How to verify if embedding is of good quality for health care related data?

One approach of checking if the obtained embeddings are good of good quality is to do clustering on obtained embedding vectors and then visualise those clusters. Related embeddings will be together in clusters, but this approach will not tell if it could handle health related information properly. It might be that embedding are close because of generic words.

I am going to follow below 2 approach to check embedding quality-

1st: cosine similarity between medical keywords pair:
Because we have two more files ClinNotes.csv and MedicalConcepts.csv. One approach could be to check cosine similarity between pairs of keywords. If the two pairs are equal(synonym) then cosine similarity is 1 and if the pairs are opposite(antonym) then it is -1 and if the cosine similarity is 0 the pairs have no relation. I will calculate the cosine similarity for each pair(we have 558 unique pairs in the MedicalConcepts.csv file), then the best model would give a mean cosine similarity near to 1.

2nd approach: generate embedding for ClinNotes.csv data and draw a scatter plot:
I will generate embedding for ClinNotes.csv data and draw a scatter plot for embedding against the category column.. If the embeddings are good, all the cardiovascular related vectors should be closer in plot, all the Neurology related vectors should form another group and the same goes for the 3rd category. For creating a scatter plot, I am first transforming embeddings in 2 dimensions using T-SNE and then plotting it.

Techniques to find embeddings for health i have used multiple techniques-

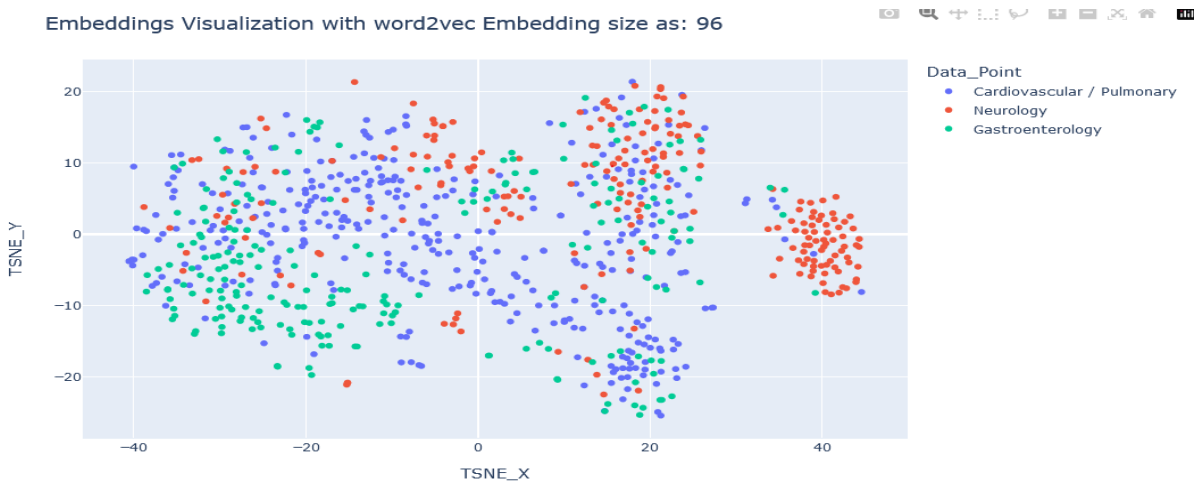
1st: Word2Vec
I have started off with the simplest semantic similarity approach based embedding technique, which is word2vec. I have used spacy for getting embedding using wrod2vec. Sapcy has multiple models using which word2vec embeddings can be obtained and these model are already trained on huge corpus of data. I have used en_core_web_sm and en_core_web_lg models. en_core_web_sm is small model and give 96 length long embedding and en_core_web_lg is a larger model which gives 300 word long embedding.

Though I have used Word2vec for comparison purposes , it has a major problem and that is, it can not identify the context in which the word appears in a sentence. Second problem is that this model is trained on generic corpus and might not give good embedding for health related keywords.
Below is the result of cosine similarity on medical keywords pairs using word2vec small model.

```
Total number of pairs =   558
Total cosine_similarity =   239.9576534051448
mean cosine_similarity =   0.43003153800964355
```

Please note the last value “mean cosine similarity = 0.43”, which is not very good. we want similarity closer to 1.

