CSCE 771: Computer Processing of Natural Languages
Prof. Biplav Srivastava, Fall 2022

Quiz 2 / Instructions

- This is a programming quiz. Code has to be submitted in a directory of your GitHub called “Quiz2” with sub-dir for code, data and doc. Code will have your source code, data will have any input or output generated, and doc will have a .pdf of this file (called Quiz2-CSCE771-answers.pdf) along with any answers
- Complete quiz by 9:00 am on Monday, Oct 3, 2022 by sending an email to biplav.s@sc.edu confirming completing the quiz and attaching your Quiz2-CSCE771-answers.pdf.
- Total points = 50 + 20 + 30 = 100
- Obtained =

Student Name:

Brendan Reidy

Question 1: Contextual word embedding and TF-IDF
[5 + 5 + 20 + 10 + 10 = 50 points]

(a) What is the benefit of using a counting based vector representation like TF-IDF over a learning based representation like Word2Vec? [5 points]

TF-IDF is more human readable compared to Word2Vec and therefore the results are more interpretable

(b) What are the advantages of character-based representation like fasttext over word-based representation like Word2Vec? [5 points]

Character-based approaches like fasttext can understand words out of vocabulary (such as typos) where as word based representations cannot

(c) In sample-code/l13-llm-quiz folder in course github, you will find a file called “projs.txt” containing the list of projects in the course. Do the following:

(i) Consider each line as a document and represent words in TF-IDF. [20 points]

(ii) Identify your project name and identify all projects similar to yours. Use a cosine similarity of 0.9 [10 points]

Output:

```
(image captioning using transformer models) with 1(evolving firearm regula
tions) is = [[0.]]
(image captioning using transformer models) with 2(crime analysis in south
carolina) is = [[0.]]
(image captioning using transformer models) with 3(target aspect based sen
timent analysis for urban neighborhoods) is = [[0.]]
(image captioning using transformer models) with 4(extracting synthesis pr
ocedure from solar cell perovskite based scientific publications.) is = [[
0.]]
(image captioning using transformer models) with 5(entity recognition : wa
ter data regulations) is = [[0.]]
(image captioning using transformer models) with 6(tos: banks' terms of se
rvices summary) is = [[0.]]
(image captioning using transformer models) with 7(water regulation summar
ization) is = [[0.]]
(image captioning using transformer models) with 8(predicting the 2022 gub
ernatorial election of south carolina using sentiment analysis of twitter.
) is = [[0.07753848]]
(image captioning using transformer models) with 9(scientific artical summ
arization) is = [[0.]]
(image captioning using transformer models) with 10(new fasttext [with ele
ction data]) is = [[0.]]
(image captioning using transformer models) with 11(chatbot to answer ques
ries regarding who water regulations ) is = [[0.]]
(image captioning using transformer models) with 12(verifying various food
s connection to improve diabetes using nlp techniques ) is = [[0.08344243]
]
(image captioning using transformer models) with 13(summarization of terms
and conditions) is = [[0.]]
(image captioning using transformer models) with 14(chatbot for elections
faq - state of mississippi) is = [[0.]]
(image captioning using transformer models) with 15(image captioning using
transformer models) is = [[1.]]
(image captioning using transformer models) with 16(specialist doctor reco
mmendation system) is = [[0.]]
(image captioning using transformer models) with 17(application of artific
ial neural networks (ann) to automatic speech recognition (asr) on a novel
dataset created using youtube) is = [[0.06449659]]
(image captioning using transformer models) with 18(detecting and rating s
everity of urgency in short, one-time crisis events vs. ongoing ones) is =
[[0.]]
(image captioning using transformer models) with 19(water regulations - ar
izona) is = [[0.]]
(image captioning using transformer models) with 20(damaged doc. predictio
n (10%)) is = [[0.]]
(image captioning using transformer models) with 21(visual question answer
ing) is = [[0.]]
```

According to the output, there are no projects with cosine similarity >= 0.9 to mine

(iii) Identify clusters of projects along the same theme, based on similarity of project names. (Hint: you just have to reuse your code from (ii) above) [10 points]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
0	1	0	0	0	0.18	0	0	0	0	0	0.13	0	0	0	0	0	0	0	0.24	0	0
1	0	1	0.13	0	0	0	0	0.31	0	0	0	0	0	0	0	0	0	0.11	0	0	0
2	0	0.13	1	0.1	0	0	0	0.16	0	0	0	0	0	0.12	0	0	0	0	0	0	0
3	0	0	0.1	1	0	0	0	0	0.16	0	0	0	0	0	0	0	0	0	0	0	0
4	0.18	0	0	0	1	0	0.17	0	0	0.19	0.21	0	0	0	0	0	0.11	0	0.38	0	0
5	0	0	0	0	0	1	0	0.11	0	0	0	0	0.27	0.07	0	0	0.05	0.05	0	0	0
6	0	0	0	0	0.17	0	1	0	0.28	0	0.13	0	0.23	0	0	0	0	0	0.23	0	0
7	0	0.31	0.16	0	0	0.11	0	1	0	0.11	0	0.05	0.13	0.1	0.08	0	0.11	0.07	0	0	0
8	0	0	0	0.16	0	0	0.28	0	1	0	0	0	0.22	0	0	0	0	0	0	0	0
9	0	0	0	0	0.19	0	0	0.11	0	1	0	0	0	0	0	0	0	0	0	0	0
10	0.13	0	0	0	0.21	0	0.13	0	0	0	1	0.08	0	0.13	0	0	0.06	0	0.28	0	0
11	0	0	0	0	0	0	0	0.05	0	0	0.08	1	0	0	0.08	0	0.1	0	0	0	0
12	0	0	0	0	0	0.27	0.23	0.13	0.22	0	0	0	1	0.09	0	0	0.05	0.17	0	0	0
13	0	0	0.12	0	0	0.07	0	0.1	0	0	0.13	0	0.09	1	0	0	0.04	0.04	0	0	0
14	0	0	0	0	0	0	0	0.08	0	0	0	0.08	0	0	1	0	0.06	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
16	0	0	0	0	0.11	0.05	0	0.11	0	0	0.06	0.1	0.05	0.04	0.06	0	1	0.03	0	0	0
17	0	0.11	0	0	0	0.05	0	0.07	0	0	0	0	0.17	0.04	0	0	0.03	1	0	0	0
18	0.24	0	0	0	0.38	0	0.23	0	0	0	0.28	0	0	0	0	0	0	0	1	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

