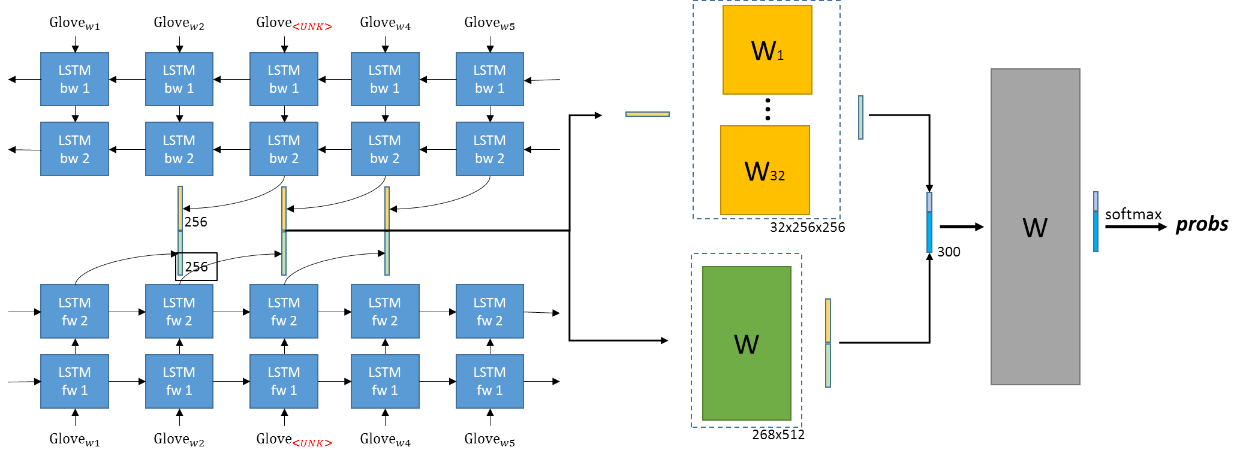
Environment

Linux, Intel i7, GTX 960M(4G memory)

