

## Generate Poems and Letters Using an Iterative Neural Network (Poem Generating using LSTM)

**Ashwaq Katham Mtasher<sup>1\*</sup>, Wasnaa Kadhim Jawad<sup>2</sup>, Noor Abdul Khaleq Zghair<sup>3</sup>**

Submitted: 05/09/2022      Accepted: 23/12/2022

### Abstract

Creating a linguistic text using the famous type of recursive neural network: From the well-known section as a section of the main branches of artificial intelligence, we can see that one of the main tasks that fall under this section is to generate a real chat using deep intelligence models and because of its distinctive ability, a text will be generated very close to the level of texts written by real humans and almost The text generated using the model cannot be distinguished as an industrial text because it is very similar to the text written by humans and we see the great future benefit that this new knowledge will achieve in solving many of the various tasks facing the world today for example, translating from language to language, generating novels and poems on the level of art, writing a summary of a long text using intelligence models, enabling the robot to write and generating sounds later. All these tasks will be solved with the help of models of deep intelligence. A modern method will be presented in generating poetry writing using deep model and identification, an iterative neural network that learns through the sequence and remembers it for a short period of time and then generates a similar sequence of letters that will form sentences bearing a poetic meaning.

**Keywords:** RNN, poem generating, fake text, LSTM.

### INTRODUCTION

A simple brief definition of Recurrent Neural Networks (RNN)

The type of networks that we will present in this paper is the type we see in most projects related to “Natural Language Processing” and it is visible in most of the uses that we see for intelligence projects in daily life. Recursive networks are a solid type and able to solve many problems in the field of natural languages and their tasks Like many networks and other machine learning algorithms, this type of neural network is somewhat old and was first introduced in the eighties of the last century, and its ability to solve difficult problems was not shown until recently. The increase in the size of data in the world added additional problems that must be dealt with for that The old school methods were resorted to because they always dealt with the foundations that are the roots of solving modern problems. This is what makes this type of network shine again.

What distinguishes this type of network is its ability to deal with data that contains a pattern or sequence in it

and its ability to remember from the last several sequences processed by the network

Which allows it to predict the next sequence in the data according to the last several sequences present in memory. This distinction made it able to deal with several very difficult tasks on other type of deep networks. Some of the concepts that help in solving this algorithm are the tasks that deal with Data that contains a pattern or is defined by time, sound, text, or stock exchange data

In the field of deep neural networks, we can say that this type of network is responsible for modeling the sequence in one of the forms of data that we have previously mentioned. This mechanism in action is derived from the way the human brain works.

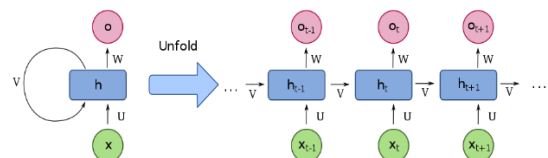


Figure 1: the mechanism of RNN (source: wiki)

### THE PROBLEM STATEMENT

We are in the process of presenting a model that helps in generating novels and poems in the Arabic language, as there is similar work in several other languages. Therefore, we will present a recursive neural network in

<sup>1</sup>Department of Community Health, Health and Medical Techniques College of Kufa, Al\_Furat Al\_Awsat Technical University (ATU), 31003, Iraq.

<sup>2</sup>University of Information Technology and Communications, Baghdad, Iraq

<sup>3</sup>Computer Engineering Department, University of Technology- Iraq

a project to create and generate narratives that are almost unable to distinguish them from those written by a person or even look better than what he can provide. A person who does not have specialization in the field of writing poetry and prose and writing short novels

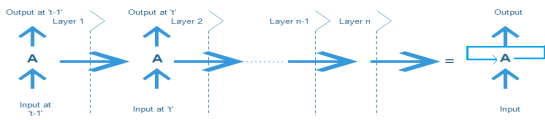


Figure 2: the way of create output using NN (source: net)

### SOFTWARE\_USED

These are some of the necessary libraries used in the field of text generation and novel writing:

- NLTK.
- Gensim
- sklearn
- pandas
- PDF
- TEXT
- KERAS
- TENSORFLOW

Most of them were used, and the Python language was dealt with to achieve the project, in addition to an HP laptop with a decent processing power with NVIDIA processors.

