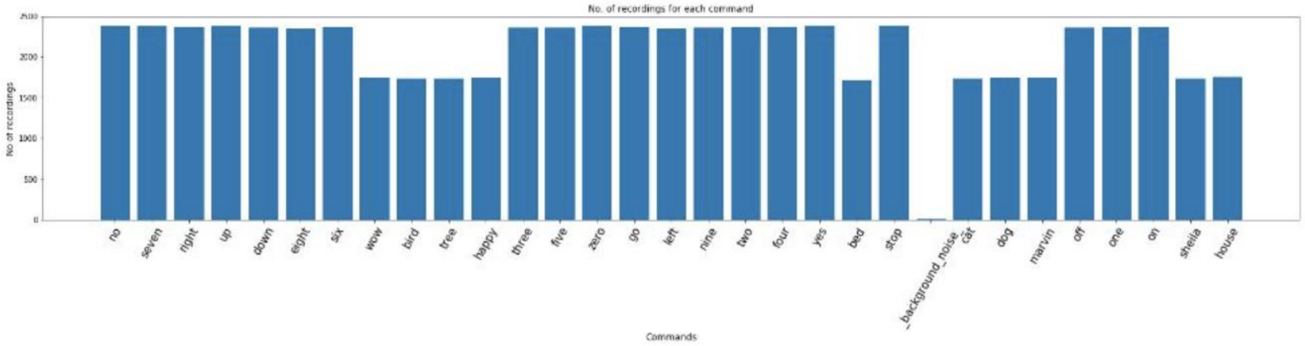

Figure 9: Number of audio recording per label

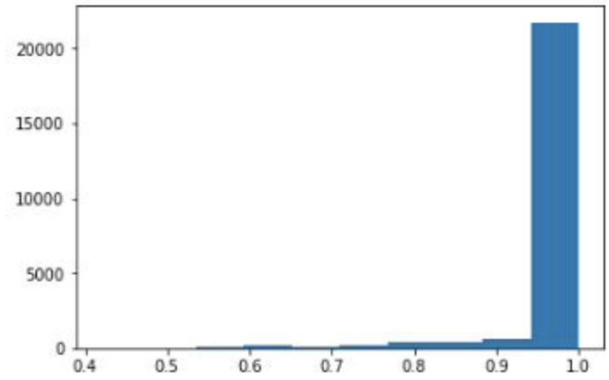

Figure 10: Duration of the audio recordings in the dataset

7.2. DATA EXPLORATION AND VISUALISATION

For better understanding of the audio data, the audio signals are visualised in time series domain.

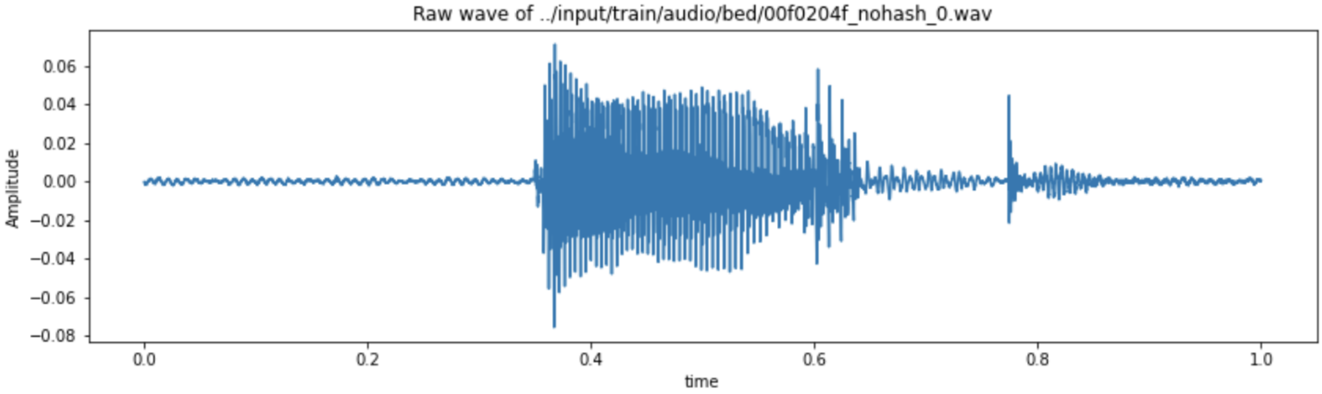

Figure 11: Raw waveform of the audio labelled 'bed'

7.3. DATA PREPROCESSING

1. Resampling - The signal's sampling rate is 16,000 Hz. It is visible from the above data exploration that there are a few recordings with a duration of less than one second and a sample rate that is far too high. Because 8000 Hz is the most prevalent speech-related frequency, the audio is resampled to that frequency.

