FREQUENTLY ASKED QUESTIONS EXTRACTOR

INTERNSHIP REPORT

By:- Siddhant Swaroop Dash

Objective:

For a given dataset of questions write a code to extract the most frequently asked questions relating to the given topics.

Approach:

The approach for this problem is to find out keywords from the given questions in the dataset and find the words that might be interrelated. Metadeta is extracted for easier correlation.

The dataset given consists of relatively small length sentences. The already present keyword extraction algorithms such as RAKE do not perform well on these kind of sentences, since they are based on frequency of a word which is usually higher in a paragraph and not in few sentences. Hence a part of the concept of BAG OF WORDS has been used along with SIMILAR STRING MATCHING. The question dataset and the topic dataset are stored in dataframes and are visualised first. Then the unwanted data is removed and metadata is extracted.This algorithm is as good as the NLTK pos\_bag()

THE CODE:

FAQ.ipynb

---------------

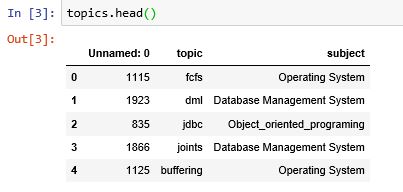
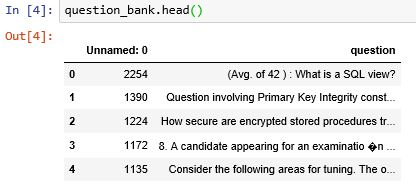
Importing required libraries