### RESEARCH\_IMPORTANCE

Learning sequence and dealing with sequential data is a great challenge for neural networks of all kinds without the recurrent neural network, which was able to outperform other types of neural networks by dealing with sequential data. It tries to predict a studied text at the character level through training the recurrent neural network

With the occurrence of many developments in the field of artificial intelligence, the tasks have become easier on several levels, and a very good percentage of quality accuracy has been obtained for several tasks such as the task of eliminating long text content in addition to artificially generating texts for characters in a movie.

This is something that will be the talk of the future in the development of applications that depend on sequence prediction and detection of important sequences in the text.

The importance of this paper is to provide a deep model that provides ease and comfort to the writer of novels or the poet who writes several characters together and saving him time and effort together. Also with intense training, we will maintain the writer's style and method of writing, that is, our model will learn the style in which the writer deals and as a final benefit of the proposed

model We note that it is difficult for people to write with the emotion of a particular writer or the feelings of another poet. This model will try to present a beautiful poem in the style of the writer whose data has been worked and trained.

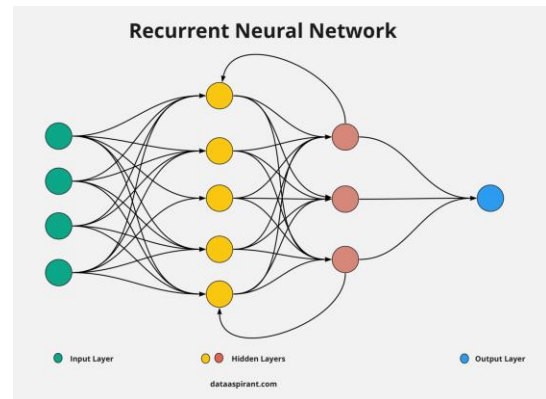


Figure 3: The presentation of RNN (source: wiki)

### Power of RNN NETWORK

RNN is a group of deep neural networks. The strength of this network lies in the fact that it does a dynamic job During the process of learning, and training on data, this can

The type of network that handles a file full of sequences. As income for her in the first layer, the main difference in this. The type of network that uses a forward feed or feedback. In that this type is used in addition to the foreground feedback. This additional feed results in an additional output whose idea is to modify the shape of the current output. With another effect that came out of the last step, and here lies the idea of memory in this type of network. An additional type of recursive network can be used

In addition to RNN, RNN can be used with LSTM. The aim of it is to treat the data in a more parallel way, which is more effective the reason is that it contains two oblivion gates and one of the type (in and out). Moreover, the regression problem in this type of recurrent neural network has been solved.

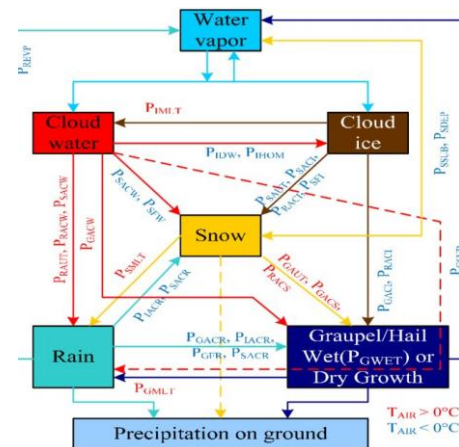


Figure 4: Example of recursive using RNN (source: wiki)

## RELATED WORK

There are many problems and tasks that recurrent neural networks solve in the field of natural language processing and, especially in tasks that need to deal with sequences, it is famous for its short and long-term memory that it has of its two types

Some of the tasks solved by recurrent neural networks that fall under the work related to our research are generating different texts in German, generating texts in Chinese, generating different texts in Persian, generating different texts in English and generating different texts in French. Summarizing long texts in English, which includes many research papers in this field, and here we provide the generation of poems and novels in Arabic and English.