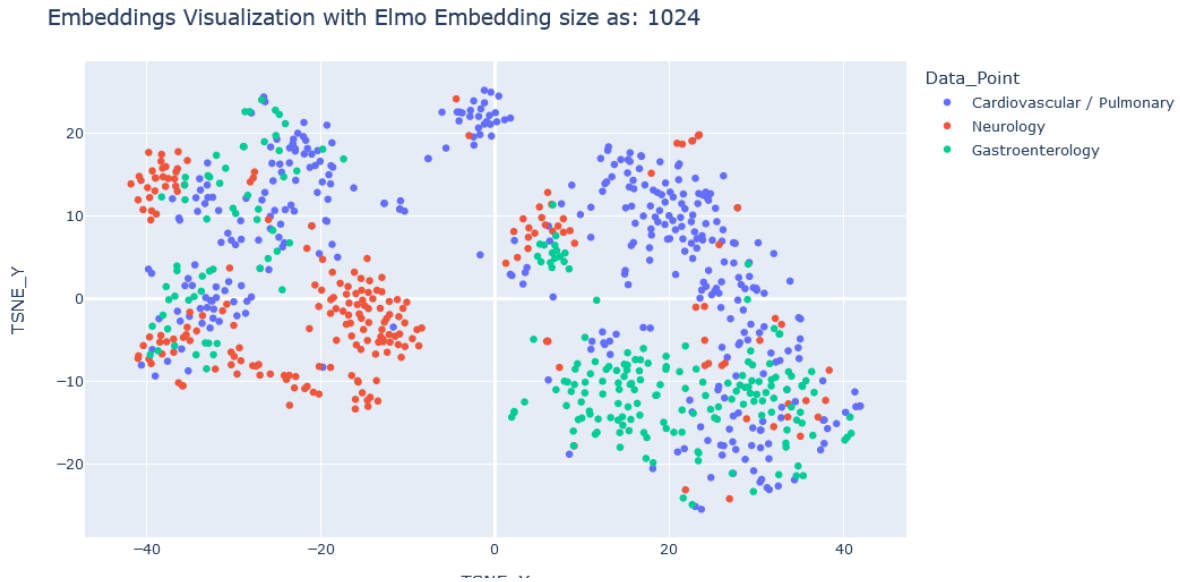
Now below is the result obtained when I fed ClinNotes.csv data in word2vec model and plot it against category column.



As it can be seen clearly that embeddings are not good. No clear group can be seen(though there is a small group of red points at the very right side). Mostly, all data points are mixed up

2nd: ELMO

Second technique was to get embedding from the Elmo model. This model is better than word2vec and can capture the context of a sentence pretty well. But again this model too is not specifically trained on health related data and might not give good embedding for health related keywords as other models(BioBERT, ClinicalBERT) can give. Below is the result when i fed ClinNotes.csv data to ELMO and plot obtained vectors against category



I think the results are far better than the word2vec. We can see clearer groups in data. Let's check cosine similarity on keyword pair

```
Total number of pairs =   558
Total cosine_similarity =   308.23337239027023
mean cosine_similarity =   0.5523895
```

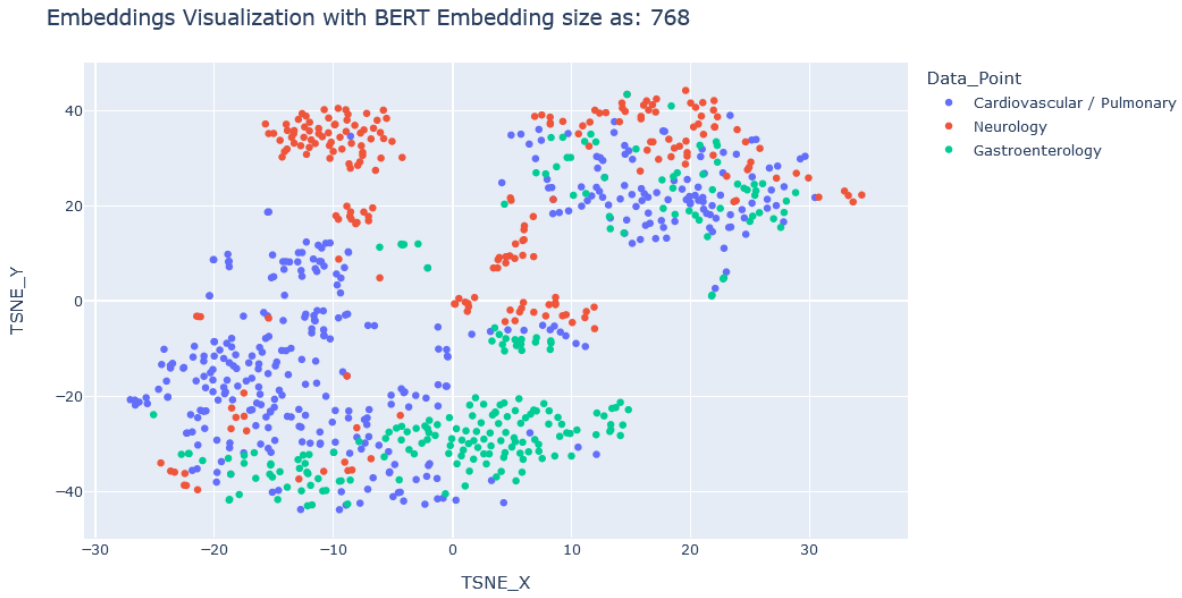
The mean cosine similarity is 0.55, which is not very good. As stated above, ELMO did perform better than word2vec. One of the possibilities, mean cosine similarity is low because ELMO gives 1024 length long embedding and length of embedding decreases similarity score. This score could be slightly higher if the embedding size would have been around 768. Nonetheless, ELMO is again not specifically trained on health data and we have other models which are trained on various types of health related data only.

