NLP - Exercise 4

Student ID: 211406343

Part 1:

Word Pairs I used and the distances I got:

1. girl - boy:	0.93271995
2. man - woman:	0.8860338
3. playing - games:	0.7456378
4. dancing - love:	0.67933095
5. fast - slow:	0.87440497
6. sweet - sugar:	0.67736775
7. smoke - haze:	0.5486138
8. food - hungry:	0.58667403
9. east - west:	0.95427144
10. door - open:	0.655653

From the results above, I can see that opposite words have got the highest values for similarity. However, words with strong relation between them like “food” and “hungry” have got small similarity between them. This leads us to understand that similarity between two words is measured by replacing the two given words in the sentences they appeared and see how many times the new sentences\ subsentences appeared. Since replacing opposite words gives a meaningful and sensible sentence in most of the times, they get the highest values of similarity. After replacing the opposite words, I will probably get a totally different meaning for the sentence, but the sentence would still be sensible. On the other hand, pairs with words related to each other get a smaller value for similarity but it is still a good value like the one for “door” and “open”.

Part 2:

1. Most of the sentences I got for the new song were okay, most of the nouns were replaced with their plural\singular form. Thus, the meaning did not change in most of the sentences despite the wrong form I have got. In other sentences where I replaced adjectives, I got opposite meaning but we still got sensible sentences like when I replaced “my pool warm” with “my pool cool”.

2. The worst sentences I have got were by replacing a verb in the sentence, changing the verb of a sentence would totally change its meaning and in most cases the sentence would make no sense.

Part 3:

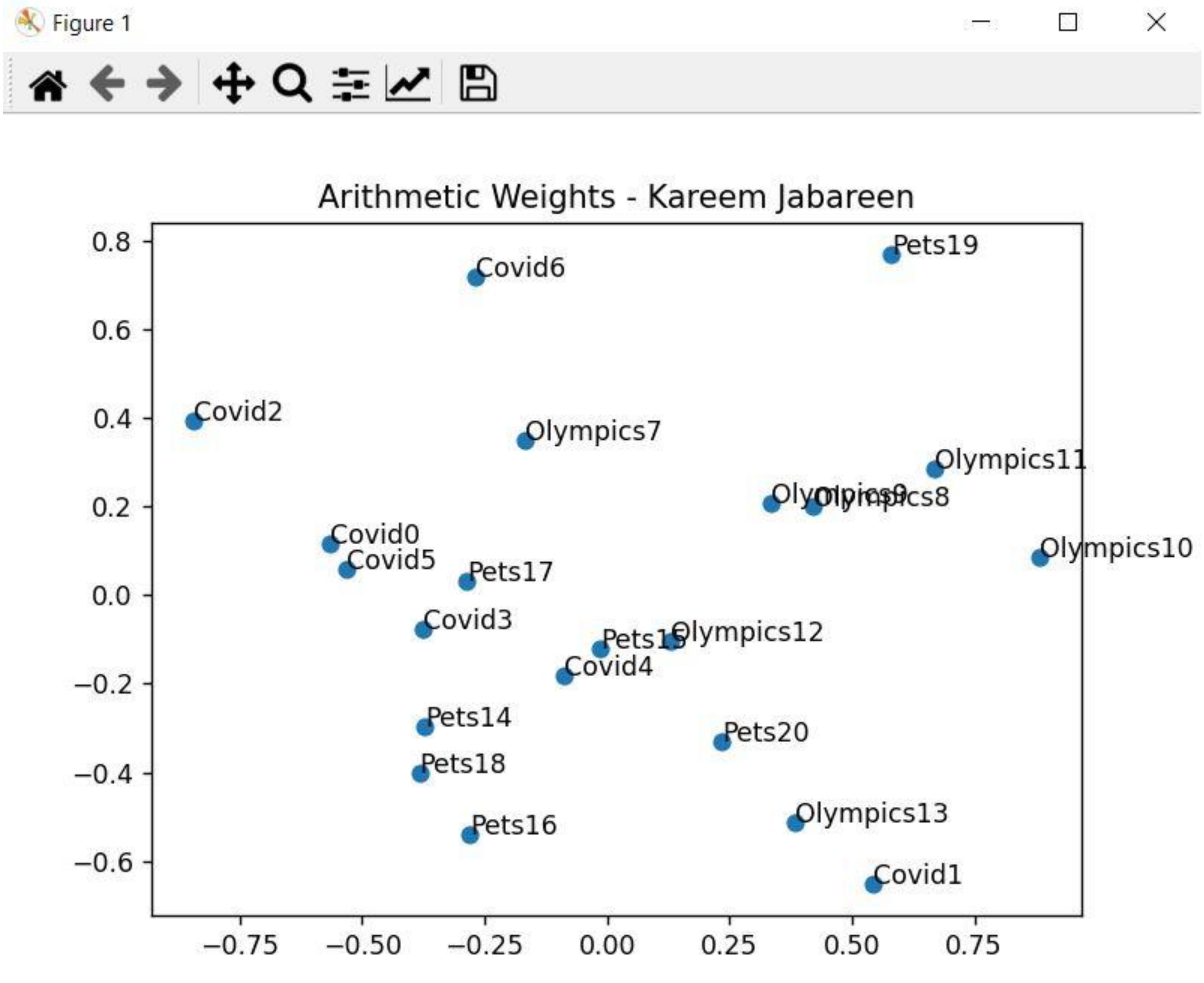
For the custom weight function, I created two lists of words, one is called “weak_words” and the other one is called “strong_words”.