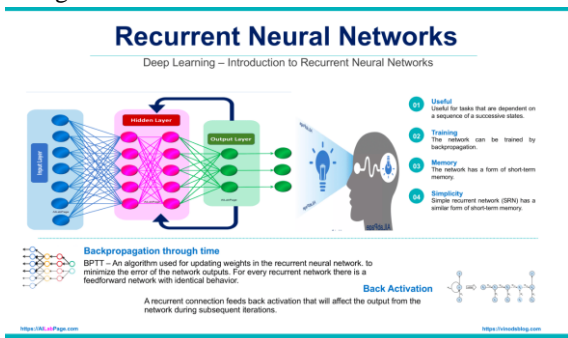


Figure 5: The deep RNN figure (source: wiki)

## MODEL ARCHITECTURE

This is the structure of the model that we will use to achieve the required task, and we can note the simplicity of the basic structure that can achieve the required, and this is due to the strength of the iterative young network that we have already talked about

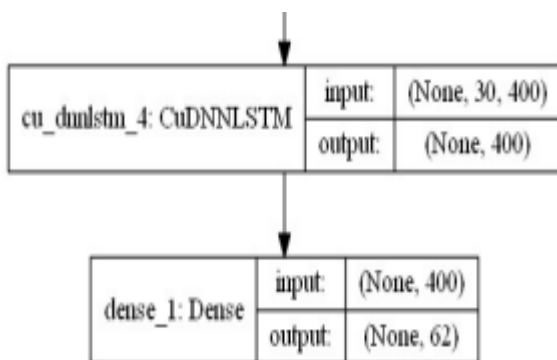


Figure 6: summary present layers of the model (source: wiki)

Below is another architecture that has not been used, but will be compared with it, which is the architecture used by a team that authored the following research paper "Text Generation Using Recurrent Neural Networks"

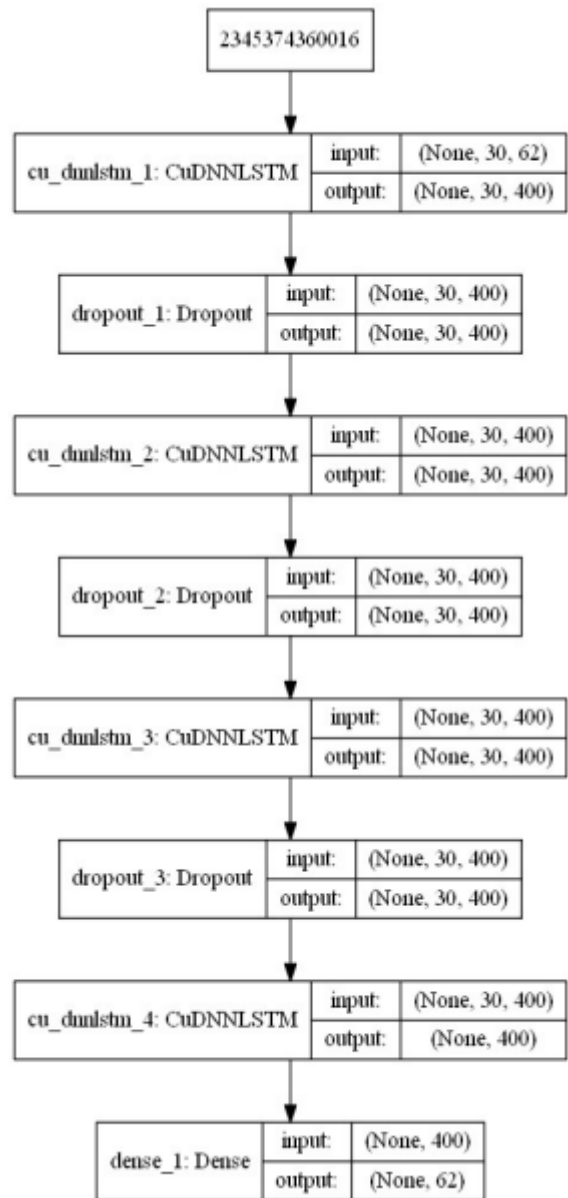


Figure 7: another proposed summary present the layers of a deep model RNN (source: wiki)

## DATA SET

After looking at several types of data that can be used in order to achieve this task, we decided to choose several types of text files that contain a group of spoken speech or can contain text sentences of the required language (French, Persian, German, Chinese, Hindi and Arabic). A special function has been created to read and process each data separately. The processing methods used in processing the Arabic language differ from the processing methods used in processing the Chinese language and the Indian language ... etc.

