

REPORT

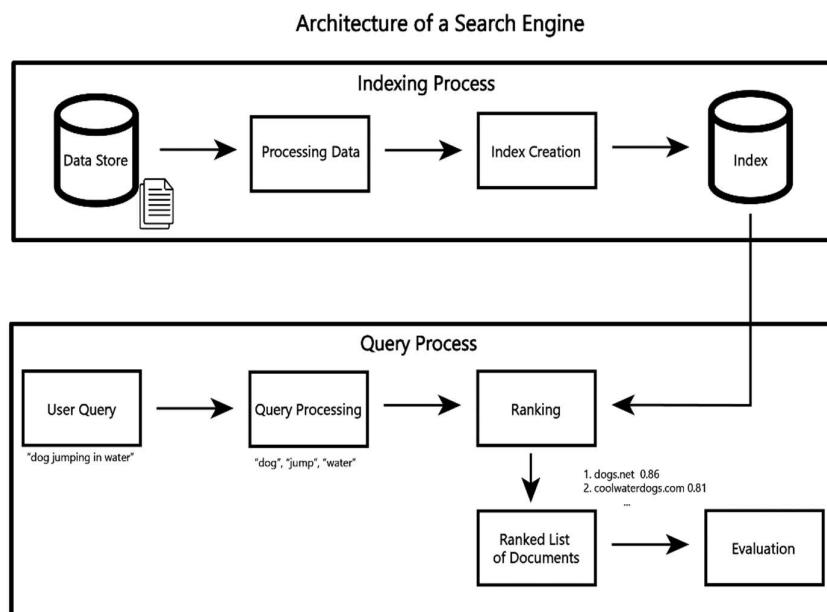
TITLE & CORPUS : Implementing Search Engine, using a “Russian Tweets” corpus

INTRODUCTION :

In today's digital age, the importance of search engines cannot be overstated. The ability to quickly and accurately retrieve relevant information from a vast pool of data is essential for individuals and organizations alike. The purpose of this project is to implement a search engine that incorporates both Boolean and similarity-based retrieval models. The search engine will use an inverted index to efficiently store and retrieve information. The text content will be pre-processed to improve the accuracy of the retrieval process, and the system will support phrase queries through the use of bi-word and positional indexing.

The search engine will be implemented using a relevant corpus, which is a collection of documents that are representative of the type of content that the search engine will be used to retrieve. The corpus should be chosen carefully to ensure that it is representative of the target domain and contains a sufficient amount of data to produce meaningful results.

The system will incorporate two retrieval models: the Boolean model and the similarity model. The Boolean model is a simple, binary retrieval model that retrieves documents based on the presence or absence of specific keywords or phrases. The similarity model, on the other hand, retrieves documents based on



their relevance to a given query, taking into account factors such as term frequency and inverse document frequency.

To improve the accuracy of the retrieval process, the text content will be pre-processed before the creation of the inverted index. This will involve techniques such as tokenization, stop word removal, stemming, and normalization. These techniques will help to ensure that the index contains only relevant information and that the retrieval process is more efficient.

To support phrase queries, the system will use a combination of bi-word indexing and positional indexing. Bi-word indexing involves breaking down phrases into pairs of adjacent words, while positional indexing takes into account the relative positions of the words in the phrase.

The retrieved documents will appear in a ranked manner, with the most relevant documents appearing at the top of the list. Precision and recall metrics will be used to evaluate the accuracy of the system.

In addition to the implementation of the search engine, a detailed analysis report will be submitted. This report will include a justification for the pre-processing techniques used, an explanation of the data structure chosen, and a justification for the similarity scoring scheme used for the Twitter dataset.

DATASET DESCRIPTION :

The Russian Troll Tweets dataset available at <https://github.com/fivethirtyeight/russian-troll-tweets.git> contains a collection of tweets that were posted by Russian trolls on Twitter between 2015 and 2017. The dataset was collected by the cybersecurity firm FireEye and contains over 3 million tweets that were posted by 2,752 Twitter accounts that have been linked to the Internet Research Agency, a Russian "troll farm."

The corpus used in our analysis is the "Russian Troll Tweets" dataset from the FiveThirtyEight repository on GitHub. The corpus size which was initially 1.4 GB reduced to 700-800 MB after the required filtering was done.

The dataset includes information such as the tweet ID, the date and time the tweet was posted, the Twitter account that posted the tweet, and the text of the tweet itself. The text of the tweets is in raw form, meaning that it has not been pre-processed or cleaned.

The dataset is relevant for the requirements of the project as it provides a large corpus of text that can be used to train and test the search engine. The text content of the tweets will need to be pre-processed to improve the accuracy of the retrieval process, and the system will need to be able to support phrase queries, as many of the tweets contain phrases that are relevant to the target domain.

Given the nature of the dataset, the choice of similarity scoring scheme is particularly important. The tweets contain a significant amount of noise and irrelevant information, so it will be important to choose a similarity scoring scheme that is robust to noise and can accurately identify relevant information. One possible approach would be to use a term frequency-inverse document frequency (TF-IDF) weighting scheme, which takes into account the frequency of terms in the corpus and the inverse document frequency of the terms to identify relevant documents.

PRE-PROCESSING :

The code performs the following preprocessing steps on the Russian Troll Tweets dataset:

1. Loads 13 .csv files containing the tweet data and concatenates them into a single DataFrame.
2. Filters the DataFrame to include only the relevant columns: 'external_author_id', 'content', 'language', and 'publish_date'.
3. Filters the DataFrame to include only English tweets.
4. Initializes a stemmer (SnowballStemmer) and set of stop words (from the nltk package).
5. Removes URLs, mentions, and hashtags from the tweet content using regular expressions.
6. Lowercases the text, removes stop words, and stems the remaining words.
7. Removes all non-alphanumeric characters from the tweet content.
8. Converts the tweet content to lowercase and removes stop words.

9. Resets the index of the DataFrame.
10. Saves the preprocessed DataFrame as a CSV file named 'preprocessed_data.csv'.
11. Prints the first 5 rows of the preprocessed DataFrame for inspection.

These preprocessing steps are aimed at cleaning and normalizing the text data in the tweets to improve the accuracy of the retrieval process. The steps include removing noise such as URLs, mentions, and hashtags, lowercasing the text, removing stop words, stemming the remaining words, and removing non-alphanumeric characters. The resulting preprocessed DataFrame contains only English tweets and can be used to build the search engine with the required features.

JUSTIFICATION OF PREPROCESSING TECHNIQUES USED

Pre-processing is a crucial step in any Search engine IR task, including information retrieval. In this assignment, the following pre-processing techniques were used:

1. **Tokenization:** The documents were tokenized into individual words to extract meaningful terms. Tokenization is the process of breaking a text into words, phrases, symbols, or other meaningful elements, called tokens. It helps in reducing the dimensionality of the data and makes it easier to work with. In our corpus, we have used the nltk library to tokenize the text.
2. **Stopword Removal:** Stop words are the commonly occurring words in a language like "the", "is", "and", etc. They do not add much meaning to the text and can be safely removed to reduce the dimensionality of the data. In our corpus, we have used the nltk library to remove the stop words from the text. Hence, these led to improve the accuracy and efficiency of the search algorithm.
3. **Stemming:** Stemming is the process of reducing inflected (or sometimes derived) words to their word stem, base or root form. It was used to

convert the words to their base form i.e. words that are similar and to reduce the dimensionality of the feature space.

4. **Lowercasing:** All the text was converted to lowercase to treat uppercase and lowercase versions of the same word as the same. This is done to reduce the sparsity of the data and to treat the same word in different cases as the same word. In our corpus, we have chosen to lowercase all the text as the case of the words is not important in the analysis.
5. **Punctuation Removal:** Punctuation marks and special characters like hashtags, mentions, URLs, and emoticons do not add much value in the analysis and can create noise in the data. Therefore, we have removed them from the text using regular expressions.

INDEX GENERATION DETAILS:

The code performs the following steps to generate the inverted index:

1. Initializes a defaultdict named `inverted_index` to store the inverted index. defaultdict is used to handle cases where a term does not exist in the inverted index yet, making it easier to add new terms and associated documents to the index.
2. Initializes a dictionary named `doc_id_mapping` to map document ids to their content.
3. Iterates through each row in the DataFrame `df` and extracts the document id, content, and relevant metadata (`external_author_id` and `publish_date`).
4. Splits the pre-processed content into terms and adds each term and its associated document id to the inverted index using the defaultdict.
5. Writes the inverted index to a text file named '`inverted_index.txt`'. The file format lists each term followed by a colon and a space, then the list of document ids containing that term separated by commas.
6. Prints the first 10 terms and their associated document ids to inspect the inverted index.

The inverted index is a data structure that maps each term in the corpus to the list of documents that contain the term. This makes it possible to retrieve documents that contain specific terms or combinations of terms. The inverted index is generated from the pre-processed text content of the documents in the Russian Troll Tweets dataset using a defaultdict to store the index, and a dictionary to map document ids to their content and metadata. The resulting index is written to a text file and can be used to build the search engine with the required features such as Boolean Model and retrieval based on similarity, supporting phrase queries (combination of Biword index and positional indexing), and ranking of retrieved documents based on relevance.

- **JUSTIFICATION FOR CHOOSING AN APPROPRIATE DATA STRUCTURE**
- We chose to represent the corpus as a list of strings, where each string represents a document. This is a simple and straightforward way to store the text data in a format that is easy to work with. It also allows us to perform basic operations on the entire corpus, such as counting the number of documents or iterating over all the documents.
- We used dictionary of sets data structure to store inverted indices, where the processed word was the key and the value was the list of set of documents. This gives high optimization, as it is index based.
- To transform the raw text into a format that can be used for information retrieval, we used the TF-IDF vectorizer from scikit-learn. This converts the text documents into a sparse matrix of TF-IDF weights, where each row represents a document and each column represents a unique term in the corpus. This data structure allows for efficient computation of cosine similarity scores between the query and all documents in the corpus.

- Overall, the choice of data structures for this task strikes a balance between simplicity and efficiency, and allows us to perform the necessary operations for information retrieval.

BOOLEAN MODEL DETAILS :

The Boolean Model is a classic information retrieval model based on set theory, which retrieves documents that match a query using Boolean operators (AND, OR, NOT) to combine terms.

