e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
Volume:03/Issue:06/June-2021 Impact Factor- 5.354 www.irjmets.com

CRAFTING MELODY USING BI-DIRECTIONAL LSTM MODEL

Mustafa*1, Nimisha Tiwari*2, Rohit Gupta*3

*1,2,3Student, Department Of Computer Science And Engineering, Acropolis Institute Of Technology And Research, Indore, MP, India.

ABSTARCT

"Music makes the world go round, rocking and rolling".

Our desire is to fabricate a recurrent network architecture which tends to craft music which is rich in mellowness and harmonized and is permissible as composer generated music. Peculiarly, there exists a lot of work based on recurrent net amalgamated with Boltzmann machine (RNN-RBM) and substantiated with other energy based models. In the state of art, the subsisting autonomous melody generation approach is categorized as: symbolic models and raw audio but individually they both serve incapable. They train and create at the level of notes and can capture very long term ranging dependencies of the musical basal structure but they generally lags in catching the nuances and richness of raw audio generations and albeit unstructured music. The keen analysis of the past work firstly shows the absence of a one stop solution that is single model for all genre music generation. Secondly the absence of genre mixture also exists. In this paper we mix these approaches in a recurrent net architecture Long Short-Term memory network-LSTM with an attention layer to cover the past shortages and create a structured & realistic-sounding composition. Crafting a soothing tune requires rhythm ,basses, harmony ,chords, synchronization, variations and melody creations, out of all these melody generation serves as a base for the crafted music work. LSTM network is utilized to learn and scrutinize melodic structure, notes formalization, rest & duration between chords and notes and chords synchronization of different styles of music and then using this symbolic generation. The result obtained in this study depicts the improvement in music generated after every epoch and the variation of model predictions on different musical genre.

Keywords: Recurrent Network, Music Generation, LSTM- Long Short-Term Memory, Attention Layer.

I. INTRODUCTION

A couple of decades ago, technology named machine learning came into picture. Later the Darwinism of deep learning from machine learning made it significantly more powerful as it showed the power of data and the mimicking of human brains in machines and unfolded many electrifying windows of opportunity to design advanced artificial intelligent machines which can learn and enlighten themselves with simultaneous improvement without being programmed explicitly. The annals of music generation in machines is as ancient as the invention of computers. As foreseen by starry-eyed visionary Ada Lovelace, Systems can create and generate "elaborate and scientific pieces of music of any degree of complexity and ex- tent" (Lovelace 1843) [1]. The work of Hiller and Isaacson, on the composition of a musical piece using a computer program, was published as early as shortly after the introduction of the very first computer [4]. Erstwhile the advancement in deep net has made it possible to rigorously attain automatic music composition although it is a difficult task. The current techniques of music creation comprises scrupulously fabricated musical and melodic features with simple generative techniques like generative adversarial networks and Markov chain, graph based minimization technique etc., but they work on a relatively huge noise free dataset and libraries with make it highly expensive in terms of computation. Although these traditional techniques can many-a-times produce commendable compositions beauteous to listen but those musical sequences consisted of repetition of notes and absence of musical thematic structure. A teeming number of methods from these are focused on generating note level music in which the output is nothing more than the symbolic presentation i.e., a note number sequence or else a stream of events or chords like MIDI file.

In the state of art, computer based composition systems are a go-go. Effectuating a music which is pragmatic and lilting via a self-winding machine is an appealing and breathtaking piece of work because of the advancement of artificial intelligence algorithms capable of remembering things like RNN-LSTM, RNN-GRU etc. In this day and age and also antecedently the rigorous formalization and catalyzation of notes, tempo, rhythm, melody, tune and most importantly the euphonicity of the melody has made music generation gain significant enthrallment in regards to other art forms. Music is an ultimate art form surpassing other art.

The nascency of composition is well known till now due to its mellowness and fluency. In the course of the entire history of music generation, a countless number of astonishing composers and musicians have composed ravishing, mellifluous and sensible music pieces with a high prominence on the underlying basal musical structure. Even with this level- headed technology and metamorphosis of the automation industry a big question floating underneath the generation is "the possibility and practicability for a machine to learn the formalization of music structure and catalyzation of a soothing melody. Computer-based music generation (CMG) has to deal with the task of tuning rhythm and lyrics and composing melody and chords. Co-composition and notes harmonization in a fully automated background can be achieved via a plethora of methods from multifarious viewpoints. This computational effectuation can be synchronized with human creativity and can take musical works to new heights without resulting in manual composers or lyricist block. Deep neural networks is a tremendously powerful tool to enhance the accessible methods and can lead to concoction of new algorithms from its amalgamation with others.