2. Audio lasting less than one second is deleted.

7.4. MODEL ARCHITECTURE

This study implements four variations of the LSTM-based model for STT conversion.

1. LSTM - Luong - CTC

2. LSTM - Bahdanau - CTC

3. LSTM - Luong - Cross-entropy

4. LSTM - Bahdanau - Cross-entropy

7.5. EXPERIMENTAL RESULTS

To evaluate the models' performance, the projected text for the test input audio from each model is compared to the actual text label provided with the audio in the dataset. The multi-class accuracy score is used to assess the performance of the models. This metric provides an overall assessment of how well the model predicts over the whole dataset. Table 1 displays the accuracy score for each of the five models. As shown in Table 1, the LSTM network with Bahdanau attention and cross-entropy loss achieves the best accuracy for STT conversion on the test dataset, whereas the combination of Luong Attention and CTC loss with the LSTM network achieves the lowest accuracy out of the four models.

MODEL	ACCURACY SCORE
LSTM - Luong - CTC	0.8135
LSTM - Bahdanau - CTC	0.8743
LSTM - Luong - Cross-entropy	0.8801
LSTM - Bahdanau - Cross-entropy	0.8971

Table 1: Performance of the models on the test audio data evaluated using accuracy score

8. CONCLUSION

In this article, I built and analysed four LSTM-based models for STT conversion that differed in their attention mechanisms and loss functions. TensorFlow Speech Commands Datasets were used to train and test the models. The multi-class accuracy score was used to evaluate the performance of each model. Among the four models, the LSTM network with Bahdanau attention and cross-entropy loss achieved the best accuracy, whereas the LSTM network with Luong attention and CTC loss achieves the lowest accuracy.

REFERENCES

1) Chorowski, J., Bahdanau, D., Cho, K., & Bengio, Y. (2014, December 4). End-to-end Continuous Speech Recognition Using Attention-based Recurrent NN: First Results. arXiv.org. https://arxiv.org/abs/1412.1602.

2) Chorowski, J., Bahdanau, D., Serdyuk, D., Cho, K., & Bengio, Y. (2015, June 24). Attention-Based Models for Speech Recognition. arXiv.org. https://arxiv.org/abs/1506.07503.

3) Long Short-term Memory Recurrent Neural Network-based Acoustic Model Using Connectionist Temporal Classification on a Large-scale Training Corpus. (n.d.). Long short-term memory recurrent neural network-based acoustic model using connectionist temporal classification on a large-scale training corpus. https://ieeexplore.ieee.org/abstract/document/8068761.

4) Luong, M., Pham, H., & Manning, C. D. (2015, August 17). Effective Approaches To Attention-based Neural Machine Translation. arXiv.org. https://arxiv.org/abs/1508.04025.

5) Sak, H., Senior, A., Rao, K., & Beaufays, F. (2015, July 24). Fast and Accurate Recurrent Neural Network Acoustic Models for Speech Recognition. arXiv.org. https://arxiv.org/abs/1507.06947.

6) Sherstinsky, A. (2018, August 9). Fundamentals of Recurrent Neural Network (RNN) And Long Short-Term Memory (LSTM) Network. arXiv.org. https://arxiv.org/abs/1808.03314.

7) Soltau, H., Liao, H., & Sak, H. (2016, October 31). Neural Speech Recognizer: Acoustic-to-Word LSTM Model for Large Vocabulary Speech Recognition. arXiv.org. https://arxiv.org/abs/1610.09975.

8) Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Kaiser, Ł., Gouws, S., Kato, Y., Kudo, T., Kazawa, H.,... Dean, J. (2016, September 26).

9) Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. arXiv.org. https://arxiv.org/abs/1609.08144.

10) Yin, W., Kann, K., Yu, M., & Schütze, H. (2017, February 7). Comparative Study of CNN and RNN for Natural Language Processing. arXiv.org. https://arxiv.org/abs/1702.01923.

11) Zeyer, A., Irie, K., Schlüter, R., & Ney, H. (2018, May 8). Improved Training of End-to-end Attention Models for Speech Recognition. arXiv.org. https://arxiv.org/abs/1805.03294.