To implement the Boolean Model, we first preprocess the corpus by tokenizing and cleaning the text data using regular expressions. Then, we create an inverted index data structure that maps each term to a set of documents that contain that term.

Next, given a query, we pre-process it and split it into terms. We then retrieve the sets of documents that contain each term in the query, and combine them using the specified Boolean operators. Finally, we return the text content of the matching documents.

The implementation of the Boolean Model can be divided into two main parts: pre-processing and the Boolean retrieval function.

In pre-processing, we tokenize and clean the text data, and create the inverted index.

In the Boolean retrieval function, we pre-process the query, split it into terms, retrieve the sets of documents that contain each term in the query, and combine them using the specified Boolean operators.

The Boolean retrieval function takes as input a query and the inverted index, pre-processes the query, retrieves the sets of documents that contain each term in the query, and combines them using the specified Boolean operators. Finally, it returns the text content of the matching documents.

Overall, the Boolean Model is a simple but effective method for information retrieval, and can be useful in cases where the user wants to retrieve documents that match specific combinations of terms.

“WILD CARD” AND “PHRASE QUERY” IMPLEMENTATION

DETAILS :

The code implements a search application that handles wild card queries and phrase queries.

The handle_wildcard_query function takes a query string as input, replaces any "" characters with a regular expression ".", compiles the resulting regular expression, and searches the inverted index for terms that match the regular expression. It then returns the set of document ids that contain any of the matching terms.

The handle_phrase_query function takes a query string as input, removes any URLs and punctuation, splits the remaining string into terms, and searches the corpus for documents that contain the terms in the same order as the original query. It then returns a list of document indices that match the query.

The calc_precision_recall function takes two sets of document ids, one representing the relevant documents and one representing the retrieved documents, and calculates the precision and recall of the retrieved documents. Precision is the fraction of retrieved documents that are relevant, while recall is the fraction of relevant documents that were retrieved.

The query_app function takes two query strings as input, one for a wild card query and one for a phrase query, and uses the handle_wildcard_query and handle_phrase_query functions to find the documents that match each query. It then prints the query and the matching document ids for each query.

The query_pr_app function is similar to query_app, but it also takes a list of relevant document ids as input, and it calculates and prints the precision and recall of the wild card and phrase queries using the calc_precision_recall function.

RANKING OF RETRIEVED RESULTS:

The ranking of retrieved results depends on the retrieval algorithm used. In this case, the two functions **retrieve_using_cosine_similarity** and **log_likelihood** use different algorithms to retrieve and rank the documents.

retrieve_using_cosine_similarity retrieves documents using the inverted index and calculates the cosine similarity between the query and candidate

documents. It then sorts the candidate documents by cosine similarity in descending order to get the top documents. The rank of each document is based on its cosine similarity score with the query terms.

log_likelihood retrieves documents that contain any of the query tokens and computes the likelihood of each retrieved document. It then ranks the retrieved documents by their likelihood. The rank of each document is based on its likelihood score.

Both functions provide a way to rank retrieved documents based on their relevance to the query, but the ranking criteria are different. Cosine similarity measures the similarity between the query and document vectors, while likelihood measures the probability of the query given the document. The choice of algorithm and ranking criteria depends on the specific task and dataset.

- **JUSTIFICATION FOR CHOOSING A TYPICAL SIMILARITY SCORING SCHEME**
- Cosine similarity was used as a similarity scoring scheme to compare the query with the documents. Cosine similarity is a measure of similarity between two non-zero vectors of an inner product space. It measures the cosine of the angle between two vectors in a multi-dimensional space.
- Cosine similarity is a popular choice for similarity scoring in information retrieval tasks, including document search and recommendation systems. It is easy to compute and does not require any assumptions about the distribution of the data.
- In our case, we are using TF-IDF vectors, which are a more sophisticated version of term-frequency vectors. TF-IDF takes into account not only the frequency of a term in a document but also the inverse frequency of the term in the entire corpus. By using cosine similarity with TF-IDF vectors, we can get a good measure of similarity between a query and a

document in the corpus. This makes cosine similarity an appropriate similarity score for our retrieval system.

ADVANCE FEATURE RELEVANCE FEEDBACK AND RERANKING OF RESULTS (ADDITIONAL FEATURES IMPLEMENTED) :

- This code defines a function that retrieves documents from a corpus using cosine similarity with relevance feedback and re-ranks the results using the Rocchio algorithm. The corpus is defined as a list of documents in a Data-Frame. The function takes a query, the number of documents to retrieve, and three parameters alpha, beta, and gamma that control the weights assigned to the original query, relevant documents, and non-relevant documents, respectively.
- The function first transforms the query using a TF-IDF vectorizer and calculates the cosine similarity between the query and all documents in the corpus. It then sorts the documents by cosine similarity and retrieves the top documents. The user is prompted to provide feedback on the relevance of each document, and the relevant and non-relevant documents are used to calculate a new query vector using the Rocchio algorithm. The cosine similarity is then calculated between the new query vector and all documents in the corpus, and the top documents are reranked based on the new cosine similarity scores.
- When the function is called with the query "donald trump", it retrieves the top 5 documents most similar to the query using cosine similarity and displays them along with the reasons for their high similarity scores. The user is prompted to provide feedback on the relevance of each document. Based on the feedback, a new query vector is calculated using the Rocchio algorithm, and the top 5 documents are reranked and displayed along with the reasons for their high similarity scores.
- Based on the code provided, the program uses TF-IDF vectorization and cosine similarity to retrieve documents that are most similar to a given query. It then asks the user for relevance feedback on the top documents and re-ranks them using the Rocchio algorithm to improve the search results.
- For the given query "donald trump", the program retrieves and displays the top 5 documents that are most similar to the query based on their

cosine similarity scores. It then asks the user to provide relevance feedback on each of the documents and re-ranks them using the Rocchio algorithm. Finally, it displays the top 5 reranked documents that are most similar to the new query.

```
# Preview the DataFrame  
print(df.head())
```

```
[2] ... external_author_id content \
0 90600000000000000000 sit democrat us senat trial corrupt bare heard...
1 90600000000000000000 marshawn lynch arriv game anti trump shirt jud...
2 90600000000000000000 daughter fallen navi sailor deliv power monolo...
3 90600000000000000000 presid trump dedic presid cup golf tournament ...
4 90600000000000000000 19 000 respect nation anthem standforouranthem

language publish_date
0 English 10/1/2017 19:58
1 English 10/1/2017 22:43
2 English 10/1/2017 22:50
3 English 10/1/2017 23:52
4 English 10/1/2017 2:13
```

- It is important to note that the success of this program depends on the quality of the corpus and the relevance of the user feedback. Additionally, there are other techniques and algorithms that can be used to improve the search results further, such as semantic analysis, query expansion, and machine learning models.

RESULTS :

Results are shown in the screenshots with appropriate docs retrieved with accurate tradeoffs between precision and recall.

SCREENSHOTS OF THE ASSIGNMENT ON EVALUATION

PARAMETERS

- ## 1. Pre-processing of raw data

```

results = boolean_model(query)
print(f"{len(results)} matching documents found:\n")
i = 0

# Printing top 25 docs with contents
for result in results:
    print(result + "\n")
    i += 1
    if(i == 25):
        break

```

[16] Python

... Output exceeds the [size limit](#). Open the full output data [in a text editor](#)

3397 matching documents found:

Leading Conservatives Predicted This Last Week: Obama Is Behind Illegal Leaks on Trump and Russia <https://t.co/LDk8u5RHSr>

#chaWITCH HUNT : Russia Probe Includes Tenants From Trump Tower <https://t.co/tSPCpp4vBT> <https://t.co/rt0Z1CJEjs>

NEW VIDEO: Trump still hasn't answered the Impeachable question. Did he or his minions coordinate with Russia? <https://t.co/aC>

A Democratic Congressman Just Caught Trump In 2 Massive Russia Lies <https://t.co/gpvlgsgsy> <https://t.co/PUSNUhUVUU>

This is the lie Donald Trump told- refuted by his own son- that makes me believe he is criminally tied to Russia. <https://t.co>

2. A) Generate Inverted Index (variation in data structures)

The complete inverted index file of our project is provided with this link : [Click Here](#)