3rd: BERT

Third model that I tried was BERT which is a transformer based model and can capture context of the sentence pretty well. There are many variations of the BERT model, few of them are specifically trained on health related data. I have first started off with the BERT base uncased model which is trained on generic corpus. Below is the result of cosine similarity on keyword pairs

```
Total number of pairs =   558
Total cosine_similarity =   319.710486933589
mean cosine_similarity =   0.57295823097229
```

BERT base model has performed even better than ELMO on keyword similarity tasks. Now lets plot clinic notes using bert embeddings



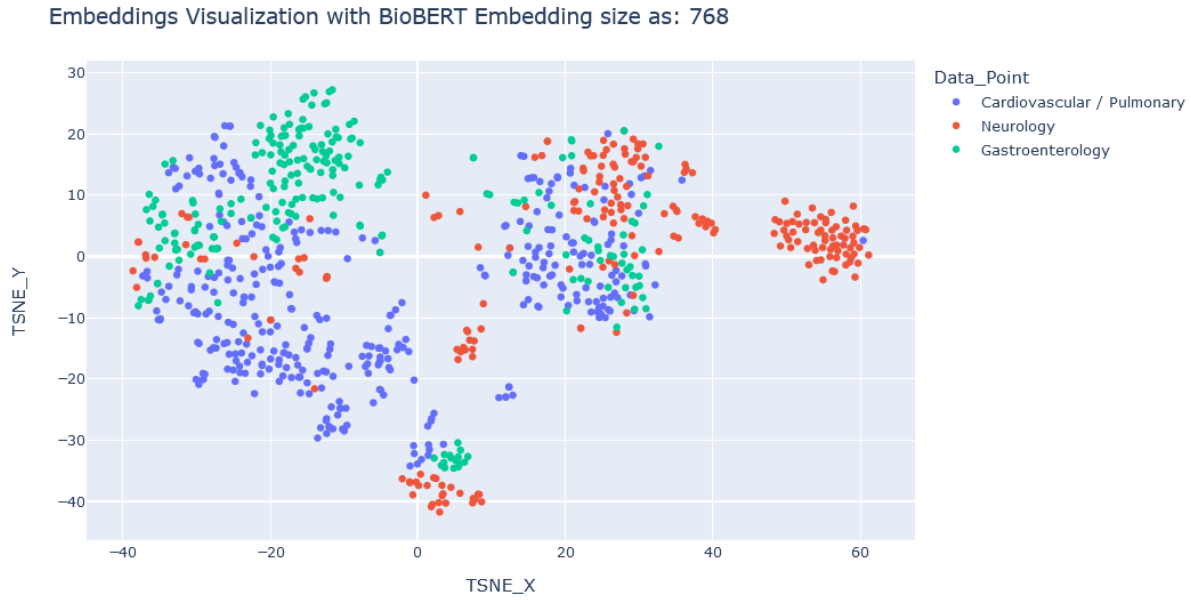
We can see the groups getting formed in above plot, but again they are not very good. Cosine similarity too is not very good.

