COMPSCI 723(001)	Forouzan Naderi
ASSIGNMENT 1	10/17/2023

Corpus:

The CoNLL-2000 corpus is a corpus of English text that has been annotated with part-of-speech tags and chunk tags. It was created for the CoNLL-2000 shared task on shallow parsing, which was held at the Conference on Natural Language Learning (CoNLL) in 2000.

The corpus contains approximately 270,000 words and 10948 sentences of Wall Street Journal text, divided into a training set and a test set. The training set contains sections 15-18 of the Wall Street Journal, and the test set contains section 20.

The part-of-speech tags in the CoNLL-2000 corpus are based on the Penn Treebank tagset. The chunk tags are used to identify noun phrases, verb phrases, and prepositional phrases.

The CoNLL-2000 corpus is a valuable resource for research in natural language processing and computational linguistics. It has been used in a variety of studies, including:

- Training and evaluating machine learning models for shallow parsing and other natural language processing tasks
- Developing new algorithms for shallow parsing
- Studying the relationship between part-of-speech tags and chunk tags
- Developing new natural language processing tools and resources

Before training a word2vec model, we performed some preprocessing steps on this corpus. First, we used a sentence tokenizer, then we applied a word tokenizer, then we converted all words to lowercase, and finally we applied a lemmatizer. To use these functions, we downloaded the punkt and wordnet packages. Additionally, because this corpus includes a bunch of numbers, we removed them before word embedding using the re library.

```
sentences = [[re.sub(r'\d+', '', word) for word in sentence] for sentence
in sentences]
nltk.download('punkt')
nltk.download('wordnet')
```

Methods:

In this assignment, we performed word embedding using two different methods: skip-gram (sg=1) and CBOW (sg=0). To better understand the difference between these two methods, we kept all other parameters the same for both methods.

```
m1 = models.Word2Vec(sentences, sg=1, epochs=300, vector_size= 100,
min_count=5)
```

visualizing word embedding:

I chose these 20 words:

```
["market", "share", "improvement", "increase", "economy", "country",
"decrease", "city", "food", "desk", "inflation", "human", "finance",
"school", "demand", "cash", "bookshelves", "capitaslism", "charge",
"power"] to visualize the two set of embeddings.
```

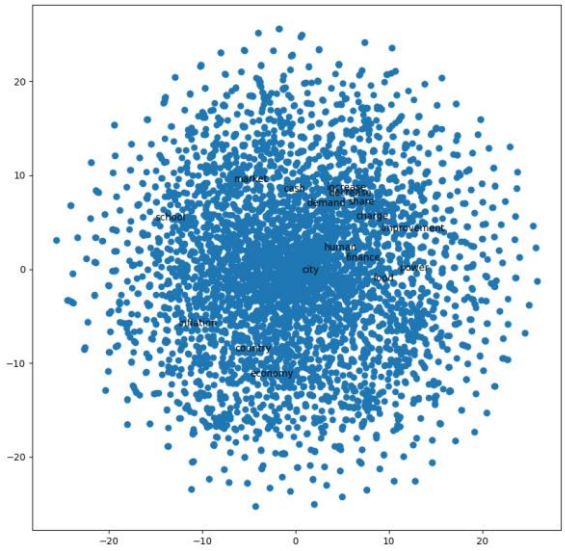

Figure 1 Word embedding visualization using model 1 (SkipGram)

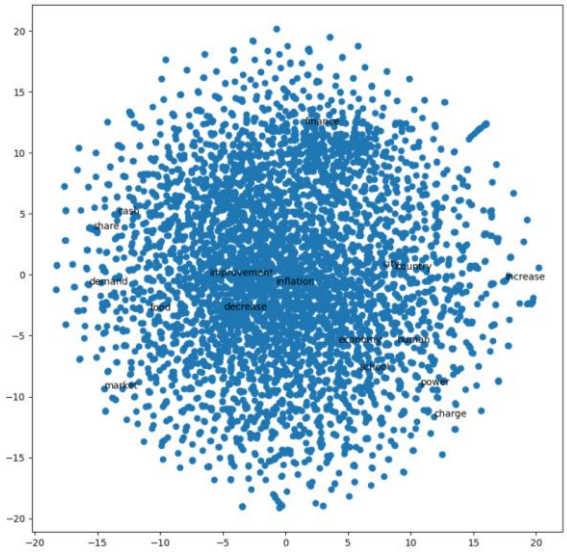

Figure 2 Word embedding visualization using model 2 (CBOW)

In Model 1's market, 'cash,' 'share,' and 'charge' are closely related to each other and have proximity to 'demand,' 'increase,' 'decrease,' and 'improvement.' Likewise, in Model 2 or the CBOW model, 'cash' and 'share,' as well as 'city' and 'country,' form good pairs. It's evident that the word embeddings in these two models are entirely different.

