International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2022): 7.942

A Deep-Learning based Approach for Automatic Lyric Generation

Tanmoy Debnath[1], Suvvari Sai Dileep[2]

[1, 2]Department of Computer Science and Engineering, B.M.S. College of Engineering, Basavangudi, Bangalore, India - 560019

[1]*tanmoy.cs17[at]bmsce.ac.in*
[2]*saidileep.cs17[at]bmsce.ac.in*

Abstract: ***Writing as a task is characteristic of humans, given its inherent requirement for creativity and grammatical abilities. However, with rapid advancements in artificial intelligence, there has been tremendous progress in automating writing tasks. In this paper, we study the effectiveness of a Long Short-Term Memory (LSTM) model, utilizing a Markov model, in automatically generating lyrics, a form of writing. We further analyze a pre-trained GPT-2 model in its performance of the same task and evaluate its results against those of the Markov-LSTM. For the evaluation, we leverage BLEU scores and assessments by humans. The results of both evaluations show that the Markov-LSTM model delivered better results than the pre-trained GPT-2.***

Keywords: deep learning, long short term memory, lyric generation, GPT-2

1. Introduction

The task of writing is one that involves creativity and imagination and, thus, is intrinsically human. Over the years, however, with the rise and improvement in technology, there have been several attempts to automate this process that would otherwise have irrefutably been characteristic of humans alone. The developments in artificial intelligence, catapulted by rapid advancements in neural networks in particular and the field of natural language generation in general, have taken this endeavor a step further. Machines, in the form of computer systems, suddenly find themselves capable of automating to the extent that their performances can be comparable to humans. Moreover, this advancement in automation has been extended to various forms of text generation, which include, but are not limited to, prose, poetry, and lyrics for songs.

The newly created advanced text generation models have shown promise in producing original long-form prose content from minimal initial text. Natural Language Generation applied to generate song lyrics, while an exciting task, has yet to obtain much concentration within the research community. The conventional tasks in the field of NLG, such as prose generation, obey specific rules which define a strict structure and semantics that are relatively superficial. In comparison to writing regular prose, the process of automating lyric writing involves many difficulties. These include structural differences for a given genre of music, which can be further broken down into differences like the length or flow of any given line, coherence maintained across the different lines, their inherent meaning, and differences in rhyming patterns, among others. Thus, the task of generating lyrics is not simply about automating the process of writing but doing so while being conscious of various musical and artistic features.

Our work seeks to tackle this problem of automating the process of writing lyrics for songs by developing and leveraging the abilities of an LSTM model, utilizing a Markov model, and a pre-trained GPT-2 model, that has been fine-tuned as per our dataset, to generating meaningful lyrics, with an emphasis on rhyming and coherence. They were chosen due to their encouraging results in other forms of text generation. The rest of the paper is structured as follows: Section II sheds some light on related work. Section III provides a detailed description of our approach. Section IV discusses the dataset, and the results derived from both models, followed by an evaluation of the generated lyrics of both models. Finally, section V ends with a conclusion.

2. Related Work

Historically, attempts at generating lyrics have followed a particular set of rules and have primarily been about maintaining a rhyming pattern. Watanabe et al. (2014) [1] proposed a probabilistic model that realizes topic transitions from one paragraph to another in order to generate lyrics. There was, however, a shortcoming in terms of consistency of meaning.

In recent years, however, with tremendous advancements still being made in the field of Deep Learning, text generation has found a number of fruitful approaches. For instance, (Sutskever, 2011) [2], which has shown the effectiveness of Recurrent Neural Networks (RNNs) in this context.