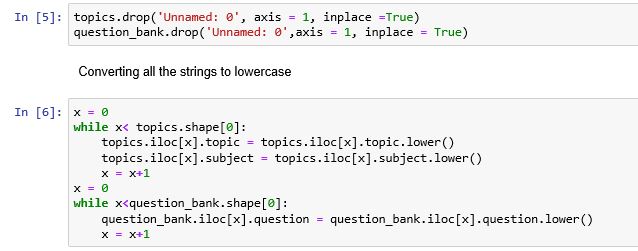
Reading CSV files



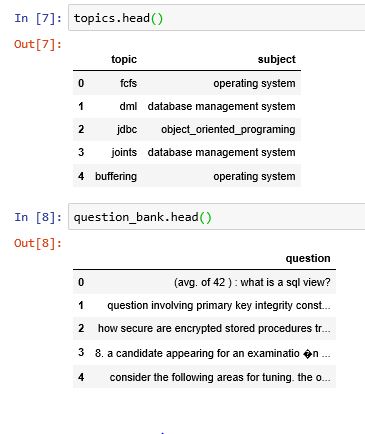
Preview Before Cleaning the data



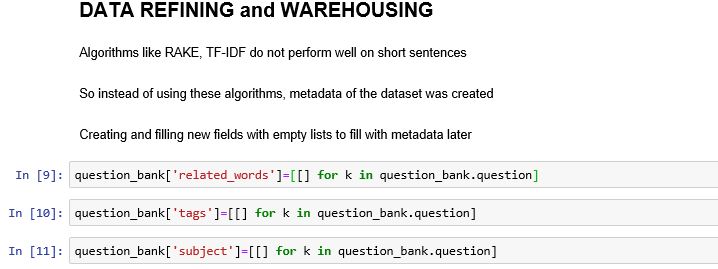
Removing unwanted data



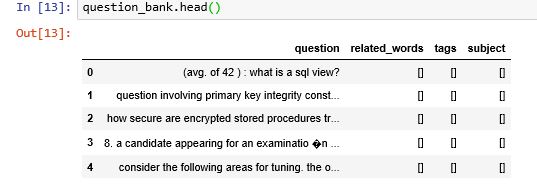
After cleaning preview

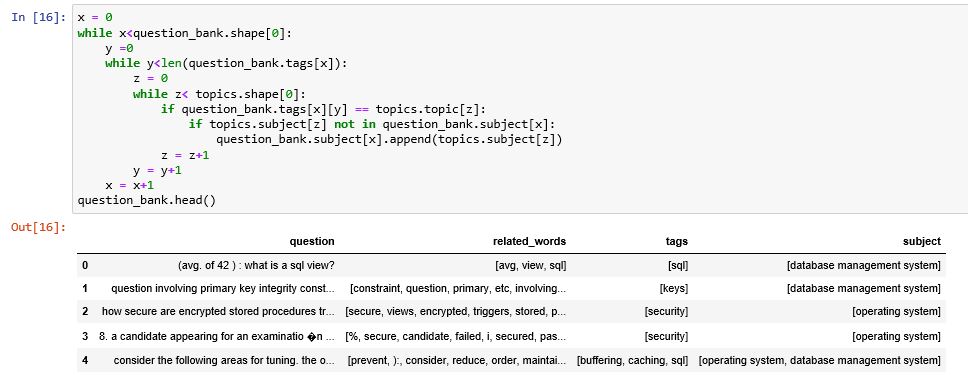
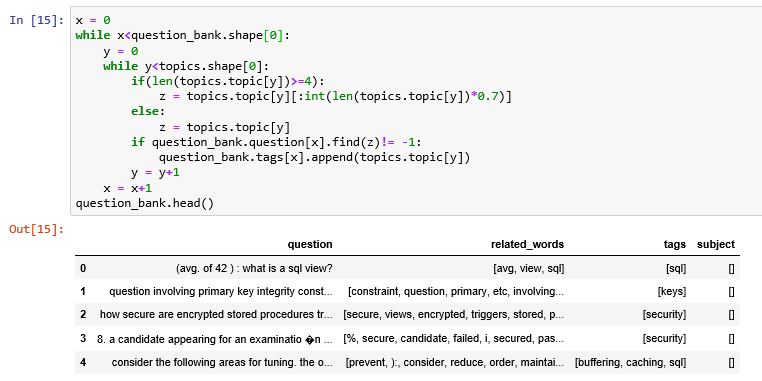
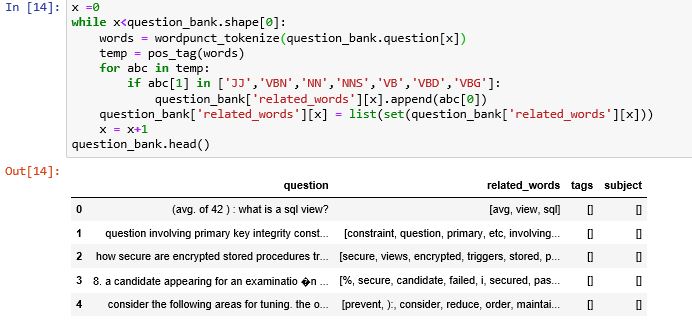


Refining and Warehousing the data



Adding New fields for metadata



Creating metadata

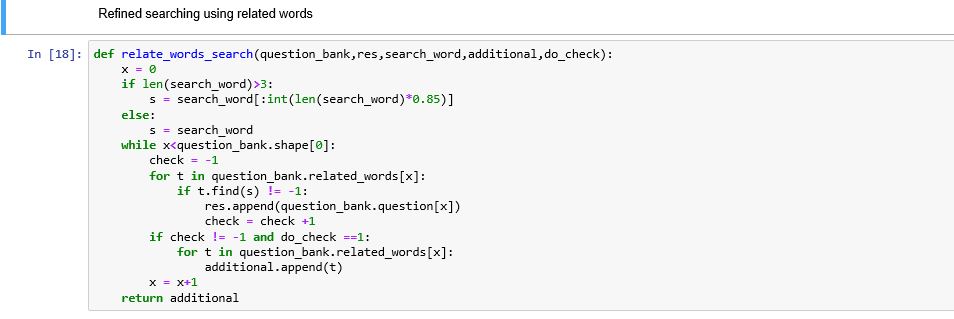
Searching Part:

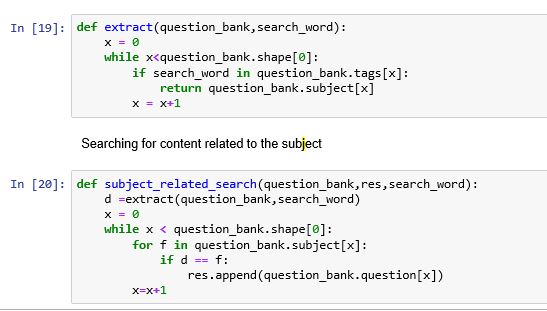
Searching was divided into 3 parts;

Tag wise search

Related words wise search

Subject wise search

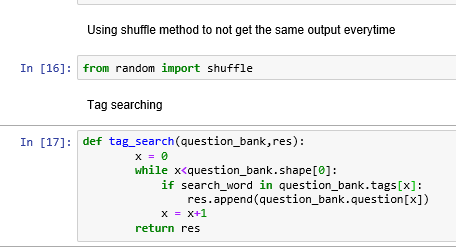




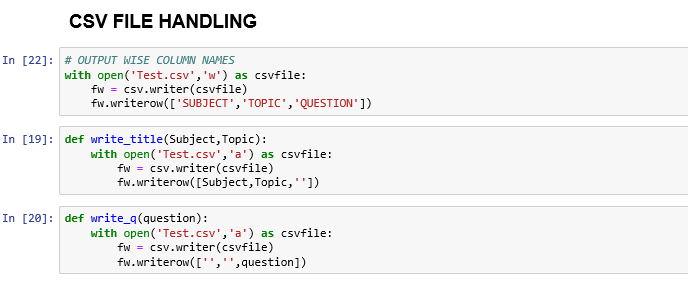
Tag search

The main searching algorithm is a simple tag matching algorithm. The other algorithms act as a failsafe if the number of outputs is very less. The other two delve into deep matching, i.e. searching for matching keywords/related keywords and finally matching questions based on the topic background.

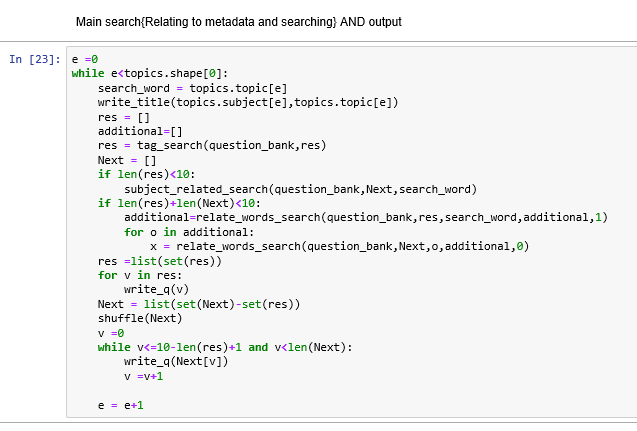
Using shuffle to get different outputs every time

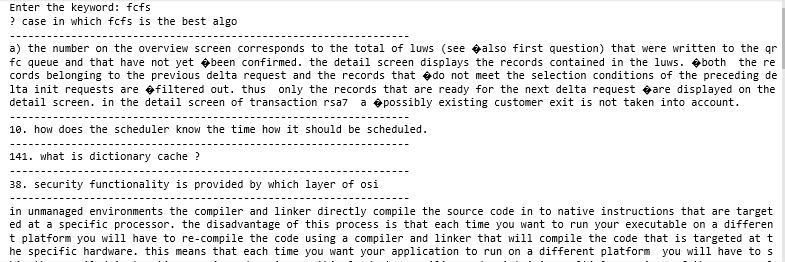


CSV part:



Main Searching:



Output:(sample output when fcfs was searched) 