4th: BioBERT

Now I have used another variant of BERT called BioBERT. This model is trained on a huge corpus of biomedical literature. I believe this model should be able to perform better than any other model that we have used so far. Below is the cosine similarity result from this model-

```
Total number of pairs =   558
Total cosine_similarity =   447.63191509246826
mean cosine_similarity =   0.8022072911262512
```

As you can see, mean cosine similarity is 0.80, which is way higher than any other model we used so far. Let's plot clinic notes data with this model's embedding



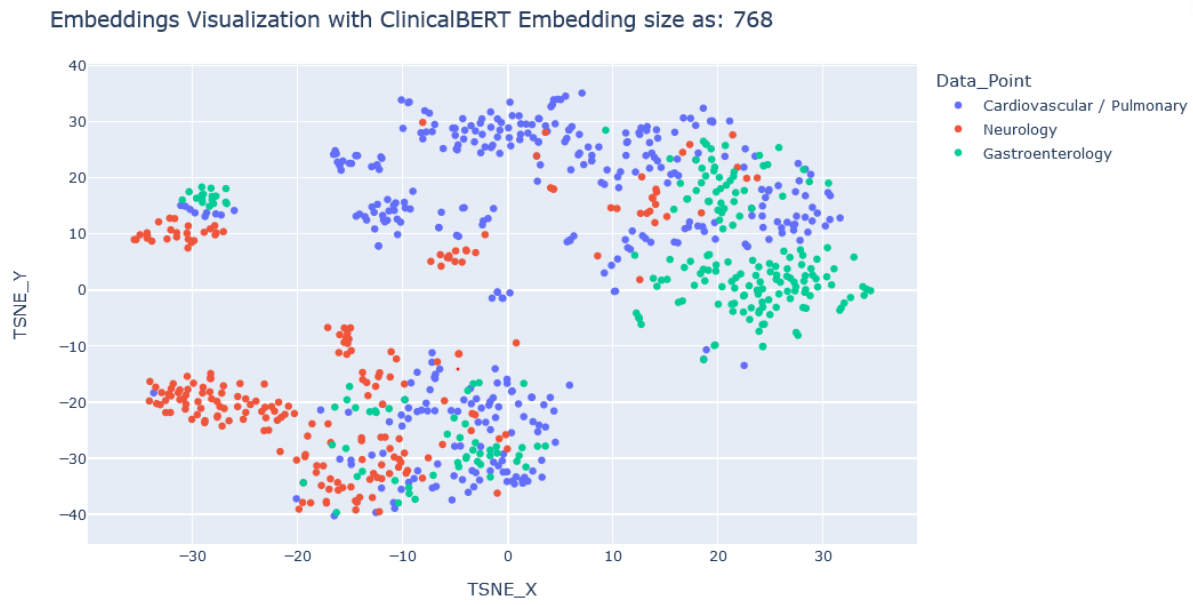
This plot looks better than the BERT base model's plot. Results from this model are good but let's explore other models as well.

5th: ClinicalBERT

Another variant of BERT is ClinicalBERT. This model is trained in a huge corpus of clinic notes data, which can be advantageous as we too are dealing with clinic notes data. Below is the cosine similarity result from this model-

```
Total number of pairs =   558
Total cosine_similarity =   453.973201751709
mean cosine_similarity =   0.8135718107223511
```

ClinicalBERT has given even better results than BioBERT. Lets plot ClinNotes data with model's embedding

