Natural Language Processing

Assignment 1, CSE 628

Shilpa gupta

This task is a part of Assignment 1 of CSE 628 course. The aim was to understand the internals of word embedding generation using cross entropy and noise contrastive estimation**.**

1. **Hyper Parameter Explored** :

a. **Batch Size :**

Batch Size defines number of samples that are going to propagate through the neural network. We take our data into small batches and train the model on these batches one after another. The smaller the batch the less accurate estimate of the gradient. But the model trains faster with small batch size. If we take the batch size to be higher it can take longer time and more memory to train the model.

b. **Embedding Size :**

Embedding size is basically the number of dimensions in which we are representing each word. This is equivalent to the output layer of the neural network. If the embedding size is very small It may not capture the full context of the given word but if it is very big it will make our model equivalently bigger and complex. Also it can get more specific to the word rather than its context.

c. **Skip window/No. of skips :**

Skip window is the window span that we consider during the generation of the word pair of target and context words for the training. No of skips is the number of pairs that we fetch from each window. Larger window tends to capture the topic or domain information more while the smaller window on the other hand capture the information about the words, the words which are functionally similar.

d. **Learning rate :**

Learning rate is the size of step that we take towards the goal in the process of gradient descent. If the learning rate is very high then we may diverge and miss the goal so there will be no learning at all. On the other hand if learning rate is very low then the model will take longer time to diverge.

e. No. of samples :

No. of samples signifies the number of negative words to be considered for noise contrastive estimation. Increasing the number of noise words can decrease the efficiency of the model because It will decrease the ratio of valid words for in the training.

f. Iterations :

Iterations is the number of times we are iterating in the network to adjust the parameters in order to minimize the error. Increasing number of iterations might help in better training of model but up to a certain limit. After a limit increasing number of iterations might not help in training also can increase the risk of overfitting in the model.

2. **Results** :

Cross Entropy:

Below are few different set of combinations of hyper parameters and their performance on developing.txt that we tried.

Batch Size :

Batch Size	Embed Size	Skip window	No. of skips	iterations	learning rate	Accuracy	Most illustrative	Least Illustrative
152	128	2	4	100001	1	31.40%	32.10%	30.60%
128	128	2	4	100001	1	34.30%	33.70%	34.80%
100	128	2	4	100001	1	34.20%	32.90%	35.50%
80	128	2	4	100001	1	33.00%	32.80%	33.10%

In the cross entropy model I tried with different values of batch size from 80 to 152. As we can see changing the batch size in both the directions resulted in decreasing the accuracy. So we considered the 128 to be the best value for the batch size.

Embedding Size :

Batch Size	Embed Size	Skip window	No. of skips	iterations	learning rate	Accuracy	Most illustrative	Least Illustrative
128	128	2	4	100001	1	33.00%	32.20%	33.80%
128	100	2	4	100001	1	29.80%	29.20%	30.40%
128	152	2	4	100001	1	30.90%	31.80%	30.00%

Similar to batch size we varied the embedding size from 100 to 152 it also resulted in decrement of the accuracy in both directions.

Skip Window :

Batch Size	Embed Size	Skip window	No. of skips	iterations	learning rate	Accuracy	Most illustrative	Least Illustrative
128	128	1	2	100001	1	32.10%	33.80%	30.30%
100	128	1	2	100001	1	34.70%	32.80%	36.60%
200	128	1	2	100001	1	32.50%	32.90%	32.10%

This time we tried the skip window of 2 with different batch size values and and as we see the with 128/128 combination skip window of 1 was increasing the accuracy a bit.

Learning Rate :

Batch Size	Embed Size	Skip window	No. of skips	iterations	learning rate	Accuracy	Most illustrative	Least Illustrative
128	128	2	4	100001	1	34.30%	33.70%	34.80%
100	128	1	2	100001	0.5	33.10%	31.00%	35.20%
100	128	1	2	100001	1.5	32.20%	31.60%	32.80%

We tried to increase and decrease the learning rate as well. But the best accuracy was coming with learning rate 1.

