assignment4

October 22, 2023

[ ]:
```
import nltk
from nltk.tokenize import word_tokenize
import numpy as np
import os
import torch
import transformers
from transformers import AutoModelForQuestionAnswering, AutoTokenizer, pipeline

transformers.logging.set_verbosity_error()

!pip install torch
!pip install transformers
nltk.download('punkt')
```

```
Requirement already satisfied: torch in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (2.1.0)
Requirement already satisfied: typing-extensions in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from torch) (4.3.0)
Requirement already satisfied: jinja2 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from torch) (2.11.3)
Requirement already satisfied: sympy in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from torch) (1.10.1)
Requirement already satisfied: networkx in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from torch) (2.8.4)
Requirement already satisfied: fsspec in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from torch) (2023.10.0)
Requirement already satisfied: filelock in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from torch) (3.6.0)
Requirement already satisfied: MarkupSafe>=0.23 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from jinja2->torch) (2.0.1)
Requirement already satisfied: mpmath>=0.19 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from sympy->torch) (1.2.1)
Requirement already satisfied: transformers in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (4.34.1)
Requirement already satisfied: huggingface-hub<1.0,>=0.16.4 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from transformers) (0.17.3)
Requirement already satisfied: regex!=2019.12.17 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from transformers)
```

```
(2022.7.9)
Requirement already satisfied: numpy>=1.17 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from transformers) (1.21.5)
Requirement already satisfied: tokenizers<0.15,>=0.14 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from transformers) (0.14.1)
Requirement already satisfied: tqdm>=4.27 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from transformers) (4.64.1)
Requirement already satisfied: packaging>=20.0 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from transformers) (21.3)
Requirement already satisfied: filelock in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from transformers) (3.6.0)
Requirement already satisfied: requests in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from transformers) (2.28.1)
Requirement already satisfied: safetensors>=0.3.1 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from transformers) (0.4.0)
Requirement already satisfied: pyyaml>=5.1 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from transformers) (6.0)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from huggingface-
hub<1.0,>=0.16.4->transformers) (4.3.0)
Requirement already satisfied: fsspec in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from huggingface-
hub<1.0,>=0.16.4->transformers) (2023.10.0)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from
packaging>=20.0->transformers) (3.0.9)
Requirement already satisfied: certifi>=2017.4.17 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from
requests->transformers) (2022.9.24)
Requirement already satisfied: charset-normalizer<3,>=2 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from
requests->transformers) (2.0.4)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from
requests->transformers) (1.26.11)
Requirement already satisfied: idna<4,>=2.5 in
/opt/homebrew/anaconda3/lib/python3.9/site-packages (from
requests->transformers) (3.3)

[nltk_data] Downloading package punkt to
[nltk_data]     /Users/magnusde93/nltk_data…
[nltk_data]   Package punkt is already up-to-date!
```

[ ]:
```
True
```

[ ]:
```
def levenshtein_distance(s1, s2):
    matrix = [[0 for _ in range(len(s2) + 1)] for _ in range(len(s1) + 1)]
```

```
    for i in range(len(s1) + 1):
        matrix[i][0] = i
    for j in range(len(s2) + 1):
        matrix[0][j] = j

    for i in range(1, len(s1) + 1):
        for j in range(1, len(s2) + 1):
            cost = 0 if s1[i - 1] == s2[j - 1] else 1
            matrix[i][j] = min(matrix[i - 1][j] + 1,
                               matrix[i][j - 1] + 1,
                               matrix[i - 1][j - 1] + cost)

    return matrix[-1][-1]

word1 = ["apple", "flour", "hamburger", "python", "moon"]
word2 = ["banana", "flower", "cheeseburger", "java", "stars"]
for i in range(len(word1)):
    distance = levenshtein_distance(word1[i], word2[i])
    print(f"The Levenshtein distance between '{word1[i]}' and '{word2[i]}' is␣
 ↪{distance}")
```

```
The Levenshtein distance between 'apple' and 'banana' is 5
The Levenshtein distance between 'flour' and 'flower' is 2
The Levenshtein distance between 'hamburger' and 'cheeseburger' is 5
The Levenshtein distance between 'python' and 'java' is 6
The Levenshtein distance between 'moon' and 'stars' is 5
```

[ ]:
```
def tokens_file(file_path):
    with open(file_path, 'r') as file:
        text = file.read()
        tokens = word_tokenize(text)
        vocabulary = set(tokens)
    return vocabulary


def detect_misspelled_words(sentence, vocabulary):
    words = sentence.split()
    misspelled_word = ''

    for word in words:
        if word.lower() not in vocabulary:
            misspelled_word = word.lower()

    return misspelled_word

file_path = ("/Users/magnusde93/University-Assignments/Semester-3/
 ↪Introduction-to-lanugae-technology/of-mice-and-men.txt")
```

```
tokens = tokens_file(file_path)


correct_sentence = "Evening of a hot day started the little wind to moving␣
 ↪among the leaves"
misspelled_words1 = detect_misspelled_words(correct_sentence, tokens)
print("This sentence should have zero misspelled words but if there are any,␣
 ↪they are: ", misspelled_words1)

incorrect_sentence = "Evening of a hot day started the little wind to moving␣
 ↪among the leavs"
misspelled_words2 = detect_misspelled_words(incorrect_sentence, tokens)
print("Misspelled word in the same sentence with one word missing a letter: ",␣
 ↪misspelled_words2)
```

```
This sentence should have zero misspelled words but if there are any, they are:
Misspelled word in the same sentence with one word missing a letter:  leavs
```

