Advanced NLP Assignment 1 Report

Tanish Lad (2018114005)

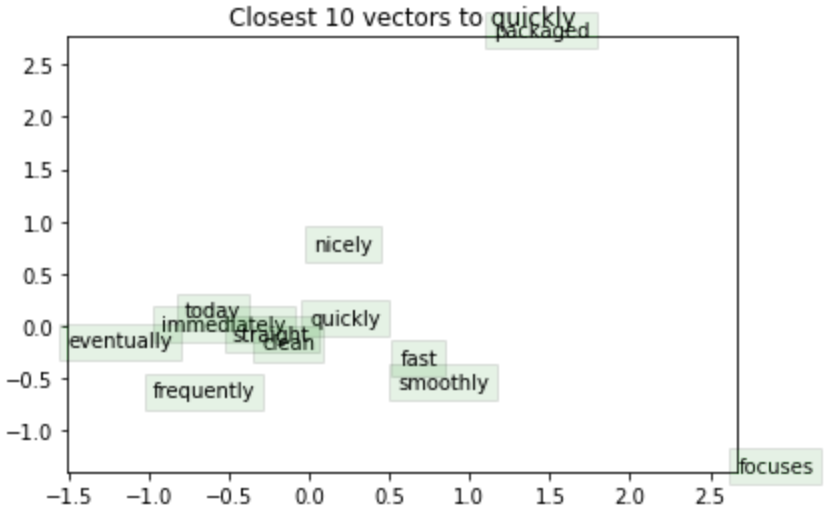
CBOW

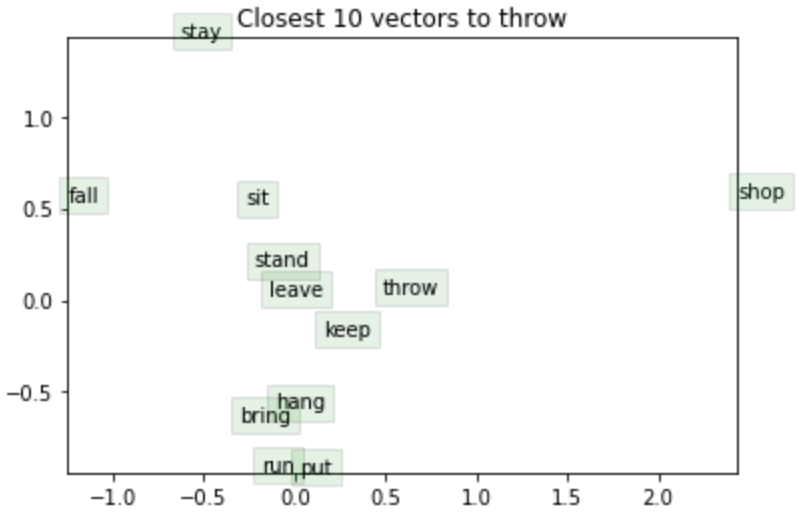
Parameters and Hyperparameters:

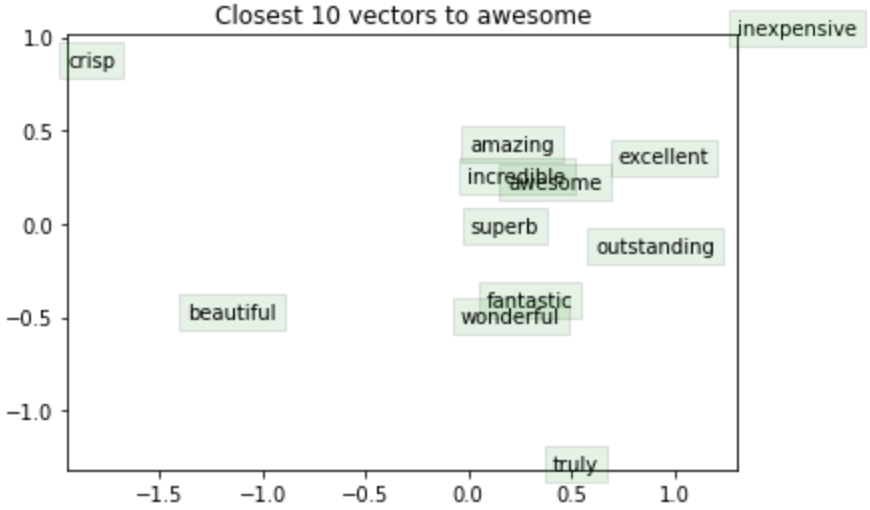
- Embedding Size - 100 Dimensions
- Window Size - 3
- Learning Rate - 0.03 (with decay of ⅔)
- Dropout - 0.6
- Dataset - 50k reviews
- 2 epochs
- No Negative Sampling
- Full Softmax