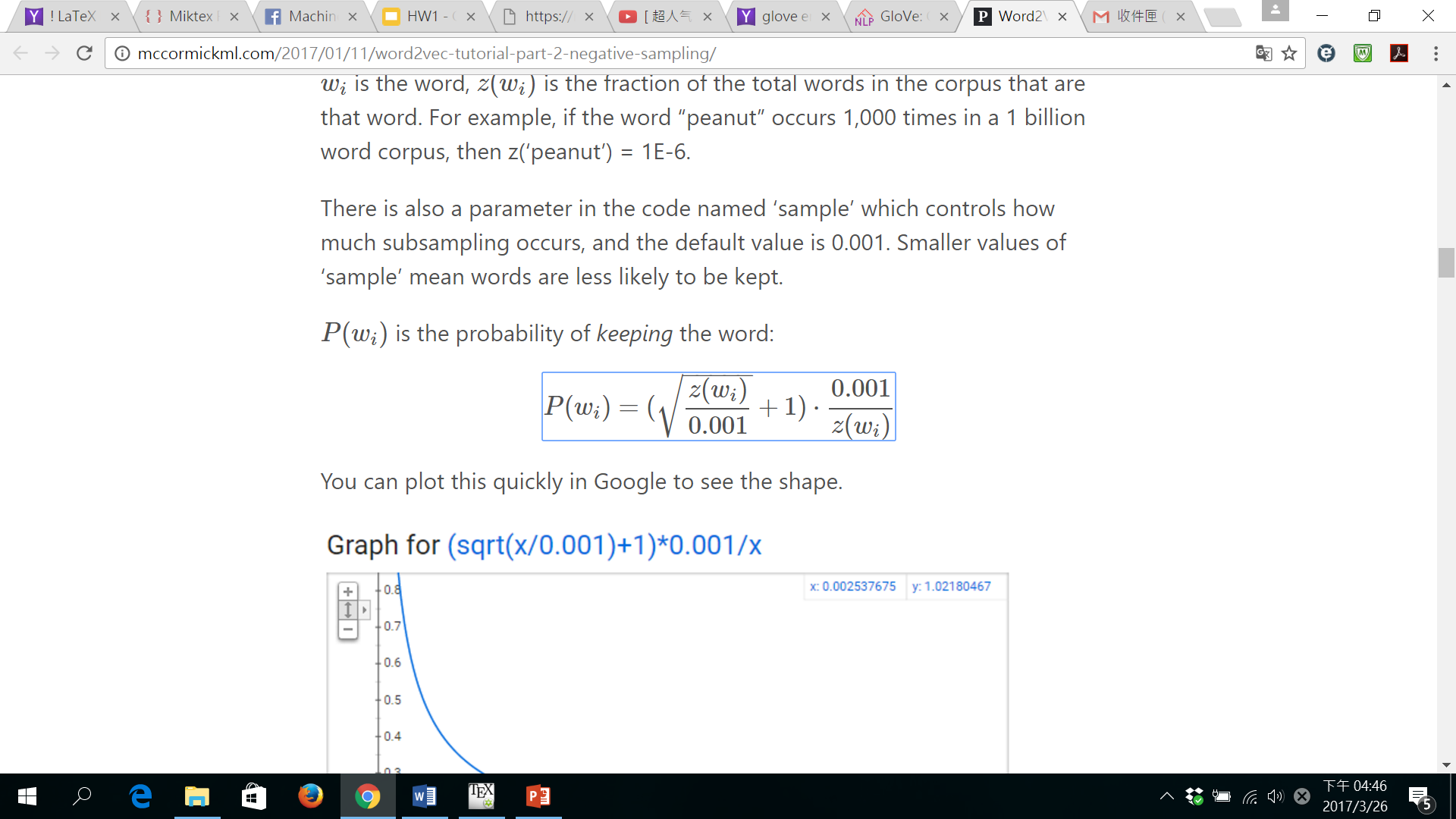
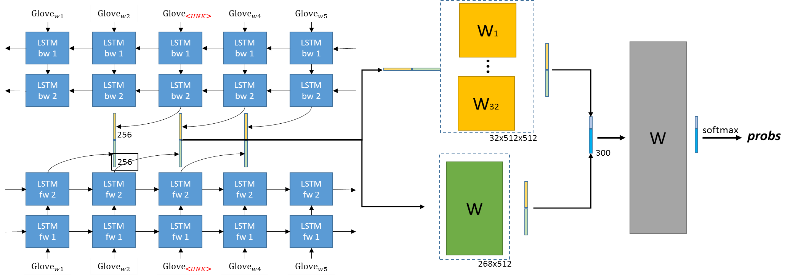
Package: tensorflow, nltk, genism, numpy

Model description



glove (<https://nlp.stanford.edu/projects/glove/>)

Performance

1. Subsample target words according to frequency (~10%up)  
   (<http://mccormickml.com/2017/01/11/word2vec-tutorial-part-2-negative-sampling/>)
2. Finetune pre-trained embedding (~5%up)
3. Stack up to 2 BiRNN (~5%up)
4. Increase diversity of feature interaction by adding (~5%up)  
   (李弘毅老師3/24上課內容)
5. In inference time, sum up scores of neighbor words (5 words left and 5 words right) (~5%up)

Experiment settings and results

**# Model Structure**

Voc\_Size = 60000

Hidden\_Size\_per\_Single\_LSTM\_Cell = 256

Basic\_Cell = tf.contrib.rnn.BasicLSTMCell(Hidden\_Size, forget\_bias=0.1)

Num\_Stack\_Layer = 2

Pre\_Trained\_Ebd = “Glove”