## WHAT IS "LSTM" AND HOW IT WORKS

The regression problem of RNNs is a very well-known problem

To solve it, a very common type of this network was developed called LSTM, and this type is much better than the first type, as it overcame the recursive network

in points that need a long range. This structure was designed in a mathematical way that seems somewhat strange in calculating the operations in the hidden layers. We note that its basic design contains an input and an exit and a layer that acts as a gate to forget and a gate that acts as a masculine for the last output, and this is somewhat unconventional.

The figure below shows the structure of the LSTM:

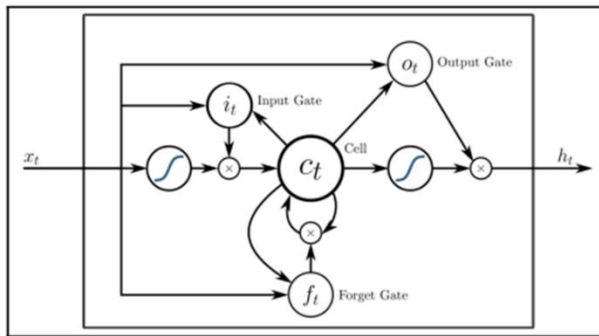


Figure 8: the connection between gates in RNN (source: wiki)

How LSTMs work:

As for the LSTM mechanism, its distinctive design contains several gates of different types “forgetting, learning, excreting, remembering the last output.” What distinguishes its design is the type of long-term memory. In addition to a short-term memory. If we want to divide the work mechanism into steps, it will be as follows:

Step 1: When the income reaches the LSTM, it chooses one of the portals that were presented above. The long-term data is then transmitted to a forgotten gateway, and information that is not relevant to the basic information is discarded

As for short-term data, it enters a learning-type portal, and the information that is extracted is extracted It will be saved until the next step.

Step 2: The information from the first step is sent to the implementation gate, which performs the process of redefining the types of memories, and the result of the network is displayed from this gate, i.e. we see the design output from this gate.

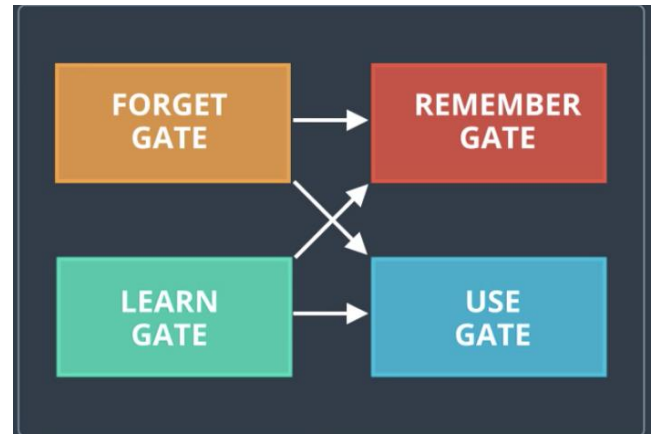


Figure 9: the different types of gates in RNN (source: wiki)

### THE IDEA OF THE WORK

Achieving the task of generating poems or novels in a way that is very similar to the way a writer or poet writes. By utilizing recursive neural networks. Where it is known about these networks that they have the ability to predict and retain the last content that was trained and used in the process of influencing the current output. For example, in the figure below, it can be seen the formation of the sequence word after the word, keeping the sequence, and finally displaying it in the form of a narrative text or a poem resembling the style of one poets.

