CSE 538 NLP Assignment 1 Report

111462188 Mihir Chakradeo

1 and 2: Hyperparameter experiments and analogy results

[I] Cross Entropy Experiments

embedding_size = 128, valid_size = 16, valid_window = 100

Hyperparameters	**Loss**	**Accuracy Least illustrative**	**Accuracy Most illustrative**	**Overall Accuracy**	**Top 20 Similar**	**Comments**
max_num_steps = 200001 batch_size = 128 skip_window = 4 num_skips = 8 learning_rate = 1	4.82	35.0	32.6	33.8	**'american':** ['german', 'british', 'english', 'its', 'carmen', 'usurped', 'russian', 'french', 'canadian', 'anatomical', 'bioavailability', 'ceiling', 'oncifelis', 'his', 'united', 'projectiles', 'integral', 'ww', 'hobbled', 'reject'], **'would':** ['could', 'will', 'must', 'might', 'can', 'did', 'should', 'does', 'may', 'began', 'was', 'seems', 'had', 'do', 't', 'appears', 'is', 'made', 'argued', 'householder'], **'first':** ['last', 'same', 'most', 'largest', 'main', 'latter', 'following', 'ismailis',	Default hyperparameters

					'original', 'gevarus', 'control', 'next', 'because', 'actual', 'due', 'steadfastly', 'rotary', 'callings', 'best', 'hume']	
max_num_steps = 200001 batch_size = 128 skip_window = 8 num_skips = 16 learning_rate = 0.001	4.87	35.6	32.4	34.0	**'american':** ['german', 'british', 'french', 'english', 'italian', 'its', 'war', 'russian', 'european', 'eu', 'international', 'of', 'irish', 'canadian', 'borges', 'united', 'trade', 'd', 'other', 'player'], **'would':** ['not', 'could', 'that', 'will', 'been', 'we', 'said', 'must', 'might', 'they', 'do', 'does', 'who', 'did', 'you', 'to', 'seems', 'if', 'should', 'may'], **'first':** ['last', 'name', 'following', 'during', 'most', 'original', 'second', 'same', 'until', 'end', 'after', 'best', 'before', 'book', 'city', 'united', 'main', 'next', 'beginning', 'title']	The accuracy of least illustrative increases as we increase the skip window and num skips
max_num_steps = 200001 batch_size = 256 skip_window = 4 num_skips = 8	5.56	35.6	32.4	34.0	**'american':** ['german', 'british', 'french', 'english', 'italian', 'its', 'war', 'russian',	On increasing the batch size and decreasin

learning_rate = 0.001					'european', 'eu', 'international', 'of', 'irish', 'canadian', 'borges', 'united', 'trade', 'd', 'other', 'barzani'], **'would':** ['not', 'could', 'that', 'will', 'been', 'we', 'said', 'must', 'might', 'they', 'do', 'does', 'who', 'did', 'you', 'to', 'seems', 'if', 'should', 'may'], **'first':** ['last', 'name', 'following', 'during', 'most', 'original', 'second', 'same', 'until', 'end', 'after', 'best', 'before', 'book', 'city', 'united', 'main', 'next', 'beginning', 'title']	g the skip_window and num_skips size, the loss was more but the result did not change
max_num_steps = 800001 batch_size = 64 skip_window = 8 num_skips = 16 learning_rate = 0.001	4.17	35.6	32.5	34.0	**'american':** ['german', 'british', 'french', 'english', 'italian', 'its', 'russian', 'war', 'european', 'eu', 'international', 'of', 'irish', 'borges', 'canadian', 'united', 'trade', 'd', 'player', 'writer'], **'would':** ['not', 'could', 'will', 'that', 'been', 'we', 'said', 'must', 'might', 'they', 'do', 'does',	Decreasing batch size and increasing number of training steps reduced the loss, but the final result of the model did not change