**# Training**

Train\_Iters = 90000 # ~ 7 epochs = ~ 15 hours

Batch\_Size = 64

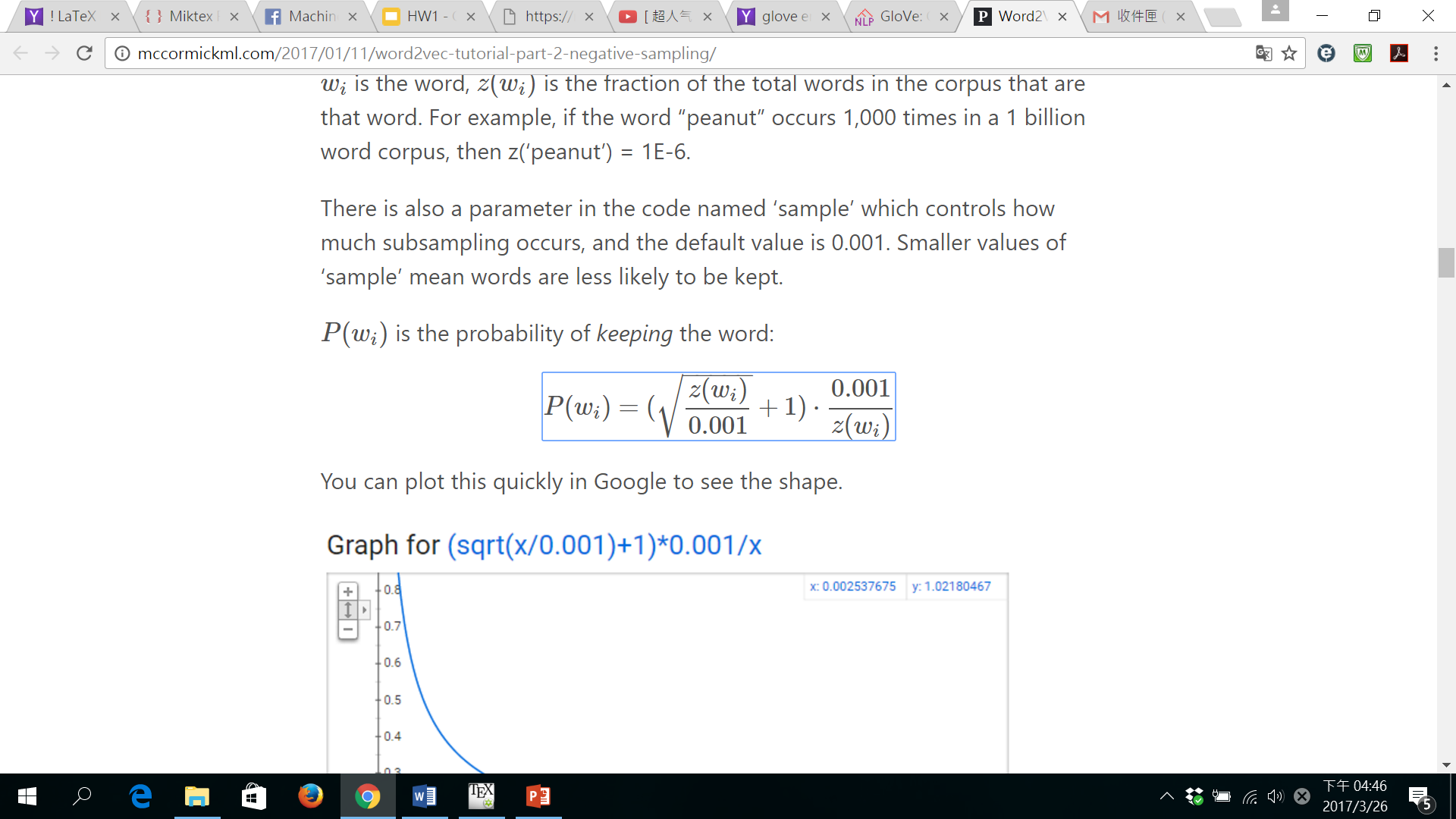
Loss = “Sampled\_Softmax\_Loss”

Sampled\_Softmax\_Loss\_Num\_Sampled = 1024

Optimizer\_Type = “Adam”

Learning\_Rate = 0.01

Learning\_Rate\_Decay\_Step = 39000 # ~3 epochs

Finetune\_Embedding\_Rate = 0.05 \* current\_learning\_rate

SubSample\_Rate = 0.001 #

**# Results**

**Public set: 0.62308 Private set: 0.65385**