Diving further into RNNs, Long Short-Term Memory (LSTM) networks are a type of RNN capable of learning order dependence in sequence prediction problems and have proven incredibly successful. For example, Fan et al. (2019) [3] made use of LSTM models to tackle the reliance on sequences of long words. Wu et al. [4] utilized an LSTM in conjunction with a hierarchical attention model that grasps the context of generated text at both a sentence level and a document level to generate lyrics for Chinese songs. It takes a line of lyrics as input, based on which it generates the following line. Potash et al. (2015), in Ghostwriter: Using an LSTM for automatic rap lyric generation [5], successfully generated lyrics that incorporated a rhyming pattern.

However, given their model was trained on a specific artist, their results faced limitations in its extent. We extend the research done with LSTMs and, having prepared a sizable dataset that accounts for diversity in genres and artists, directly tackling one consideration which was seen lacking in some notable prior works, we explore text generation in the context of generating lyrics. We also make it a point to account for rhythmic patterns when generating lyrics.

Additionally, GPT-2 has been found applicable in a wide variety of forms. There have been attempts to use it for generating text in the form of poetry and prose both. Liao et al. [6] used a basic GPT model with no human-crafted rules or characteristics to generate forms of Chinese poetry, and they were able to produce poems that were significantly well-formed and meaningful than those produced by previous RNN-based approaches. Coupled with the brilliance of GPT-2 discussed in the context of Natural Language Processing tasks by Radford et al. (2019) [7], and the success it has found in terms of rendering poetry, we believe leveraging GPT-2 will turn out to be fruitful for generating lyrics of songs.

3. Methodology

To elucidate the effectiveness of LSTMs, we must first understand that RNNs take in the output from a prior step and use this as input for the following step of the network. This works in a continuous loop. Thus, owing to their capability to, at any given step, hold the results of all the previous inputs, RNNs are successful at remembering information and use it for modifications on the output. When the sequences are too long, the gradients either explode or vanish, which makes it difficult for RNNs to learn. The appeal of LSTMs is found in the gaps of information for context. LSTMs tackle the issues with the gradient by expanding the single tanh unit. Thus, where the capabilities of RNNs to learn long sequences of data is hampered, LSTMs are far more successful.

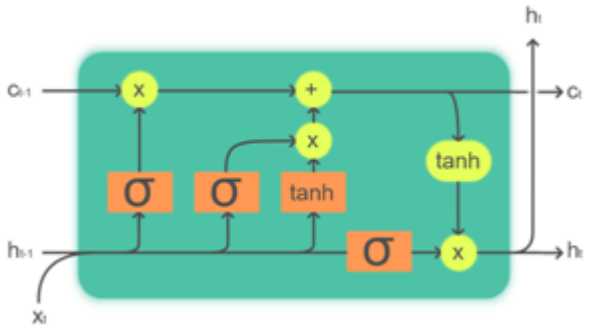

Figure 1: The LSTM cell [8]

The primary model that we are using for lyric generation is the Markov-LSTM model. Markov models have been used in multiple areas. One of them is text generation, wherein it has shown encouraging outcomes in short text generation. The Markov-LSTM considers any given word and, based on this, decides the next word. This model generates lyrical verses based on each word's likelihood, using Markovify functions. As a result, if the current word is "perfect" and the likelihood that the word "love" should appear after it is 20%, and the probability that the word "crime" should come after it is 25%, the word "crime" will be chosen, and so on. Now, the opening lyrics to any given verse were chosen at random. The output verses were additionally encoded and fed into the LSTM model as input. The LSTM memory cell, which is specific to the LSTM model, defines the architecture for the hidden transformation. The presence of an input gate, output gate, forget gate, and cell/cell memory, which appear in the model as activation vectors, is the crucial component of the LSTM memory cell. The hidden layer at each time-step is now a complex nonlinear composition of the gate, cell, and hidden vectors, with each of these gates/cells having its own bias vector.