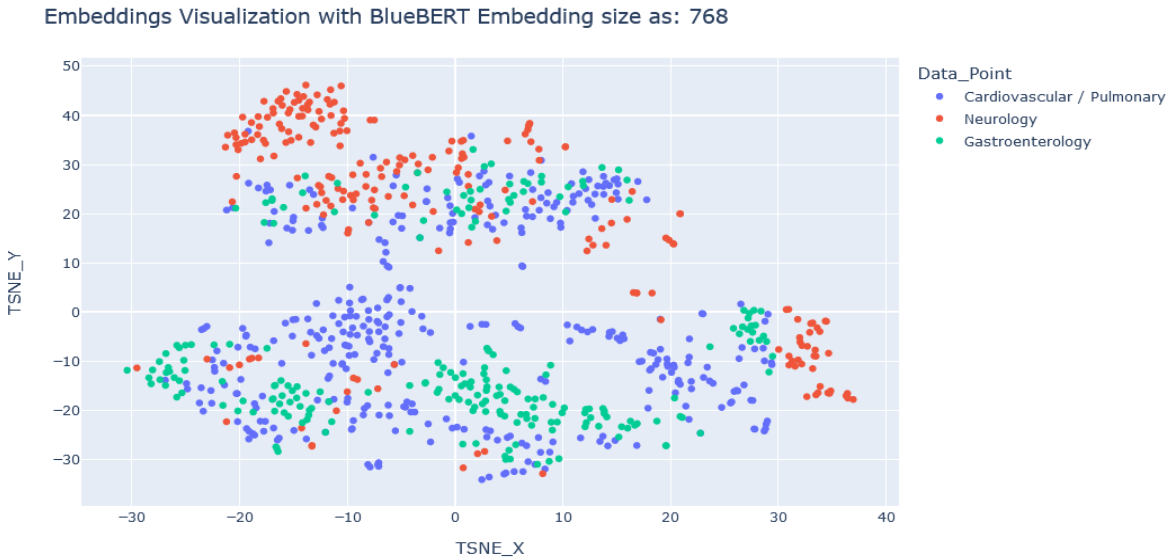


This plot is even cleaner than BioBERT. This is the best model so far, but let's explore more models.

6th BlueBERT:
This is another variation of BERT, which again trained on biomedical and clinical notes, both types of data. Let's check the cosine similarity obtained from this model-

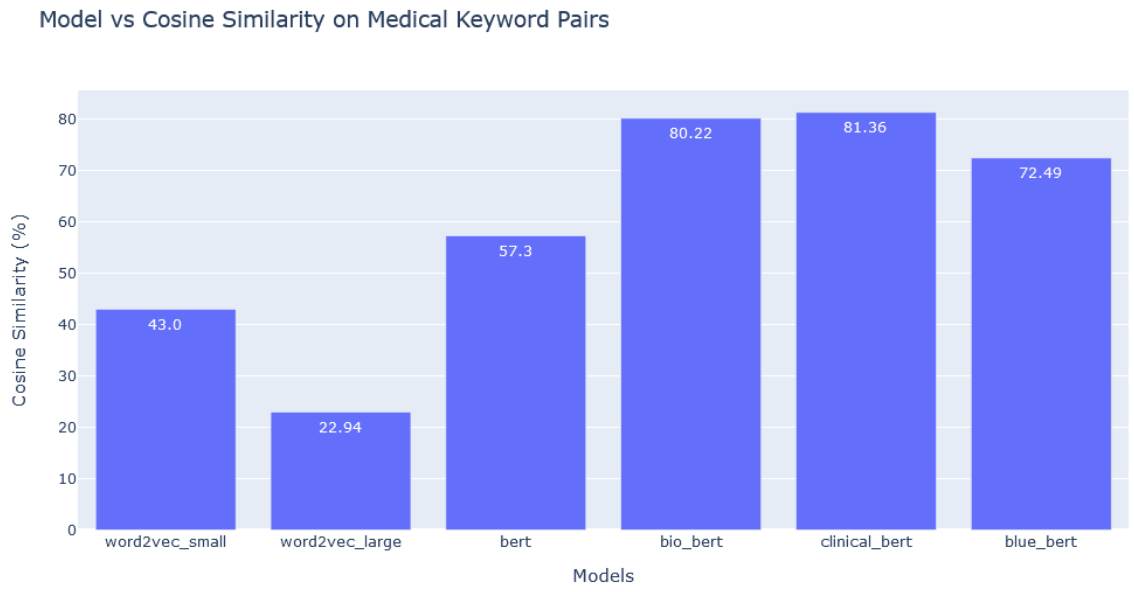
```
Total number of pairs =   558
Total cosine_similarity =   404.4722881615162
mean cosine_similarity =   0.7248604893684387
```

This model is not as good as BioBert and ClinicalBERT. Lets also verify the embedding performance on ClincNotes data as well-



