Assignment 1 - E1 246: Natural Language Understanding (2019)

Apoorv Saxena
apoorvumang@gmail.com

Abstract

This document contains the report for Assignment 1 of course E1 246: Natural Language Understanding (2019).

1 Data preprocessing

The data is taken from the Reuters Corpus. For preprocessing, the following steps were taken:

1. Removing punctuation and converting all words to lower case
2. Removing numbers. With numbers, the vocabulary was of the order 44k. As is done Word2vec (Mikolov et al., 2013), we removed numbers and replace them with the token ### which resulted in a much smaller vocabulary.

No lemmatization or stemming was done. The resulting corpus had the following statistics:

- 1,323,158 tokens
- 34,055 unique words(types)

2 Generating training data

To generate the training data ie [word, context] pairs, we used different window sizes - ± 2, ± 3 and ± 4 words. We also did subsampling of frequent words (as done by Mikolov et al.) to reduce the size of the training data. This was done using the following formula:

$$P(w_i) = 1 - \sqrt{\frac{t}{f(w_i)}}$$

where $P(w_i)$ is the probability of **discarding** the word w_i in the training data, $f(w_i)$ is the word's frequency in the corpus and t is a threshold set to 0.00001.

The statistics of the training data are given in Table 1

Window size	# of training pairs (word, context)
± 2 words	1,773,468
± 3 words	2,058,628
± 4 words	2,638,849

Table 1: Statistics of training data for different window sizes

Hyperparameter	Values
Context window size	± 2, ± 3, ± 4 words
Embedding dimension	50, 150, 300
# of negative samples	5, 15, 20

Table 2: Different hyper parameters that we experimented with

3 Hyperparameters

The hyperparameters that we tuned on were context window size, embedding dimension and number of negative samples used during negative sampling (details about negative sampling later).

Table 2 shows the different hyperparameters we tried.

4 Training the model

The objective function for training was

$$\log \sigma({v'_{w_O}}^\top v_{w_I}) + \sum_{i=1}^{k} \mathbb{E}_{w_i \sim P_n(w)} \left[\log \sigma(-{v'_{w_i}}^\top v_{w_I}) \right]$$

where w_O is the context word, w_I is the centre word, v' the context embedding and v is the word embedding. w_i are the negative words ie non-context words that are sampled from the corpus according to negative sampling probability $P_n(w) \propto f(w)^{\frac{3}{4}}$

The objective function was optimized using stochastic gradient descent. Training was done until training error convergence. No regularization was done.

5 Performance

Performance was measured using the **SimLex-999 similarity score**(Hill et al., 2014). SimLex-999 is a relatively hard statistic - it provides a way of measuring how well models capture **similarity**, rather than **relatedness or association**. For example, although 'clothes' and 'closet' are *related* words, they are *not similar* ie they do not mean the same thing. Hence they have a low SimLex-999 score but higher scores in other metrics like WordSim-353. So, for comparison purposes, we have also included the WordSim-353 score as well although the final metric that we consider is SimLex-999.

After training, we get 2 embeddings for each word: word embedding and context embedding. If w and c are the respective embedding vectors for a certain word, then we use the following 4 embeddings during evaluation:

1. w
2. c
3. $\frac{w+c}{2}$
4. $w.c$ ie the concatenated embedding (size 2*embedding dim size)

Finally, we report the score of a trained model as the **Pearson coefficient** between the SimLex-999 scores and the cosine distances between the corresponding word embeddings.

$$cosineDistance(w_1, w_2) = \frac{< w_1, w_2 >}{||w_1||||w_2||}$$

6 Results

Figures 1,2 and 3 show how the performance varies with embedding dimension, window size and number of negative samples.

Table 3 shows hyperparameters for the overall best performing models

7 Empirical Analysis

In almost all cases, the concatenated embedding ie w.c performs the best. Table 4 shows the closest words to certain handpicked words from the corpus on our best model.

Best Model	SimLex-999	WordSim-353
Window size	3	3
Embedding Dim	150	300
# Neg samples	20	20
Embedding vector	w.c	c
SimLex-999	**0.165**	0.139
WordSim-353	0.245	*0.304*

Table 3: Hyperparameters for the best performing models

7.1 Analogical reasoning

The given corpus is very small - **about 0.1%** the size of the corpus used by Mikolov et al. (1.3 million tokens vs 1 billion tokens). Hence, analogical relations like Germany : Berlin :: France : Paris are hard to answer since the number of occurrences of these words are very small. For example, *the word 'woman' is not even present* in the corpus! So we have excluded analysis on semantic analogical tasks.