					'who', 'did', 'you', 'to', 'seems', 'if', 'should', 'may'], **'first':** ['last', 'name', 'following', 'during', 'most', 'original', 'second', 'same', 'until', 'end', 'after', 'best', 'before', 'book', 'city', 'united', 'next', 'main', 'beginning', 'title']	
max_num_steps = 700001 Embedding_size = 512 batch_size = 256 skip_window = 16 num_skips = 32 learning_rate = 0.001	5.77	37.5	37.1	37.3	**'american':** ['taragarh', 'translates', 'citymayors', 'horehound', 'comden', 'mythological', 'schwyz', 'lenoir', 'bada', 'kvatter', 'giulietta', 'vesi', 'chamavi', 'koto', 'tests', 'consequence', 'wares', 'squabbling', 'osa', 'ngc'], **'would':** ['delacroix', 'supercomputer', 'mainframe', 'vedea', 'gac', 'duffy', 'flattop', 'distinction', 'dreamlike', 'blackberries', 'viadana', 'africanists', 'courted', 'istar', 'paladins', 'raskin', 'fluid', 'jags', 'enveloping', 'halo'],	Increasing the embedding size gave high loss, better accuracy. But the similar words do not capture valid information

					'first': ['isoprene', 'cathartidae', 'malcontent', 'lethe', 'parallelism', 'norther', 'yvonne', 'antivirus', 'salg', 'dullness', 'deliverance', 'leonid', 'satemization', 'flamel', 'empiricist', 'claiborne', 'uf', 'erikson', 'predominating', 'shoulders']	
max_num_steps = 500001 Embedding_size = 128 batch_size = 128 skip_window = 2 num_skips = 4 learning_rate = 0.001	4.23	35.6	32.4	34.0	**'american':** ['german', 'british', 'french', 'english', 'italian', 'its', 'war', 'russian', 'european', 'eu', 'international', 'of', 'irish', 'canadian', 'borges', 'united', 'trade', 'd', 'other', 'barzani'], **'would':** ['not', 'that', 'could', 'will', 'been', 'we', 'said', 'must', 'might', 'they', 'do', 'does', 'who', 'did', 'you', 'to', 'seems', 'if', 'should', 'may'], **'first':** ['last', 'name', 'following', 'during', 'most', 'original', 'second', 'same', 'until', 'end', 'after', 'best',	This model gave the best performance. The idea was using the result from third experiment (smaller window gives better results for analogy) and reducing the batch size. Moreover, the small learning rate helped to

					'before', 'book', 'city', 'united', 'main', 'next', 'beginning', 'title']	reduce the loss as well (compared to experiment 3).

Some observations:

1. Increasing the embedding size led to increase in the loss, but the accuracy came out to be better. However, the top 20 similar words failed to capture useful information
2. On decreasing the window size (for more emphasis on local information), the loss was more than the one with default hyperparameters. However, the accuracy slightly improved and the similar words captured better information.

[II] NCE Experiments

Hyperparameters	**Loss**	**Accuracy Least illustrative**	**Accuracy Most illustrative**	**Overall Accuracy**	**Top 20 Similar**	**Comments**
max_num_steps = 200001 batch_size = 128 skip_window = 4 num_skips = 8 learning_rate = 1	1.27	33.9	32.6	33.3	**'american':** ['factions', 'kekeke', 'pustaka', 'surakarta', 'kennealy', 'nymphs', 'negligent', 'returns', 'redox', 'tabla', 'observes', 'vengeful', 'slip', 'rona', 'androgens', 'midori', 'domiciled', 'coe', 'ukasiewicz', 'nox'], **'would':** ['esclusa', 'catharist', 'husayni', 'xos', 'devonian', 'subtractions',	Default setting (without pretrainedmodel)

					'selfridge', 'patti', 'guided', 'pwned', 'seis', 'imipramine', 'muff', 'weft', 'mizrahim', 'leguati', 'intracranial', 'dimension', 'torturing', 'marl'], **'first':** ['luisa', 'bleaches', 'breed', 'toei', 'prospective', 'nanometers', 'jamie', 'ddd', 'dancing', 'saravane', 'fcptools', 'sir', 'esc', 'ledeen', 'chinese', 'instill', 'azk', 'mvps', 'iici', 'unearthed']	
max_num_steps = 200001 embedding_size = 128 batch_size = 128 skip_window = 4 num_skips = 8 learning_rate = 0.001	1.66	36.1	33.6	34.8	**'american':** ['british', 'german', 'english', 'french', 'war', 'its', 'united', 'european', 'states', 'eu', 'brought', 'international', 'borges', 'civil', 'italian', 'other', 'bioavailability', 'agave', 'of', 'century'], **'would':** ['not', 'been', 'they', 'will', 'could', 'who', 'that', 'we', 'might', 'said', 'only', 'to', 'but', 'must', 'did',	Just by reducing the learning rate I got a bigger loss, but I was able to get a better overall accuracy. That means that the model may generalize well