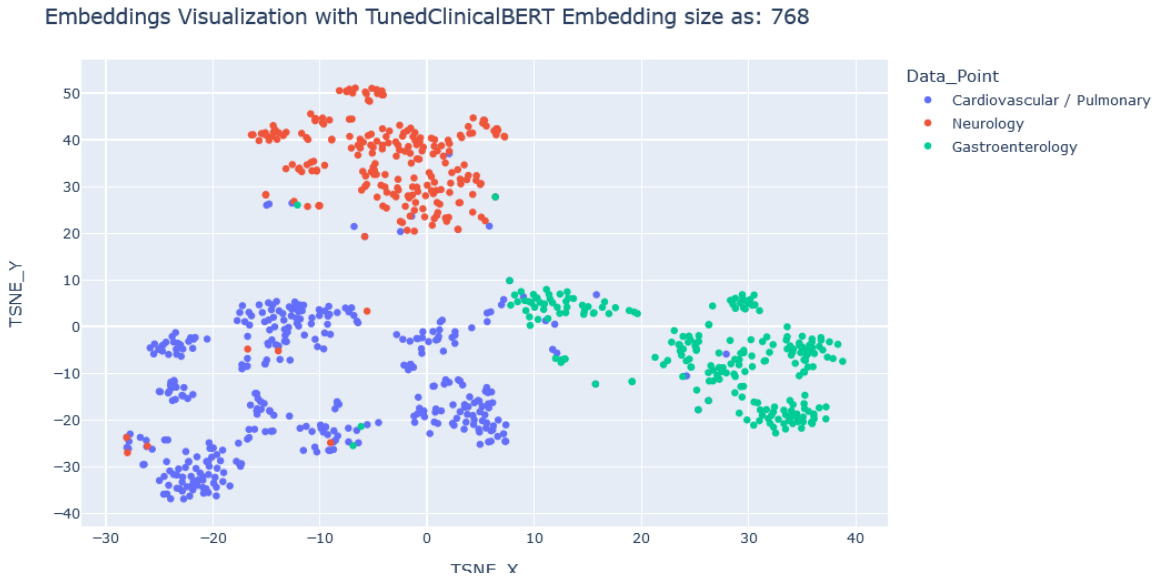
Scatter plot result too is poor than BioBERT and ClinicalBERT.

Now Let's check all models' performance together on cosine similarity task..



ClinicalBERT is best with 81.36% similarity score.

Because Clinical BERT outperformed any other model, I thought of fine tuning it even further. I trained it on ClinNotes.csv data to predict the category given notes to it. It was a 3 class classification problem. Before feeding it into the model, I performed some preprocessing steps such as removing special symbols and multiple spaces and dividing lengthy notes in smaller chunks. New chunks created had new rows with the same category. Note that chunking was important, otherwise we might have lost important information while training with BERT as it can process only 512 tokens in one go, the rest of the tokens would have been ignored. Once this model was trained, i used it to get embeddings for clinic notes data and mapped them against category below is the result i got-



This fine tuned model has formed the best groups and the reason is obvious, i have trained it on similar kinds of data. With this model I noticed that the similarity score has reduced a little. Below is the similarity score on keyword pair.

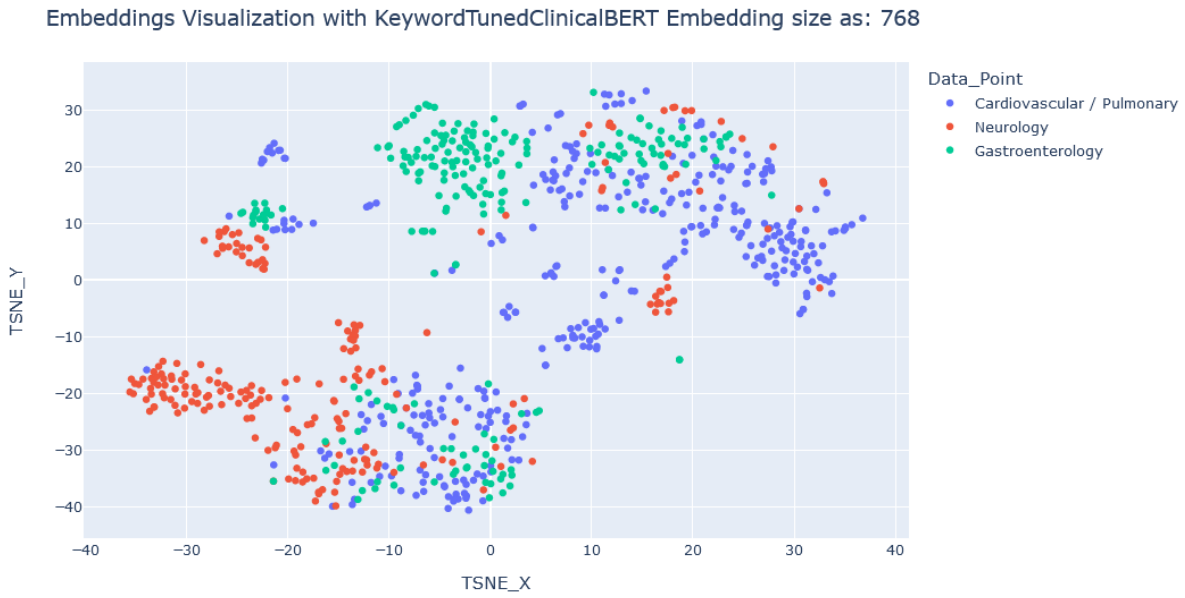
```
Total number of pairs =   558
Total cosine_similarity =   433.7339897155762
mean cosine_similarity =   0.7773011922836304
```