Hitherto and pronto, the growing computational power and the advancing development in the field of recurrent neural network architecture has made it possible for music generation to be on large scale corpuses. The lion's share of recurrent network for modelling long term dependencies i.e., a long rum memorizing model is the Long Short-Term Memory (LSTM) network. The technique of Gated Recurrent Units as stated by cho et al.[8] is acquiring high utilization in efficaciously modelling long term dependencies in generic sequence modelling tasks of various variety. In accordance to the moment of fact, the recurrent architecture of LSTM network can help us in improving the formalization and musical thematic structure of these melodic pieces and can enhance the generated composition by producing it as a brain-child and making it unique and soothing with musical coherency. They can productively model sequence of notes but hitherto proposed architectures focus on the generation of sequences per se, without considering constraints [4].

In this work , we scrutinized various aspects of music and tried to endeavor a model using Bi-directional LSTM with incorporation of an attention layer in it. We tried to fabricate our work using the past models but addressing the absence of Rest and Duration that is present between notes and chords and catalyzation of notes formalization through the incorporation of an attention layer to the LSTM network and prevention of overfitting in the model. We desire to fabricate a architecture that can use a set of params and is cheaper in computation but is completely realistic in audibility. The architectural variations in the network are capable of generating ballads, pop music, orchestral music with high melliflicity and harmonization.

One of the challenges in music generation is deciding the right representation of the data. We chose to use the midi files representation. Our data consists of midi files from different genres of music that are classical, hip-hop and jazz from various renowned artists. The utilized corpus consists of 54 classical midi files, 25 hip-hop and 98 jazz midi files. All the data was collected manually from the web in midi format.

In order for us to train our model, the midi files needed to be converted into a structure that can be coded into numerical values so we could easily feed them to our model. The first step was to examine the data we were working with. The data splits into 2 object types: Notes and Chords. Notes objects having details regarding the pitch, octave and offset of the Note and Chord objects are essentially a container for a set of notes that are played at once. The concept of Rest and Duration (Rhythm) was also incorporated between each Notes and Chords.

The main idea is to take our midi files and convert them into a list of Stream objects containing notes, chords and rest objects, with an associated duration using the Music21 library of python. Then we read each midi file and append each note, chord and rest-duration combo into an array. We append the pitch of nodes and rest duration using their string notations. Another advantage of using string notation of the pitch is that the most significant parts of the notes can be recreated using it. And every chord is appended by assigning an id to each note in the chord together into a single dot separated string.

This array is then splitted into a 100-note/chord sample each, and a mapping function is defined to map the note, chords and rest-duration string based data to integer based numerical valued data since training LSTM on numeric data is easier.

We create an input sequence and the respective output for our network. The output of each input sequence is defined as the first note or chord that comes after the sequence of notes in the input sequence as described in the list of notes. The network is trained by feeding the 100 note samples and predicting the next node and comparing that with the actual next node in the sequence.

Finally, before feeding the prepared data into our network the input is normalized and the output is encoded using One-hot encoding.

II. TECHNICAL APPROACH AND MODEL ARCHITECTURE

The proposed model consists of a single module for predicting the next sequence. After a lot of research, we finalized that the best methodology to follow would be to use a LSTM model and incorporate the use of Attention. Also, we could have used a Generative Adversarial Network (GAN) but GANs are generally more difficult to successfully train. To further break down the model, we define RNN, LSTM and Attention in terms of their success in generating contextual text given a prior prompt, and the same concept would help us to be able to generate good music with repeated melodic structures.

Recurrent Neural Networks (RNN) are essential models as they possess the capability to remember their most recent calculations (outputs) and use them along with the new input to generate the desired results. They accomplish this by using feedback loops that provides them the capability to implement a small memory structure. The problem with RNN is that the gradients of the loss function decays exponentially over time. When the gradients are taken over and over again, the model tends to forget more and more information faster.