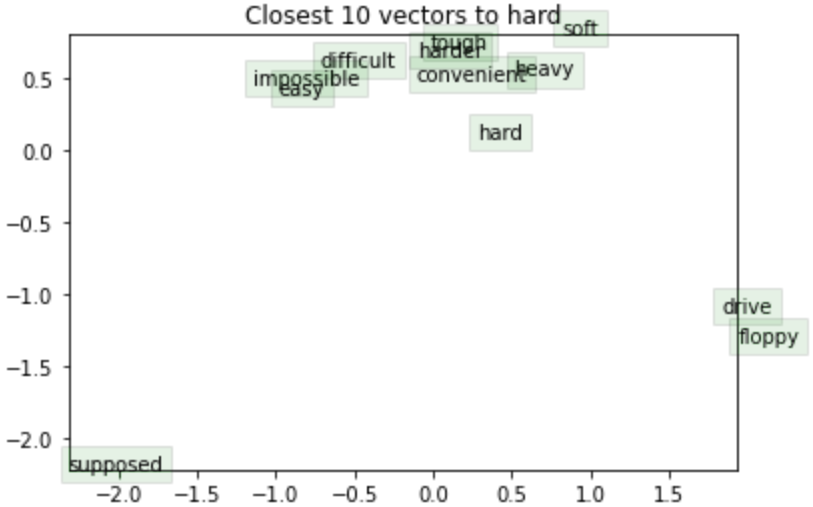
Display the top-10 word vectors for 5 different words (a combination of nouns, verbs, adjectives etc) using the above pre-trained model.

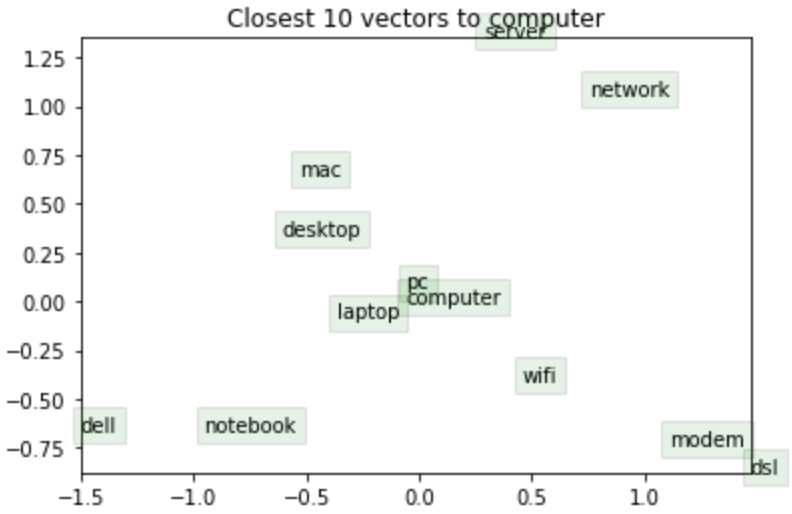
Words List - ["quickly", "throw", "awesome", "hard", "computer"]




Closest 10 vectors to throw
stay
fall
sit
shop
stand
leave
throw
keep
hang
bring
run
put


Closest 10 vectors to awesome
inexpensive
crisp
amazing
excellent
incredible
awesome
superb
outstanding
beautiful
fantastic
wonderful
truly


Closest 10 vectors to hard
soft
tough
difficult
harder
impossible
convenient
heavy
easy
hard
drive
floppy
supposed



What are the top 10 closest words for the word ‘camera’ in the embeddings generated by your program. Compare them against the pre-trained word2vec embeddings.

My Model (excluding the word “camera” itself) (in descending order)

['dslr', 'canon', 'slr', 'film', 'sensor', 'nikon', 'body', 'olympus', 'lense', 'rebel']

Gensim (excluding the word “camera” itself) (in descending order)

[‘cameras', 'cam', 'rebel', 'camcorder', 'digicam', 'lens', ‘nikon’, 'lense', ‘canon', 'slr']

Comparison between Embeddings from my model and Gensim generated:

- Many words are exactly the same (in a slightly different order)
- There is a difference because Gensim’s hyperparameter tuning and number of epochs may be better than mine. Gensim was also very highly optimized.
- Words generated by my model that are not in Gensim also do in a way do make sense and they cannot be classified as wrong.