2. B) Boolean Model :

[3] Python

... sit: [0, 78, 82, 1137, 3159, 3363, 4025, 4462, 5859, 6653, 7216, 7354, 7354, 7845, 8383, 8648, 10373, 10931, 11629, 11787, 1430, democrat: [0, 80, 88, 145, 191, 238, 250, 6364, 6421, 6535, 6710, 8742, 8842, 8847, 9124, 9285, 9287, 9330, 9419, 9519, 9536, 9611, us: [0, 11, 45, 45, 176, 193, 194, 303, 355, 356, 366, 438, 449, 487, 491, 503, 543, 566, 618, 768, 783, 796, 799, 805, 827, 1111, 1121, 1131, 1141, 1151, 1161, 1171, 1181, 1191, 1201, 1211, 1221, 1231, 1241, 1251, 1261, 1271, 1281, 1291, 1301, 1311, 1321, 1331, 1341, 1351, 1361, 1371, 1381, 1391, 1401, 1411, 1421, 1431, 1441, 1451, 1461, 1471, 1481, 1491, 1501, 1511, 1521, 1531, 1541, 1551, 1561, 1571, 1581, 1591, 1601, 1611, 1621, 1631, 1641, 1651, 1661, 1671, 1681, 1691, 1701, 1711, 1721, 1731, 1741, 1751, 1761, 1771, 1781, 1791, 1801, 1811, 1821, 1831, 1841, 1851, 1861, 1871, 1881, 1891, 1901, 1911, 1921, 1931, 1941, 1951, 1961, 1971, 1981, 1991, 2001, 2011, 2021, 2031, 2041, 2051, 2061, 2071, 2081, 2091, 2101, 2111, 2121, 2131, 2141, 2151, 2161, 2171, 2181, 2191, 2201, 2211, 2221, 2231, 2241, 2251, 2261, 2271, 2281, 2291, 2301, 2311, 2321, 2331, 2341, 2351, 2361, 2371, 2381, 2391, 2401, 2411, 2421, 2431, 2441, 2451, 2461, 2471, 2481, 2491, 2501, 2511, 2521, 2531, 2541, 2551, 2561, 2571, 2581, 2591, 2601, 2611, 2621, 2631, 2641, 2651, 2661, 2671, 2681, 2691, 2701, 2711, 2721, 2731, 2741, 2751, 2761, 2771, 2781, 2791, 2801, 2811, 2821, 2831, 2841, 2851, 2861, 2871, 2881, 2891, 2901, 2911, 2921, 2931, 2941, 2951, 2961, 2971, 2981, 2991, 3001, 3011, 3021, 3031, 3041, 3051, 3061, 3071, 3081, 3091, 3101, 3111, 3121, 3131, 3141, 3151, 3161, 3171, 3181, 3191, 3201, 3211, 3221, 3231, 3241, 3251, 3261, 3271, 3281, 3291, 3301, 3311, 3321, 3331, 3341, 3351, 3361, 3371, 3381, 3391, 3401, 3411, 3421, 3431, 3441, 3451, 3461, 3471, 3481, 3491, 3501, 3511, 3521, 3531, 3541, 3551, 3561, 3571, 3581, 3591, 3601, 3611, 3621, 3631, 3641, 3651, 3661, 3671, 3681, 3691, 3701, 3711, 3721, 3731, 3741, 3751, 3761, 3771, 3781, 3791, 3801, 3811, 3821, 3831, 3841, 3851, 3861, 3871, 3881, 3891, 3901, 3911, 3921, 3931, 3941, 3951, 3961, 3971, 3981, 3991, 4001, 4011, 4021, 4031, 4041, 4051, 4061, 4071, 4081, 4091, 4101, 4111, 4121, 4131, 4141, 4151, 4161, 4171, 4181, 4191, 4201, 4211, 4221, 4231, 4241, 4251, 4261, 4271, 4281, 4291, 4301, 4311, 4321, 4331, 4341, 4351, 4361, 4371, 4381, 4391, 4401, 4411, 4421, 4431, 4441, 4451, 4461, 4471, 4481, 4491, 4501, 4511, 4521, 4531, 4541, 4551, 4561, 4571, 4581, 4591, 4601, 4611, 4621, 4631, 4641, 4651, 4661, 4671, 4681, 4691, 4701, 4711, 4721, 4731, 4741, 4751, 4761, 4771, 4781, 4791, 4801, 4811, 4821, 4831, 4841, 4851, 4861, 4871, 4881, 4891, 4901, 4911, 4921, 4931, 4941, 4951, 4961, 4971, 4981, 4991, 5001, 5011, 5021, 5031, 5041, 5051, 5061, 5071, 5081, 5091, 5101, 5111, 5121, 5131, 5141, 5151, 5161, 5171, 5181, 5191, 5201, 5211, 5221, 5231, 5241, 5251, 5261, 5271, 5281, 5291, 5301, 5311, 5321, 5331, 5341, 5351, 5361, 5371, 5381, 5391, 5401, 5411, 5421, 5431, 5441, 5451, 5461, 5471, 5481, 5491, 5501, 5511, 5521, 5531, 5541, 5551, 5561, 5571, 5581, 5591, 5511, 5521, 5531, 5541, 5551, 5561, 5571, 5581, 5591, 5601, 5611, 5621, 5631, 5641, 5651, 5661, 5671, 5681, 5691, 5611, 5621, 5631, 5641, 5651, 5661, 5671, 5681, 5691, 5701, 5711, 5721, 5731, 5741, 5751, 5761, 5771, 5781, 5791, 5711, 5721, 5731, 5741, 5751, 5761, 5771, 5781, 5791, 5801, 5811, 5821, 5831, 5841, 5851, 5861, 5871, 5881, 5891, 5811, 5821, 5831, 5841, 5851, 5861, 5871, 5881, 5891, 5901, 5911, 5921, 5931, 5941, 5951, 5961, 5971, 5981, 5991, 5911, 5921, 5931, 5941, 5951, 5961, 5971, 5981, 5991, 6001, 6011, 6021, 6031, 6041, 6051, 6061, 6071, 6081, 6091, 6011, 6021, 6031, 6041, 6051, 6061, 6071, 6081, 6091, 6101, 6111, 6121, 6131, 6141, 6151, 6161, 6171, 6181, 6191, 6111, 6121, 6131, 6141, 6151, 6161, 6171, 6181, 6191, 6201, 6211, 6221, 6231, 6241, 6251, 6261, 6271, 6281, 6291, 6211, 6221, 6231, 6241, 6251, 6261, 6271, 6281, 6291, 6301, 6311, 6321, 6331, 6341, 6351, 6361, 6371, 6381, 6391, 6311, 6321, 6331, 6341, 6351, 6361, 6371, 6381, 6391, 6401, 6411, 6421, 6431, 6441, 6451, 6461, 6471, 6481, 6491, 6411, 6421, 6431, 6441, 6451, 6461, 6471, 6481, 6491, 6501, 6511, 6521, 6531, 6541, 6551, 6561, 6571, 6581, 6591, 6511, 6521, 6531, 6541, 6551, 6561, 6571, 6581, 6591, 6601, 6611, 6621, 6631, 6641, 6651, 6661, 6671, 6681, 6691, 6611, 6621, 6631, 6641, 6651, 6661, 6671, 6681, 6691, 6701, 6711, 6721, 6731, 6741, 6751, 6761, 6771, 6781, 6791, 6711, 6721, 6731, 6741, 6751, 6761, 6771, 6781, 6791, 6801, 6811, 6821, 6831, 6841, 6851, 6861, 6871, 6881, 6891, 6811, 6821, 6831, 6841, 6851, 6861, 6871, 6881, 6891, 6901, 6911, 6921, 6931, 6941, 6951, 6961, 6971, 6981, 6991, 6911, 6921, 6931, 6941, 6951, 6961, 6971, 6981, 6991, 7001, 7011, 7021, 7031, 7041, 7051, 7061, 7071, 7081, 7091, 7011, 7021, 7031, 7041, 7051, 7061, 7071, 7081, 7091, 7101, 7111, 7121, 7131, 7141, 7151, 7161, 7171, 7181, 7191, 7111, 7121, 7131, 7141, 7151, 7161, 7171, 7181, 7191, 7201, 7211, 7221, 7231, 7241, 7251, 7261, 7271, 7281, 7291, 7211, 7221, 7231, 7241, 7251, 7261, 7271, 7281, 7291, 7301, 7311, 7321, 7331, 7341, 7351, 7361, 7371, 7381, 7391, 7311, 7321, 7331, 7341, 7351, 7361, 7371, 7381, 7391, 7401, 7411, 7421, 7431, 7441, 7451, 7461, 7471, 7481, 7491, 7411, 7421, 7431, 7441, 7451, 7461, 7471, 7481, 7491, 7501, 7511, 7521, 7531, 7541, 7551, 7561, 7571, 7581, 7591, 7511, 7521, 7531, 7541, 7551, 7561, 7571, 7581, 7591, 7601, 7611, 7621, 7631, 7641, 7651, 7661, 7671, 7681, 7691, 7611, 7621, 7631, 7641, 7651, 7661, 7671, 7681, 7691, 7701, 7711, 7721, 7731, 7741, 7751, 7761, 7771, 7781, 7791, 7711, 7721, 7731, 7741, 7751, 7761, 7771, 7781, 7791, 7801, 7811, 7821, 7831, 7841, 7851, 7861, 7871, 7881, 7891, 7811, 7821, 7831, 7841, 7851, 7861, 7871, 7881, 7891, 7901, 7911, 7921, 7931, 7941, 7951, 7961, 7971, 7981, 7991, 7911, 7921, 7931, 7941, 7951, 7961, 7971, 7981, 7991, 8001, 8011, 8021, 8031, 8041, 8051, 8061, 8071, 8081, 8091, 8011, 8021, 8031, 8041, 8051, 8061, 8071, 8081, 8091, 8101, 8111, 8121, 8131, 8141, 8151, 8161, 8171, 8181, 8191, 8111, 8121, 8131, 8141, 8151, 8161, 8171, 8181, 8191, 8201, 8211, 8221, 8231, 8241, 8251, 8261, 8271, 8281, 8291, 8211, 8221, 8231, 8241, 8251, 8261, 8271, 8281, 8291, 8301, 8311, 8321, 8331, 8341, 8351, 8361, 8371, 8381, 8391, 8311, 8321, 8331, 8341, 8351, 8361, 8371, 8381, 8391, 8401, 8411, 8421, 8431, 8441, 8451, 8461, 8471, 8481, 8491, 8411, 8421, 8431, 8441, 8451, 8461, 8471, 8481, 8491, 8501, 8511, 8521, 8531, 8541, 8551, 8561, 8571, 8581, 8591, 8511, 8521, 8531, 8541, 8551, 8561, 8571, 8581, 8591, 8601, 8611, 8621, 8631, 8641, 8651, 8661, 8671, 8681, 8691, 8611, 8621, 8631, 8641, 8651, 8661, 8671, 8681, 8691, 8701, 8711, 8721, 8731, 8741, 8751, 8761, 8771, 8781, 8791, 8711, 8721, 8731, 8741, 8751, 8761, 8771, 8781, 8791, 8801, 8811, 8821, 8831, 8841, 8851, 8861, 8871, 8881, 8891, 8811, 8821, 8831, 8841, 8851, 8861, 8871, 8881, 8891, 8901, 8911, 8921, 8931, 8941, 8951, 8961, 8971, 8981, 8991, 8911, 8921, 8931, 8941, 8951, 8961, 8971, 8981, 8991, 9001, 9011, 9021, 9031, 9041, 9051, 9061, 9071, 9081, 9091, 9011, 9021, 9031, 9041, 9051, 9061, 9071, 9081, 9091, 9101, 9111, 9121, 9131, 9141, 9151, 9161, 9171, 9181, 9191, 9111, 9121, 9131, 9141, 9151, 9161, 9171, 9181, 9191, 9201, 9211, 9221, 9231, 9241, 9251, 9261, 9271, 9281, 9291, 9211, 9221, 9231, 9241, 9251, 9261, 9271, 9281, 9291, 9301, 9311, 9321, 9331, 9341, 9351, 9361, 9371, 9381, 9391, 9311, 9321, 9331, 9341, 9351, 9361, 9371, 9381, 9391, 9401, 9411, 9421, 9431, 9441, 9451, 9461, 9471, 9481, 9491, 9411, 9421, 9431, 9441, 9451, 9461, 9471, 9481, 9491, 9501, 9511, 9521, 9531, 9541, 9551, 9561, 9571, 9581, 9591, 9511, 9521, 9531, 9541, 9551, 9561, 9571, 9581, 9591, 9601, 9611, 9621, 9631, 9641, 9651, 9661, 9671, 9681, 9691, 9611, 9621, 9631, 9641, 9651, 9661, 9671, 9681, 9691, 9701, 9711, 9721, 9731, 9741, 9751, 9761, 9771, 9781, 9791, 9711, 9721, 9731, 9741, 9751, 9761, 9771, 9781, 9791, 9801, 9811, 9821, 9831, 9841, 9851, 9861, 9871, 9881, 9891, 9811, 9821, 9831, 9841, 9851, 9861, 9871, 9881, 9891, 9901, 9911, 9921, 9931, 9941, 9951, 9961, 9971, 9981, 9991, 9911, 9921, 9931, 9941, 9951, 9961, 9971, 9981, 9991, 10001, 10011, 10021, 10031, 10041, 10051, 10061, 10071, 10081, 10091, 10011, 10021, 10031, 10041, 10051, 10061, 10071, 10081, 10091, 10101, 10111, 10121, 10131, 10141, 10151, 10161, 10171, 10181, 10191, 10111, 10121, 10131, 10141, 10151, 10161, 10171, 10181, 10191, 10201, 10211, 10221, 10231, 10241, 10251, 10261, 10271, 10281, 10291, 10211, 10221, 10231, 10241, 10251, 10261, 10271, 10281, 10291, 10301, 10311, 10321, 10331, 10341, 10351, 10361, 10371, 10381, 10391, 10311, 10321, 10331, 10341, 10351, 10361, 10371, 10381, 10391, 10401, 10411, 10421, 10431, 10441, 10451, 10461, 10471, 10481, 10491, 10411, 10421, 10431, 10441, 10451, 10461, 10471, 10481, 10491, 10501, 10511, 10521, 10531, 10541, 10551, 10561, 10571, 10581, 10591, 10511, 10521, 10531, 10541, 10551, 10561, 10571, 10581, 10591, 10601, 10611, 10621, 10631, 10641, 10651, 10661, 10671, 10681, 10691, 10611, 10621, 10631, 10641, 10651, 10661, 10671, 10681, 10691, 10701, 10711, 10721, 10731, 10741, 10751, 10761, 10771, 10781, 10791, 10711, 10721, 10731, 10741, 10751, 10761, 10771, 10781, 10791, 10801, 10811, 10821, 10831, 10841, 10851, 10861, 10871, 10881, 10891, 10811, 10821, 10831, 10841, 10851, 10861, 10871, 10881, 10891, 10901, 10911, 10921, 10931, 10941, 10951, 10961, 10971, 10981, 10991, 10911, 10921, 10931, 10941, 10951, 10961, 10971, 10981, 10991, 11001, 11011, 11021, 11031, 11041, 11051, 11061, 11071, 11081, 11091, 11011, 11021, 11031, 11041, 11051, 11061, 11071, 11081, 11091, 11101, 11111, 11121, 11131, 11141, 11151, 11161, 11171, 11181, 11191, 11111, 11121, 11131, 11141, 11151, 11161, 11171, 11181, 11191, 11201, 11211, 11221, 11231, 11241, 11251, 11261, 11271, 11281, 11291, 11211, 11221, 11231, 11241, 11251, 11261, 11271, 11281, 11291, 11301, 11311, 11321, 11331, 11341, 11351, 11361, 11371, 11381, 11391, 11311, 11321, 11331, 11341, 11351, 11361, 11371, 11381, 11391, 11401, 11411, 11421, 11431, 11441, 11451, 11461, 11471, 11481, 11491, 11411, 11421, 11431, 11441, 11451, 11461, 11471, 11481, 11491, 11501, 11511, 11521, 11531, 11541, 11551, 11561, 11571, 11581, 11591, 11511, 11521, 11531, 11541, 11551, 11561, 11571, 11581, 11591, 11601, 11611, 11621, 11631, 11641, 11651, 11661, 11671, 11681, 11691, 11611, 11621, 11631, 11641, 11651, 11661, 11671, 11681, 11691, 11701, 11711, 11721, 11731, 11741, 11751, 11761, 11771, 11781, 11791, 11711, 11721, 11731, 11741, 11751, 11761, 11771, 11781, 11791, 11801, 11811, 11821, 11831, 11841, 11851, 11861, 11871, 11881, 11891, 11811, 11821, 11831, 11841, 11851, 11861, 11871, 11881, 11891, 11901, 11911, 11921, 11931, 11941, 11951, 11961, 11971, 11981, 11991, 11911, 11921, 11931, 11941, 11