To overcome the concept of LSTM is used, that brings the concept of long term memory which allows our input to propagate further into the network. This is accomplished by the use of a special internal mechanism called "gates" which regulates the flow of information. Also, it incorporates a "memory cell" , which has gates that can learn which data in a sequence is important to keep or throw away. By doing that, it can pass relevant information down the long chain of sequences to make predictions. But we face two main issues with this, one is the fixed length that limits the depth of understanding and the point that the vector is a final calculation, meaning we can't understand a portion from any other point of view other than the end of the input.

Attention is a more contemporary development that indeed helped us to solve our core problem. Attention mechanisms are basically a technique to non-uniformly weigh the influence of input feature vectors so as to optimize the process of learning some target. The term attention suggests a selection process in which certain inputs are particularly more important than others. Using Attention allows us to attend to a certain part of input at any instant and use those parts to generate the final results rather than using only the final calculations. In a task like music generation incorporation of Attention can have a lot of success.

At last, using the information the best model we could build a network with following layers in it;

Bidirectional LSTM layer → Attention Layer → LSTM layer.

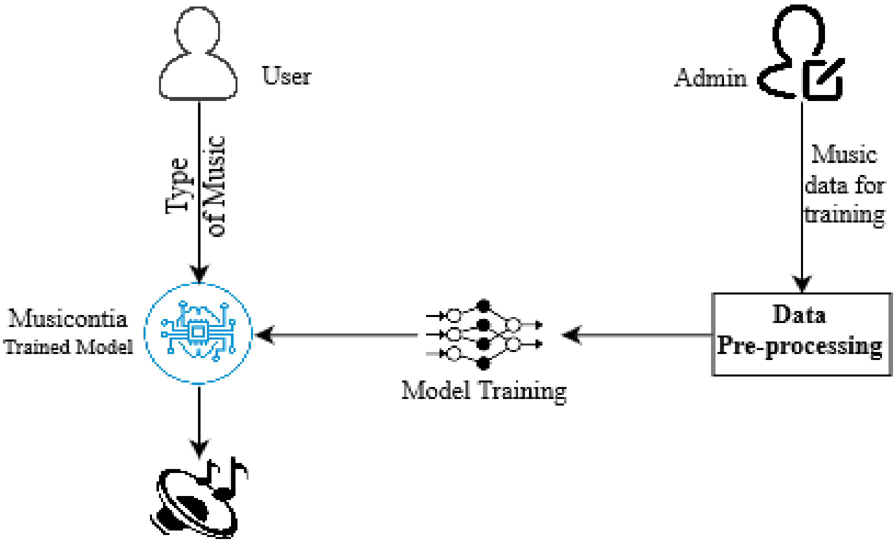

Figure. 1 Architectural Overview

The last layer is an intentional LSTM Layer. We came up to choose these layers because we thought that after the LSTM and Attention has at first made relationship with understanding the data provided, it needed another layer just to further develop the inferences it recognized before going to the Dense fully connected layers. The input sequence is fed to the 512 nodes Bi-Directional layer, then into a Attention layer, then another LSTM layer with 512 nodes and finally into a 256 node Dense layer which further propagates the output an approx. 3000 node Dense layer with SoftMax prediction. The 3000 denotes the many unique possible note, chord, rest-duration combinations the input sequence had. Further some Drop-out functions were added to each layer as a regularization technique to prevent the model from overfitting.

To add non-linearity at the final layer for prediction we use a SoftMax activation function to provide us with a final probability for a specific combination that should be used to play the next node sequence and the one with the highest probability is used.

Since, the problem is of classification or finding the best note sequence from the available ones. Hence, during the training of the model we make use of categorical cross-entropy function as a loss function in order to accurately predict the next combination as the training of our model proceeds and RMSprop as the optimizer just because of the mere fact that most of the RNN model uses it for generating better results.

The model was trained on three separate data files, data being midi files of three different genres of music. For each type, a separate model with the same architecture was trained. Due to lack of computational resources, the model training took place in a non-GPU environment. Each model was trained for a total of 20 epochs with a batch-size of 64. The model weights were stored after every 10 epochs, which were then used in future to generate results from unseen data.

III. EXPERIMENTAL RESULTS

In this architecture for music generation as the time passes by, the model accuracy tends to improve and the gradual decrease in loss can be visualized in the loss vs epoch plot. With every epoch, the loss falls off for the classical genre as seen in Figure.2(a).

Although a slight fluctuation can be visualized in the later training of hip-hop genre. Which can be due to lack of data set and use of multifarious instruments.