In “weak_words” I put words with no meaning but are essential to almost every sentence like “in”, “the” and the word “and”. These words can be found in most of English sentences. Therefore, I decided to call them “weak words” and to give their vectors a small weight (0.5) since they cannot tell us anything about the topic of the sentence.

In “strong_words” I put words that have a meaning and are related somehow to one of the three topics we have (Covid, Olympics and Pets). These words might indicate the topic of the sentence, so I gave their vectors a special high weight in my function which is 10.

The rest of the words have got a random weight in the range [1, 9]. Note that the range is nearly in the middle of the smallest weight value and highest one. Thus, the rest of the words will have a good influence on the classification, but it will not be as great as the impact of “strong words”.

Here are the graphs we have got after applying dimension reduction using PCA and running the classification algorithm with 3 different weight functions:



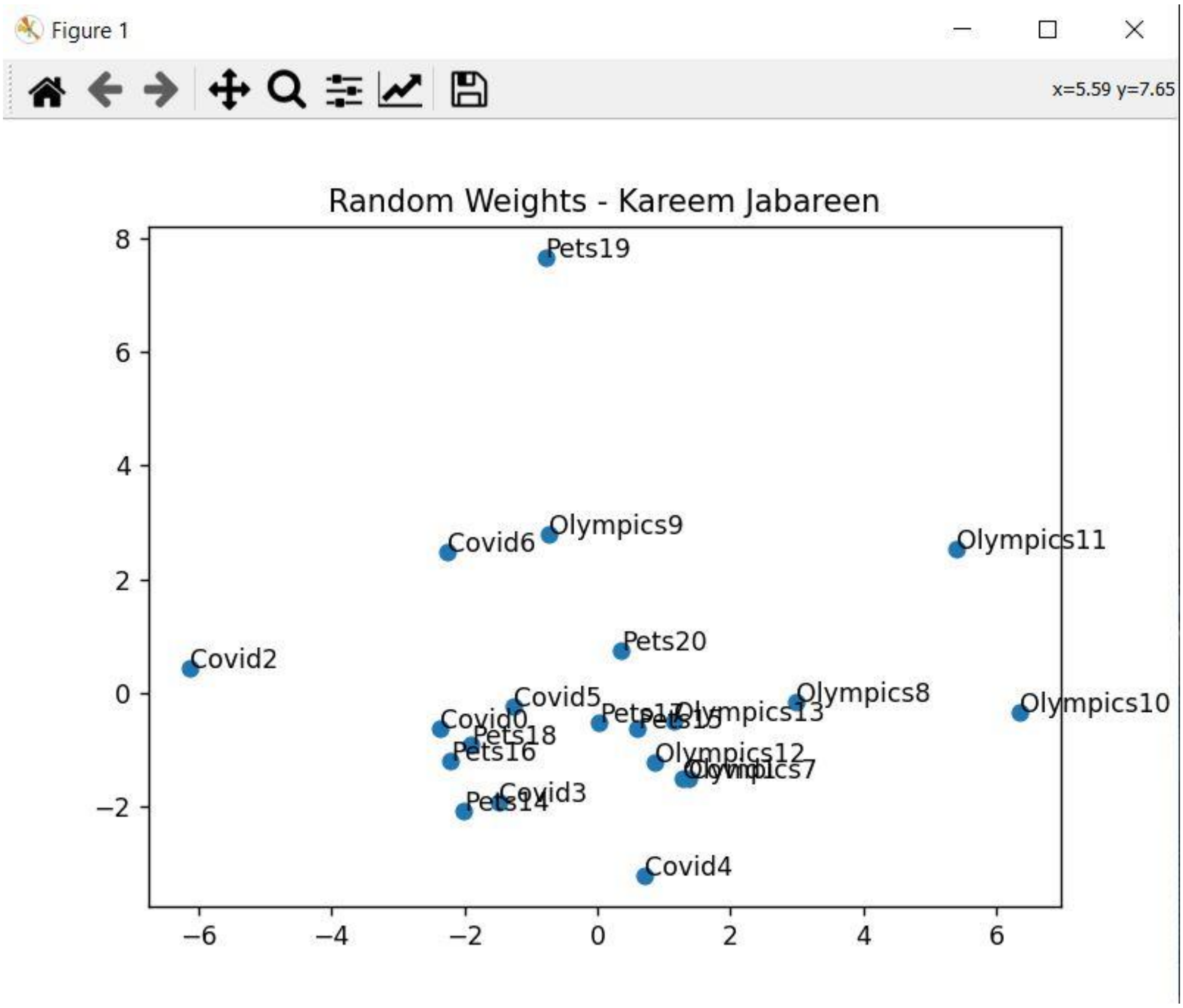
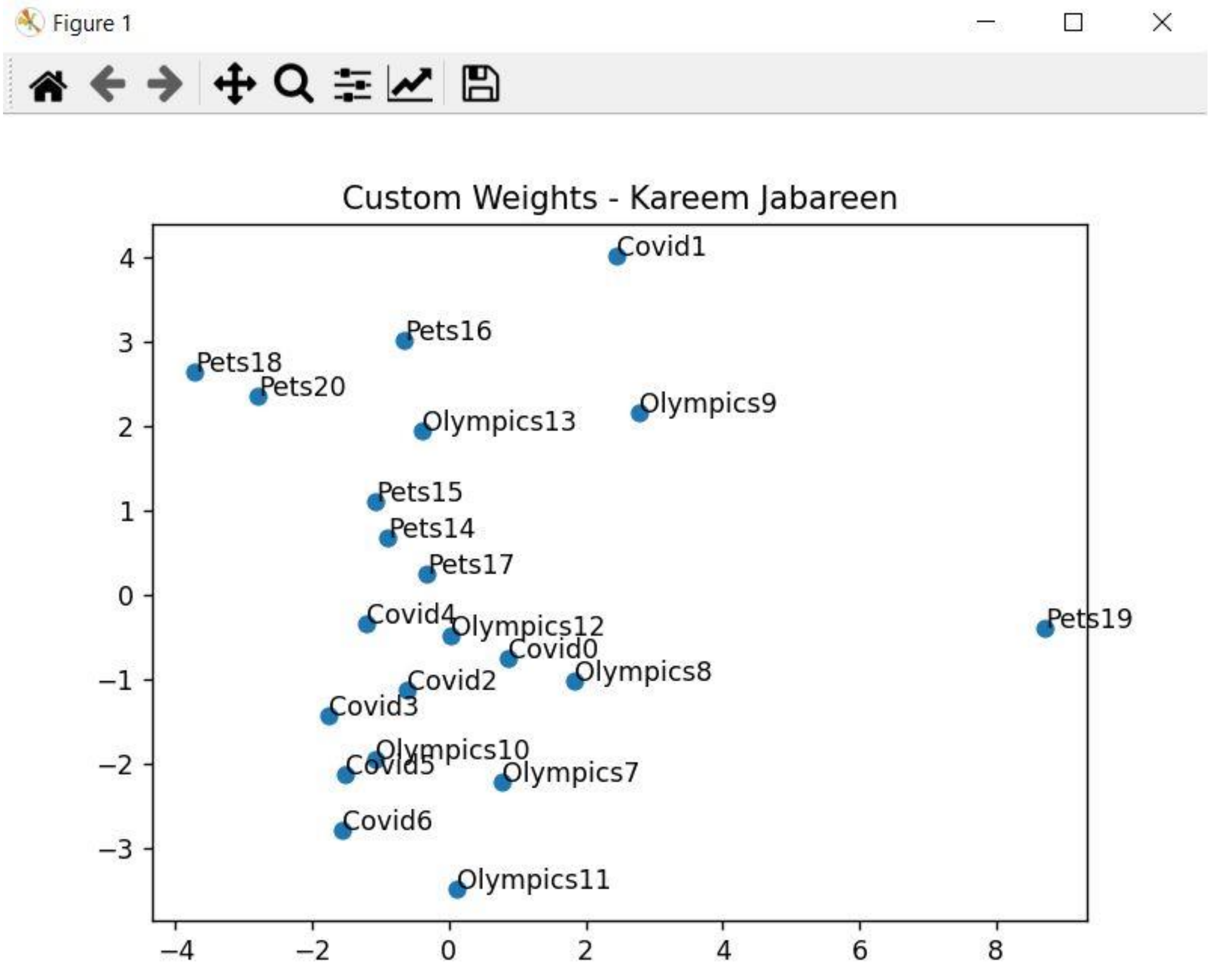


Figure 1
x=5.59 y=7.65
Random Weights - Kareem Jabareen
8
6
4
2
0
−2
−6
−4
−2
0
2
4
6
Pets19
Covid6
Olympics9
Olympics11
Pets20
Covid2
Covid5
Olympics8
Covid0
Pets17
Olympics13
Pets15
Olympics10
Pets18
Pets16
Olympics12
Covid1
Olympics7
Pets14
Covid3
Covid4



1. The best results we I got where for the weight function I created myself. However, it still has got outliers.

2. The best separated tweets are the Olympics ones, that is because in the Pets tweets we can see several words related to Covid. So, the points indicating Pets tweets are not well-separated from the tweets in the topic of Covid. On the other hand, the Olympics tweets talk mostly about sports only and most of them are long tweets which makes the classification easier.

3. We can build classifier by words vectors, but the results do not seem to be accurate. The classification might be great when dealing with two output classes, but when talking about several classes, the task becomes extremely harder and we might need extra information about the tweet – like the tweet's author and his details like his job, gender and age - not just its content in order to get a fair classifier.