# Data preprocessing

To make it practical we actually could try to get the email or chat history as the training set. But here to make it simpler I just use Kafka’s metamorphosis. It has about 100 thousand words and has two thousand and five hundred unique words.

The first step is to clean the text, like delete the line break and other stuff.

And the next step is tokenization. because LSTM can only understand real numbers, so we need to assign a unique integer to each unique word.

# Build the model

And then we could build our model. Here I use 2 lstm layers with 1000 units, and 2 dense layers. There are also callbacks to save the best parameters and for tensorboard visualization.

# Evaluate the model

And here’s the result of the model. We could see that although the accuracy of the training set is constantly increasing, the accuracy of the validation set remains rather poor. And the same is for the loss. I think the reason might be the input. Since I use a novel as the training data, there is some evolution of the plots, so there’s actually not too much similarity between the training and validation set. So it is only good to predict certain language structures in metamorphosis.

# Predict with the model

and then to use the model to predict, we just need to use the default function and transform the integers to words. As I said before, clearly this model is only useful when we try some phrases likely in the novel. For example, what a strenuous, career.

# Interesting takeaways: data size and overfitting

I just rushed through our model because I want to use more time to share some of the interesting takeaways I got from the troubleshooting process. I started with following some instructions online, but many of them are just an illustration, so they just show the “good result”, for example do not use cross validation and only display the loss if accuracy does not look good. So this inspired me to change some parameters and use 3 models to separately train on 3 different data sets. I do get some interesting comparisons. they might be straightforward but I’m new to machine learning and I am very excited to seeing the differences.

So the first really small dataset I used for trying is belling the cat. What’s displayed here is just all the texts. It only has 109 unique words. And we could see here on the result that although after 30 epochs, the model has a great performance on our training set, it does poorly and unstably on the validation set.

Another dataset I used is the adventures of Sherlock Holmes. It has a relatively bigger corpus. And the accuracy of both the training and validation set looks reasonable. Although somehow the model loss still has a confusing pattern.

# Next step: results of different models

And another thing that interests me is how different models can have rather different results. Here I have to admit I only roughly understand what each layer does, but do not know exactly how to improve it according to the result. So for example the first model I used has 1 LSTM layer with 128 units, and one dense layer. I used it on the metamorphosis data, and it looks like overfitting. And then I tried model2, with 2 LSTM layers, each with 50 units. And also 2 dense layers. Somehow the accuracy looks good, although the validation loss is still increasing. So I named this part as the next step, cuz I still do not know how the change of model results in this and that’s definitely what I want to look into next.

and also the running time varies hugely between the 2 models. Both running the Sherlock data, model1 could take twice the time as model2.

# appendix

Belling the cat model 1

A picture containing chart

Description automatically generatedChart, line chart

Description automatically generated

Belling the cat model 2

Text

Description automatically generatedChart, line chart

Description automatically generated

Sherlock model 1

Chart, line chart

Description automatically generatedChart, line chart

Description automatically generated

Sherlock model 2

Chart, line chart

Description automatically generatedChart, line chart

Description automatically generated

Metamorphosis model 1

Chart, line chart

Description automatically generatedChart, line chart

Description automatically generated

Metamorphosis model2

Chart, line chart

Description automatically generatedChart, line chart

Description automatically generated

# References

Verma, S. (2021, October 5). *Understanding Input and Output shapes in LSTM | Keras*. Medium.<https://shiva-verma.medium.com/understanding-input-and-output-shape-in-lstm-keras-c501ee95c65e>

K, B. (2020, August 22). *Next Word Prediction with NLP and Deep Learning*. Medium.<https://towardsdatascience.com/next-word-prediction-with-nlp-and-deep-learning-48b9fe0a17bf>

*Next Word Prediction Model*. (n.d.). Retrieved December 7, 2021, from<https://thecleverprogrammer.com/2020/07/20/next-word-prediction-model/>

*Exploring the Next Word Predictor! | by Dhruvil Shah | Towards Data Science*. (n.d.). Retrieved December 7, 2021, from<https://towardsdatascience.com/exploring-the-next-word-predictor-5e22aeb85d8f>