Also, the model fails to capture the patterns in the jazz genre.

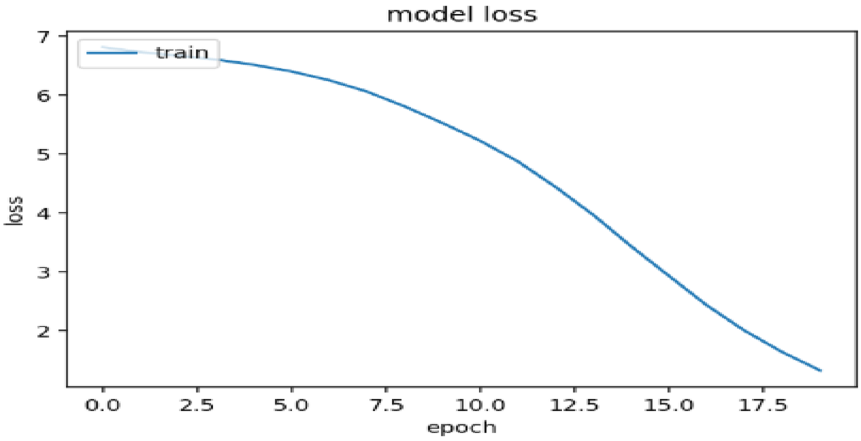

Figure. 2 (a) Loss v/s epoch plot for classical data

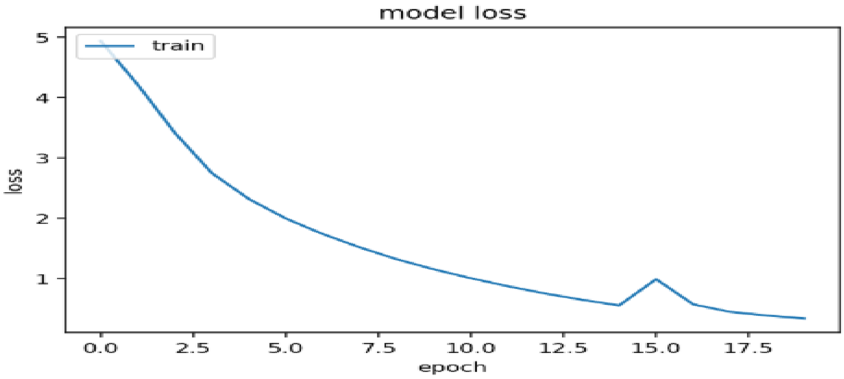

Figure. 2 (b) Loss v/s epoch plot for hip hop data

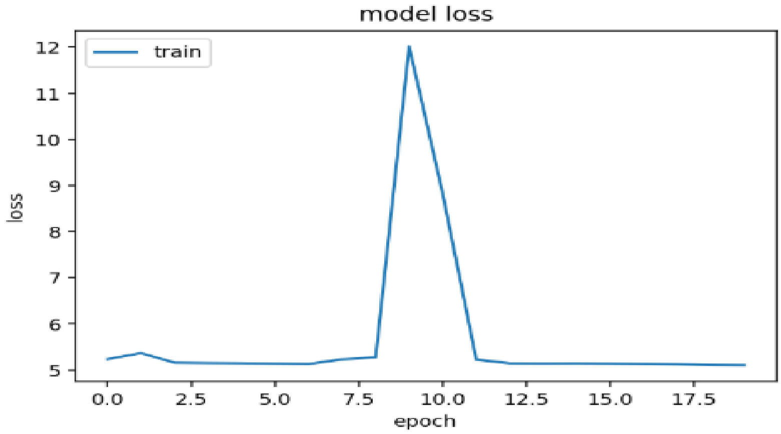

Figure. 2(c) Loss v/s epoch plot for jazz data

The output generated after the training by the model for the three different input genres is depicted below in their sheet music depiction. Although the final output is midi audio file.