					'these', 'does', 'from', 'do', 'had'], **'first':** ['most', 'during', 'name', 'after', 'at', 'was', 'following', 'one', 'and', 'of', 'last', 'in', 'on', 's', 'he', 'which', 'nine', 'to', 'from', 'is']	
max_num_steps = 200001 embedding_size = 128 batch_size = 128 skip_window = 4 num_skips = 8 learning_rate = 0.001 num_sampled = 32	1.38	35.8	33.0	34.4	**'american':** ['german', 'british', 'english', 'french', 'war', 'its', 'italian', 'european', 'russian', 'international', 'eu', 'united', 'borges', 'states', 'other', 'participation', 'd', 'irish', 'trade', 'between'], **'would':** ['not', 'been', 'could', 'will', 'they', 'we', 'said', 'might', 'must', 'who', 'did', 'does', 'do', 'that', 'you', 'only', 'but', 'these', 'seems', 'if'], **'first':** ['name', 'last', 'during', 'most', 'following', 'after', 'original', 'until', 'same', 'end', 'second',	Reducing the number of negative samples did not affect the accuracy

					'at', 'before', 'was', 'th', 'book', 'best', 'united', 'when', 'world']	
max_num_steps = 200001 embedding_size = 128 batch_size = 128 skip_window = 4 num_skips = 8 learning_rate = 0.001 num_sampled = 16	0.98	35.6	32.5	34.0	**'american':** ['german', 'british', 'french', 'english', 'italian', 'its', 'war', 'russian', 'european', 'international', 'other', 'united', 'eu', 'states', 'd', 'irish', 'canadian', 'between', 'union', 'borges'], **'would':** ['not', 'will', 'could', 'that', 'been', 'we', 'said', 'must', 'might', 'they', 'who', 'do', 'does', 'did', 'you', 'if', 'but', 'only', 'seems', 'so'], **'first':** ['last', 'name', 'following', 'during', 'most', 'original', 'second', 'until', 'after', 'same', 'end', 'before', 'at', 'book', 'city', 'was', 'of', 'best', 'th', 'united']	Reducing the number of negative samples even further. Loss decreases, but the overall accuracy does not increase
max_num_steps = 200001 embedding_size = 128	2.82	36.0	33.6	34.8	**'american':** ['of', 'war', 'its', 'three', 'nine', 'four', 'five',	I tried increasing the number of negative samples, the loss

batch_size = 128 skip_window = 4 num_skips = 8 learning_rate = 0.001 num_sampled = 128					'that', 'seven', 'by', 'eight', 'UNK', 'six', 'in', 'two', 's', 'for', 'from', 'one', 'are'], **'would':** ['not', 'that', 'they', 'who', 'been', 'to', 'will', 'from', 'four', 'but', 'three', 'five', 'seven', 'eight', 'with', 'he', 'which', 'only', 'this', 'it'], **'first':** ['most', 'at', 'was', 'after', 'and', 's', 'one', 'on', 'he', 'in', 'which', 'to', 'from', 'of', 'three', 'during', 'eight', 'is', 'nine', 'by']	did not decrease much, but the accuracy improved a bit. However, the similar words were not satisfactory
max_num_steps = 600001 embedding_size = 128 batch_size = 128 skip_window = 4 num_skips = 8 learning_rate = 0.001 num_sampled = 128	2.81	36.1	33.5	34.8	**'american':** ['of', 'three', 'nine', 'four', 'five', 'war', 'its', 'seven', 'eight', 'UNK', 'that', 'by', 'six', 's', 'two', 'in', 'from', 'for', 'are', 'one'], **'would':** ['not', 'that', 'they', 'who', 'been', 'to', 'will', 'from', 'but', 'with', 'which', 'he', 'four', 'only', 'three', 'seven', 'this', 'it', 'eight', 'five'],	Similar model as earlier, just increased number of training steps, and still the same result holds. The similar words do not make any sense

					'first': ['most', 'at', 'was', 'after', 'on', 's', 'he', 'which', 'and', 'during', 'from', 'to', 'by', 'one', 'is', 'in', 'three', 'of', 'eight', 'that']	