[ ]:
```
model = pipeline('fill-mask', model='bert-base-uncased', top_k=10)
sentence = "That ranch we're goin' to is right down there about a quorter mile"
misspelled_word = detect_misspelled_words(sentence, tokens)
new_sentence = sentence.replace(misspelled_word, "[MASK]")

pred = model(new_sentence)
Levenshtein_words = []
for i in range(10):
    Levenshtein_words.append(pred[i]["token_str"])
print("The top 10 recommendations for the masked word are: ", Levenshtein_words)

for i in Levenshtein_words:
    print(f"The distance between {misspelled_word} and {i} is␣
 ↪{levenshtein_distance(i, misspelled_word)}")
```

```
The top 10 recommendations for the masked word are:  ['quarter', 'half',
'square', 'hundred', 'full', 'whole', 'couple', '½', 'nautical', 'mile']
The distance between quorter and quarter is 1
The distance between quorter and half is 7
The distance between quorter and square is 4
The distance between quorter and hundred is 5
The distance between quorter and full is 6
The distance between quorter and whole is 5
The distance between quorter and couple is 6
The distance between quorter and ½ is 7
The distance between quorter and nautical is 7
The distance between quorter and mile is 6
```

[ ]:
```
model_name = "deepset/roberta-base-squad2"
model = AutoModelForQuestionAnswering.from_pretrained(model_name)
tokenizer = AutoTokenizer.from_pretrained(model_name)

nlp = pipeline('question-answering', model=model_name, tokenizer=model_name)

context = "Rowling was born outside of Bristol England to Peter James Rowling,␣
 ↪an aircraft engineer, and Anne Rowling, a science technician. As a child,␣
 ↪Rowling often wrote fantasy stories and was very precocious. Rowling␣
 ↪attended secondary school at Wyedean School and College, where her mother␣
 ↪worked. Rowling then attended the University of Exeter, studying French and␣
 ↪Classics. After, Rowling worked as a researcher and bilingual secretary in␣
 ↪London for Amnesty International. She later moved to Portugal to teach␣
 ↪English at night, and to write during the day. There, she met Portuguese␣
 ↪journalist Jorge Arantes. They married two years later, and their daughter,␣
 ↪Jessica, was born a year after that. The couple separated a few months after␣
 ↪Jessica's birth, and Rowling moved with her infant daughter to Edinburgh,␣
 ↪Scotland. Rowling, who had gotten the idea for Harry Potter in 1990, wrote␣
 ↪the first book while completing a teacher training course. Rowling then␣
 ↪finished Harry Potter in 1995. Initially, only 1,000 copies were printed.␣
 ↪Five months later, the book won its first award, and in early 1998, an␣
 ↪auction was held in the United States for the rights to publish the novel,␣
 ↪which was won by Scholastic Inc. for $105,000. Harry Potter became a␣
 ↪sensation, growing larger with each book and shattering sales records. Harry␣
 ↪Potter is now a global brand worth an estimated $15 billion, and the books␣
 ↪have been adapted into record-breaking films as well. In 2001, Rowling␣
 ↪remarried and had a second child. She has also become a noted␣
 ↪philanthropist, donating significant money to combat poverty, social␣
 ↪inequality, and MS, or multiple sclerosis, a disease from which her mother␣
 ↪passed away. She continues to write, and has written several crime novels␣
 ↪under a pen name, Robert Galbraith."

questions = [
    "Where was J.K. Rowling born?",
    "What did J.K. Rowling study at the University of Exeter?",
    "Who did J.K. Rowling marry, and how many children do they have?",
    "When did the idea for Harry Potter first come to J.K. Rowling?",
    "Besides the Harry Potter series, what other type of novels has J.K.␣
 ↪Rowling written under a pen name?"
]


for question in questions:
    result = nlp(question, context)
    print(f"Question: {question}")
    print(f"Answer: {result}\n")
```

```
Question: Where was J.K. Rowling born?
Answer: {'score': 0.8303226232528687, 'start': 28, 'end': 43, 'answer': 'Bristol
England'}

Question: What did J.K. Rowling study at the University of Exeter?
Answer: {'score': 0.9836851954460144, 'start': 350, 'end': 369, 'answer':
'French and Classics'}

Question: Who did J.K. Rowling marry, and how many children do they have?
Answer: {'score': 0.5860772728919983, 'start': 591, 'end': 604, 'answer': 'Jorge
Arantes'}

Question: When did the idea for Harry Potter first come to J.K. Rowling?
Answer: {'score': 0.9831283688545227, 'start': 870, 'end': 874, 'answer':
'1990'}

Question: Besides the Harry Potter series, what other type of novels has J.K.
Rowling written under a pen name?
Answer: {'score': 0.6309955716133118, 'start': 1733, 'end': 1738, 'answer':
'crime'}
```

[ ]:
```
from urllib.request import urlopen
import string

harry_url = "https://raw.githubusercontent.com/bobdeng/owlreader/master/ERead/
  ↪assets/books/Harry%20Potter%20and%20The%20Half-Blood%20Prince.txt"
harry_open = urlopen(harry_url)
harry_utf8 = harry_open.read().decode('utf-8')

question = "Why was Harry Potter gay?"

answer = nlp(question, harry_utf8)
print(answer)
```

```
{'score': 0.1077936589717865, 'start': 905081, 'end': 905088, 'answer': 'Old
age'}
```

1 Part 4

For this part it answered right for all questions related to context I fed it. Nothing wrong. Only problem is having a question in two parts. If the question has two parts it only answers the first part.

The answer for the question that is not related to the context the answer it gave was hilarious. It is obviously very much incorrect but so funny nonetheless.

2 Part 5

I talked to you in person about my final project. Looking forward to actually start working on it. Hopefully it's doable!