After trying and testing different methods I finally concluded that the below set of hyper parameters were working best for me.

Batch Size : 100

Embed Size : 128

Skip Window : 1

No of Skips : 2

Iterations : 100001

Learning Rate : 1

Noise Contrastive Estimation :

Batch Size	Embed Size	Skip window	No. of skips	iterations	learning rate	no. of samples	Accuracy	Most illustrative	Least Illustrative
128	128	2	4	100001	1	64	32.30%	31.50%	33%
128	128	1	2	100001	1	64	33.20%	33.10%	33.20%
128	128	2	4	100001	1	32	31.70%	31.70%	31.60%
128	128	2	4	100001	1	48	32.60%	33.30%	31.90%
100	128	1	2	100001	1	64	32.10%	31.50%	32.70%

We tried and tested noise contrastive estimation of different set of hyper parameters. As we can see that changing the batch or embedding size was not making much difference so I kept both to be default and only changed the skip window and No. of skips in the model which gave me the most accuracy.

Below is the setup giving highest accuracy :

Batch Size : 128

Embed Size : 128

Skip Window : 1

No of Skips : 2

Iterations : 100001

Learning Rate : 1

No of samples : 64

3. Top 20 similar words :

Cross Entropy Loss :

First : same, second, nicodemou, kaufen, cinematography, balumnia, izvinite, last, handax, antipatterns, ayase, landport, seriche, engrish, aground, united, witherspoon, schemel, wyggeston, psilotsin,

American : telemarketers, asymptomatic, ulema, sprachgeschichte, english, dulas, aboiteau, debauched, echos, farafra, chthonian, downright, stomata, carnosauria, hjeuk, imbrication, burgard, cource, geoworks, phane,

Would : can, could, will, may, should, might, had, to, was, illyricum, gaelscoil, must, were, macdraw, darby, romel, exhultation, maginnes, cannot, succeding,

Noise Contrastive loss :

First : reuptake, from, setting, in, on, deccani, and, lives, to, between, anat, same, biodiversidad, cephalorhyncus, browns, word, including, during, early, wildstorm,

Would : is, were, was, may, could, are, can, has, they, bnetd, macabi, be, unindustrialized, shares, styleheads, s, will, his, although, in,

American : and, of, thz, for, nkvd, although, with, pulau, busan, subsistence, paneer, jel, hearths, informal, portola, pigeons, recitative, but, in, sanford,

As we can see the words coming similar to would are all kind of making sense, as they are all coming into same class of words. But for others it is showing a little of gibberish words. Most of them are not making sense may be because of the input text.

4. **Summary of NCE loss :**

Noise Contrastive Estimation is a way of calculating loss function while building a model to correctly predict the context of a given word. The idea behind NCE is to reduce the density estimation problem to a probabilistic binary classification problem. The basic idea is to train a logistic regression classifier to discriminate between samples from the data distribution and samples from some "noise" distribution, based on the ratio of probabilities of the sample under model and the noise distribution. We can thus reduce the problem of predicting the correct word to a binary classification task, where the model tries to distinguish positive, genuine data from noise samples.

The Advantage of NCE is that it allows to fit models that are not explicitly normalized which makes our training time independent of the vocabulary size.

To learn the distribution of words for some specific context h, we create a auxiliary binary classification problem treating the training data as positive examples and samples from noise distribution as negative examples. We can use the unigram distribution of the training data as the noise distribution. we assume that noise samples are k times more frequent then data samples. We fit the model by maximizing the log posterior probability of the correct labels averaged over the data and noise samples.

Here we are summing over k noise samples in place of summing over the full vocabulary making the NCE training time linear in the number of noise samples and independent of vocabulary size. NCE is less sensitive to the mismatch of data noise distribution than importance sampling.