7.2 Biases

As observed in Table 4, we can see a few biases that have crept into our embeddings. Since the training data is about new reports, we see that 'killed' is the 8th closest word to 'black' with cosine distance 0.320. On the other hand, the cosine distance between 'white' and 'killed' is 0.191.

We can also see high similarity between 'men' and 'jets' along with high similarity between 'women' and 'barber' as seen in Table 4.

Preparing References:

Include your own bib file like this:
`\bibliographystyle{acl_natbib}`
`\bibliography{acl2019}`

where `acl2019` corresponds to a acl2019.bib file.

References

Felix Hill, Roi Reichart, and Anna Korhonen. 2014. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *CoRR*, abs/1408.3456.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

would	black	white	men	women
be	miners	house	***jets***	martell
will	tpa	fitzwater	sending	wk
could	pbt	derivatives	surcharges	stabilized
should	nui	syria	targetting	lived
might	shale	veg	baghdad	ceremony
if	exception	cityquest	guesstimating	rafsanjani
allow	pegasus	bread	nationals	rio
probably	***killed***	ttl-f	fishing	bbls
not	mountains	follows	sabah	macandrews
must	regulate	linseed	concerns	cadillac
expected	11th	smoothed	1980s	refer
did	employed	yemen	oil-producing	returning
continue	agrimont	ussr	their	denver
unlikely	northgate	non-denatured	dryness	wellemeyer
soon	hulk	guesstimating	khamenei	13-week
that	migrant	commons	unwilling	ivaco
likely	compounded	rebate	port's	alvite
able	u.s.-ussr	whites	staging	milds
formally	near-term	regulation	addressed	***barber***