Some Observations:

1. As we increase the number of negative examples, the accuracy increases, but the similarity task does not give good words. Tried the same configuration with more training steps, but still the similarity task produced bad results.
2. Decreasing the number of negative samples helps in achieving smaller loss values, but the accuracy on analogy part does not change much

3. Top 20 similar words

Based on the best models used

I. **Cross entropy**

first	'last', 'name', 'following', 'during', 'most', 'original', 'second', 'same', 'until', 'end', 'after', 'best', 'before', 'book', 'city', 'united', 'main', 'next', 'beginning', 'title'
american	'german', 'british', 'french', 'english', 'italian', 'its', 'war', 'russian', 'european', 'eu', 'international', 'of', 'irish', 'canadian', 'borges', 'united', 'trade', 'd', 'other', 'barzani'
would	'not', 'that', 'could', 'will', 'been', 'we', 'said', 'must', 'might', 'they', 'do', 'does', 'who', 'did', 'you', 'to', 'seems', 'if', 'should', 'may'

II. **NCE**

first	'most', 'during', 'name', 'after', 'at', 'was', 'following', 'one', 'and', 'of', 'last', 'in', 'on', 's', 'he', 'which', 'nine', 'to', 'from', 'is'
american	'british', 'german', 'english', 'french', 'war', 'its', 'united', 'european', 'states', 'eu',

	'brought', 'international', 'borges', 'civil', 'italian', 'other', 'bioavailability', 'agave', 'of', 'century'
would	'not', 'been', 'they', 'will', 'could', 'who', 'that', 'we', 'might', 'said', 'only', 'to', 'but', 'must', 'did', 'these', 'does', 'from', 'do', 'had'

Observations:

The similarities capture the following meaning:

American: the models are able to capture different nationalities, that is why the most similar words which show up are german, british, french, etc.

First: the models up to some extent are capturing the different positions, hence words like last, after, following, before, beginning, etc. are most similar. There are some instances of antonyms being detected as well, example last, end, after.

Would: the models capture modal verbs, hence the examples like might, must, may, etc. are most similar.

For all the words, in both models, there are some outliers, like the letter 'd' in american, or the 'century' in american. This happens because of some inherent noise in the training data.

4. Summary of NCE Loss

The idea behind NCE is reducing problem of density estimation to a binary classification problem, where we train a logistic regression classifier to classify samples which come from the data distribution and samples which come from a noisy distribution (negative samples). The NCE loss can be optimized faster because we do not have to normalize the samples over the entire Vocabulary.

Given some context *h*, we define the predicted word's distribution as:

$$P_\theta^h(w) = \frac{\exp(s_\theta(w,h))}{\sum_{w'} \exp(s_\theta(w',h))}.$$

As NCE allows us to ignore the normalization term, we can just use the numerator term in the distribution.

For learning the distribution, we model the problem as a binary classification task, where the positive samples are the training samples, and the negative samples come from a noisy distribution $P_n(w)$.

So, we can calculate the probability that the sample came from data as:

$$P^h(D=1|w,\theta) = \frac{P_\theta^h(w)}{P_\theta^h(w) + kP_n(w)} = \sigma\left(\Delta s_\theta(w,h)\right)$$

Here, we have used *k* times $P_n(w)$ because of the assumption that frequency of the negative samples is *k* times more than the data samples.

We optimize this function by maximizing the log posterior of the labels

$$\begin{aligned} J^h(\theta) =& E_{P_d^h}\left[\log P^h(D=1|w,\theta)\right] + kE_{P_n}\left[\log P^h(D=0|w,\theta)\right] \\ =& E_{P_d^h}\left[\log \sigma\left(\Delta s_\theta(w,h)\right)\right] + kE_{P_n}\left[\log\left(1-\sigma\left(\Delta s_\theta(w,h)\right)\right)\right] \end{aligned}$$

Empirically, the Expectation is approximated by considering samples of the data. So, the final function which is to be optimized looks like:

$$J(\theta, Batch) = \sum_{(w_o,w_c)\in Batch} -\left[\log Pr(D=1,w_o|w_c) + \sum_{x\in V^k} \log(1-Pr(D=1,w_x|w_c))\right]$$