The LSTM model we build consists of two layers and 256 units to generate the lyrics. It further takes an embedding of size 60. We add a Dense layer of 128 neurons and a Relu activation layer on top of the two layers. The model has an output layer that is a Softmax layer, which has a size equivalent to the number of words in our dataset. The training of the model was done using stochastic gradient descent with a momentum of 0.9, categorical cross-entropy, and a learning rate of 0.002. Owing to their long-term memory, LSTMs can better anticipate the characteristics of the following verse, such as its rhyming pattern. Therefore, the LSTM approach will be able to choose a suitable new lyric verse from among all the previous verses rendered by Markovify.

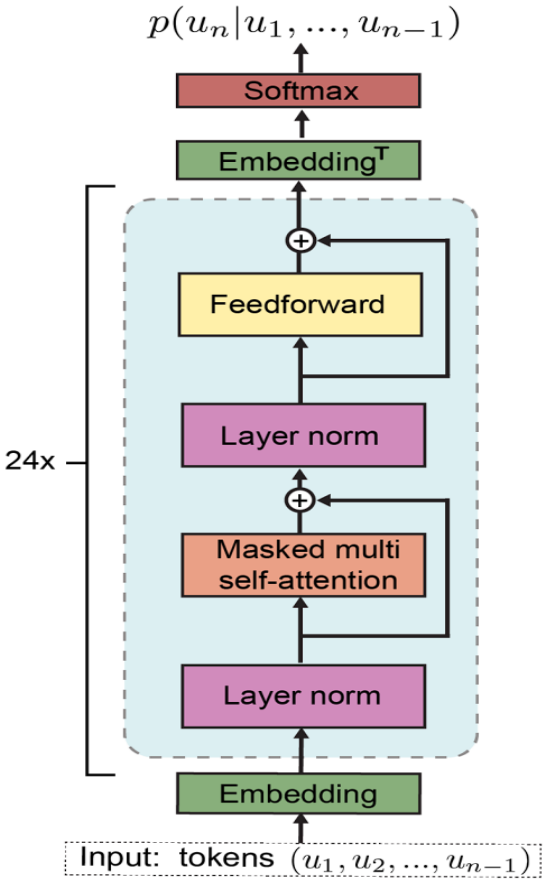

Figure 2: GPT-2 architecture[9]

GPT-2 (Generative Pre-trained Transformer 2), developed by OpenAI, is a large-scale unsupervised transformer-based generative language model. It is a machine learning model that predicts the following word at any given point of a sentence by using probability distributions. In order to enable it to make this level of prediction, GPT-2 was trained

on a large corpus (WebText) containing 40 GB of text [8]. GPT-2 uses BPE (byte-pair encoding) (Sennrich et al., 2015) in order to perform encoding. Now, the GPT-2 has four versions: an extra-large version (1.5 Billion Parameters), a large version (774 Million Parameters), a medium version (355 Million Parameters), and a small version (117 Million Parameters).

We have chosen the small version (117 Million Parameters), due to our limited computing capacity and lack of comprehensiveness in our dataset. The model is fine-tuned to fit our purposes. The data provided as input to the model is a single text file, which is what the model requires. We utilize the Finetune function provided in order to fine-tune the pre-trained GPT-2 model on our relevant dataset. The parameters for the function are set as discussed. The "steps" parameter was set at 2000. The "restore from" parameter is set to "fresh", in order to start training from the base GPT-2. For training, the learning rate parameter was set at 5e-4. After fine-tuning the model on our dataset, we use the generate function to generate the song lyrics. A crucial parameter of this function is "temperature". This parameter determines how greedy the model will be. The higher the temperature, the more unique, but likely syntactically incorrect, the output will be. We set it to 0.8. It is recommended to keep the temperature value between 0.7 and 0.9 [10].

4. Experiment

A. Dataset

We have prepared a custom dataset of English song lyrics, primarily consisting of data derived from the Genius API [11]. Our dataset ended up containing the lyrics to 10-15 songs each of 1750 artists of varying genres. Duplicates, most easily identified by a given song and its remix version having the same name, were removed. Some of the artists we included are Dua Lipa, Taylor Swift, Ed Sheeran, BTS, Justin Bieber, among others. Some information about the dataset is displayed in Table 1. The Training to Validation split was set to a 80:20 ratio.