3. Handling wild card and phrase queries Query passed as (“WILDCARD QUERY”, “PHRASE QUERY”)

```
# Passing queries
query_app("trump*", "donald trump")
query_app("krishna*", "hare krishna")
query_app("russia*", "russia moscow")
```

Python

```
Wild card query: trump*, matching doc ids: {524288, 1, 3, 5, 1572870, 524295, 8, 1048584, 12, 524300, 1572878, 1572879, 16, 52
Phrase query: donald trump, matching doc ids: [5, 198, 242, 262, 555, 1433, 1870, 2100, 4010, 5874, 5893, 6115, 6152, 6387, 65
Wild card query: krishna*, matching doc ids: {934921, 812042, 1308181, 1163291, 446493, 830497, 403491, 836643, 58407, 201770,
Phrase query: hare krishna, matching doc ids: [201770, 359272, 423479]
Wild card query: russia*, matching doc ids: {1048592, 1048593, 1572891, 29, 393248, 33, 1048618, 1048620, 655417, 1572923, 104
Phrase query: russia moscow, matching doc ids: [387251, 946212, 1435440, 1661301, 1691724, 1774579, 1892940, 2007453, 2018460]
```

```
# Sample queries for precision and recall :-
wildcard_query = 'trump'
phrase_query = 'donald trump'

# Sample relevant documents
relevant_docs = [5, 198, 242, 262, 555, 1433, 1870, 2100, 2270, 2736, 2750, 4010, 5874, 5893, 6115, 6152, 6387, 6511, 6687,
# Perform queries and evaluate
query_pr_app(wildcard_query, phrase_query, relevant_docs)
```

Python

```
Wild card query: trump*, matching doc ids: {524288, 1, 3, 5, 1572870, 524295, 8, 1048584, 12, 524300, 1572878, 1572879, 16, 52
Phrase query: donald trump, matching doc ids: [5, 198, 242, 262, 555, 1433, 1870, 2100, 4010, 5874, 5893, 6115, 6152, 6387, 65
---
Evaluation:
Number of relevant documents: 111
Wild card query precision: 0.00028750638088141244, recall: 0.44144144144144143
Phrase query precision: 0.0029934144881261227, recall: 0.40540540540540543
```

4. Retrieve relevant text using similarity index

```
# Example usage
query = 'hare krishna'
top_documents, precision, recall = retrieve_using_cosine_similarity(query, num_docs = 5)
print(f"Precision: {precision:.4f}")
print(f"Recall: {recall:.4f}")

query = 'vladimir putin'
top_documents, precision, recall = retrieve_using_cosine_similarity(query, num_docs = 5)
print(f"Precision: {precision:.4f}")
print(f"Recall: {recall:.4f}")

1 Showing top 5 documents that are most similar to the query 'hare krishna':
2
3 Rank 1 (Cosine Similarity: 0.2016):
4 Document ID: 201770
5 Growing Hare Krishna congregation to open $3.6 million temple in Baltimore County https://t.co/cT0k9snZs6 https://t.co/cT0k9snZs6
6 Reason: The document has a high cosine similarity score with the query terms.
7
8 Rank 2 (Cosine Similarity: 0.1677):
9 Document ID: 423479
10 When I re enter my body upon waking, the first two words out my mouth are mother goddess (Hare) & Krishna.
11 Reason: The document has a high cosine similarity score with the query terms.
12
13 Rank 3 (Cosine Similarity: 0.1567):
14 Document ID: 446493
15 Me And My VP Are On A Mission From God, Or Mohammed, Or Krishna ... Or Oprah. #RejectedPrezCampaignSlogans http://t.co/446493
16 Reason: The document has a high cosine similarity score with the query terms.
17
18 Rank 4 (Cosine Similarity: 0.1516):
19 Document ID: 845461
20 .@SimpsonArnold1 SimpsonArnold @Joessweetpea Krishna @ifanaureylian Sofeefee @bowiescheekbone bourgeoisophie http://t.co/845461
21 Reason: The document has a high cosine similarity score with the query terms.
22
23 Rank 5 (Cosine Similarity: 0.1440):
24 Document ID: 1905328
25 #Miami Conviction upheld for Krishna Maharaj in 1986 double murder #local
26 Reason: The document has a high cosine similarity score with the query terms.

27
28 Precision: 0.4000
29 Recall: 0.6667
30 Showing top 5 documents that are most similar to the query 'vladimir putin':
31
32 Rank 1 (Cosine Similarity: 1.0000):
33 Document ID: 245945
34 Vladimir Putin. https://t.co/lHbXvTEBXU
35 Reason: The document has a high cosine similarity score with the query terms.
36
37 Rank 2 (Cosine Similarity: 0.5064):
38 Document ID: 1616422
39 Vladimir Putin – not THAT Vladimir Putin – nabbed in Florida
40 Reason: The document has a high cosine similarity score with the query terms.
41
42 Rank 3 (Cosine Similarity: 0.4249):
43 Document ID: 1550504
44 Trump on Vladimir Putin: 'We've had some very, very good talks' https://t.co/EnTavPfFrA https://t.co/UVT07yweXw
45 Reason: The document has a high cosine similarity score with the query terms.
46
47 Rank 4 (Cosine Similarity: 0.4127):
48 Document ID: 1056003
49 Vladimir Putin: A look at Russia's president https://t.co/tqHPyy6gKn #news \| https://t.co/3d02FUvg2D https://t.co/3d02FUvg2D
50 Reason: The document has a high cosine similarity score with the query terms.
```

```
49 Vladimir Putin: A look at Russia's president https://t.co/tqHPyy6gKn #news \| https://t.co/3d02FUVg2D https://t.co/3d02FUVg2D
50 Reason: The document has a high cosine similarity score with the query terms.
51
52 Rank 5 (Cosine Similarity: 0.3723):
53 Document ID: 314695
54 Your new friend Vladimir Putin, GOP... https://t.co/XeaTUsX55H
55 Reason: The document has a high cosine similarity score with the query terms.
56
57 Precision: 1.0000
58 Recall: 0.0133
59 |
```