Table 4: Top 20 closest words to certain hand-picked words from the corpus

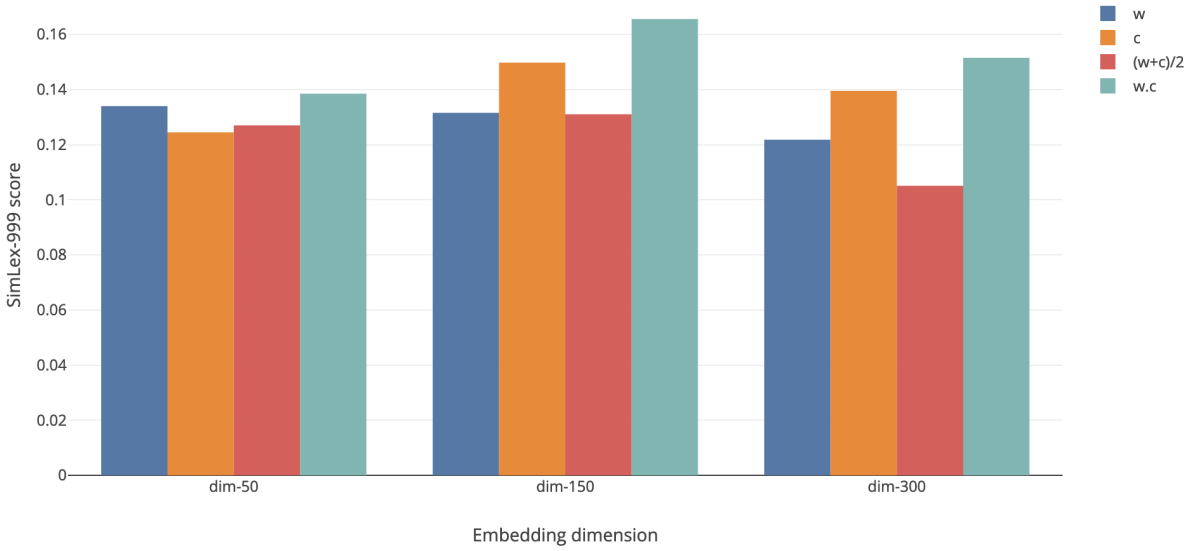

Figure 1: Best performance across different embedding dimensions

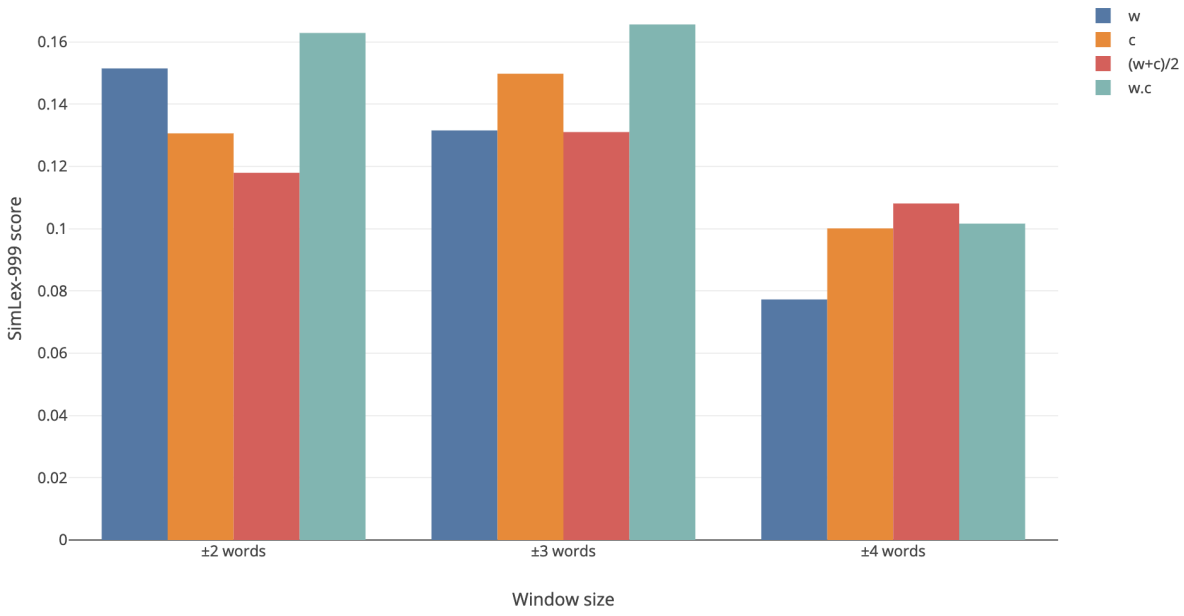

Figure 2: Best performance across different context window sizes

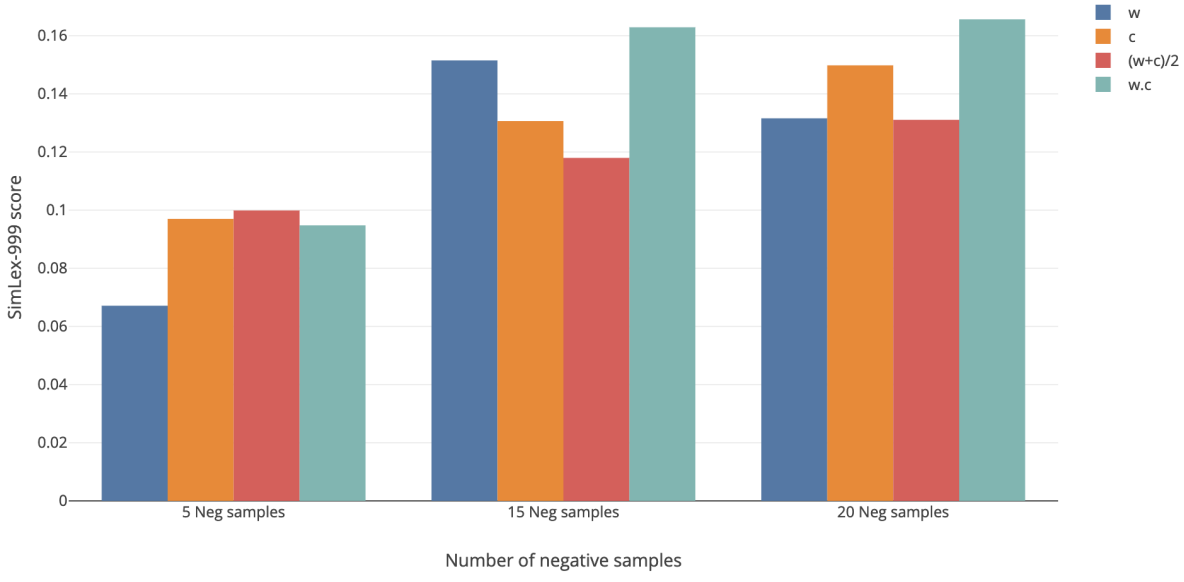

Figure 3: Best performance across number of negative samples