Table 1: Information regarding custom dataset

Number of Songs	23,200
Number of Lines	956, 304
Number of Characters	38,280,000
Mean number of lines per song	41.22

B. Results

A sample of the lyrics generated by the LSTM model:

> Can't have a better honey than sweet love
> Pre-perfection, anything could go wrong
> I could fall and you could fall too
> We'd be all sleeping on the floor
> Maybe nothing would go wrong
> but I don't know what's worth saving
> I never thought I'd love somebody so perfect for me

Another sample of the lyrics generated by the LSTM model-

> Go on and say what you want
> But you can never break me down
> I'm always gonna be good as new
> No one's ever gonna love you like me
> No one else could make it better than me
> Nothing could ever bring you down like me
> And I know what nobody else would see in us

On observing the resultant lyrics, the Markov-LSTM was found to generate meaningful lyrics with a song-like flow. While frequent, as is typical in lyrics, word repetitions were not as marked. As the forget gate determines how much of the prior states are to be preserved in the present state, LSTMs find it difficult to remember information that existed even a bit more than a few sentences ago. Thus, long-term repetitions are less frequent, but nevertheless, the coherence in the generated lyrics is notable.

A sample of the lyrics generated by the pre-trained GPT-2 model-

> Been gone now for nine days and nine years
> And I have not been well through any of it
> I caught a glimpse of the album cover once again
> For the 100th time as I keep telling myself,
> Thinking, 'did I do this to you?'
> On my bed staring at the ceiling again
> I know that she'll understand, she knows better than me

Another sample of the lyrics generated by the pre-trained GPT-2 model-

> Always been so dependable and sweet,
> I swear it's almost like you were mine
> But now I'm missing her soul and heart
> I'm not just missing her, I'm missing her touch
> The places my shoulders found comfort in her so much
> And that makes me sad but it makes me glad too

As the generated lyrics themselves might suggest, lyric generation with GPT-2 is less impressive than expectations might have demanded. The primary reasoning behind this is that GPT-2 was trained on a dataset consisting mainly of prose-like text. And this kind of text differs from lyrics and how they're structured. Moreover, the inherent rules dictating prose formation differ immensely from lyrics. For instance, rhyming does not occur in prose as much as it does in lyrics.

Furthermore, while frequent in lyrics at times, word repetitions are not encouraged in prose. Thus, to make the GPT-2 more impressive in terms of lyric writing, one would have to train it such that it leaned less towards prose and more towards lyrics. But as prose is entrenched in the very foundation of the GPT-2, it would take an incredible amount of computational resources to make it more accustomed to lyric writing. We were, however, lacking such resources.

C. Evaluation

The first evaluation we subject the generated lyrics to is through an assessment of Bilingual Evaluation Understudy

(BLEU) scores [12]. BLEU is a metric often used to automatically evaluate the results of text generation tasks. The BLEU score can be a value between 0 and 1, which measures the similarity between, in this scenario, generated lyrics and high-quality lyrics used as reference.

To calculate BLEU scores, we need to specify the number of grams - which can be uni-gram, bi-gram, 3-gram, or 4-gram. For the purpose of this evaluation, we derive uni-gram, bi-gram and 3-gram BLEU scores. A score towards the lower end of the scale denotes that the generated lyrics are of low quality. In contrast, a score nearing 1 signifies that the generated lyrics are of higher quality.

The score is calculated for every line of the generated lyrics, using lyrics from the validation set as a reference. Table II shows the results of BLEU evaluation of the generated lyrics.

