

Word2Vec

Word2Vec is a 2 layer Neural Network that processes text by vectorizing words based on their contexts so that they can be numerically computed further in complex neural networks. I have implemented the SkipGram architecture whose aim is to predict the context of the word given a word. SkipGram architecture has an advantage of storing multiple contexts of the same word. SubSampling is done by removing words based on their occurrence frequencies. I updated the weights for the correct example, but only a small number of incorrect, or noise, examples (Negative Sampling). Finally after preprocessing, I trained it for 8539 unique words for 12 epochs with a batch size of 250 using an Adam optimiser and printed the loss after every 100 sets of batches. The output is as follows:-

Training

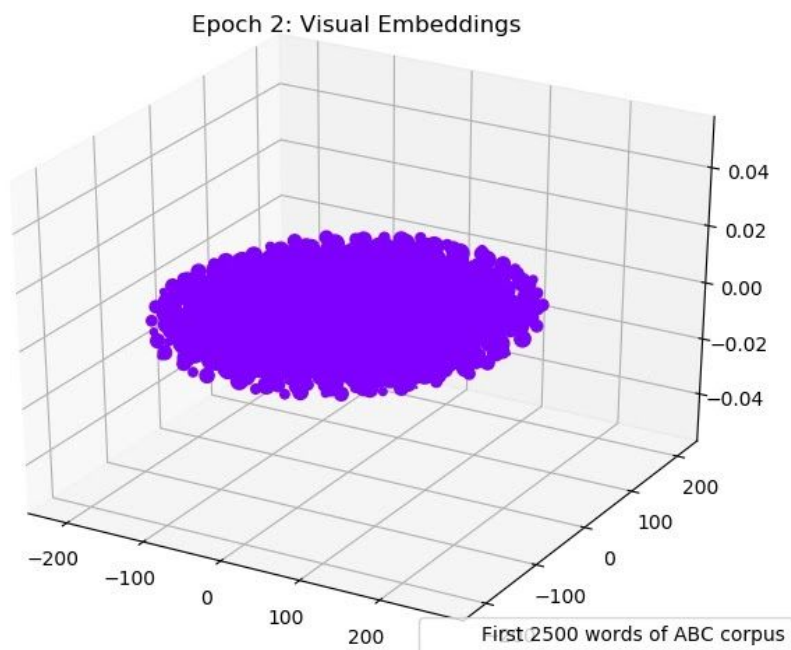
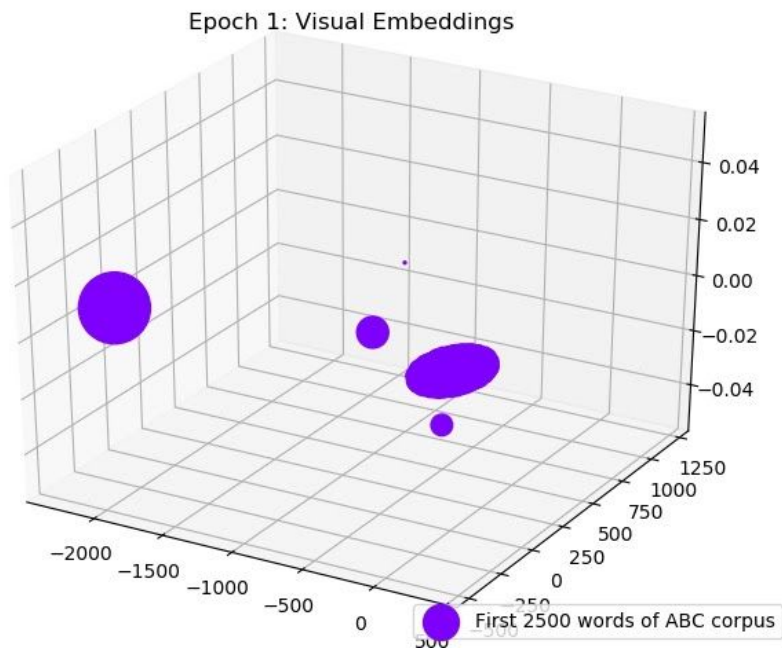
```
Total words: 125891
Unique words: 8539
Training starts...
Epoch: 1/12 | Steps: 100/501 | Loss: 13.989673614501953
Epoch: 1/12 | Steps: 200/501 | Loss: 12.431231498718262
Epoch: 1/12 | Steps: 300/501 | Loss: 11.979936599731445
Epoch: 1/12 | Steps: 400/501 | Loss: 10.949665069580078
Epoch: 1/12 | Steps: 500/501 | Loss: 10.045424461364746
Creating Visual Embeddings...
Epoch: 2/12 | Steps: 100/501 | Loss: 8.599141120910645
Epoch: 2/12 | Steps: 200/501 | Loss: 7.6827392578125
Epoch: 2/12 | Steps: 300/501 | Loss: 7.545987129211426
Epoch: 2/12 | Steps: 400/501 | Loss: 6.76023530960083
Epoch: 2/12 | Steps: 500/501 | Loss: 6.1564717292785645
Creating Visual Embeddings...
Epoch: 3/12 | Steps: 100/501 | Loss: 5.837783336639404
Epoch: 3/12 | Steps: 200/501 | Loss: 5.414849758148193
Epoch: 3/12 | Steps: 300/501 | Loss: 5.267870903015137
Epoch: 3/12 | Steps: 400/501 | Loss: 4.920295715332031
Epoch: 3/12 | Steps: 500/501 | Loss: 4.532724380493164
Creating Visual Embeddings...
```

```
Epoch: 4/12 | Steps: 100/501 | Loss: 4.554084777832031
Epoch: 4/12 | Steps: 200/501 | Loss: 4.076140403747559
Epoch: 4/12 | Steps: 300/501 | Loss: 3.94814133644104
Epoch: 4/12 | Steps: 400/501 | Loss: 3.8579635620117188
Epoch: 4/12 | Steps: 500/501 | Loss: 3.6268656253814697
Creating Visual Embeddings...
Epoch: 5/12 | Steps: 100/501 | Loss: 3.6735002994537354
Epoch: 5/12 | Steps: 200/501 | Loss: 3.3728280067443848
Epoch: 5/12 | Steps: 300/501 | Loss: 3.373048782348633
Epoch: 5/12 | Steps: 400/501 | Loss: 3.1976301670074463
Epoch: 5/12 | Steps: 500/501 | Loss: 3.040092706680298
Creating Visual Embeddings...
Epoch: 6/12 | Steps: 100/501 | Loss: 3.1056792736053467
Epoch: 6/12 | Steps: 200/501 | Loss: 2.7930028438568115
Epoch: 6/12 | Steps: 300/501 | Loss: 2.904707908630371
Epoch: 6/12 | Steps: 400/501 | Loss: 2.6933116912841797
Epoch: 6/12 | Steps: 500/501 | Loss: 2.609455108642578
Creating Visual Embeddings...
Epoch: 7/12 | Steps: 100/501 | Loss: 2.7511048316955566
Epoch: 7/12 | Steps: 200/501 | Loss: 2.559837579727173
Epoch: 7/12 | Steps: 300/501 | Loss: 2.497427225112915
Epoch: 7/12 | Steps: 400/501 | Loss: 2.4121668338775635
Epoch: 7/12 | Steps: 500/501 | Loss: 2.247138500213623
Creating Visual Embeddings...
Epoch: 8/12 | Steps: 100/501 | Loss: 2.446154832839966
Epoch: 8/12 | Steps: 200/501 | Loss: 2.257319450378418
Epoch: 8/12 | Steps: 300/501 | Loss: 2.235351324081421
Epoch: 8/12 | Steps: 400/501 | Loss: 2.257925510406494
Epoch: 8/12 | Steps: 500/501 | Loss: 1.9993691444396973
Creating Visual Embeddings...
```