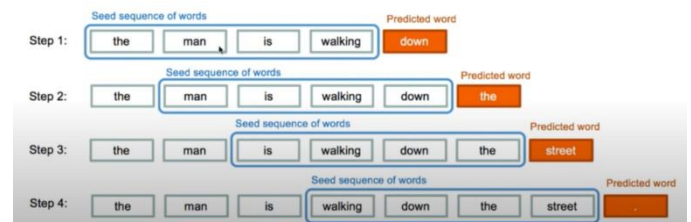


Figure 10: the predictions of RNN (source: paper)

### IMPLEMENTATION

For the application process:

We will divide the implementation process ... after the basic process in the implementation process, which is to download the necessary libraries, of course.. We have used the functions that we have defined that are specialized in the cleaning process for each type of data. In a way that fits the network. And then we will build the previously explained structure that consists of a LSTM layer and a dense layer from which we will receive the final output.

Therefore, we will convert the income into a form that the network will receive through the famous techniques used, such as one hot encoding or word empaded.

After this observation and the process of preparing the data into the required form, we have everything ready to receive the income in the required standard form and we compile the model to start the fit process

The model was implemented using Keras with TensorFlow and we will train to a sufficient number of time epochs with leaks that will improve the training and divide the data into blocks in order to develop the training process in each block and we will use the famous adam optimizer.

Below is the proposed method that we implement the model on it :

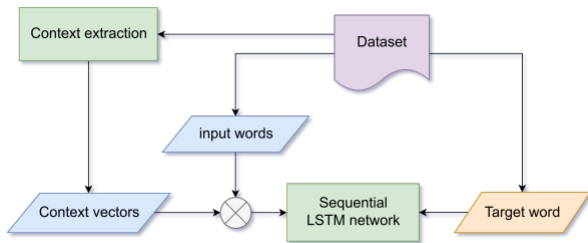


Figure 11: the methodology from A to target (source: wiki)

This is the model in detail how the processing process is carried out and pass to the next step:

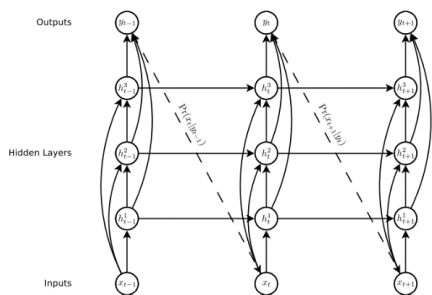


Figure 12: return sequences using LSTM (source: wiki)

### RESULT

In order to monitor the results generated or generated by the model, several tests are conducted to detect the similarity of the generated synthetic text, one of which is (POS) and the other is the cosine, which is the best measure for detecting similarity and evaluating the output of the model since we are dealing with Natural Language Processing

First this is the model accuracy on " tensorboard ":

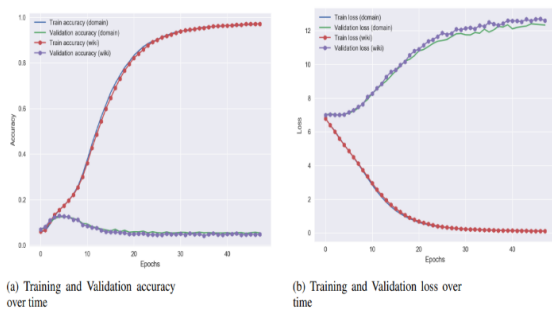
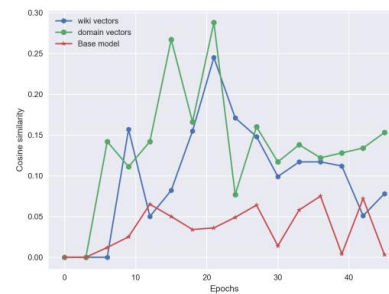


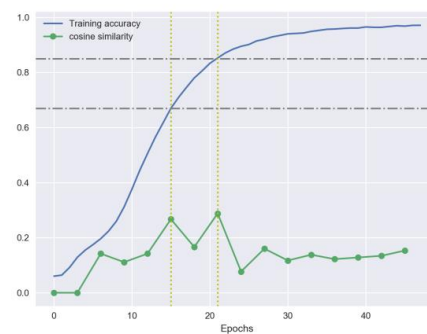
Figure 13: result from tensorboard

And the output of the model can be seen below and how it was measured on the cosine scale:



Cosine similarity of the generated text with context over training time

After looking at the previous chart, you can notice some negative values in the chart that was created And of course, we will neglect these values by calculating the arithmetic mean of the entire values, and we will see the following chart, whose results are clearly very good. After the previous process:



A portion of the generated text can be seen in the image below:

Generated text:

of something akin to fear had begun to be a sound of his door and a man in the man of the compants and the comins of the compants of the street. i could he could he married him to be a man which i had a sound of the compant and a street in the compants of the companion, and the country of the little to come and the companion and looked at the street. i have a man which i shall be a man of the cominstance to a some of the man which i could he said to the house of the comins and the man of street in the country and a sound and the c

Here is a glimpse of how the proposed structure works for character generation character by character across an iterative neural network:

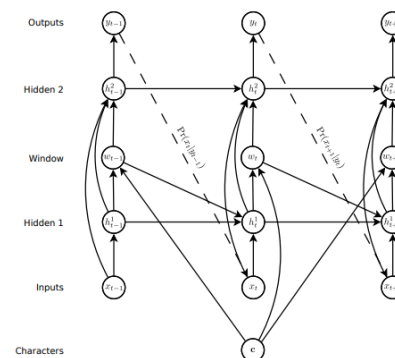


Figure 14: The methodology for predict characters

## COMPARISON TABLE

Below is a comparison table between our model and model

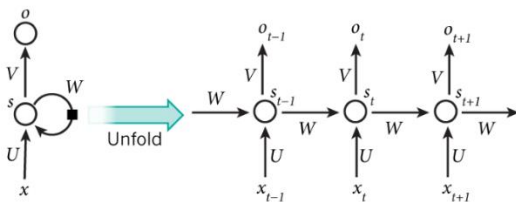
That had solved this task and generate text with 81% accuracy

Model	Model1(the standard model)	Our model
Accuracy	81%	80%
Text similarity	94%	89%

## RECOMMENDATIONS

The structure of the basic model can be changed, adding some extra layers ("more LSTM-, extra coarse layers") and using leakage to solve the training problem without result. The data size can be increased or processed in different ways in order to increase the model speed or increase the training period

Change the batch\_size and monitor the training if there is an improvement in the model's performance



A recurrent neural network and the unfolding in time of the computation involved in its forward computation. Source: Nature

## CONCLUSION

In the content of this paper, we presented a new task that shows the power of recursive neural networks in doing the process of generating typefaces and texts, and what distinguishes our work is to add the Arabic language to see poems and novels generated in an industrial way, which is very similar to the way the poet thinks and the way the writer writes in it, and this is a very powerful thing As it will expand the horizons for applications that are in the Arabic language and open the way for them to use applications and algorithms of intelligence in them. The model has been tested and trained in several languages, including Arabic, Hindi, German, Chinese .... etc.

## REFERENCES

[1] Jia Wei; Quiang Zhou; YiciCai; Poet-based Poetry Generation: Controlling Personal Style with Recurrent Neural Networks; International Conference on Computing, Networking and Communications (ICNC); 2018.

[2] Zejian Shi; Minyong Shi; Chunfang Li; The prediction of character based on recurrent neural network language model; IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS); 2017

[3] Kyuyeon Hwang; Wonyong Sung; Character-level language modeling with hierarchical recurrent neural networks; IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP); 2017

[4] Tzu-Hsuan Tseng; Tzu-Hsuan Yang; Chia-Ping Chen; Verifying the long-range dependency of RNN language models; International Conference on Asian Language Processing (IALP); 2016

[5] CONTEXT BASED TEXT-GENERATION USING LSTM NETWORKS 2020

[6] omas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhersch, and Armand Joulin. Advances in pretraining distributed word representations. In Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018), 2018.

[7] Generating Steganographic Text with LSTMs 2017

[8] Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. 2016. Exploring the limits of language modeling. arXiv preprint arXiv:1602.02410 .

[9] Denis Volkhonskiy, Ivan Nazarov, Boris Borisenko, and Evgeny Burnaev. 2017. Steganographic generative adversarial networks. arXiv preprint arXiv:1703.05502 .

[10] Alex Wilson and Andrew D Ker. 2016. Avoiding detection on twitter: embedding strategies for linguistic steganography. Electronic Imaging 2016(8):1-9.