Table II: BLEU Scores

GRAMS	MARKOV-LSTM	PRE-TRAINED GPT-2
1	0.8924	0.8488
2	0.7851	0.7439
3	0.6852	0.5917

As we had stated in the introduction of this paper, writing is inherently a task characteristic of humans. So there will be discrepancies between how a machine evaluates generated lyrics versus how a human considers the same. To alleviate this inconsistency, we also conducted human evaluations on the same generated lyrics we had obtained BLEU scores of previously. For this evaluation, the chosen representatives were given three criteria to work with: meaning, rhyme, rhythm. They were asked to assign a score between 1 and 3 to each, where 1 denotes lyrics of lower quality and 3 signifies high-quality lyrics. Table III shows the results of human evaluation of the generated lyrics.

Table III: Human Evaluation

	MARKOV-LSTM	PRE-TRAINED GPT-2
Listenability	2.5	2
Rhyming	2.4	1.9
Meaning	2	1.6

As we can observe, the Markov-LSTM model scored better for both sets of evaluations than the pre-trained GPT-2 model.

5. Conclusion and Future Work

In this paper, we demonstrated the usefulness of an LSTM model in creating unique lyrics that are meaningful. This study presents two approaches for generating lyrics. A Markov-LSTM model was developed and a pre-trained GPT-2 model was considered after fine-tuning as per the data. We used the BLEU score to evaluate the performance of lyric generation and Human evaluation of three criteria: Listenability, Rhyming and Meaning. According to the scores, the Markov LSTM model outperformed the pre-trained GPT-2 model. There are still some Grammatical rules, logical word sequences, and their interconnectivity that were absent from the created lyrics. We were mostly constrained by computational power, computer memory, and time. Our long-term aim is to lower execution time and enhance the grammatical rules of the models' produced lyrics. This study also provides the groundwork for future work where we will be able to succeed with less data and compose better lyrics that are indistinguishable from those performed by human artists.

References

[1] Kento Watanabe, Yuichiroh Matsubayashi, Kentaro Inui, and Masataka Goto. Modeling structural topic transitions for automatic lyrics generation. In Proceedings of the 28th Pacific Asia Conference on Language, Information and Computing, pages 422-431, Phuket,Thailand, December 2014. Department of Linguistics, Chulalongkorn University

[2] Ilya Sutskever, James Martens, and Geoffrey E Hinton. 2011. Generating text with recurrent neural networks. In Proceedings of the 28th International Conference on Machine Learning (ICML-11), pages 1017–1024.

[3] Zhifang Fan, Zhen Wu, Xin-Yu Dai, Shujian Huang, and Jiajun Chen. 2019. Target-oriented Opinion Words Extraction with Target-fused Neural Sequence Labeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2509–2518, Minneapolis, Minnesota. Association for Computational Linguistics.

[4] Xing Wu, Zhikang Du, Y. Guo, and H. Fujita. Hierarchical attention based long short-term memory for Chinese lyric generation. Applied Intelligence, 49:44—52, 2018.

[5] Peter Potash, Alexey Romanov, and Anna Rumshisky. 2015. Ghostwriter: Using an lstm for automatic rap lyric generation. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 1919–1924.

[6] YiLiao, Yasheng Wang, Qun Liu, and Xin Jiang. Gpt-based generation for classical chinese poetry. CoRR, abs/1907.00151, 2019.

[7] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019). Language models are unsupervised multitask learners. OpenAI blog, 1(8):9.

[8] Heilbron et al., 2019 available at arXiv:1909.04400

[9] This visualization is freely available and is licensed under the CC-BY License, by Guillaume Chevalier. For more information, visit https://github.com/guillaume-chevalier/ Linear-Attention-Recurrent-Neural-Network/tree/master/inkscape_drawings.

[10] Utane, N. (2020, April 17). Complete guide to build and deploy a tweet generator app into production. Retrieved December 22, 2020, from https://towardsdatascience.com/complete-guide-tobuild-and-deploy-a-tweet-generator-app-into-production-5006729e583c

[11] Genius Lyrics API (https://docs.genius.com/).

[12] Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting of the Association for Computational Linguistics, pages 311–318