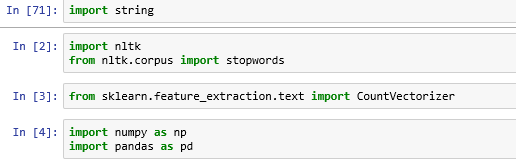
Finally using this program, we can find questions related to any word available in the the dataset and not only the topics and subjects. It gives the question that best matches the given search word by following the algorithm,i.e. Tag search and then subject wise search and deep matching

Scoring.ipynb

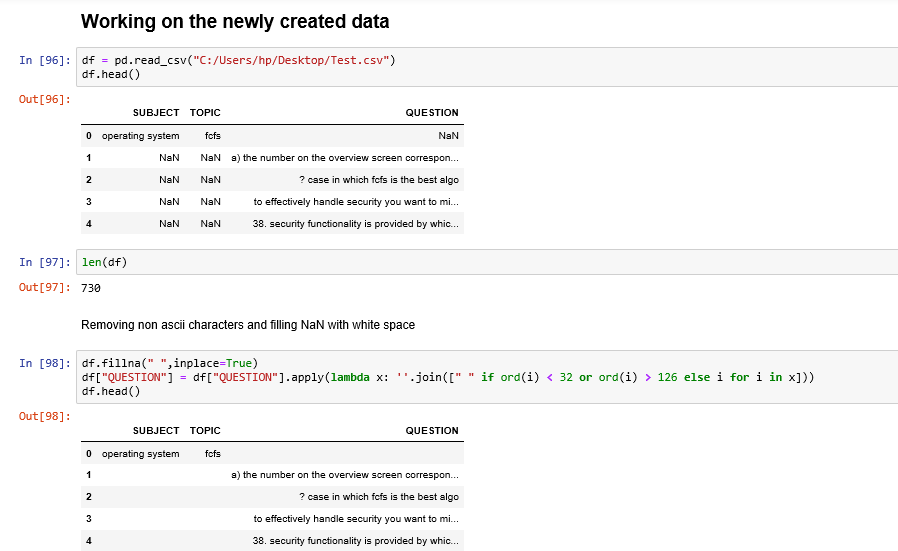
--------------------------------

We have extracted questions related to each topic. We will now find the frequency of words that occur in the set of questions for a particular topic and not the whole dataset.

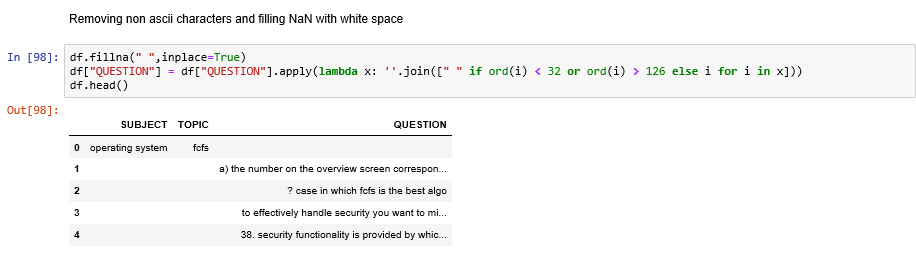
Imported necessary libraries



From the csv file created in FAQ.ipynb



Removing non ascii characters and filling the table with NaN

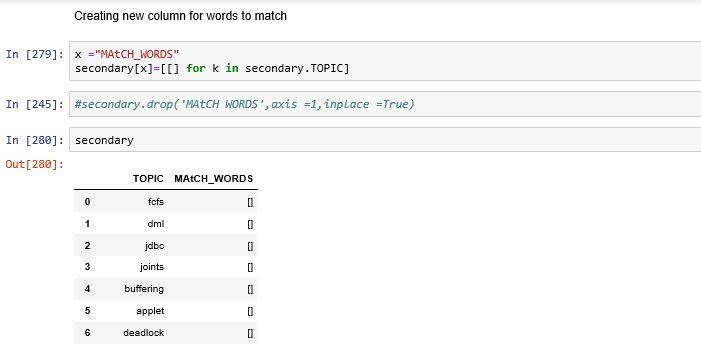


Removing numbers and punctuations. Also making each sentence into its vocabulary. All the questions related to a topic will be stored in the dictionary as list with key as topic name.