5. Retrieve relevant text using likelihood language model

```
# Define the query
query = "harekrishna"
log_likelihood(query, 5, [740353, 1593358, 814038, 836643, 840009, 840013, 28564, 567158, 934921, 1078196, 1380963, 1498746
print("-----")
query = "vladimir putin"
log_likelihood(query, 5, [306, 307, 481, 481, 482, 487, 495, 498, 513, 516, 518, 711642, 714984, 715043, 715760, 716572, 716
print("-----")
query = "petrol price"
log_likelihood(query, 5, [], 0)
```

Python

```
1 Document ID: 1593358
2 Likelihood: -1.9459
3 Jamaica and Haiti especially, we haven't forgotten you. #Prayers #HareKrishna #108Plus
4 relevant
5
6 Document ID: 740353
7 Likelihood: -2.0794
8 When you touch ground and the Bhagavad Gita touches a heart. #HareKrishna #Everywhere https://t.co/S3CHRAYGvZ
9 relevant
10
11 Precision: 0.4000
12 Recall: 0.1538
13 -----
14 Document ID: 961860
15 Likelihood: -0.3466
16 By Putin? https://t.co/Rt82GwlKtJ
17 not relevant
18
19 Document ID: 1300311
20 Likelihood: -0.3466
21 Сейчас в Омске #Putin https://t.co/X7tAHMAuhj
22 not relevant
23
24 Document ID: 245945
25 Likelihood: -0.6931
26 vladimir Putin. https://t.co/lHbXvTEBUX
```

```
26 vladimir Putin. https://t.co/lHbXvTEBUX
27 not relevant
28
29 Document ID: 311547
30 Likelihood: -0.8959
31 Putin during the debate https://t.co/G7515hBTwv
32 not relevant
33
34 Document ID: 1017681
35 Likelihood: -0.8959
36 Trump to Putin:
37 not relevant
38
39 Precision: 0.0000
40 Recall: 0.0000
41 -----
42 Document ID: 1167943
43 Likelihood: -0.3466
44 What's your price? https://t.co/t0vPp2UjXm
45 not relevant
46
47 Document ID: 1299082
48 Likelihood: -0.3466
49 What's your price? https://t.co/t0vPp2UjXm
50 not relevant
51
52 Document ID: 512755
53 Likelihood: -0.3466
54 What's your price? https://t.co/t0vPp2UjXm
55 not relevant
56
57 Document ID: 898368
58 Likelihood: -0.3466
59 What's your price? https://t.co/t0vPp2UjXm
60 not relevant
61
62 Document ID: 1169224
63 Likelihood: -0.3466
64 What's your price? https://t.co/t0vPp2UjXm
65 not relevant
66
67
```

6. Ranking of retrieved documents

The ranking is done in TASK 3, 4 and 5

7. Advanced search: relevance feedback, semantic matching, reranking of results, finding out query intention

```

# Print the reranked top documents
print(f"\nShowing top {num_docs} reranked documents that are most similar to the query '{query}':\n")
for i, (text, cosine_sim) in enumerate(top_documents):
    print(f"Rank {i+1} (Cosine Similarity: {cosine_sim:.4f}):")
    print(text)
    print("Reason: The document has a high cosine similarity score with the reranked query.\n")

retrieve_using_cosine_similarity_with_feedback("dональд трамп")

```

Output exceeds the [size limit](#). Open the full output data [in a text editor](#)
 Showing top 5 documents that are most similar to the query 'dональд трамп':

```

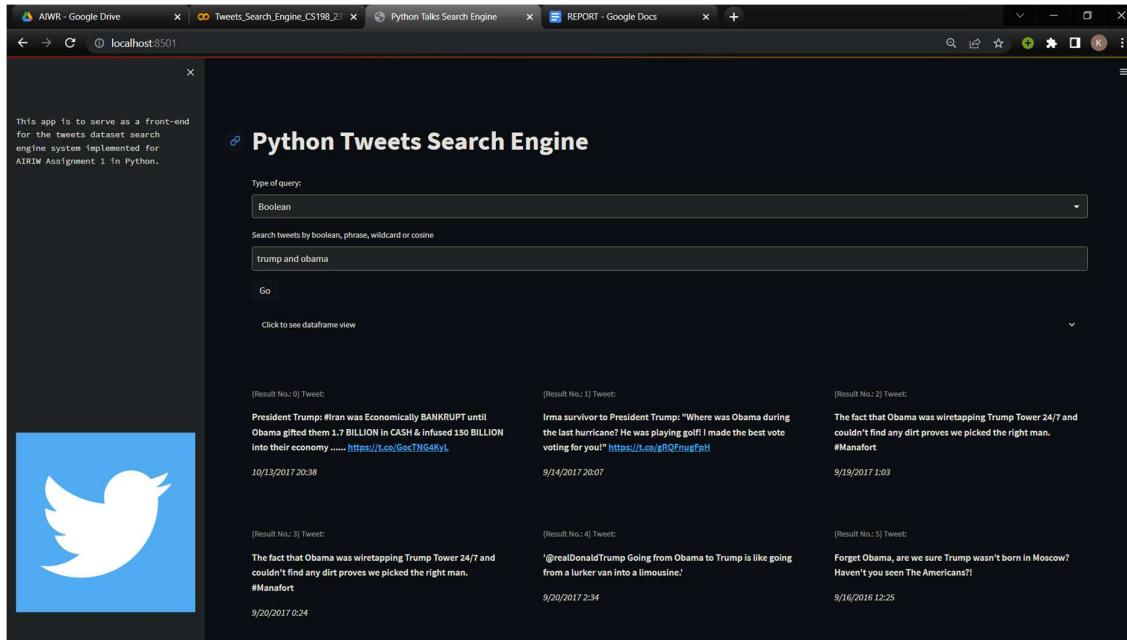
1 Showing top 5 documents that are most similar to the query 'dональд трамп':
2
3 Rank 1 (Cosine Similarity: 1.0000):
4 DONald Trump
5 Reason: The document has a high cosine similarity score with the query.
6
7 Rank 2 (Cosine Similarity: 1.0000):
8 or Donald Trump https://t.co/h5R6WGnykE
9 Reason: The document has a high cosine similarity score with the query.
10
11 Rank 3 (Cosine Similarity: 1.0000):
12 Donald Trump by Donald Trump
13 Reason: The document has a high cosine similarity score with the query.
14
15 Rank 4 (Cosine Similarity: 0.8506):
16 The Donald
17 Reason: The document has a high cosine similarity score with the query.
18
19 Rank 5 (Cosine Similarity: 0.8380):
20 Donald Trump Threatens To Sue Donald Trump For Being Donald Trump https://t.co/e0Ml3GbFYv #donaldtrump https://t.co/XADkm5W0p7
21 Reason: The document has a high cosine similarity score with the query.
22
23 Is document 1 relevant? (y/n): n
24 Document 1 was not marked as relevant.
25 Is document 2 relevant? (y/n): y
26 Is document 3 relevant? (y/n): n
27
28 Document 2 was marked as relevant.
29 Document 3 was not marked as relevant.
30 Is document 4 relevant? (y/n): y
31 Document 4 was marked as relevant.
32 Is document 5 relevant? (y/n): n
33 Document 5 was not marked as relevant.
34
35 Showing top 5 reranked documents that are most similar to the query 'dональд трамп':
36
37 Rank 1 (Cosine Similarity: 0.9919):
38 or Donald Trump https://t.co/h5R6WGnykE
39 Reason: The document has a high cosine similarity score with the reranked query.
40
41 Rank 2 (Cosine Similarity: 0.9919):
42 DONald Trump
43 Reason: The document has a high cosine similarity score with the reranked query.
44
45 Rank 3 (Cosine Similarity: 0.9919):
46 Donald Trump by Donald Trump
47 Reason: The document has a high cosine similarity score with the reranked query.
48
49 Rank 4 (Cosine Similarity: 0.9100):
50 The Donald
51 Reason: The document has a high cosine similarity score with the reranked query.
52
53 Rank 5 (Cosine Similarity: 0.8216):
54 Donald Trump Threatens To Sue Donald Trump For Being Donald Trump https://t.co/e0Ml3GbFYv #donaldtrump https://t.co/XADkm5W0p7
55 Reason: The document has a high cosine similarity score with the reranked query.
56

```

GUI for the Search Engine

As part of the assignment, we have implemented a minimalist GUI for the search engine in Streamlit, which can be opened via a browser, that supports the basic operations an end-user looking to search tweets can use. Screenshots depicting the GUI follow this:

Boolean queries:



This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.

Python Tweets Search Engine

Type of query:

Boolean

Search tweets by boolean, phrase, wildcard or cosine

trump and obama

Go

Click to see data frame view

external_author_id	content	language	publish_date
43	President Trump: Iran was Economically BANKRUPT until Obama gifted them 1.7 BILLION in CASH & infis	English	10/13/2017 20:38
153	Irma survivor to President Trump: "Where was Obama during the last hurricane? He was playing golf! I made th	English	9/14/2017 20:07
219	The fact that Obama was wiretapping Trump Tower 24/7 and couldn't find any dirt proves we picked the right n	English	9/19/2017 1:03
238	The fact that Obama was wiretapping Trump Tower 24/7 and couldn't find any dirt proves we picked the right n	English	9/20/2017 2:24
259	'@realDonaldTrump Going from Obama to Trump is like going from a lurker van to a limousine.'	English	9/20/2017 2:34
484	Forget Obama, are we sure Trump wasn't born in Moscow? Haven't you seen The Americans??!	English	9/16/2016 12:25
8955	#abi Obama Left Trump a WELL FUNDED NUCLEAR MESS in Iran https://t.co/SelipKgMnC #ABISS https://t.co/L	English	8/13/2017 16:28
9074	#abi WHOA Trump Just Shredded the HELL out of Obama https://t.co/Bdy7rx3Ax #ABISS https://t.co/QUWage	English	8/15/2017 17:14
9236	#abi COMPARISON VIDEO: Trump Condemns Hate While Obama Defends it https://t.co/4vtb5mMMeo #ABISS In	English	8/18/2017 5:59
9450	#abi Massive Obama Era Food Stamp Fraud EXPOSED and ENDED By Trump! https://t.co/1sXkXzubB# #ABISS In	English	8/9/2017 2:28

Twitter logo

Windows taskbar

This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.