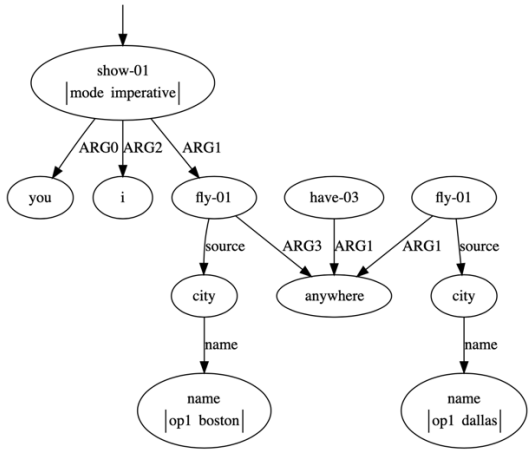
Similarity matrix (row = project number, column = project number, matrix = similarity of row to column)

Question 2: Semantics

[8 + 12 = 20 points]

Consider text = “show me flights from Boston to anywhere that has flights to Dallas”

(a) Using the online AMR tool at http://amparser.coli.uni-saarland.de:8080/, find the AMR structure of the example text. Paste it below.



(b) The AMR refers to specific variant of **show**, **fly** and **have**. Use pennbank and show the predicate, its arguments and its meaning. Use a propbank visualizer like https://verbs.colorado.edu/verb-index/index.php.

Show: *show.01* , cause to see, *Source:* , *vncls:* , *framnet:*

Roles:

Arg0-CAU: *shower* (vnrole: 29.5-2-agent, 37.1.1-1-1-agent, 40.3.2-agent, 78-1-1-cause, 48.1.2-agent)

Arg1-PPT: *thing seen/shown* (vnrole: 29.5-2-theme, 29.5-2-predicate, 37.1.1-1-1-topic, 40.3.2-patient, 78-1-1-topic, 48.1.2-theme)

Arg2-GOL: *seer* (vnrole: 37.1.1-1-1-recipient, 40.3.2-recipient, 78-1-1-recipient, 48.1.2-recipient)

Fly: *fly.01* , fly through the air, travel via air, fly in a flock.„ *Source:* , *vncls:* , *framnet:*

Roles:

Arg0-PAG*:* pilot, agentive entity capable of flight (like a bird) *(vnrole: 51.3.2-2-1-agent, 11.5-1-agent, 51.4.2-agent)*

Arg1-PPT*:* passenger, cargo *(vnrole: 11.5-1-agent, 51.4.2-theme)*

Arg2-VSP*:* aircraft flown, flight number, steed, non-agentive thing in motion *(vnrole: 51.3.2-2-1-theme)*

Arg3-VSP*:* type of flight plan, mission, cognate object (like 'a flight' or 'sorties')

Arg4-GOL*:* airline

Have: *have.03* , own, possess, *Source:* , *vncls:* , *framnet:*

Arg0-PAG: *owner* (vnrole: 100.1-pivot, 39.4-agent)

Arg1-PPT: *possession* (vnrole: 100.1-theme, 39.4-patient)

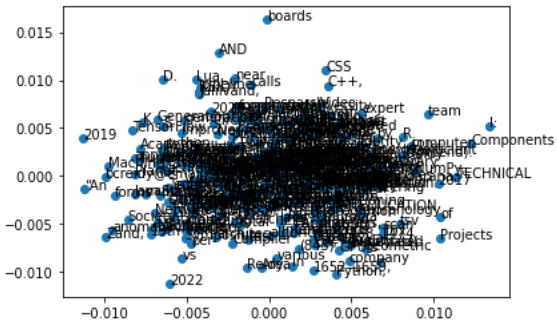
Question 3: Word2Vec

[10 + 10 + 10 = 30 points]

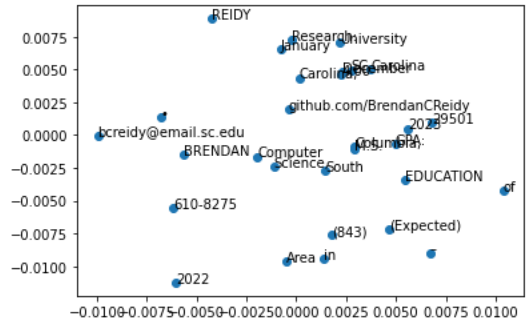
(a) Take your latest resume (must be more than 1 page). Create a word2vec representation for it using genism and print statistics of embeddings.

```
Statistics for word embeddings:
mean -6.0352636e-06 median 1.0376073e-06 min -0.0050356956 max 0.005061689
```

(b) Visualize the embedding using PCA.



First 30 words (more interpretable)



(c) Now create and visualize the embedding of the projects listed in the file - sample-code/l13-llm-quiz/projs.txt.