CSCI 59000 NLP Spring 2024 Homework 1

January 11, 2024

Instructions

We will be using Canvas to collect your assignments. Please read the following instructions to prepare your submission.

1. Submit your solution in a pdf file and a zip file (<yourLastName_FirstName>.pdf/zip). Your write-up must be in pdf. Your code must be in the zip file.
2. In your pdf file, the solution to each problem should start on a new page.
3. Latex is strongly encouraged to write your solutions, e.g., using Overleaf (https://www.overleaf.com/). However, scanned handwritten copies are also acceptable. Hard copies will not be accepted.
4. You may discuss the problems and potential directions for solving them with another student. However, you need to write your own solutions and code separately, and not as a group activity. Please list the students you collaborated with on your submission.

Problem 1 (10 points)

Provide answers to the following operations. If it is not possible to calculate, write "invalid."

(a)

$$\begin{bmatrix} 2 & 7 & 7 \\ 4 & 3 & 1 \end{bmatrix} \begin{bmatrix} 0 & 2 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}$$

(b)

$$\begin{bmatrix} 0 & 6 \\ 9 & 2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 1 & 7 \\ 2 & 3 \\ 5 & 1 \end{bmatrix}^{\top}$$

(c)

$$\begin{bmatrix} 2 & 0 & 2 \end{bmatrix}^{\top} \begin{bmatrix} 4 & 1 & 4 \\ 4 & 4 & 0 \\ 1 & 0 & 3 \end{bmatrix}^{\top}$$

(d)

$$\begin{bmatrix} 3 & 1 & 0 \end{bmatrix}^{\top} \begin{bmatrix} 1 & 4 & 5 \end{bmatrix}$$

(e)

$$\begin{bmatrix} 1 & 4 \\ 7 & 0 \end{bmatrix}^{\top} \begin{bmatrix} 6 & 8 \end{bmatrix}$$

Problem 2 (20 points)

Write regular expressions for the following cases.

(a) One or more words (only with lowercase alphabets) separated by spaces
e.g., "red blue green white"

(b) Title case sentences, assuming all words are capitalized. Note that the text can contain numbers and punctuation.
e.g., "Why Sleep Is So Important To Your Health?"

(c) Strings that contain the word "ice" without matching the words that contain "ice" such as "icecream" or "ice-bucket"

(d) The set of all lowercase alphabetic strings ending in a "b"

(e) All strings that start at the beginning of the line with an integer and that end at the end of the line with a word

Problem 3 (20 points)

(a) Pick a language other than English, which you are familiar with. Your mother tongue will be good if it is not English. If your mother tongue is English only, you can pick any language you are most familiar with.

i. Which language did you pick?

ii. What kind of challenges do you expect when processing texts in that language and building NLP models? List two or three challenges. Describe each challenge concisely. If you use an example, please translate it into English.

For example, I pick Korean.

- Korean is a free-order language that uses case markers to specify grammatical roles (i.e., to indicate a word is used as a subject, a subject case marker is appended at the end of a word without a space), which means that words are not separated by spaces in many cases. This can make the tokenization process harder.
- Korean is a pro-drop language where certain word classes can be omitted in a sentence when they can be inferred by context. This can be a challenge when building an NLP model because each sentence might not contain all necessary parts to understand the sentence and inferences should be made.

(b) Find one publicly available English dataset for an NLP task. Describe the following attributes.

i. Motivation (e.g., why was the corpus collected? by whom? who funded it?)

ii. Situation: In what situation was the text written?

iii. Collection process: if it is a subsample how was it sampled? Was there consent? What kind of pre-processing was done?

iv. Annotation process: is it annotated? If so, which labels? what was the annotation process?

Problem 4 (Programming involved, 50 points)

Download the Diplomacy training dataset (train.jsonl): `https://sites.google.com/view/qanta/projects/diplomacy`

We will only use the language data in the "messages" field in the JSON format. First, read the dataset (train.jsonl) and extract only the "messages" using a JSON parser. Write a new file ("data.txt") which contains messages in each line. We will use **data.txt** from now on.

Use the NLTK package (`https://www.nltk.org/`) to split the data into sentences. Use the `sent_tokenize()` function.

(a) How many sentences are there? Note that you need to ignore empty lines and empty sentences.

Now let's find words. First, split the sentences using the python `split(' ')` function.

(b) How many tokens are there?

This time, use NLTK's `word_tokenize()` function to split into words.

(c) How many tokens are there now?

Lowercase all the words.

(d) How many tokens and types now?

(e) Compare the number of tokens from (b), (c), and (d). Why are they different?

Lastly, make a dictionary of word type counts. The dictionary contains word type as its key, and frequency as its value. Sort the dictionary.

(f) What is the most frequent word type?

(g) What is the 5th most frequent word type?

(h) Using the sorted dictionary, draw a graph to see if this data shows the Zipf's law. X-axis: ranked words, Y-axis: frequencies. Submit your graph and discuss it (in your pdf). Submit your code as well.