(Result No.: 6) Tweet:

#abi Obama Left Trump a WELL FUNDED NUCLEAR MESS in Iran https://t.co/SelipKgMnC #ABISS https://t.co/L1Nr#mtWYv

8/13/2017 16:28

(Result No.: 7) Tweet:

#abi WHOA Trump Just Shredded the HELL out of Obama https://t.co/Bdy7rx3Ax #ABISS https://t.co/QUWage7nL

8/15/2017 17:14

(Result No.: 8) Tweet:

#abi COMPARISON VIDEO: Trump Condemns Hate While Obama Defends it https://t.co/4vtb5mMMeo #ABISS https://t.co/1dCUWBRRN

8/18/2017 5:59

(Result No.: 9) Tweet:

#abi Massive Obama Era Food Stamp Fraud EXPOSED and ENDED By Trump! https://t.co/1sXkXzubB# #ABISS https://t.co/KorWbPYSk

8/8/2017 2:28

(Result No.: 10) Tweet:

king_covefieTuckerCarlson Did u hold Obamas Administration accountable? Like your bio says about Trump? What did you do about Iran deal?

8/8/2017 3:26

(Result No.: 11) Tweet:

Obama and the Left Push for Hate Against the Police. Trump is Pushing Back! https://t.co/3XE3WdULr https://t.co/Wsz40TXY

7/30/2017 11:05

(Result No.: 12) Tweet:

BOOM! Trump ERASES Another Piece of Obama's Failed Legacy https://t.co/4MSay98tVt https://t.co/bxFc8PGiBE

7/30/2017 3:00

(Result No.: 13) Tweet:

Look Who's Calling on Trump to Investigate Obama's IRS https://t.co/NP9um3g27S https://t.co/8tdsyhICaYs

8/1/2017 23:15

(Result No.: 14) Tweet:

Trump Just CRUSHED Obama's First 6 Months Fundraising Number! https://t.co/ZKIRlyOhpG https://t.co/NMRhL4dp11

8/2/2017 21:40

(Result No.: 15) Tweet:

Trump Corrects Obama and Tells the World Who the REAL Dreamers are https://t.co/cmb5BPARao https://t.co/9EapPCxjol

8/2/2017 21:40

(Result No.: 16) Tweet:

Gen McMaster is FIRING Trump Loyalists While PROTECTING Obama Loyalists https://t.co/5Z1svd5DFp https://t.co/y7hrCv6DF8

8/3/2017 21:35

(Result No.: 17) Tweet:

#acapa Obama Left Trump a WELL FUNDED NUCLEAR MESS in Iran! https://t.co/YirRk8eMnDp https://t.co/yGvzhd4Tmw

8/13/2017 16:28

Twitter logo

Windows taskbar

This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.

7/30/2017 3:00 8/1/2017 23:15 8/2/2017 21:40

(Result No.: 15) Tweet:
Trump Corrects Obama and Tells the World Who the REAL Dreamers are <https://t.co/cmbSBPRAro> <https://t.co/lBEqPCXioL>

(Result No.: 16) Tweet:
Gen McMaster is FIRING Trump Loyalists While PROTECTING Obama Loyalists <https://t.co/sZisvdSDP> <https://t.co/y7brCvAFB>

(Result No.: 17) Tweet:
#acapa Obama Left Trump a WELL FUNDED NUCLEAR MESS in Iran! <https://t.co/YirRiseMnDp> <https://t.co/vGvh64Tmw>

8/2/2017 21:40 8/3/2017 21:35 8/13/2017 16:28

(Result No.: 18) Tweet:
#acapa WHOA! Trump Just Shredded the HELL out of Obama <https://t.co/04e3KcBqj> <https://t.co/pCo31bmSXh>

(Result No.: 19) Tweet:
#acapa COMPARISON VIDEO: Trump Condemns Hate While Obama Defends It <https://t.co/yNEXnToOXC> <https://t.co/wq2ectw6bh>

(Result No.: 20) Tweets:
#acapa Food Stamps Under Trump VS Obama – The Results Will SHOCK YOU <https://t.co/XWILEBTepG> <https://t.co/VqlobWaaf2>

8/15/2017 17:10 8/18/2017 6:06 8/8/2017 18:54

(Result No.: 21) Tweet:
mackette52 glennbeck But I was upset that he didn't support Trump at first. He didn't know him...but he hated obama Hillary.

(Result No.: 22) Tweet:
Trump continues to overturn key Obama-era landmarks, enraging Native American and environmental protectionists. <https://t.co/5728y1yKWA>

(Result No.: 23) Tweet:
Israel settlements grew on Obama's watch. They may be poised for a boom on Trump's. <https://t.co/yTm8RvIEWj> via @theMEU

8/9/2017 2:51 8/30/2017 7:06 1/4/2017 7:39



This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.

SEARCH: president or trump

Click to see dataframe view

10/1/2017 22:43 10/1/2017 23:52 10/1/2017 2:47

(Result No.: 0) Tweet:
Marshawn Lynch arrives to game in anti-Trump shirt. Judging by his sagging pants the shirt should say Lynch vs. belt <https://t.co/mJH13gj72>

(Result No.: 1) Tweet:
JUST IN: President Trump dedicates Presidents Cup golf tournament trophy to the people of Florida, Texas and Puerto Rico. <https://t.co/izwVa4d0jAE>

(Result No.: 2) Tweet:
Dan Bongino: "Nobody trolls liberals better than Donald Trump." Exactly! <https://t.co/bigV3saC8j>

10/1/2017 3:47 10/10/2017 20:57 10/10/2017 23:42

(Result No.: 3) Tweet:
As much as I hate promoting CNN article, here they are admitting EVERYTHING Trump said about PR relief two days ago. <https://t.co/lZmSeA48oh>

(Result No.: 4) Tweet:
Sarah Sanders destroys NBC reporter: "Trump made a joke, maybe you guys should get a sense of humor." <https://t.co/1coJ8scE9G>

(Result No.: 5) Tweet:
First lady Melania Trump visits infant opioid treatment center in Huntington, West Virginia. We love your grace and style @FLOTUS! ❤️ ❤️ <https://t.co/hLAkPSOr0d>

(Result No.: 6) Tweet:
Because he hates women so much, President Trump just nominated another woman, Kirstjen Nielsen, to be in his Cabinet.

(Result No.: 7) Tweet:
Pres. Trump: "We want lower taxes, bigger paychecks, and more jobs for American truckers and for American workers." <https://t.co/gZc8mvpays>

(Result No.: 8) Tweet:
Pres. Trump just dropped a NUKE-truth-bomb: "The NFL should have suspended Kaepernick for one game and he would have never done it again" <https://t.co/qkYPwhsglu>



This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.

(Result No.: 9) Tweet:
So we've gone from "Russia literally hacked the election for Trump" to "Russians helped Trump get elected by hacking Pokemon GO." Amazing! <https://t.co/5IN05y54U>
10/12/2017 20:01

(Result No.: 10) Tweet:
Eminem is a Trump supporter!! 😊 <https://t.co/nCUJ7senU>
10/12/2017 21:38

(Result No.: 11) Tweet:
A cute 10 yr girl had a Trump dress on. Hannity asked her why she liked him. She said because he's doing great things for the country! ❤️ <https://t.co/QzXsuPgH28>
10/12/2017 20:02

(Result No.: 12) Tweet:
President Trump: #Iran was Economically BANKRUPT until Obama gifted them 1.7 BILLION in CASH & infused 150 BILLION into their economy <https://t.co/GocTNG4KyL>
10/13/2017 20:38

(Result No.: 13) Tweet:
After posting their support of President Trump, Arizona restaurant forced to close down due to death threats. This is the "peaceful" left. <https://t.co/GPPQCSYNe>
10/14/2017 1:09

(Result No.: 14) Tweet:
#FakeNews MSM has been out to get President Trump since he was a candidate. Be warned—We The People are fighting back & you're going down.
10/14/2017 2:09

(Result No.: 15) Tweet:
President Trump tweeted this back in 2012. <https://t.co/HZ8dJlykXt>
10/14/2017 2:21

(Result No.: 16) Tweet:
President Trump tweeted this back in 2012. <https://t.co/IW8sy7vD7>
10/14/2017 2:26

(Result No.: 17) Tweet:
President Trump and first lady Melania Trump lead White House staff in a moment of silence for #LasVegas. <https://t.co/SnRtLaTHc>
10/2/2017 19:24

(Result No.: 18) Tweet:
@CNN in hysterics as Puerto Rico FULLY EMBRACES President Trump w/ Love and Cheers of Praise !! Fake News Narrative CRUSHED <https://t.co/Mn8Jt7IO98>
10/3/2017 21:51

(Result No.: 19) Tweet:
Pres. Trump to Las Vegas first responders: "You showed the world—and the world is watching—and you showed what professionalism is all about" <https://t.co/fwSHainh>
10/4/2017 19:18

(Result No.: 20) Tweet:
Las Vegas victim meets Trump, stands up on wounded leg: "I will always respect my country, my President, my flag!" You won't see this on CNN <https://t.co/zHEFAS1Lyv>
10/5/2017 0:51

This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.