The results may not align with our expectations, but it appears that the root of this problem could be related to reducing the dimension. Alternatively, we can consider increasing the number of epochs to enhance the models.

Correlation Scores (comparing three models):

For model 1:

```
(PearsonRResult(statistic=0.7459154757040682,
pvalue=0.14774546504117553),
SignificanceResult(statistic=0.7999999999999999,
pvalue=0.10408803866182788), 50.0)
```

The correlation between the word pairs and the word embeddings from Model 1 is moderately strong (0.746). The p-value of 0.148 suggests that there is a 14.8% chance that this correlation occurred by random chance, which is slightly higher than a typical significance level of 5%. Therefore, the correlation is not statistically significant at the 5% level but is moderately strong.

For second model:

```
(PearsonRResult(statistic=0.3602401128200361,
pvalue=0.5514512506013286), SignificanceResult(statistic=0.3,
pvalue=0.6238376647810728), 50.0)
```

The correlation between the word pairs and the word embeddings from Model 2 is relatively weak (0.360). The high p-value of 0.551 indicates that this correlation is likely to have occurred by random chance, and it is not statistically significant at the 5% level.

For Google_news model:

```
(PearsonRResult(statistic=0.6706414448335718,
pvalue=0.0687093518597792),
SignificanceResult(statistic=0.6904761904761906,
pvalue=0.057990318164572716), 20.0)
```

The correlation between the word pairs and the word embeddings from the Google News Model is moderately strong (0.671). The p-value of 0.069 suggests that there is a 6.9% chance that this correlation occurred by random chance, which is slightly higher than a typical significance level of 5%. However, it is statistically significant at the 20% level, indicating a reasonable level of confidence in the correlation.

In summary, Model 1 shows the highest correlation, though it's not statistically significant at the 5% level. Model 2 demonstrates a weak and statistically insignificant correlation. The Google News Model presents a moderately strong correlation, which is statistically significant at the 20% level. This suggests that the Google News Model performs relatively better on our dataset compared to the other two models.

Comparing three models in finding most similar words:

Model	Dollar	market	bank	trade	Release
M1(sg=1)	[('mixed', 0.561781), ('finishing', 0.471044), ('rebounded', 0.47009), ('intraday', 0.46121), ('pound', 0.43685)]	[('stock', 0.64442), ('markets', 0.49350), ('junk-bond', 0.48690), ('volatility', 0.48257), ('equities', 0.45603)]	[('banks', 0.48037), ('central', 0.46699), ('holding', 0.46620), ('Chemical', 0.43901), ('banking', 0.43645)]	[('deficit', 0.5708), ('Community', 0.5125), ('figures', 0.4394), ('U.S.', 0.4272), ('Operating', 0.4187)	[('plane', 0.4479), ('news', 0.4212), ('selected', 0.4150), ('Sunday', 0.4144), ('tomorrow', 0.4144)]

M2(sg=0)	[('currency', 0.36483), ('Stick', 0.35467), ('stock', 0.35372), ('season', 0.33402), ('Merkur', 0.33194)]	[('markets', 0.55381), ('prices', 0.49265), ('price', 0.38374), ('uncertainty', 0.35539), ('volatility', 0.35528)]	[('company', 0.41497), ('Odeon', 0.35232), ('equity', 0.35005), ('long-term', 0.34885), ('HealthVest', 0.34318)]	[('Societe', 0.3922), ('beta', 0.3652), ('confirm', 0.3577), ('Community', 0.3377), ('Elsewhere', 0.3343)]	[('Gorbachev', 0.3924), ('flag', 0.3588), ('remarks', 0.3582), ('poll', 0.3421), ('Lorenzo', 0.3367)]
Google_news	[('greenback', 0.74667), ('currency', 0.68092), ('Dollar', 0.66069), ('Japanese_Yen', 0.65041), ('loonie', 0.62662)]	[('markets', 0.76660), ('marketplace', 0.69553), ('themarket', 0.59914), ('maket', 0.59208), ('mkt', 0.58593)]	[('banks', 0.74407), ('banking', 0.69016), ('Bank', 0.66986), ('lender', 0.63422), ('banker', 0.60929)]	[('trading', 0.5736), ('trades', 0.5639), ('Trade', 0.5276), ('traded', 0.5155), ('TRADE', 0.4945)]	[('releasing', 0.710), ('releases', 0.7096), ('released', 0.6614), ('relase', 0.6244), ('Releasing', 0.573)]

Model M1 (Skip-Gram) shows mixed results, with some word pairs having relevant similarities and others being less related. It may benefit from further training or adjustments.

Model M2 (CBOW) generally performs less well compared to the other models, with weaker or unrelated word pairs.

The pre-trained Google News model consistently provides strong and relevant word pair similarities, especially for financial terms like 'Dollar,' 'market,' 'bank,' 'trade,' and 'Release.' This suggests that the Google News model has captured a rich semantic understanding of these financial terms.