Report for CIS 530 Homework 4

Yixuan Meng

September 2021

1 Preamble and Setup

To avoid error in installation, I use 'wheel==0.35.0', and comment out the lines starting with "from wheel.pep425tags" and "platform = ".

2 Character-Level Recurrent Neural Networks

The RNN model I use consists of two linear layers as the one from the tutorial. I experiment with the learning rate, number of iterations, size of the hidden layer and different optimizers (SGD and Adam).

As shown in Table 1, the model is sensitive to learning rate, the initial learning rate from the tutorial will lead to NAN losses, making the model gives a bad performance by guessing the same category for all cities, the accuracy is 11% in this case. When the learning rate is decreased to 0.002 and lower, the performance of the model gets better.

I also try to increase the number of iterations for the training, which improves the performance until the model becomes over-fitting. I try to set the size of the hidden layer to be 32, 64, 128, and 256, and compare SGD and Adam optimizers (the SGD optimizer gives a slightly better result on development dataset with learning rate 0.0005, 400,000 iters and 128 hidden states according to Table 1).

The highest accuracy my model achieved was 54.56% when using a learning rate of 0.0005, 400,000 iterations, 256 hidden layer size, and SGD optimizer.

Learning Rate	Iterations (thousands)	Hidden Size	Optimizer	Validation Accuracy (%)
0.005	250	128	SGD	11.00
0.002	250	32	SGD	48.00
0.002	250	64	SGD	49.44
0.0005	250	64	SGD	50.78
0.0005	400	64	SGD	53
0.0005	400	128	SGD	53.11
0.0005	400	128	Adam	52.89
0.0005	350	256	SGD	54.11
0.0005	400	256	SGD	**54.56**

Table 1: Validation results for the models with different parameters

A plot of training and validation loss as my best model is training is shown in Figure 1.

A plot of validation accuracy as my best model is training is shown in Figure 2.

As shown in the figures, the losses on both datasets decrease and the variance increases as the number of iterations goes up, and the validation accuracy increases fast at the beginning, but becomes more stable after about 200 thousand iterations. There is no severe over-fitting since the loss on the development dataset does not increase significantly.

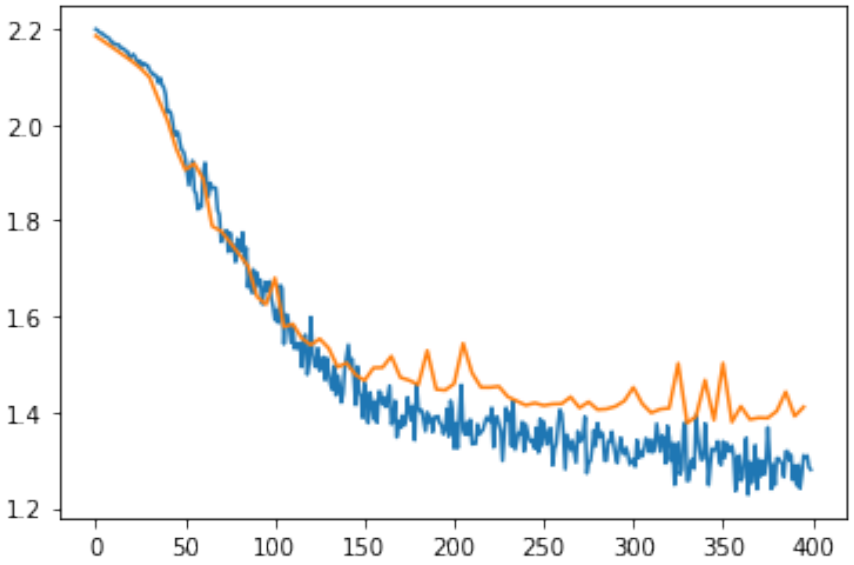

Figure 1: Plot of train loss and validation loss during training for best classification model.

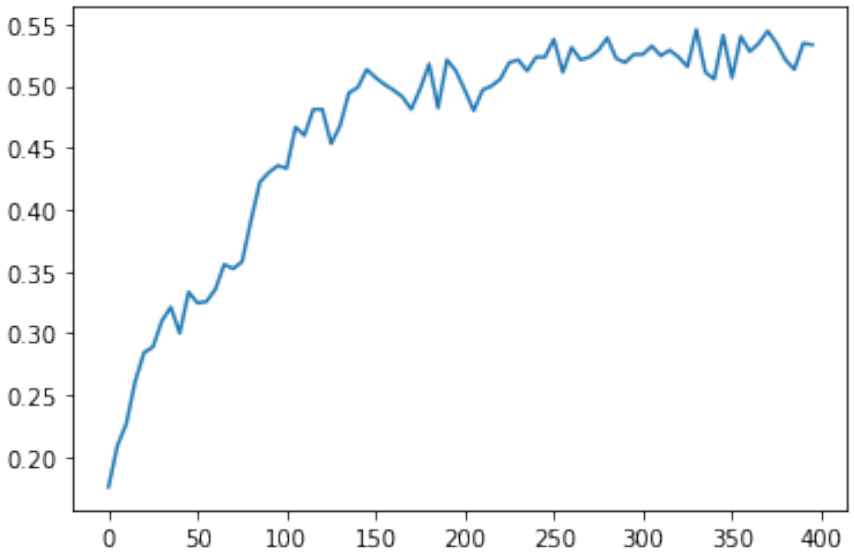

Figure 2: Plot of validation accuracy during training for best classification model.

The confusion matrix for the best model can be found in Figure 3. We can see that cities from af are often mistaken for cities from ir. Cities in these two languages both have ' in it, like delbajin pa'in (af), zoratu pa'in (ir) and both have dash-delimited words like deh-e (af) daraq-e (ir).