Python Tweets Search Engine

Type of query: Boolean
Search tweets by boolean, phrase, wildcard or cosine
trump and clinton and not obama
Go
Click to see data frame view

(Result No.: 0) Tweet:
Top GOP official Ed Rollins says Trump's White House Chief of Staff John Kelly voted for Hillary Clinton <https://t.co/RD31OBQwww>
9/21/2017 1:09

(Result No.: 1) Tweet:
"What Clinton & Trump need to do to win 'epic' debate." @feldmike & I go behind what happens in debate prep [@cosnews">https://t.co/UXFNUseM">@cosnews](https://t.co/UXFNUseM)
9/23/2016 15:25

(Result No.: 2) Tweet:
Donald Trump has said he wants to ban Muslims, but Hillary Clinton has already bombed Muslims. #tytlive
10/22/2016 13:28

(Result No.: 3) Tweet:
21 out of 21 white players - Trump ✅ 20 out of 22 black players - Clinton ✅ #BRmag NFL straw poll <https://t.co/5H9srXQePC>
10/6/2016 17:50

(Result No.: 4) Tweet:
New #Clinton Ad Featuring Cast Of 'Empire' Targets #Trump, Highlights The Importance Of Voting <https://t.co/EK8AZNThz> @NewsOneNow
10/7/2016 0:04

(Result No.: 5) Tweet:
Trump on 1993 suicide of top Clinton aide Vince Foster: "Very fishy" <https://t.co/14rAnqm7ow> <https://t.co/jm8eCNBax>
5/24/2016 13:19

This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.

How Trump could beat Clinton: <https://t.co/q4CoMQboFB>
<https://t.co/7o4d4lsTa5k>

5/8/2016 13:38

Trump collected \$3.1 million for his campaign in May... about what Clinton gets in 3 days: <https://t.co/9KsF4qL4Qb>
<https://t.co/HM0iVPh75j>

6/21/2016 12:54

They're not saying they want Trump, but that Clinton isn't a given. Work to be done there, Dems. @elonjames @MsPuckyetti @HillaryClinton

9/5/2016 16:15

(Result No.: 9) Tweet:
Clinton, Trump, Brexit, Terror #2016In4Words
<https://t.co/17z8P>

12/28/2016 17:09

(Result No.: 10) Tweet:
Graham Argues Clinton's Crimes Should Be Overlooked Since She Was Trump's "Rival" <https://t.co/lB9f1oqcVT>
<https://t.co/hrfbqnMm01>

7/26/2017 16:11

(Result No.: 11) Tweet:
This woman is all of us, as she reacts to Trump calling Clinton the "devil": #Debates #Debates2016 <https://t.co/fNDLDvawJX>
<https://t.co/10t06Dw>

10/10/2016 17:55

(Result No.: 12) Tweet:
Things got ugly — and then uglier: Top takeaways from Clinton-Trump II: The two nominees squared ... [#BogardThat](https://t.co/h7jHx0fr)
<https://t.co/1o0iV>

10/10/2016 5:51

(Result No.: 13) Tweet:
Clinton has unprecedented 20-point advantage over Trump among college-educated white women: <https://t.co/V5oQgGPivV> <https://t.co/64xF5sxyvf>

10/9/2016 11:47

(Result No.: 14) Tweet:
Clinton has unprecedented 20-point advantage over Trump among college-educated white women: <https://t.co/V5oQgGPivV> <https://t.co/64xF5sxyvf>

11/18/2016 7:07

(Result No.: 15) Tweet:
Trump's Proving That Those Who Called Clinton "The Lesser Of Two Evils" Didn't Know What Evil Looked Like <https://t.co/cB6Cxc8egJ>

11/23/2016 11:26

(Result No.: 16) Tweet:
Clinton has unprecedented 20-point advantage over Trump among college-educated white women: <https://t.co/V5oQgGPivV> <https://t.co/64xF5sxyvf>

12/30/2016 21:07

(Result No.: 17) Tweet:
Bernie Sanders has more potential than Hillary Clinton to beat Trump <https://t.co/uRmbmavE5>
<https://t.co/c3UjqcXmsy>

5/15/2016 14:49

Phrase Queries:

This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.

Type of query:

Search tweets by boolean, phrase, wildcard or cosine

Go

Click to see data frame view

(Result No.: 0) Tweet:
#adee RT Jen_L_yam: BREAKING NEWS: Vladimir Putin Retaliates After New Sanctions Are Slapped On Russia <https://t.co/D9yAPln> #rtt

7/30/2017 23:02

(Result No.: 1) Tweet:
Professor Stephen Cohen: Vladimir Putin is potentially America's most valuable national security asset <https://t.co/j8afJVn0Mh> <https://t.co/pDAlygB0>

12/24/2016 21:57

(Result No.: 2) Tweet:
Funny how WikiLeaks only seems to target people who oppose Vladimir Putin. What an incredible coincidence. <https://t.co/gP8sfhLiUD>

13/19/2016 7:13

(Result No.: 3) Tweet:
American Neo-Nazis Believe Vladimir Putin is the Heralded Savior of White People—Here's Why <https://t.co/SHWG9WudDy>

12/21/2016 12:01

(Result No.: 4) Tweet:
Vladimir Putin's cataclysmic endgame once elevating Donald Trump proves to have been a mistake <https://t.co/pMvFh3qsnV> via @PalmerReport

3/4/2017 9:03

(Result No.: 5) Tweet:
Retweeted John Aravosis (@aravosis): Said Vladimir Putin. <https://t.co/zpemijSn3R> <https://t.co/HASac81xob>

7/31/2017 0:56

This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.

UPDATE: NYPD says it's taken down the Vladimir Putin banner unfurled over Manhattan Bridge <https://t.co/wv1LPdfQ8j> <https://t.co/MfQnHKCKb7>

10/6/2016 22:48

(Result No.: 9) Tweet:

•@SeanHannity: "How did [Barack Obama and Hillary Clinton] ever think it's a good idea to give 20% of America's uranium to Vladimir Putin?" <https://t.co/dCe0thfB>

10/27/2017 2:07

(Result No.: 10) Tweet:

•Has Vladimir Putin killed more people than #HillaryForPrison? [@maddow #Arkanicide #LegitimatePresident](https://t.co/maddow)

1/4/2017 5:31

(Result No.: 11) Tweet:

RT SethAbramson: (THREAD) In 2002, Donald Trump crowned Vladimir Putin's girlfriend Miss Universe. Now it's possible that Mueller's....

8/18/2017 5:35

(Result No.: 12) Tweet:

US sends special forces to Russian border as NATO is poised to strike back against Vladimir Putin's 'aggression' <https://t.co/SILTPtUH>

1/4/2017 19:25

(Result No.: 13) Tweet:

U.S. leaders insult Vladimir Putin too much, Trump says in Politico interview <https://t.co/Od3bufeXF> <https://t.co/JUTjMyzka>

10/18/2016 6:51

(Result No.: 14) Tweet:

A banner with the Russian & Syrian flags & Vladimir Putin in a suit with the phrase "Peace Maker" appeared on Manhattan Bridge in #N#USA <https://t.co/lHMf0MJuJz>

10/6/2016 22:06

(Result No.: 15) Tweet:

A very large Vladimir Putin banner has appeared hung on the Manhattan bridge. This is so #HighEnergy. PRAISE KEK ☹️ <https://t.co/RWkE5DIA9k>

10/6/2016 22:10

(Result No.: 16) Tweet:

#Nedelya conducted a poll among its readers ➤ 84% of Ukrainian desired Vladimir Putin as President of #Ukraine Only 1% voted for #Poroshenko <https://t.co/q3wTljdq4E>

11/15/2016 2:20

(Result No.: 17) Tweet:

"A Society Which Cannot Defend Its Own Children Has No Future". President Vladimir Putin Shreds European Elites <https://t.co/xyawUcd8ba>

11/27/2016 6:37



This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.

1/4/2017 18:25

10/18/2016 6:51

10/6/2016 22:06

(Result No.: 15) Tweet:

A very large Vladimir Putin banner has appeared hung on the Manhattan bridge. This is so #HighEnergy. PRAISE KEK ☹️ <https://t.co/RWkE5DIA9k>

10/6/2016 22:10

(Result No.: 16) Tweet:

#Nedelya conducted a poll among its readers ➤ 84% of Ukrainian desired Vladimir Putin as President of #Ukraine Only 1% voted for #Poroshenko <https://t.co/q3wTljdq4E>

11/15/2016 2:20

(Result No.: 17) Tweet:

"A Society Which Cannot Defend Its Own Children Has No Future". President Vladimir Putin Shreds European Elites <https://t.co/xyawUcd8ba>

11/27/2016 6:37

(Result No.: 18) Tweet:

April 2006: DNC Planned Fake Donald Trump/Vladimir Putin Narrative #DNCLeak2 <https://t.co/bvb1Bgtx> <https://t.co/5Iaomvnp4>

11/7/2016 8:08

(Result No.: 19) Tweet:

The Russian Government, Russia, The Russians, Vladimir Putin, The Kremlin Russian Hack #FakeNews Blame the RUSSIANS! <https://t.co/4ad2aezeZm>

12/10/2016 14:06

(Result No.: 20) Tweet:

Vladimir Putin, Bashar al Assad one of these days you will pay for what you're doing to the helpless people of Aleppo

12/15/2016 2:18

(Result No.: 21) Tweet:

Key ally of Vladimir Putin: Shadowy NATO forces masterminded the assassination of Russian ambassador Andrey Karlov! <https://t.co/oSRycOnndD>

12/20/2016 10:37

(Result No.: 22) Tweet:

#Russia <https://t.co/bzvYQPKWg> TRUMP RELEASES PUTIN LETTER: Russia's Vladimir Putin Asks for Better Relationship with U.S.

12/24/2016 6:00

(Result No.: 23) Tweet:

WATCH - Vladimir Putin DESTROYS Trump-Hating Reporters With a BRILLIANT Question <https://t.co/DUw1Bg01c8> <https://t.co/c9LEPjh1st>

12/26/2016 12:12



This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRIM Assignment 1 in Python.