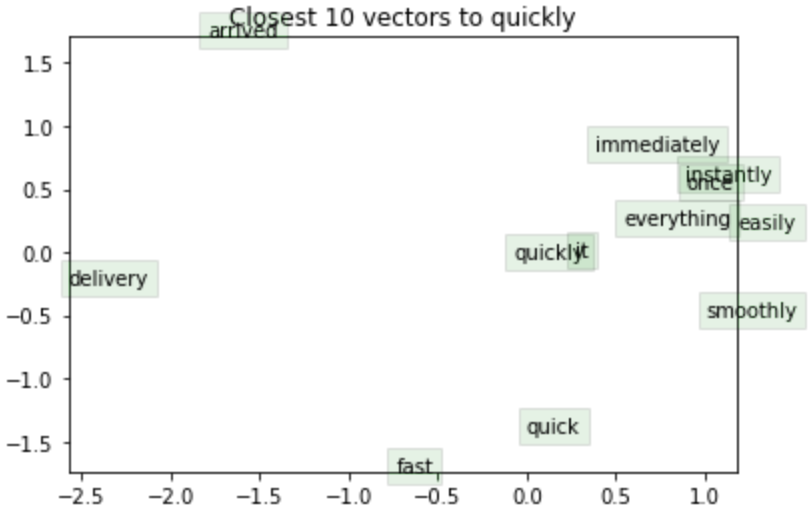
Skip-Gram

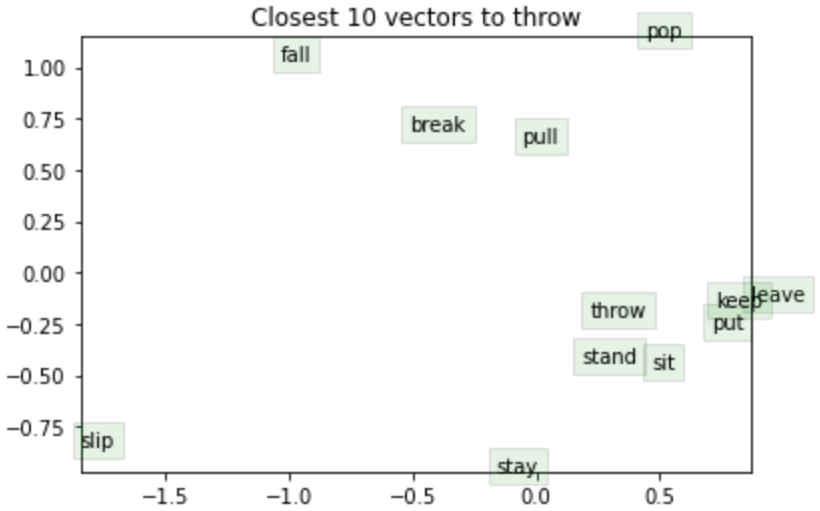
Parameters and Hyperparameters:

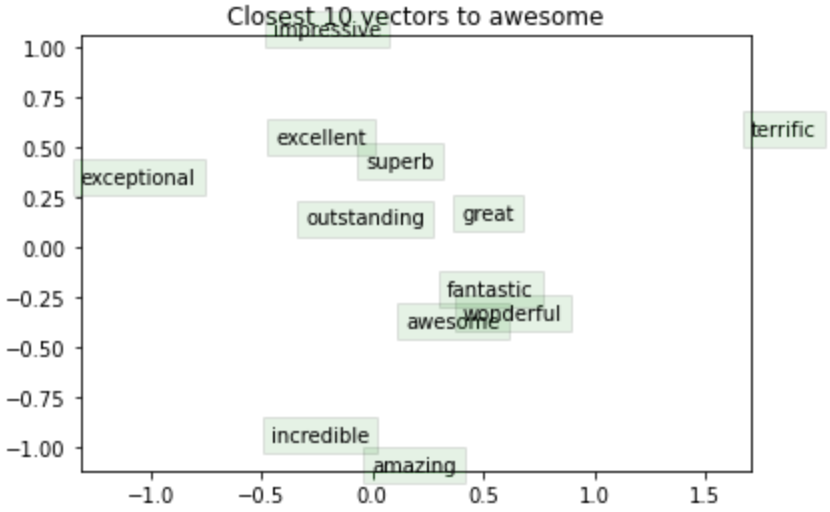
- Embedding Size - 100 Dimensions
- Window Size - 3
- Learning Rate - 0.03 (with decay of ⅔)
- No Dropout
- Dataset - 50k reviews
- 2 epochs
- Negative Sampling (20:4 ratio)
- Sigmoid

