Automatic Generation of Lyrics in Bob Dylan’s Style
Dongzhuo Li (lidongzh@stanford.edu), Chao Liang (chao2@stanford.edu), Tianze Liu (tianze@stanford.edu)

1. Introduction
Bob Dylan was recently awarded the Nobel Prize "for having created new poetic expressions within the great American song tradition". It is interesting to see if machine could learn his poetic style by looking at his lyrics. In this project, we use N-grams and Recurrent Neural Network (RNN) with Long Short Term Memory (LSTM) to model Dylan’s lyrics, and eventually use the algorithms to generate samples of lyrics in Bob Dylan’s style.

2. Data
The data we use includes lyrics of 465 songs downloaded from Bob Dylan’s official website (728KB in total). In order to make it easier for machines to process the data, we preprocessed the data by lowering case, removing punctuations from words.

3. N-grams

4. RNN with LSTM
LSTM: Memory that allows RNN to learn long-term dependencies.

5. Word vs. Character level RNN
- In practice, RNNs work at either word level or character level.
- To make word level RNN more efficient, words are mapped into high dimensional vector in which semantically similar words are close to each (word embedding), before fed to RNN.
- In our experiments, character level RNNs performs significantly better than word level RNNs, likely due to the small number of words our data contains.
- In the following sections all results are generated by character level RNNs.

6. Metrics for RNN Performance
We use perplexity as the metrics for performance of RNNs, which is defined as:

$$\text{perp} = \exp\left(\frac{1}{N} \sum_{i=1}^{N} \log P_{\text{target,}}\right)$$

$P_{\text{target,}}$ is the probability of the $i$th word output by softmax regression following RNN.

7. Training RNN

8. Effects of Batch Size and Hidden Layer Size

9. Regularization by Drop-out
- Drop-out means randomly disable a certain fraction of neurons in each iteration, in order to avoid over-fitting.
- In our experiments, a non-zero drop-out rate is very effective in preventing over-fitting.

10. Sample lyrics

11. Discussion

12. References

---

**References**
Dongzhuo Li (lidongzh@stanford.edu), Chao Liang (chao2@stanford.edu), Tianze Liu (tianze@stanford.edu)
Michael A. Nielsen, Neural Networks and Deep Learning, Determination Press, 2015.

---

"..."