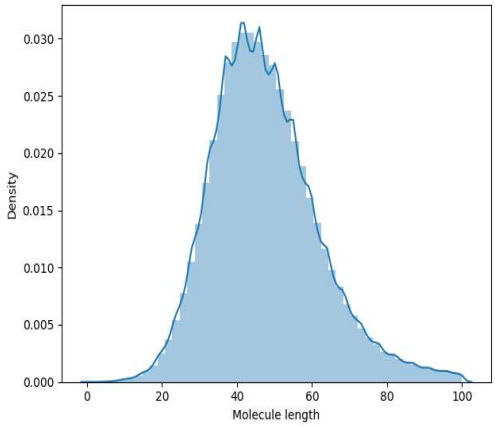
Urchade Zaratiana, k11932165

Molecule Generation Challenge

This challenge aims at developing a machine learning algorithm that can generate realistic molecules given a large set of human-designed molecules as training data set. The main task is to achieve a low Frechet ChemNet Distance (FCD) while auxiliary metrics "novelty", "validity" and "uniqueness" should be kept high (more than 90% each).

1. The dataset

The training data consists of 1272851 SMILES-coded molecules and our goal is to generate 10000 new molecules by using a neural network model.



2. Data preprocessing

We first load the data and then combine all the molecules by adding adding a special token <sos> (start of sequences) at their starts and <eos> (end of sequences) at their ends. Moreover, we create two dictionnaries, one maps characters to ids (encoder) and the other one maps the ids to characters (decoder).

3. The model

Our model is a 2-layers LSTM (Hochreiter and Schmidhuber, 1997) combined with a Softmax layer. We feed the model with a sequence of characters and for each character of the sequence, we predict its next character as illustrated in the figure 1.

Our LSTM layer contains 128 hidden size and we add a dropout (Srivastava et al., 2014) of rate 0.1 to prevent overfitting. The hyperparameter search were done manually by trying different configurations.

Our model is similar model to (Gupta et al., 2017).

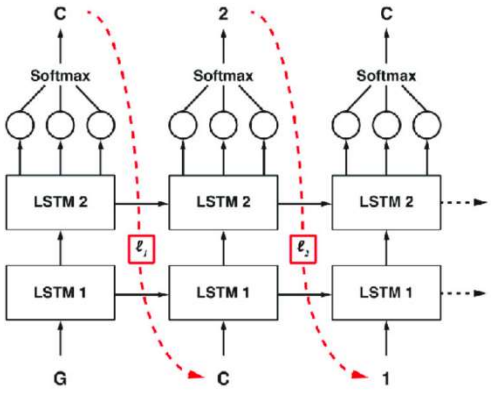


Figure 1: LSTM model. Image taken from (Gupta et al., 2017)

4. Training

During training, we feed the model with a sequence of length 100 and batch size of 128. We use a cross entropy loss and optimize it with an Adam optimizer (Kingma and Ba, 2017) with a learning rate of 0.01 for 50 epochs. Moreover, we halved the learning rate at epochs 20, 30 and 40. During training, we saved the model that got the best performance on the validation set. Our best model got a perplexity of 2.24.

5. Generating new molecules

To generate new molecules, we input the model with the special token <sos> and generate character by character until we reach the special token <eos>. The generated molecule is accepted only if it is not present in the training set and if it is a valid molecule. We test the validity of a molecule by using Rdkit library. To addition, for the molecule generation, we use a top k sampling instead of greedy search to get more diverse results. In fact, we notice that using greedy decoding, we always get C (carbon) as first sequence when we generate molecules since it is the most common start in the training set whereas with top k sampling, we sometime get O or N for example as first sequence.

6. Results and conclusion

Finally, we got more than 99% of novelty, uniqueness and validity. We believe that we could have got more than 100% score on all of these metrics by filtering out molecules that are present in the sample submission file. Concerning the Frechet ChemNet Distance, we got 4.469 which is a good score but could have been improved by doing more hyperparameter search.

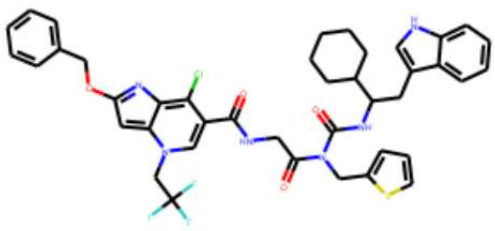


Figure 2: Example of generated molecule with our machine learning algorithm

7. Bibliography

Gupta, A., Müller, A., Huisman, B., Fuchs, J., Schneider, P., Schneider, G., 2017. Generative Recurrent Networks for De Novo Drug Design. Molecular Informatics 37. https://doi.org/10.1002/minf.201700111

Hochreiter, S., Schmidhuber, J., 1997. Long Short-Term Memory. Neural Computation 9, 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Kingma, D.P., Ba, J., 2017. Adam: A Method for Stochastic Optimization. arXiv:1412.6980 [cs].

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research 15, 1929–1958.