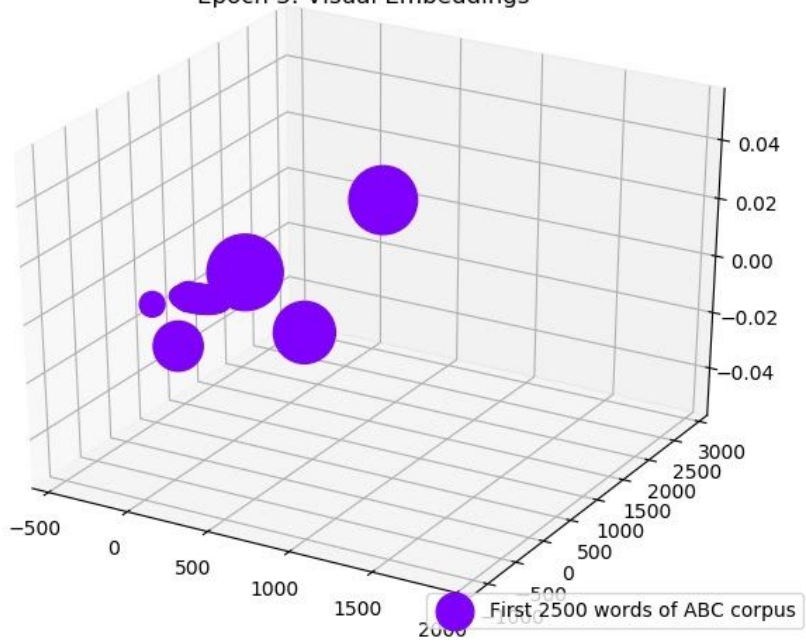
```
Epoch: 9/12 | Steps: 100/501 | Loss: 2.2680625915527344
Epoch: 9/12 | Steps: 200/501 | Loss: 2.019604444503784
Epoch: 9/12 | Steps: 300/501 | Loss: 2.025233268737793
Epoch: 9/12 | Steps: 400/501 | Loss: 2.012941837310791
Epoch: 9/12 | Steps: 500/501 | Loss: 1.9514490365982056
Creating Visual Embeddings...
Epoch: 10/12 | Steps: 100/501 | Loss: 2.057184934616089
Epoch: 10/12 | Steps: 200/501 | Loss: 1.8599034547805786
Epoch: 10/12 | Steps: 300/501 | Loss: 1.8564449548721313
Epoch: 10/12 | Steps: 400/501 | Loss: 1.8445159196853638
Epoch: 10/12 | Steps: 500/501 | Loss: 1.7836815118789673
Creating Visual Embeddings...
Epoch: 11/12 | Steps: 100/501 | Loss: 2.0289251804351807
Epoch: 11/12 | Steps: 200/501 | Loss: 1.726201057434082
Epoch: 11/12 | Steps: 300/501 | Loss: 1.776809573173523
Epoch: 11/12 | Steps: 400/501 | Loss: 1.7351410388946533
Epoch: 11/12 | Steps: 500/501 | Loss: 1.6023329496383667
Creating Visual Embeddings...
Epoch: 12/12 | Steps: 100/501 | Loss: 1.8671834468841553
Epoch: 12/12 | Steps: 200/501 | Loss: 1.718000888824463
Epoch: 12/12 | Steps: 300/501 | Loss: 1.6809892654418945
Epoch: 12/12 | Steps: 400/501 | Loss: 1.571476936340332
Epoch: 12/12 | Steps: 500/501 | Loss: 1.5326920747756958
Creating Visual Embeddings...
```