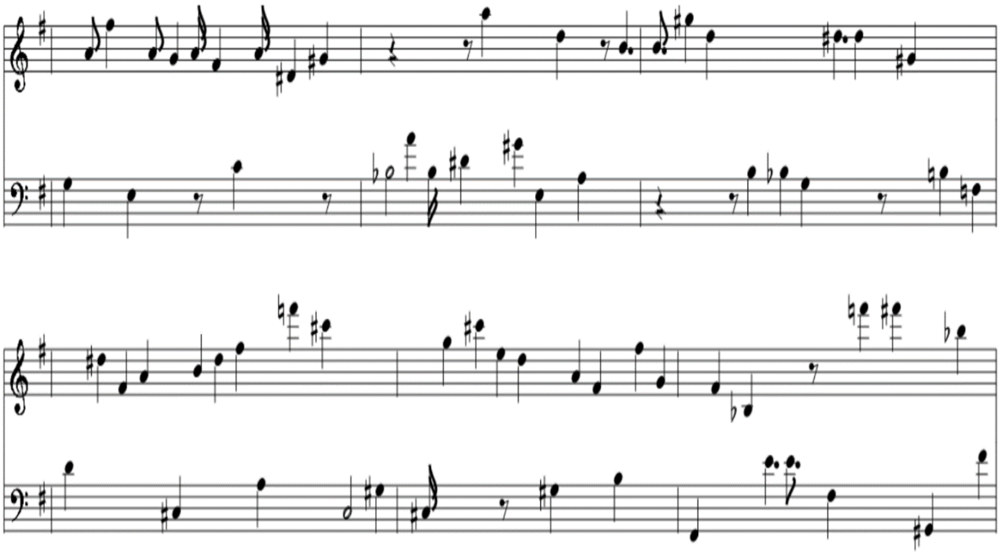

Figure. 3 (a) Sheet music depiction for classical genre

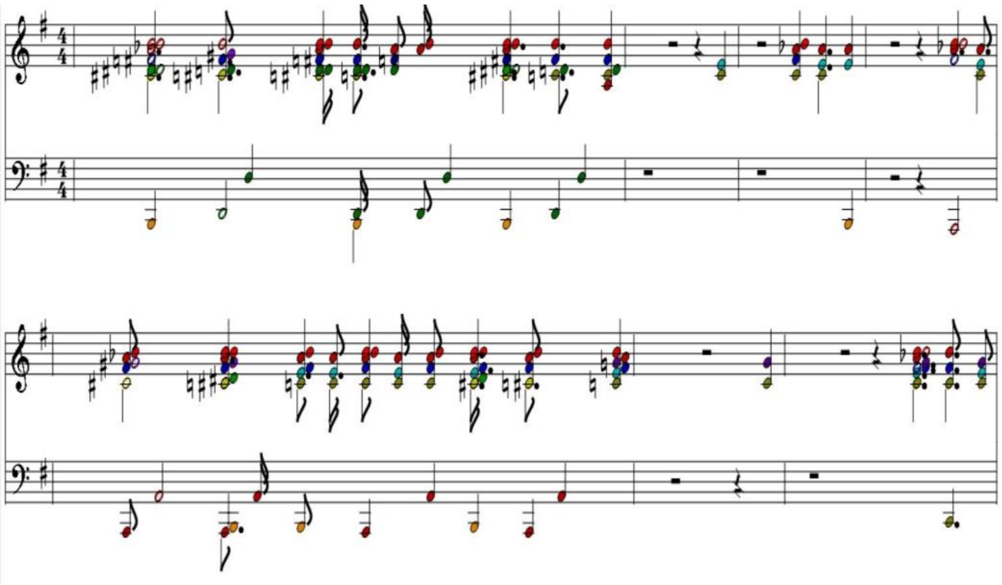

Figure. 3 (b) Sheet music depiction for hip hop genre

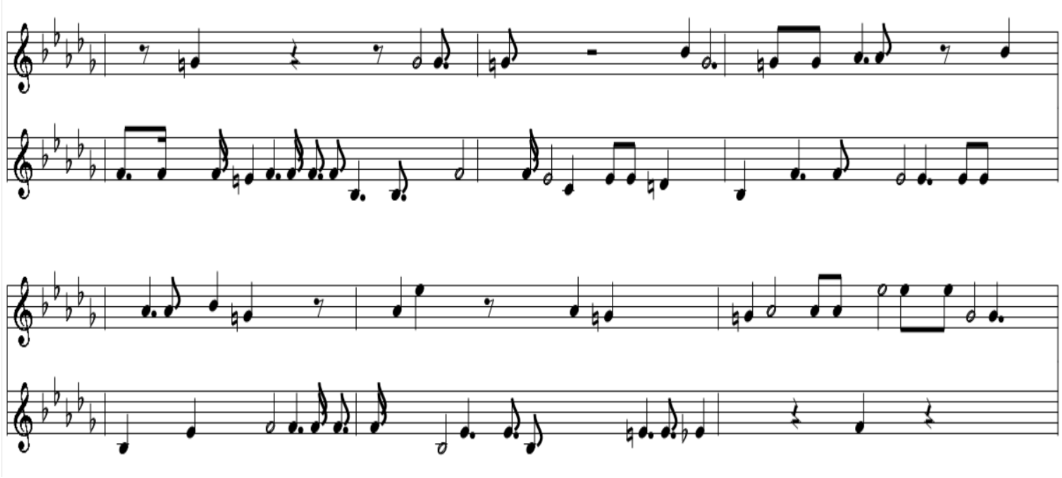

Figure. 3 (c) Sheet music depiction for jazz genre

Results from the classical genre of the training data were found to be the highest in the mellowness, and the results from the hip hop and jazz genre were quite fluctuating.

The relationship between training and amount of data depicts that on increasing the computational capacity, data and data refinement, number of epochs and computational power will ultimately result in the improvement of the melody generated.

IV. CONCLUSION

The fervor in deep learning inspired advancement and hence we can introduce highly complex data models and can capture the rest and duration between notes and chords. The system can learn the underlying musical basal structure and melodic formalization. The model doesn't exactly mimic the input sequence provided to it, rather a unique music is brainchild by it. This model creates a soothing melody using LSTM and attention layer. Although the output is not that effective due to lack of computational power on our end, it can be smoothed a lot using rigorous training and noise reduction. For further improvement in the model, we are planning first to thoroughly analyze and evaluate our contemporary model through ratings and reviews about music from composers, musicians and ardent music lovers.

FUTURE WORK

In future we can expand our model to craft multi-track and multi-phonic music via a richer and larger dataset which is highly refined. We are also analyzing the usage of reinforcement learning amalgamated with the principles of music theory and substantiated with genre recognition, theme recognition, emotion recognition and even input from other musical information retrieval models. Even we can implement our model in combination and amalgamation with multitudinous domains of music. The integration of speech and music can bring new paths to produce lyrical tunes. The different instances of our model can be trained on various musical genres with tighter interaction and integration.

