Ashish Shukla cs.ashishshukla@gmail.com (mailto:cs.ashishshukla@gmail.com)

Final production code for API for stackoveflow case study

In [1]:
```
# Importing libraries
import warnings
warnings.filterwarnings("ignore")
import numpy as np
import pandas as pd
import xml.etree.ElementTree as et
import os
import re
import csv
from nltk.corpus import stopwords
from nltk.stem import SnowballStemmer, WordNetLemmatizer, PorterStemmer
from nltk.tokenize import word_tokenize
from datetime import datetime
from sklearn.metrics.pairwise import cosine_similarity
import pickle
import tensorflow_hub as hub # Version Should be 0.8.0
```

In [6]:
```
# Loading embeddings data
try:
    word2vec_titles_array, all_dataframe_without_preprocess_df,  all_data_index_
    print("It exists")
except:
    word2vec_titles_array = np.load('all_titles_embeddings.npy')
    all_dataframe_without_preprocess_df = pd.read_csv('all_dataframe_without_prep
    with open('keyword_index_dict.pickle', 'rb') as handle:
        all_data_index_dictionary = pickle.load(handle)
    embed = hub.load("https://tfhub.dev/google/universal-sentence-encoder/4")
    print("All Loaded")
```

```
All Loaded
```

In [11]:
```
# Algorithm for detecting text
def suggest_questions_for_question_title(question_title, k):
    """
    Before calling this function we must load fallowing files
        Load Pretrained Embeddings 'all_titles_embeddings.npy' as word2vec_titles
        Load keyword_index_dict as all_data_index_dictionary
        Load all_dataframe_without_preprocess_df['Title']
    This function does fallowing tasks
        1 - preprocessing of given raw question_title
        2 - converting preprocessed question_title into (1, 512) shape numpy vect
        3 - extract all keywords from given question_title
        4 - get all indices of stored question titles in which at least one of th
        5 - take Numpy arrays of titles corresponding to all above selected indic
        6 - Calculate Cosine distance between asked question title numpy array an
        7 - Select k indices corresponding to k minimum cosine distances calculat
        8 - Select k indices from selected indices in step 4 corresponding to k m
        9 - take k question titles from all_dataframe_without_preprocess_df['Titl
            and return it as search result.

    """

    # question_title is the title of the single question

    document = question_title
    stop_words = set(stopwords.words('english'))

    stemmer = WordNetLemmatizer()
    data = document
    cleanr = re.compile('<.*?>')
    cleancode = re.compile('<code>.*?</code>')

    clean_text = re.sub(cleanr, ' ', data)
    clean_text = re.sub(cleancode, ' ', clean_text)
    clean_text = re.sub(r'[^A-Za-z]+',' ',clean_text)
    clean_text = clean_text.lower()

    striped_html_text = clean_text
    words = word_tokenize(str(striped_html_text.lower()))
    #Removing all single letter and and stopwords from question except for the le
    cleaned_document = ' '.join(str(stemmer.lemmatize(j)) for j in words if j not
    question_title_preprocessed = cleaned_document


    # embed is the loaded Universal sentence encoder from tensorflow hub
    question_title_vectorized = embed([question_title_preprocessed]).numpy()

    keywords_in_question = question_title_preprocessed.split()

    # keyword_index_dict contain keywords as keys and list of index of documents
    all_indexes = []
    for keyword in keywords_in_question:
        try:
            all_indexes.extend(all_data_index_dictionary[keyword])
        except:
            pass
```

```
    index_questions_keyword = np.array(list(set(all_indexes)))


    # word2vec_titles_array is the pre loaded array of all vectors
    # Select only those questions embeddings array which contain keyword of query
    word2vec_titles_array_selected = word2vec_titles_array[index_questions_keywo


    """
    Here all_arrays shape should be (n, d) and
    test array shape should be (1, d)
    """
    all_arrays, test_arrays = question_title_vectorized, word2vec_titles_array_s

    all_cosine_similarity = cosine_similarity(all_arrays, test_arrays)
    all_cosine_distances = (1 - all_cosine_similarity).reshape(1, -1)[0]

    # Since we have to select only k minimum distances so there is no need to sh
    index_of_least_distances = np.argpartition(all_cosine_distances, k)[:k]
    selected_k_cosine_distances =  all_cosine_distances[index_of_least_distances

    index_of_least_distances_sorted = np.argsort(selected_k_cosine_distances) # 
    final_index_of_least_distances_sorted = index_of_least_distances[index_of_le

    index_questions_keyword_similar = index_questions_keyword[final_index_of_lea
    recommendations = all_dataframe_without_preprocess_df.loc[index_questions_ke
    return recommendations
```

Testing of above function

In [12]:
```
# Test 1
query_question = "How to create a linked list in python?"
print("The query question is: ", query_question)

start = datetime.now()
recommendations = suggest_questions_for_question_title(query_question, 15)

print("Time taken is: \n", datetime.now()-start)

print("Suggestions are: ")
for recom in recommendations:
    print(recom)
```

```
The query question is:  How to create a linked list in python?
Time taken is:
 0:00:00.948463
Suggestions are:
linked list in python
Linked Lists Python 2.7
circularly linked list in python
Python; Linked list and traversing!
Concatenate Python Linked List
Circular Linked list in python
Why Python doesn't have a native Linked List implementation?
Python linked list O(1) insert/remove
doubly Linked list iterator python
single linked list reverse in python
Singly Linked List with special methods in python, stuck
Does Python use linked lists for lists? Why is inserting slow?
Faster way to create a linked list of n-length in Python
python linked list evaluation on the node of self
How to create a linked list with a given size(java)?
```

In [13]:
```
# Test 2
query_question = "How to reverse a linked list in C?"
print("The query question is: ", query_question)

start = datetime.now()
recommendations = suggest_questions_for_question_title(query_question, 15)

print("Time taken is:\n ", datetime.now()-start)

print("Suggestions are: ")
for recom in recommendations:
    print(recom)
```

```
The query question is:  How to reverse a linked list in C?
Time taken is:
  0:00:00.774923
Suggestions are:
How to reverse linked list C++
Reversing a singly linked list in C
How to reverse a linked list?
reverse a linked list?
reverse printing of linked list in c
Modifying Linked Lists in C++
Singly Linked List - C
Sort a linked list in C++
Reversing a linked list
reversing linked list
Linked List in C
More linked lists in C
linked list in C++
reverse linked list problem
Doubly linked list in C
```

In [ ]:
```

```