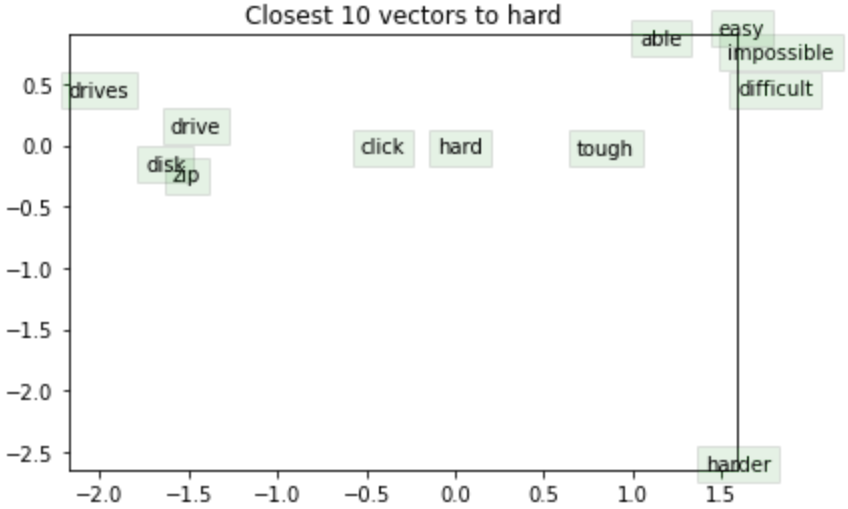
Display the top-10 word vectors for 5 different words (a combination of nouns, verbs, adjectives etc) using the above pre-trained model.

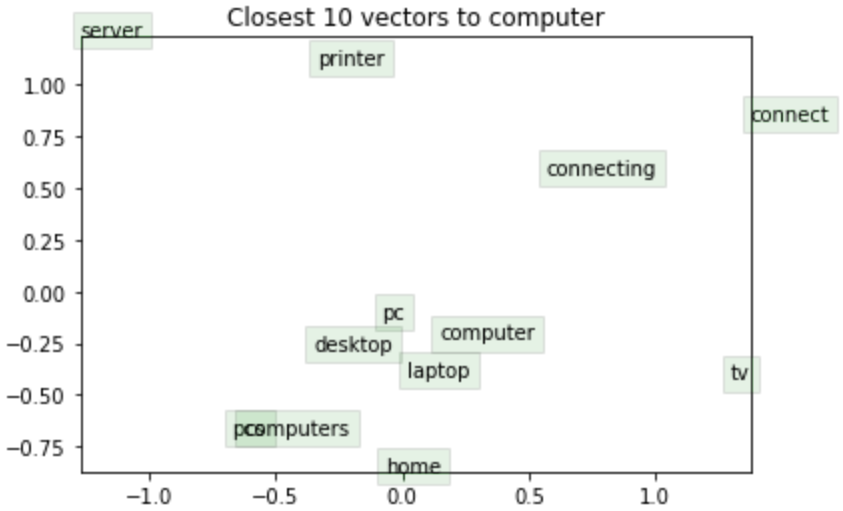
Words List - ["quickly", "throw", "awesome", "hard", "computer"]




Closest 10 vectors to throw
pop
fall
break
pull
leave
keep
throw
put
stand
sit
slip
stay
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.5
−1.0
−0.5
0.0
0.5


Closest 10 vectors to awesome
impressive
terrific
excellent
superb
exceptional
outstanding
great
fantastic
wonderful
awesome
incredible
amazing
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.0
−0.5
0.0
0.5
1.0
1.5


Closest 10 vectors to hard
able
easy
impossible
difficult
drives
drive
disk
zip
click
hard
tough
harder
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5



What are the top 10 closest words for the word 'camera' in the embeddings generated by your program. Compare them against the pre-trained word2vec embeddings.

My Model (excluding the word "camera" itself) (in descending order)

['rebel', 'dslr', 'canon', 'lense', 'film', 'lens', 'camcorder', 'slr', 'sensor', 'mm']

Gensim (excluding the word "camera" itself) (in descending order)

['cameras', 'camcorder, 'rebel, 'cam, 'slr, 'cannon, 'lens, 'monopod, 'filter', 'zoom']

Comparison between Embeddings from my model and Gensim generated:

- Many words are exactly the same (in a slightly different order)
- There is a difference because Gensim's hyperparameter tuning and number of epochs may be better than mine. Gensim was also very highly optimized.
- Words generated by my model that are not in Gensim also do in a way do make sense and they cannot be classified as wrong.

- Both models produced very good results. Gensim's outputs were slightly better.