Dataframe of the topics

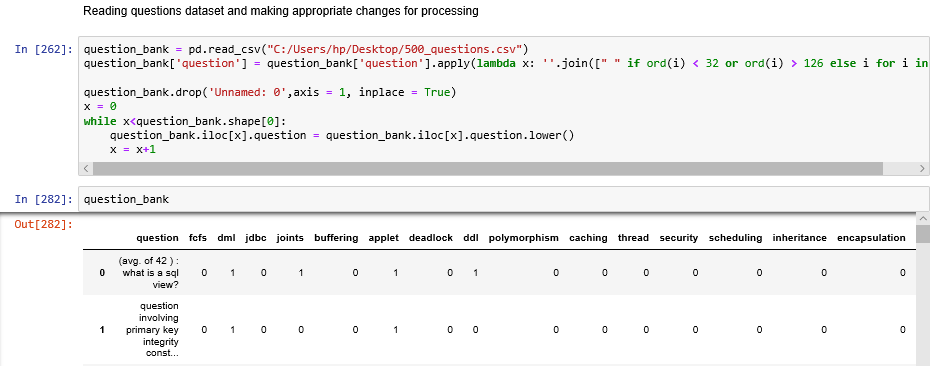
New column

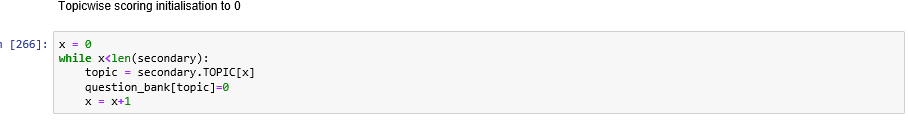


Finding the 10 most frequent words in the list for each topic



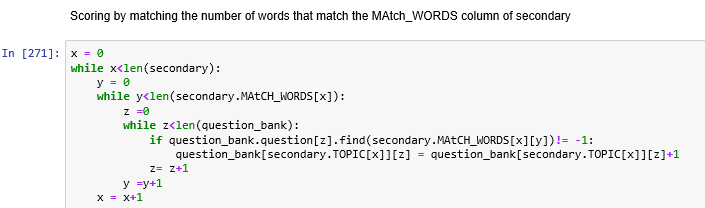
Making new columns(for scores of each topic) with the old question\_bank:



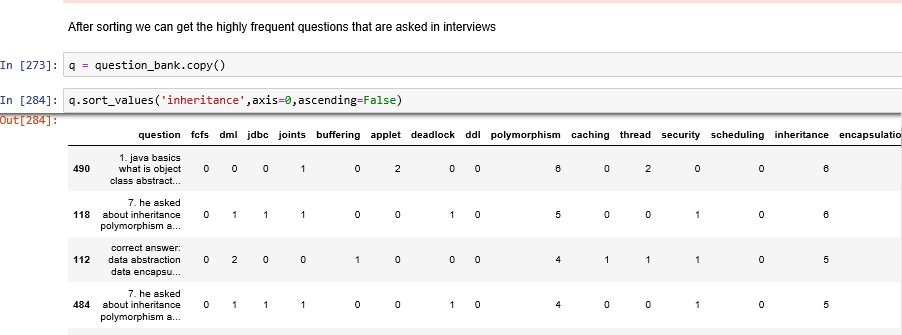
Initializing all new columns to 0  


Scoring algorithm:

The most frequent words are compared with the question bank. Each time a frequent word occurs in a question the respective topic’s score will increase by one



We can now sort in ascending order of the score for a particular topic and get the question which have highest no. of frequently asked words



IMPROVEMENTS THAT CAN BE MADE:

Stopwords set needs to be modified as there are few words in the frequently used words set that have irrelevant connection. For example, ‘questions’ and ‘question’ are some of the frequently that may not be relevant.