Stony Brook University
Department Of Computer Science

CSE 538 Natural Language Processing - Fall 2019
Instructor: Niranjan Balsubramanian

Assignment 1

Madhusmita Dash (SBU ID: **112715430**)

1. Hyperparameters explored for Cross entropy and NCE:

a. Batch_size: It controls the number of samples taken in each batch for training. If batch_size is very large, it can cause the accuracy to decrease. If the batch size is small, we can get a regularizing effect as the sample will add some noise.

b. Skip_window: It controls the window size, i.e the number of words on each side of the context word to be considered. If the skip_window is very large, then words that are far from the context word can add noise and we may lose the effect of the nearby words which are more likely to be related to the context word.

c. Num_skips: The number of samples to be taken from a window. If we take more samples from a window then the nearby words will have more effect on the context word which might be unnecessary as the context word can be represented by a fewer number of nearby words.

d. Max_num_steps: The maximum number of steps is the number of epochs for which we are training the data. If we train for a large max_num_steps the chances of overfitting also increase.

e. Num_sampled: The number of negative samples taken to normalize the NCE loss. If we take a large num_sampled then the effect of not taking the complete vocabulary is lost.

f. learning _rate: It controls how quickly or slowly the gradient descent takes steps and the model learns. Increasing the learning rate will cause the model to take larger steps and may cross the minimum thus finding no solution. Decreasing the learning rate will cause the model to learn very slowly and take a lot of time to converge.

2. Different configurations tried:

Cross entropy:

Learning Rate	Batch size	Skip window	Num skips	num_sampled	max_num_steps	Model Accuracy	Word Analogy
0.5	128	4	8	64	200001	4.8	33.8
1	128	4	4	64	200001	5.32	33.2
1	256	2	4	64	75000	5.07	33.1
1	64	4	8	64	75000	4.13	33.4
0.5	256	2	4	64	75000	5.11	33.7
1	256	4	4	64	200001	5.33	33.4

Trends: The maximum accuracy for the word analogy task is for the above configuration when we increase the batch size and decrease the learning rate.

The accuracy improved when the batch size was reduced as well as the max_num_steps decreased. Maybe the large num_steps caused overfitting. Reducing the learning rate also improved performance.

Keeping the num_skips either equal to or double the skip_window sometimes gave better accuracy and sometimes reduced the accuracy.

NCE:

Learning Rate	Batch size	Skip window	Num skips	num_sampled	max_num_steps	Model Accuracy	Word Analogy
0.5	128	4	8	64	200001	0.97	34.1
1	128	4	4	64	75000	1.98	34.5
1	256	2	4	64	75000	1.44	34.2
1	64	4	8	64	75000	1.45	32.1
0.5	256	2	4	64	75000	1.26	34.1
1	256	4	4	64	200001	1.32	34.2

Trends: The maximum accuracy for the word analogy task when we decrease the max_num_steps and the num_skips.

Reducing the learning rate increased accuracy. The max_num steps when reduced gave better performance than other configurations.
Reducing the batch size, however, decreased accuracy.

3. Top 20 similar words using Cross entropy:

From the below 20 words for each word, we can see that with first we get words like church, kingdom, city, fact etc. which looks like the term first acts as a qualifier. The first church, first kingdom etc.

For american, we get words like game, organisational, union, majority,western etc which looks like it defines the things the country america is famous for. Western culture, games (sports tournaments), big organisations etc are american.

For would, we get words like in, on, into, during, then, etc which don’t make sense to its meaning. However, they could be the stop words that appear in a sentence with would.

first	**american**	**would**
same	time	through
north	modern	in
east	film	over
city	second	within
eastern	game	on
west	organisational	into
british	battle	from
south	kodansha	then
states	union	thus
church	majority	during
soviet	king	around
uk	fly	black
fact	container	with
kingdom	swirled	both
case	western	after
region	ours	for
us	situation	by
america	campaigning	became
middle	morphology	of
process	bannister	at

Top 20 similar words using NCE:

For first, we get words like thousands, hundreds, examples, types etc. which define quantities.
For american, we get words like power, united, together etc. which look like they are associated with the country.
For would, we get words like many, his, some, her, its, an, all which are pronouns/quantities.

first	**american**	**would**
those	both	its
different	through	many
types	became	his
the	around	some
examples	within	all
kinds	over	used
thousands	black	other
because	b	her
hundreds	i	an
an	long	american
iapetus	would	would
a	into	what
them	new	how
various	together	modern
list	d	free
forms	power	up
some	united	europe
been	now	especially
names	language	a
several	my	western

4. Noise Contrastive Estimation:

The likelihood of the context and target word pairs are normalized over the entire vocabulary. When the vocabulary is huge, the computation is expensive to normalize each training example over the output of every vocabulary word.

Negative sampling: Instead of summing over each incorrect vocabulary we can select a few negative samples and normalize over them. This is called negative sampling. The number of gradients and updates reduces from embedding*Vocabulary_size to embedding*(k+1) where k +1 is the number of negative samples and the actual context, word pair.

Noise contrastive estimation uses the concept of negative sampling to create a new task of learning true distribution. The k negative samples are given a label of 0 and the true pair as label 1 which implies that the true pair is from a real distribution and the k pairs are noise.

$$logit \ = \ log(P) \ - \ log(Q)$$

We want to know the log-odds that a class is from a true distribution than a noise distribution. The term log(Q) is used to generate the k samples which we use as noise. We can take Q to be uniformly random or take the rare words in the vocabulary.

$$P(D = 1|w,) = P(w)/(P(w) + kP_n(w)) \ = \ (\Delta s(w))$$

The above equation calculates the probability that a word is from a true distribution (label D=1) assuming the sample from noise distribution is k times more frequent than the sample from D=1.

$$(\Delta s(w)) \ = \ s(w, h) - log(kP_n(w))$$

We are ignoring the normalized denominator for $s(w, h)$ as we are considering an unnormalized model as NCE objective encourages to model to be normalized if it contains the model class distribution.

We want to maximize the log-posterior probability of the correct labels D averaging over the data and noise samples.

$$J(theta, batch) \ = \ \ -\sum(log(P(D = 1, w_c|w) \ + \ \sum log(P(D = 0, w_x|w))$$

$$J(theta, batch) \ = \ - \sum(log(P(D = 1, w_c|w) \ + \ \sum log(1 \ - \ P(D = 1, w_x|w))$$

We model the unsupervised learning problem as a supervised logistic regression problem and predict the P by minimizing the loss. In this way, the model learns the log odds ratio P (true distribution). In NCE, we explicitly take into consideration the probability that the sample came from a noise distribution.

References

1. **https://blog.zakjost.com/post/nce-intro/**
2. **https://www.cs.toronto.edu/~amnih/papers/wordreps.pdf**