My model also mistaken cities from cn for cities from in. Same prefixes appears in words from both languages, like zhentou zhen (cn) minle zhen (in), and they are both space-delimited.

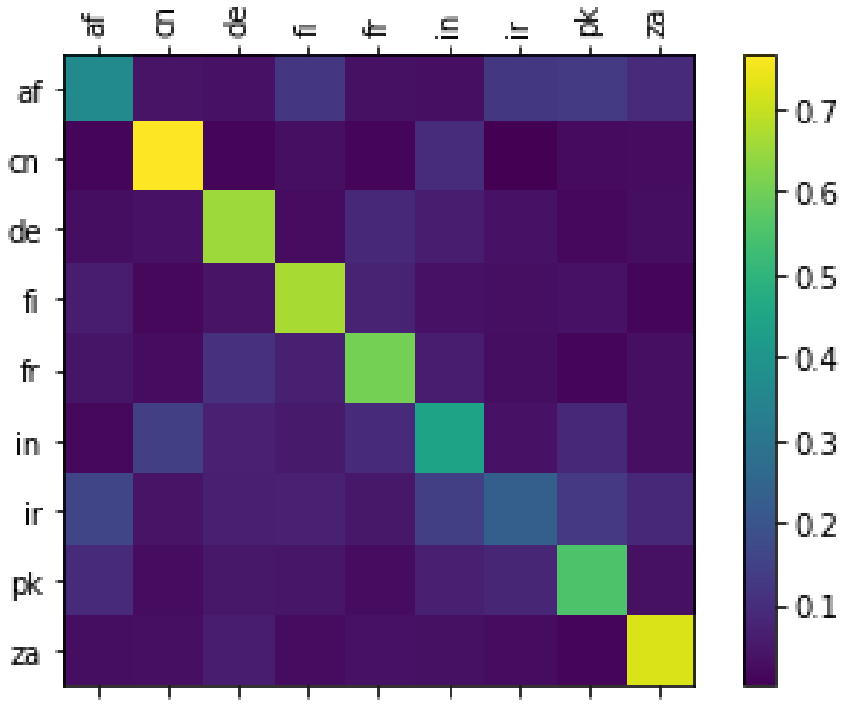

Figure 3: Confusion matrix for best model.

3 Text Generation

3.1 Model Training

I trained a generative model on a corpus of Jane Austen novels. My model consists of GRU and linear layers like the one from the tutorial. I trained the model for 2000 epoches using the Jane Austen novels dataset. The length of the dataset is 4,373,592, it contains novels of Jane Austens collected by Project Gutenberg, including 'Sense and sensibility', 'Emma' etc. The example generations are presented in Table 2.
After training 100 epoches, the model generates text like

> Whing wamy hant cof he hadte hand distuss hever por he haverVst bing nered gho she warnt was she and

The paragraph seems meaningless, but the model learns simple words like 'he'. 'was', 'and'.
After training 1000 epoches, the model generates text like

> Whereded any it, have–thee her not he dissurable sweem they moasy do now meest out be fereing of the

The model seems to learn to use comma and dash, and learned more words spellings.
After training 2000 epoches, the model generates text like

> Whind Lay was her familing for Lade, and her in the de. Brothers." "Will, such many bence, to he

The generated words makes more sense and the model seems to learn frequently appeared grammars in Jane Austen's novels ('Lay was she').

Epoches	Text
100	Whing wamy hant cof he hadte hand distuss heverpor he haverVst bing nered gho she warnt was she and
500	Whe to counsure fit much and her her havind, peally intre sun a mrom I goad she goon of Mr. Bentery ar
1000	Whereded any it, have–thee her not he dissurable sweem they moasy do now meest out be fereing of the
1500	Wherstadnest of for sheth, what I thatly?"
2000	Whind Lay was her familing for Lade, and her in the de. Brothers." "Will, such many bence, to he

Table 2: Example generations of text generation model after training certain epoches

3.2 Perplexity Computation

Since I trained the model on the Jane Austen novels dataset, I calculated the perplexity for a dissimilar dataset, the ABC music format dataset. I also calculated perplexity result on a dataset similar to the Jane Austen novels datasest - the Lord of the Rings Data from Kaggle[1]. The Lord of the Rings Data are in CSV format initially, containing script from the trilogy. The data fields in it are 'char', 'dialog','movie', where 'char' is the name of characters, 'dialog' is the script content, and 'movie' is the name of the trilogy. I processed the data into a format similar to the Shakespeare dataset. The format of the datasets are shown in Table 3.

For the ABC music format dataset, the perplexity is 1068.88, for the Lord of the Rings Data, the model reports a lower perplexity of 75.06. This makes sense because there are few words and grammars in the ABC music format dataset similar to the Jane Austen Novels, and the Lord of the Rings scripts have relatively complete sentences like in the novels, but sentence structures are not as complex as in the novels.

Jane Austen Novels	She had had a disappointment, moreover, which that book, and especially the history of her own family, must ever present the remembrance of. The heir presumptive, the very William Walter Elliot, Esq., whose rights had been so generously supported by her father, had disappointed her.
Lord of the rings	DEAGOL Oh Smeagol Ive got one! , Ive got a fish Smeagol, Smeagol! SMEAGOL Pull it in! Go on, go on, go on, pull it in! DEAGOL Arrghh!
ABC music format	P:3 cdec BcdB ABcA — A2 d4 A2 BcdB —\ c2 g4 B2 ABcA — G2 c4 A2 BcdB —]

Table 3: Example of used datasets.

[1]https://www.kaggle.com/paultimothymooney/lord-of-the-rings-data