# Rap and Text generation
# How a rap text can be generated considering the rhyming, metrical, and lexical questions

E. Partouche H. Hamila N. Perdriau

Octobre 2019

## Contents

## 1 Abstract

In this paper, we will aim to show how we can use a Long Short-Term Memory Language (LSTM) to generate a rap text or to create a “rap machine”, including the generation of diverse recurrent requirements for this kind of texts such as rhymes, rhythm and a various vocabulary.

A large part of the previous studies on the generation of text [1] have as basis a Neural network, especially Recurrent Neural Networks (RNN): when a Neural Network is a circuit of artificial neurons made to solve artificial intelligence (AI) problems, a RNN, still composed of Neural Networks, can remember them because they are recurrent but encounters some limits.

This is why the LSTM model appears to be more effective whitin the scope of text generation : it has the possibility to correct the vanishing gradient problem of the RNN and thus learning what to remember and what to forget. An interesting work was made on the use of ghostwriting through LSTM [2] : the goal is to give the impression that a rapper has produced a new song, by reproducing his style of writing.

It is necessary to base our generation on a pre-existing set of lyrics from one or several artists on which we train our model. We also considered the differents ways of predicting the next character, the next word or the next line. As they are several ways of doing so, we have shown some interest in the way Ruslan Nikolaev [3] showed the effectiveness of the character-level model in 2018. Also, Hirjee and Brown have created a rhyme detection tool based on probabilistic model.[4]

Still, methods to estimate the quality of generated lyrics are noted by Potash, Romanov and Rumshisky in 2015 and Malmi and his colleagues in 2016 [5], considering the style of the rapper and the rhyme density. Still, the text has to answer to some considerations such as the occurrence of linguistic phenomena as lexis and phonetics, even if most of the time, the program elaborated appears to give a “word salad” [6].

This is why using the CMU Pronouncing Dictionary appears to be essential [4], as it make it possible for us to capture the global meaning of a group of words. One of the goal of this research is to have a nicely generated pronunciation at the same time. Finally, Malmi in 2016 also considers the length of verses in a rap text generated and all the kind of rhymes that we can find in it.

This work, then, will be a mixture of all of these previous works, trying to produce a more precise model for rap lyrics generation, including metric and length, phonemes, question of predictions and judgment of the uniqueness of the production.

Thne, we intend to present a study of a generated goswritten text based on some Tupac's lyrics, considering the accentuation on syllables of this text, the metric and the rhymes having yet partially studied, as the lexical aspect, another of our goal being a coherent new rap text, being as coherent as the Tupac's ones.

## 2 Data and Methods

One of the main exercices we had to perfom was writing a model in Python, using the TensorFlow library and a RNN, that would take into account our needs. This program is available for free, under the GPLv3 licence, and is available with this article. In it, we have added the CMU Pronouncing Dictionnary, allowing us to read the generated text produced with the good accents, considering the word's syllable(s) and the metric of the setence, the lexical stress playing also a role in the accentuation of words. Moreover, a code allowing us to recognize the rhyming syllables at the and of lines was introduced. The intial set of lyrics came from Rap Genius, where we downloaded the lyrics thanks to a scraper [7], hoping to produce ghostwritten texts, as it Tupac had produced a new song while he's dead. In this way, we calculated the similarity fo the verses with the verses generated and the verses of Tupac, in order to see if our generated lyrics were the most novelest plus a calculation of rhyme density enabling us to produce rhyme types and rhyme frequency resembling to the artist ones, or even better to the artist's obtained average. Then, thanks to the help of a teacher from Paris-Diderot, we trained the model on a cluster of computers for about XXXX TODO XXX iterations. You can see a example of the evolution through the iterations in the poster accompagning our research, as well as an example of generated lyrics. [We don't have yet the results and then, can't make a critic]

## References

[1] I. Sutskever, J. Martens, and G. Hinton, "Generating Text with Recurrent Neural Networks," p. 8.

[2] P. Potash, A. Romanov, and A. Rumshisky, "GhostWriter: Using an LSTM for Automatic Rap Lyric Generation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, (Lisbon, Portugal), pp. 1919–1924, Association for Computational Linguistics, 2015.

[3] R. Nikolaev, "Generating Drake Rap Lyrics using Language Models and LSTMs," Apr. 2018.

[4] H. Hirjee and D. Brown, "Using Automated Rhyme Detection to Characterize Rhyming Style in Rap Music," *Empirical Musicology Review*, vol. 5, no. 4, pp. 121–145, 2010.

[5] E. Malmi, P. Takala, H. Toivonen, T. Raiko, and A. Gionis, "DopeLearning: A Computational Approach to Rap Lyrics Generation," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*, pp. 195–204, 2016. arXiv: 1505.04771.

[6] H. G. Oliveira, "Automatic generation of poetry: an overview," p. 7.

[7] F. Paupier, "RapLyrics-Scraper," Sept. 2018.