Embeddings Visualisation

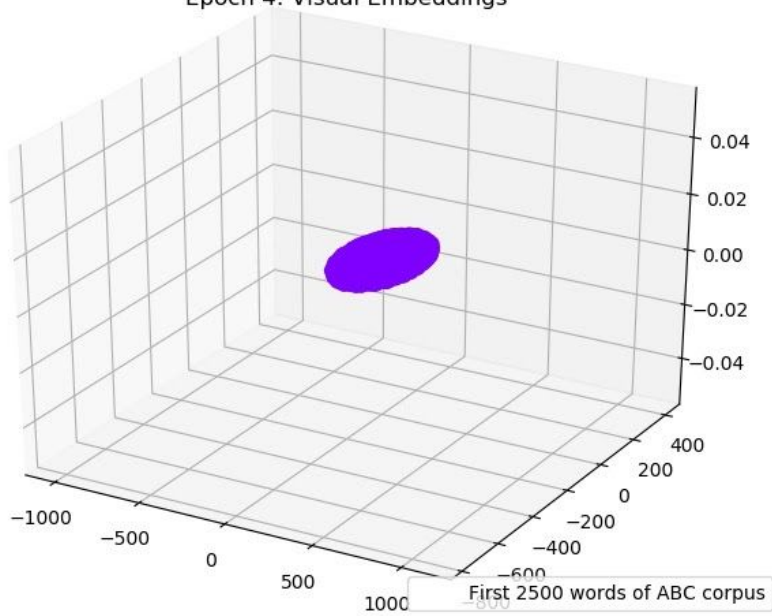
Due to computational time constraints, I visualised the embeddings of the first 2500 words from the training corpus using 3D TSNE after every epoch (12 total).