[11] Park, Y.H., Jeong, H.J., Kang, I.M., Park, C.Y., Choi, Y.S., Lee, K.J.: Automatic generation of korean poetry using sequence generative adversarial networks. In: Proceedings of the 2018 Annual Conference on Human and Language Technology, Human and Language Technology. pp. 580-583 (Oct 2018)

[12] Wang, K., Tian, J., Gao, R., Yao, C.: The machine poetry generator imitating du fu's styles. In: Proceedings of the 2018 International Conference on Artificial Intelligence and Big Data (ICAIBD 2018). pp. 261-265. IEEE, Chengdu, China (May 2018). <https://doi.org/10.1109/icaibd.2018.8396206>

[13] Zugarini, A., Melacci, S., Maggini, M.: Neural poetry: Learning to generate poems using syllables. In: Tetko, I.V., Kurková, V., Karpov, P., Theis, F.J. (eds.) Artificial Neural Networks and Machine Learning – ICANN 2019: Text and Time Series – Proceedings of the 28th International Conference on Artificial Neural Networks. Lecture Notes in Computer Science, vol. 11730, pp. 313-325. Springer International Publishing, Munich, Germany (Sep 2019). [https://doi.org/10.1007/978-3-030-30490-4\\_2](https://doi.org/10.1007/978-3-030-30490-4_2)

[14] R. M. P. J. Sanidhya Mangal, "gru\_lstm\_tv\_script," GitHub, [Online]. Available: [https://github.com/sanidhyamangal/gru\\_lstm\\_tv\\_script](https://github.com/sanidhyamangal/gru_lstm_tv_script). [Accessed June 2019].

[15] R. M. P. J. Sanidhya Mangal, "generated\_text," GitHub, [Online]. Available: [https://github.com/sanidhyamangal/gru\\_lstm\\_tv\\_script/tree/master/generated\\_text](https://github.com/sanidhyamangal/gru_lstm_tv_script/tree/master/generated_text). [Accessed July 2019].

[16] S. Koirala, "Game of Thrones Data," GitHub, [Online]. Available:

- [https://github.com/shekharkoirala/Game\\_of\\_Thrones/tree/master/Data](https://github.com/shekharkoirala/Game_of_Thrones/tree/master/Data) . [Accessed June 2019].
- [17] S. Mangal, "Got.pkl," GitHub, June 2019. [Online]. Available:  
[https://github.com/sanidhyamangal/gru\\_lstm\\_tv\\_script/blob/master/got.pkl](https://github.com/sanidhyamangal/gru_lstm_tv_script/blob/master/got.pkl). [Accessed June 2019].
- [18] pickle — Python object serialization," Python Software Foundation, [Online]. Available:  
<https://docs.python.org/3/library/pickle.html>. [Accessed June 2019]
- [19] "Magenta," Google, [Online]. Available:  
<https://magenta.tensorflow.org/>. [Accessed 29 October 2018]
- [20] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4700–4708, 2017
- [21] A. Daud, W. Khan, and D. Che. Urdu language processing: a survey. *Artificial Intelligence Review*, 47(3):279–311, 2017
- [22] Briot, Jean-Pierre, Gaëtan Hadjeres, and François Pachet. "Deep Learning Techniques for Music Generation-A Survey." arXiv preprint arXiv:1709.01620 (2017)
- [23] Johnson, Daniel D. "Generating Polyphonic Music Using Tied Parallel Networks." In *International Conference on Evolutionary and Biologically Inspired Music and Art*, pp. 128-143. Springer, Cham, 2017.
- [24] Mariya I Vasileva, Bryan A Plummer, Krishna Dusad, Shreya Rajpal, Ranjitha Kumar, and David Forsyth. 2018. Learning Type-Aware Embeddings for Fashion Compatibility. arXiv preprint arXiv:1803.09196 (2018).
- [25] Lvona Tautkute, Tomasz Trzcinski, Aleksander Skorupa, Lukasz Brocki, and Krzysztof Marasek. 2018. DeepStyle: Multimodal Search Engine for Fashion and Interior Design. arXiv preprint arXiv:1801.03002 (2018).