The reason of it is that ClinNotes data does not contain all the keywords that are there in MedicalConcepts.csv file.

I also tried improving the performance on keyword pair similarity tasks, for which I tried the technique below. Note that for this task too I am fine tuning the pre trained ClinicalBERT model.
:
For each right keyword pair , I created and wrong keyword pair. Then I mapped the right keyword pair with 1 and wrong keyword pair with 0. Now it became a binary classification problem where the task was to identify if a pair is equal to 1 or 0. I trained this model just for 1 epoch and I found a slight improvement in cosine similarity. Below is the result-

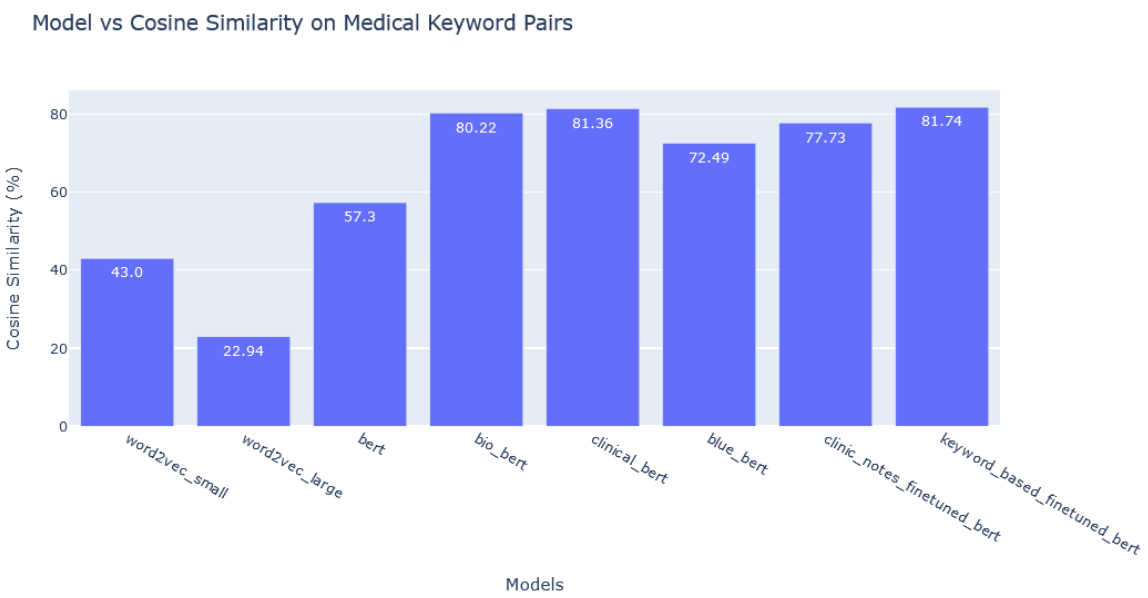
```
Total number of pairs =   558
Total cosine_similarity =   456.11888909339905
mean cosine_similarity =   0.8174178600311279
```

As you can see, mean cosine similarity is 0.8174. Note that, with the pretrained ClicalBert model, it was 0.8135. I know it's a slight improvement (I just ran it for 1 epoch), but I think results can further be improved with right hyperparameter tuning and more data. I could not perform hyperparameter tuning because of insufficient computation power.
Now lets plot the embeddings obtained from this model for ClinicNotes data.



Formed groups are actually not too bad but also not very good.

Lastly let's verify each models performance on keyword cosine similarity task together-



Keyword_based_fine_tuned_bert and given best similarity score.

According to me with the type of data that I am available with. Pertained ClinicalBERT model, ClinicalBERT fine tuned on clinic notes data and ClinicalBERT tuned on keyword pair task could be researched even further.