Question 1D (-8)

Consider the role of the parameter `words`.

Try the following code on the check:

words = ['test1', 'test2', 'test3', 'test5']

plot\_embeddings(M\_reduced\_plot\_test, word2ind\_plot\_test, words)

Question 1E and 2A are wrong consequently. Please see the correct plot. From question 1E and 2A you should have asked whether you are producing the right plots.

def plot\_embeddings(M\_reduced, word2ind, words):

""" Plot in a scatterplot the embeddings of the words specified in the list "words".

NOTE: do not plot all the words listed in M\_reduced / word2ind.

Include a label next to each point.

Params:

M\_reduced (numpy matrix of shape (number of unique words in the corpus , 2)): matrix of 2-dimensioal word embeddings

word2ind (dict): dictionary that maps word to indices for matrix M

words (list of strings): words whose embeddings we want to visualize

"""

import matplotlib.pyplot as plt

### SOLUTION BEGIN

types = word2ind.keys()

x\_coords = M\_reduced[:,0]

y\_coords = M\_reduced[:,1]

# calculate range of x and y values

x\_range = np.max(x\_coords) - np.min(x\_coords)

y\_range = np.max(y\_coords) - np.min(y\_coords)

for i,type in enumerate(types):

if type in words:

x = x\_coords[i]

y = y\_coords[i]

plt.scatter(x, y, marker='x', color='red')

plt.text(x + 0.0001, y + 0.0001, type, fontsize=9)

plt.show()

### SOLUTION END

Question 2D (-2), but I graded your homework based on part 1 and 4.

Analogy does not correspond to the equality. Technically speaking, arg max\_x cos((w+(g-m)),x), and here w+(g-m) is enough.

Question 4