LIMITATIONS

We are prompted and inspired to use deep net architecture because of its ability and training techniques to quickly learn musical styles automatically from the music corpus provided to it and then it can craft musical pieces from the distribution learned. But in spite of all these advantages certain limitations hold our leg in this field of music as deep net architectures are autistic automata without human interference and interaction and generate music completely autonomously and hence is far away from composers, it lag the objective of interaction assisting and adding to composers to refine and compose music. The major issues arising are control, structure, creativity and interactivity.

V. REFERENCES

[1] Herremans D, Chuan CH, Chew E. A functional taxonomy of music generation systems. ACM Computing Surveys (CSUR). 2017 Sep 26;50(5):1-30.

[2] Huang A, Wu R. Deep learning for music. arXiv preprint arXiv:1606.04930. 2016 Jun 15.

[3] Nayebi A, Vitelli M. Gruv: Algorithmic music generation using recurrent neural networks. Course CS224D: Deep Learning for Natural Language Processing (Stanford). 2015.

[4] Makris D, Kaliakatsos-Papakostas M, Karydis I, Kermanidis KL. Combining LSTM and feed forward neural networks for conditional rhythm composition. In International Conference on Engineering Applications of Neural Networks 2017 Aug 25 (pp. 570-582). Springer, Cham.

[5] Roads C. Research in music and artificial intelligence. ACM Computing Surveys (CSUR). 1985 Jun 1;17(2):163-90.

[6] Briot JP, Pachet F. Music generation by deep learning-challenges and directions. arXiv preprint arXiv:1712.04371. 2017 Dec 9.

[7] towardsdatascience.com/how-to-generate-music-using-a-lstm-neural-network-in-keras-68786834d4c5

[8] Cho K, Van Merriënboer B, Bahdanau D, Bengio Y. On the properties of neural machine translation: Encoder-decoder approaches. arXiv preprint arXiv:1409.1259. 2014 Sep 3.

[9] I. Goodfellow, Y. Bengio, and A. Courville. Deep Learning. MIT Press, 2016.