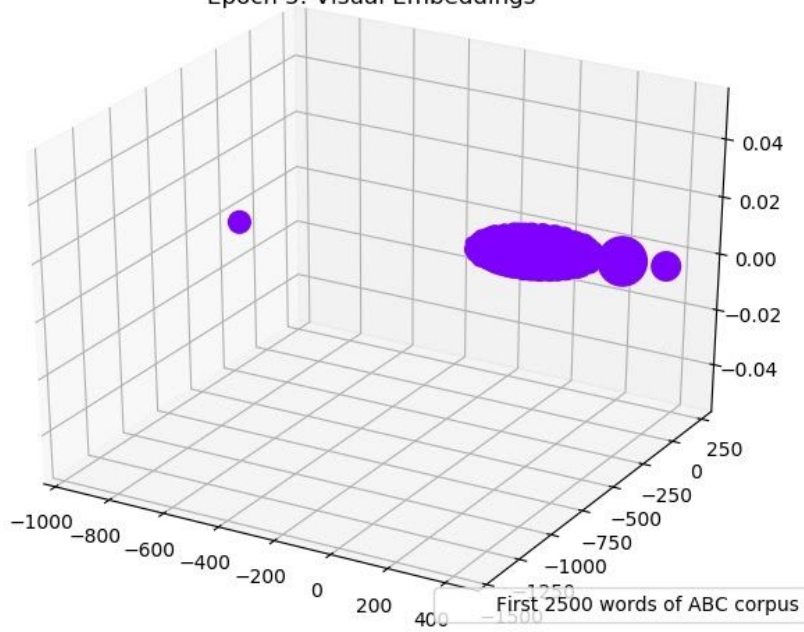
Epoch 3: Visual Embeddings



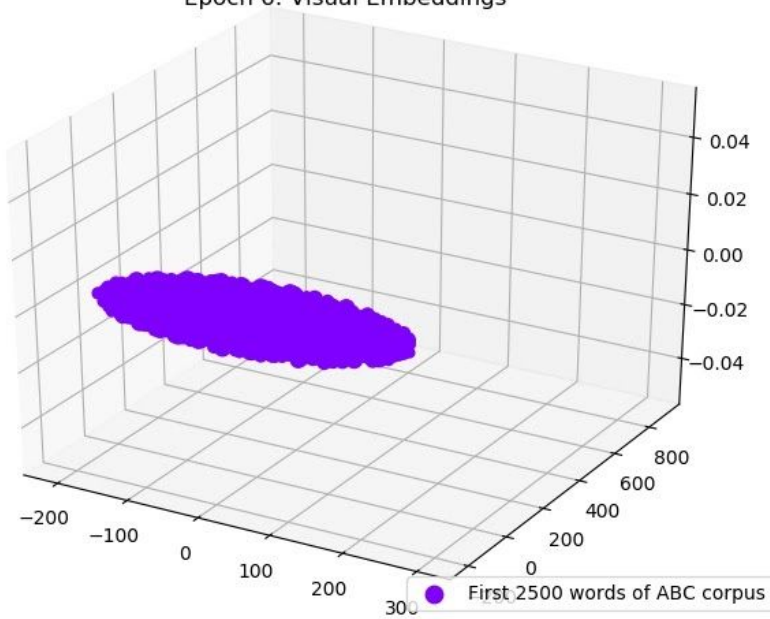
Epoch 4: Visual Embeddings



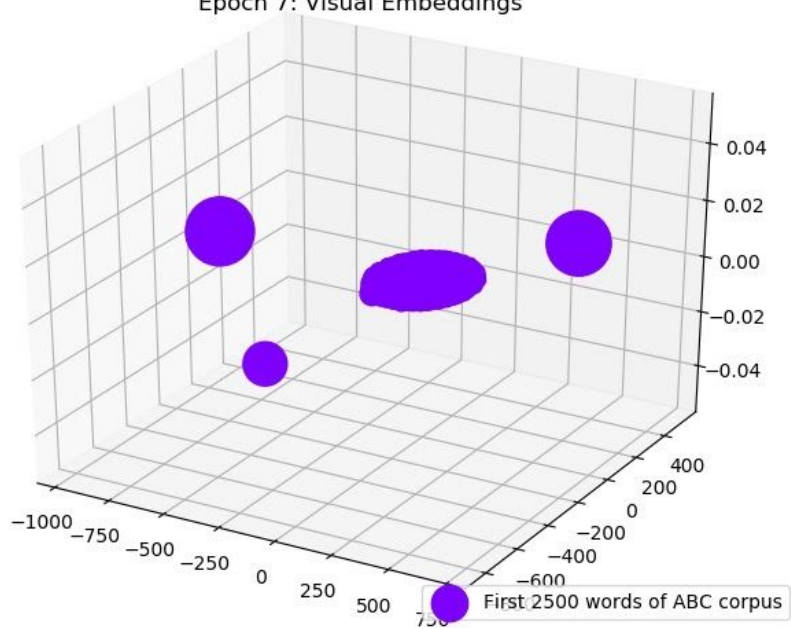
Epoch 5: Visual Embeddings



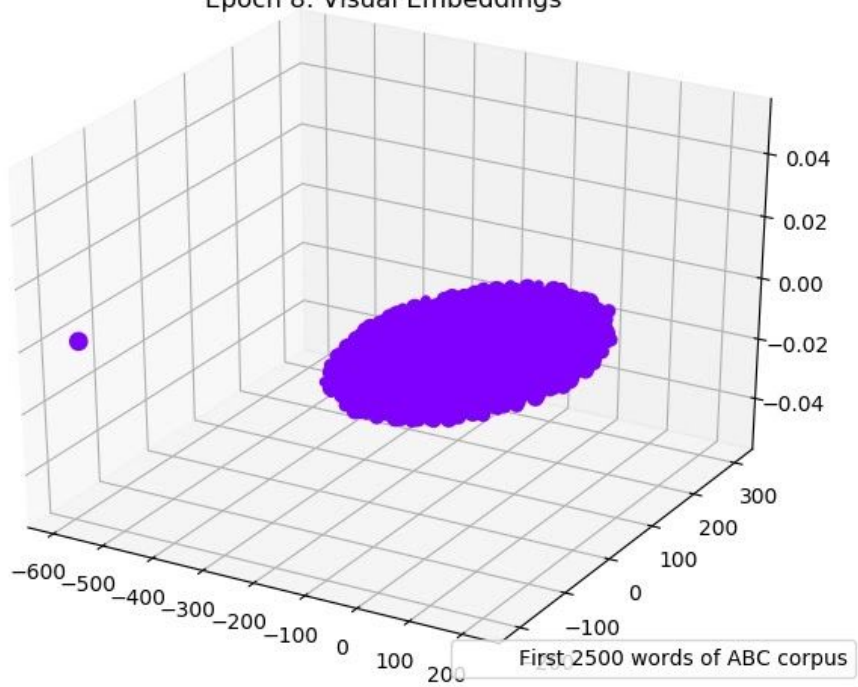
Epoch 6: Visual Embeddings



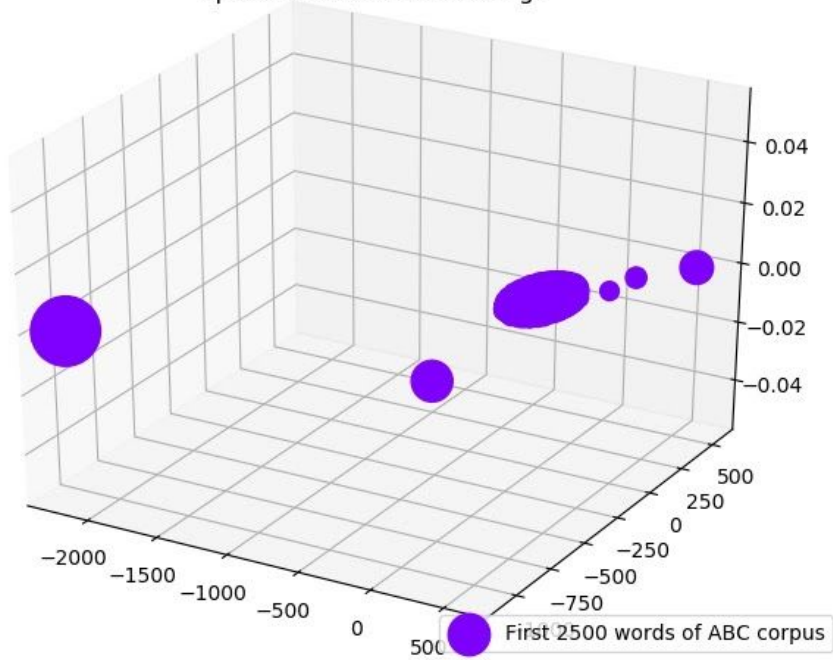
Epoch 7: Visual Embeddings



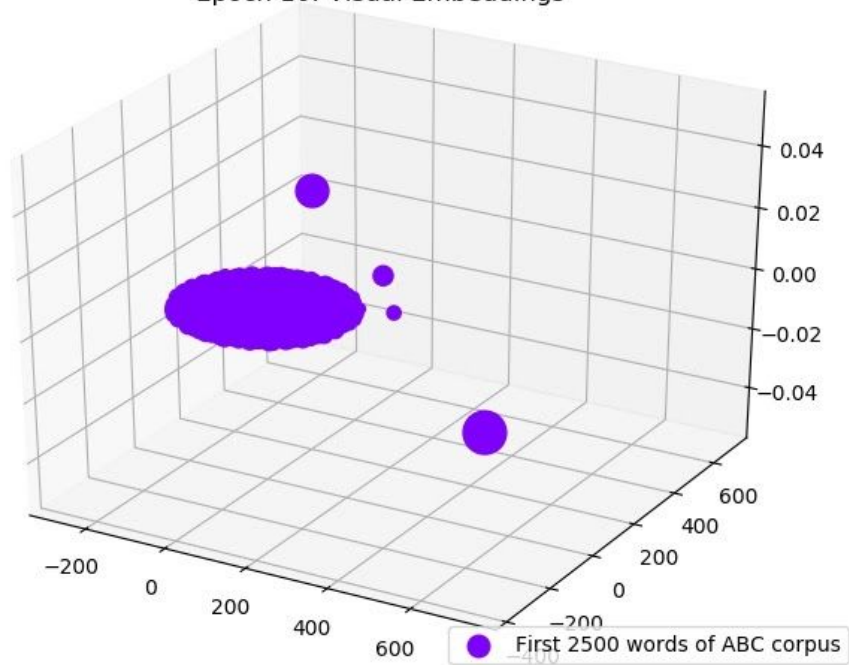
Epoch 8: Visual Embeddings



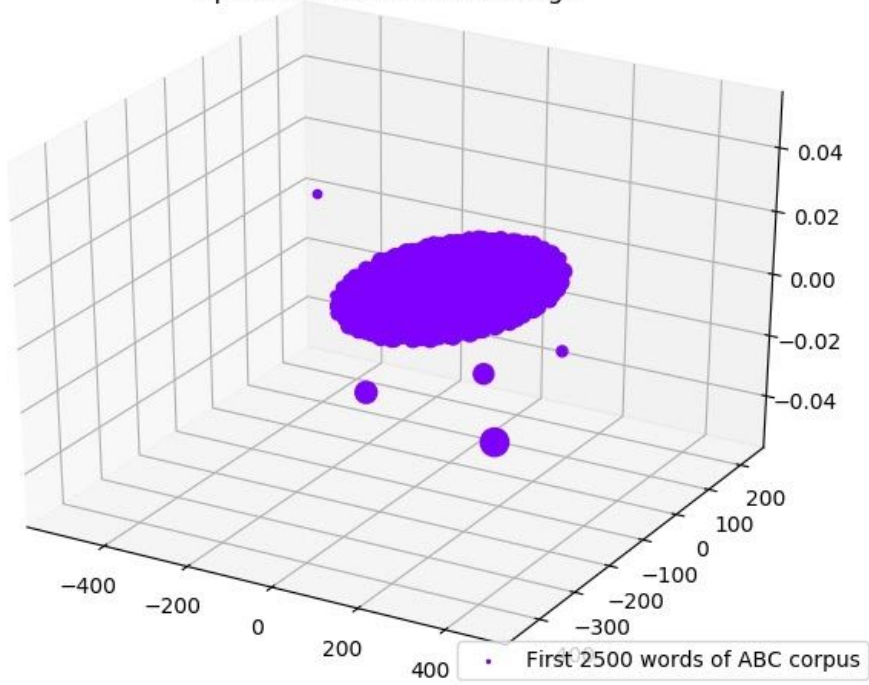
