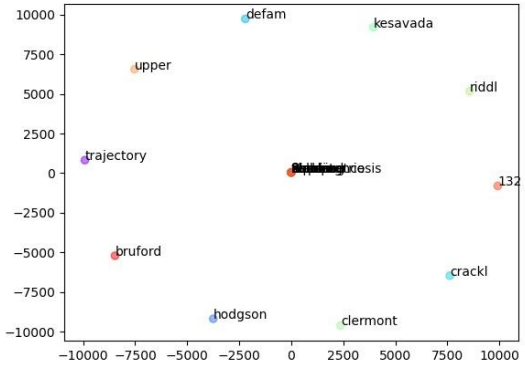
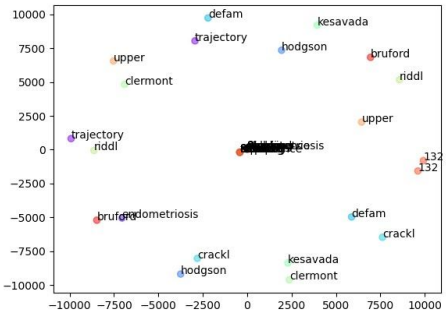
MCA Assignment 4
Abhishek Maiti
2016005

Q1. I have implemented the SkipGram Model using the context size of 2 on either side of the main word. So the context word for “This is going to **be my** last **B.Tech deadline** :(” for the word “last” is highlighted in the sentence.
I have used Stochastic Gradient Descent, removed stop words, and used Porter stemmer to stem the words.
The t-SNE embeddings for epoch are as follows.

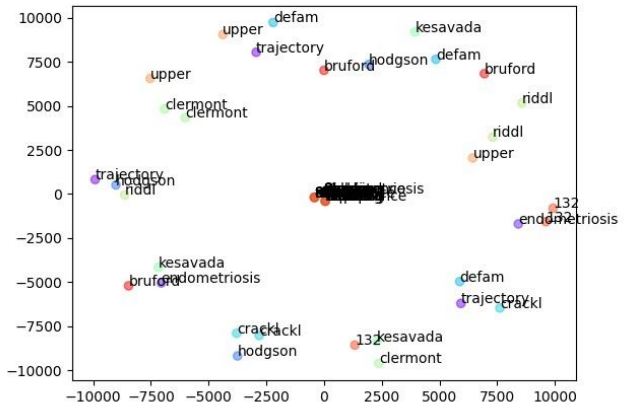
Epoch 25



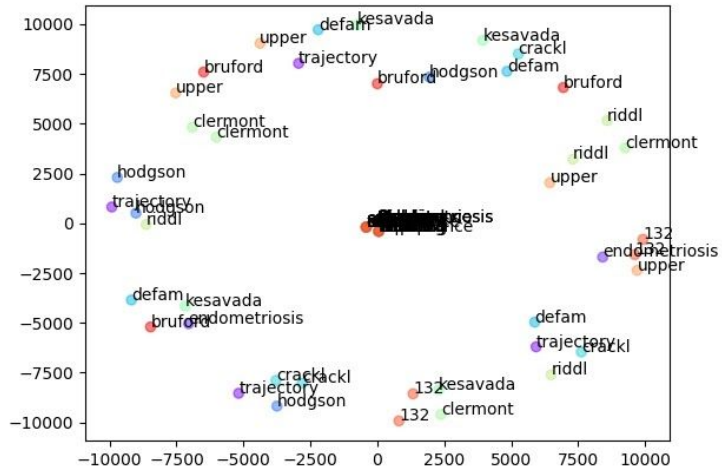
Epoch 50



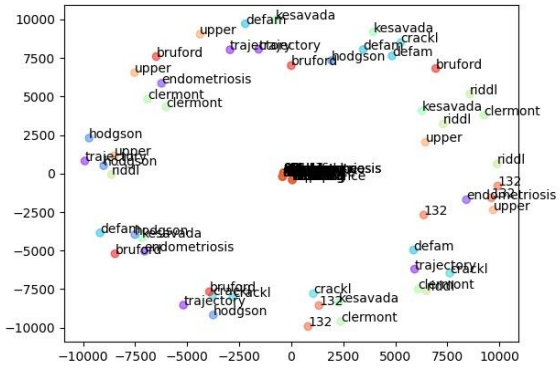
Epoch 75



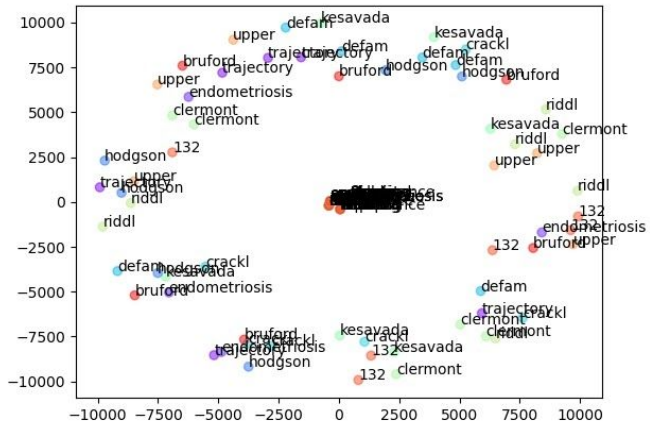
Epoch 100



Epoch 150



Epoch 200



Note: The graph has embeddings from the previous iteration also for ease of comparing. I have run the model for 200 epochs.

Q2.

Baseline Retrieval
MAP: 0.52

Retrieval with Relevance Feedback
MAP: 0.61

Retrieval with Relevance Feedback and query expansion
MAP: 0.62

We can see that relevance feedback improves significantly w.r.t. To the baseline and relevance feedback with query expansion improves slightly w.r.t to the vanilla relevance feedback.

I have tuned the hyperparameters, and the used parameters were giving the best results. For query expansion, I have extracted the term with the highest tf-idf, found 10 most similar words, and set their weights to their present synonym. To find synonyms, I computed, A.A^T where A = vec_docs, the A_{ij} element gives the similarity between the ith and the jth word.

Note: I have converted the given code to python 3 (basically added brackets for the `print` statements.