Epoch 9: Visual Embeddings



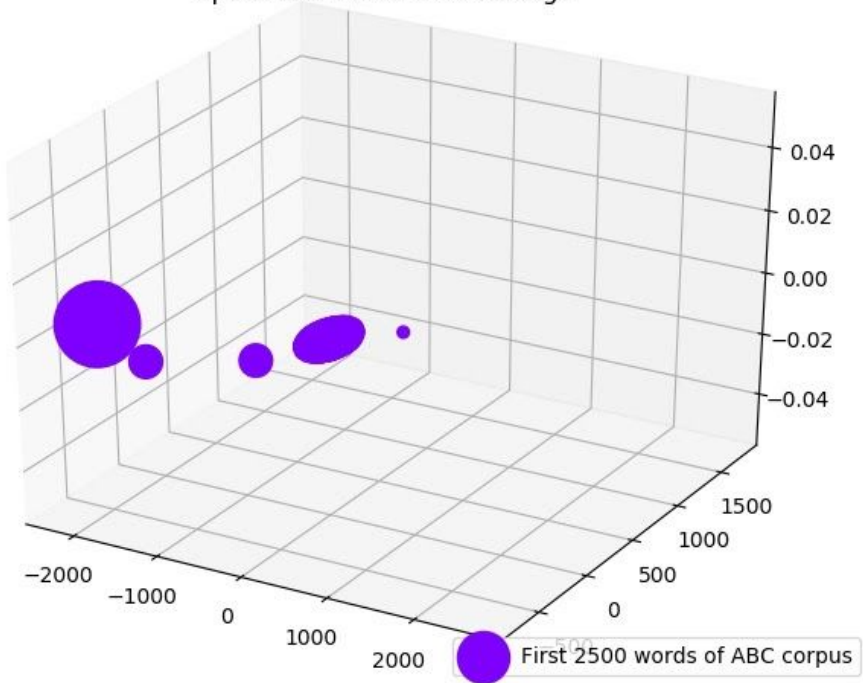
Epoch 10: Visual Embeddings



Epoch 11: Visual Embeddings

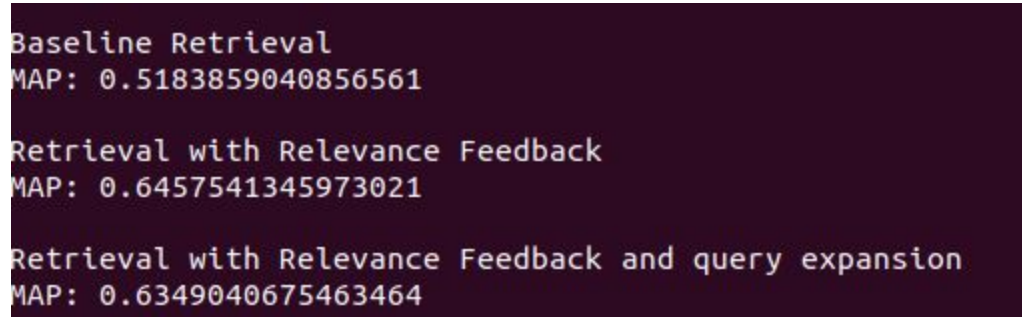


Epoch 12: Visual Embeddings



Observation: Over the course of 12 epochs, the loss roughly saturated and I observed the embeddings to be loosely clustered as it trains better with every epoch. The fact that as the epochs proceeded the clustering segregated to well defined clusters supports the correct contextual training of the word embeddings due to similar words forming some clusters instead of a complete blob after epoch 2.

Document Retrieval

A terminal window with a dark purple background and light green text. It displays three lines of output, each showing a method name followed by its MAP score.

```
Baseline Retrieval
MAP: 0.5183859040856561

Retrieval with Relevance Feedback
MAP: 0.6457541345973021

Retrieval with Relevance Feedback and query expansion
MAP: 0.6349040675463464
```

Observation: According to the results, an expected increase is observed on the relevance feedback retrieval done with 3 iterations whereas it was expected that along with the incorporation of query expansion the MAP will increase but as observed it slightly decreased. It could be my incorrect understanding of the pseudocode. I could not figure a way out to correct the query expansion part of it.