Type of query:
Phrase
Search tweets by boolean, phrase, wildcard or cosine

barack obama

Go

Click to see dataframe view

(Result No.: 0) Tweet:
Hiroshima survivors welcome Barack Obama visit
<https://t.co/wsaJDCvU1>
5/11/2016 13:44

(Result No.: 1) Tweet:
The pressed Barack Obama (to renounce me) and he gave in.
#SD17CHI #Barakhan
2/26/2017 17:56

(Result No.: 2) Tweet:
Rosa Parks sat so Martin Luther could walk., Martin Luther walked so Barack Obama could run Barack Obama ran so all the children could fly.
3/30/2017 21:53

(Result No.: 3) Tweet:
Barack Obama now has his own holiday 🎉 Starting in 2015, "Barack Obama Day" will be observed in Illinois
<https://t.co/hgnEGanwE1> <https://t.co/sYU2UErsl>
8/7/2017 17:28

(Result No.: 4) Tweet:
#Recap @Ashanti Joining First Lady Michelle Obama & Our President Barack Obama During The Final...
<https://t.co/JwVzGuA0e>
10/7/2016 0:11

(Result No.: 5) Tweet:
Dr Cornel West on Barack Obama. #blacklivesmatter
#drcornelwest <https://t.co/TZQeMfkis8>
6/7/2016 12:56

(Result No.: 6) Tweet:
[empty]

(Result No.: 7) Tweet:
[empty]

(Result No.: 8) Tweet:
[empty]

Windows Search Task View Google Chrome ENG 7:22 PM

This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRIM Assignment 1 in Python.

RT mmwtweet2: Barack Obama Negotiates With #NorthKorea...#Wuss realDonaldTrump
<https://t.co/qgDkGyc1d>
8/12/2017 0:03

A young Barack Obama as a pirate with his mother Ann Dunham. <https://t.co/Gws5Ep0N0W7> <https://t.co/yuyle78soV>
4/23/2017 2:32

Barack Obama You're Fired #TrumpCampaignSlogans
<https://t.co/1dm8sxDjf>
7/25/2015 15:29

(Result No.: 9) Tweet:
I believe that Barack Obama is absolutely the worst President that this great nation has ever seen in 250 years.
#SlaveryOfSpeech
2/25/2015 18:28

(Result No.: 10) Tweet:
RT TheresaMechelle: Barack Obama made everything about race for eight years starting with the attack on a white Pol...
<https://t.co/Bgfd0mko>
8/13/2017 12:19

(Result No.: 11) Tweet:
RT TheresaMechelle: Retweet For Barack Obama and Hillary Clinton To Be Prosecuted For Treason! Requests For Help De...
<https://t.co/MIU4riQaFy>
8/13/2017 5:13

(Result No.: 12) Tweet:
RT TheresaMechelle: Retweet If You Agree Barack Obama Should Be Tried For Treason! Obama Knew Bergdahl Was A Traitor... <https://t.co/Cx1cMoV88>
8/13/2017 5:45

(Result No.: 13) Tweet:
Barack Obama made everything about race for eight years starting with the attack on a white Police Officer. Obama...
<https://t.co/Bgfd0mko>
8/13/2017 6:49

(Result No.: 14) Tweet:
Where was the MSM and Dems calling out Barack Obama when he refused to name RADICAL ISLAMIC TERRORISM? The hypocrisy of the Left is rich.
8/14/2017 6:44

(Result No.: 15) Tweet:
RT TheresaMechelle: Barack Obama made everything about race for eight years starting with the attack on a white Pol...
<https://t.co/Bgfd0mko>
8/17/2017 0:19

(Result No.: 16) Tweet:
RT TheresaMechelle: Where was the MSM and Dems calling out Barack Obama when he refused to name RADICAL ISLAMIC TERRORISM? The hypocrisy of ...
8/18/2017 20:09

(Result No.: 17) Tweet:
RT TheresaMechelle: Retweet For Barack Obama and Hillary Clinton To Be Prosecuted For Treason! Requests For Help De...
<https://t.co/MIU4riyzsY>
8/18/2017 20:09

Windows Search Task View Google Chrome ENG 7:23 PM

This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.

Date	Content	Date	Content	Date	Content
8/13/2017 5:45	(Result No.: 15) Tweet: RT TheresaMechle: Barack Obama made everything about race for eight years starting with the attack on a white Pol... https://t.co/BYGrfd0mko	8/13/2017 6:49	(Result No.: 16) Tweet: RT TheresaMechle: Where was the MSM and Dems calling out Barack Obama when he refused to name RADICAL ISLAMIC TERRORISM? The hypocrisy of ... https://t.co/MIU4riQafy	8/14/2017 6:44	(Result No.: 17) Tweet: RT TheresaMechle: Retweet For Barack Obama and Hillary Clinton To Be Prosecuted For Treason! Requests For Help De... https://t.co/MIU4riQafy
8/17/2017 0:19	(Result No.: 18) Tweet: RT TheresaMechle: Retweet For Barack Obama and Hillary Clinton To Be Prosecuted For Treason! Requests For Help De... https://t.co/MIU4riQafy	8/18/2017 20:09	(Result No.: 19) Tweet: RT TheresaMechle: Retweet For Barack Obama and Hillary Clinton To Be Prosecuted For Treason! Requests For Help De... https://t.co/MIU4riQafy	8/18/2017 20:09	(Result No.: 20) Tweet: BOOK: The Communist: Frank Marshall Davis, Untold Story of Barack Obama's Mentor https://t.co/HdokigbSy https://t.co/cpozg02RemV
8/8/2017 0:57	(Result No.: 21) Tweet: BARACK OBAMA POSES THE GREATEST THREAT TO NATIONAL SECURITY https://t.co/0f110e3spE	8/9/2017 7:53	(Result No.: 22) Tweet: "Putin felt that Hillary Clinton would be more aggressive than Barack Obama." - Former U.S. diplomat James Jeffrey https://t.co/QaApzDrU0C	8/9/2017 7:53	(Result No.: 23) Tweet: Was Barack Obamas a GOP Plant All Along? - Even since the outgoing president, Barack Obama, helped orchestrate o... https://t.co/hfRb83iur
12/13/2015 2:32	12/15/2016 20:26	12/21/2016 10:25	12/21/2016 10:25	12/21/2016 10:25	12/21/2016 10:25

Wildcard query:

Type of query: Wildcard

Search tweets by boolean, phrase, wildcard or cosine
national*

Go

Click to see data-frame view

Date	Content	Date	Content	Date	Content
9/16/2016 15:02	(Result No.: 0) Tweet: "Nobody but us!" Putin, LePen and Trump on a poster for Russian nationalist candidate Maria Katsanova. https://t.co/POLdkx2iha	10/21/2016 12:29	(Result No.: 1) Tweet: https://t.co/D6p57dPtek #FeesMustFall #NationalShutdown https://t.co/kSQnN9q6S	11/16/2015 20:49	(Result No.: 2) Tweet: #TSA is probably the weirdest government agency. The fuck we paying for? #TakingToyFromSyo #NationalSecurity https://t.co/jCnyE7emZ
11/26/2015 20:13	(Result No.: 3) Tweet: @ShankKing @nationalbar @AttorneyCrump @robertloerzel they should be fired! https://t.co/4hsmIB71A	12/4/2015 19:45	(Result No.: 4) Tweet: Here is a #badcop cookie #NationalCookieDay #policestate #LEGO #MarioWoods https://t.co/q7hsmlB71A	5/18/2016 23:20	(Result No.: 5) Tweet: This is #NationalPoliceWeek. Dear cops, stop killing innocent people. Just fuck off. Thank you for the attention https://t.co/qKUPD0MUQ

This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.



[Result No.: 6] Tweet:
It's in the arch of my back, The sun of my smile The ride of my breasts, The grace of my style I'm a woman – Maya Angelou
#NationalPoetryDay
10/6/2016 23:04

[Result No.: 7] Tweet:
Classic. #NationalBestFriendsDay <https://t.co/FBG1O4NxC>
6/8/2016 14:09

[Result No.: 8] Tweet:
#NationalSelfieDay Hello you sexy beasts
<https://t.co/pSyohf4WBm>
6/21/2016 15:56

[Result No.: 9] Tweet:
Powerful poetry presented by poet Jackie Hill
@JackieHillPerry - 'JIG-A-BOO' <https://t.co/j7ZB84108v>
#NationalPoetryDay.
10/7/2016 11:38

[Result No.: 10] Tweet:
The emerging white nationalist alliance between European neo-Nazis, Russia, and Trump officials needs a name. Hmm: Axis of evil?
12/22/2016 5:31

[Result No.: 11] Tweet:
'@GregMontfort Exactly how will white ethno-nationalist prevent Asians/Blacks/Jews/Hispanics from entering your "white-ethno-states"?'
7/20/2017 16:03

[Result No.: 12] Tweet:
"Stephen Miller attacks Statue of Liberty poem, echoing popular white nationalist talking point" by rebekahentlago <https://t.co/h2jhvNCg01>
8/3/2017 11:18

[Result No.: 13] Tweet:
I hope you will read this. White nationalist Stephen Miller spoke from the West Wing yesterday, and confidently delivered gibberish. <https://t.co/Bz2WqLqRlk>
8/3/2017 11:18

[Result No.: 14] Tweet:
#NationalPoetryDay have to hail my fav living poet. Chosen this on fellow wordsmith M. Ali - Ali Rap. @MrChuckD <https://t.co/vwmtTvs2JE>
10/6/2016 12:59

[Result No.: 15] Tweet:
#NationalPoetryDay Respect to the late great #MuhammadAli
👉👉👉👉👉 <https://t.co/2PvDE70jV1>
10/6/2016 21:56

[Result No.: 16] Tweet:
Ok white nationalist. <https://t.co/oAOpk6eybF>
8/23/2016 17:14

[Result No.: 17] Tweet:
#NationalPoetryDay - Shoutout to 'Ye <https://t.co/AMQLwicZ4u>
https://t.co/utkMfTvs2JE
10/6/2016 19:59

This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.



Type of query:
Wildcard

Search tweets by boolean, phrase, wildcard or cosine
post*

Go

Click to see dataframe view

[Result No.: 0] Tweet:
After posting their support of President Trump, Arizona restaurant forced to close down due to death threats. This is the "peaceful" left. <https://t.co/GPQOCsFIyNe>
10/14/2017 1:09

[Result No.: 1] Tweet:
"Nobody but us!" Putin, LePen and Trump on a poster for Russian nationalist candidate Maria Katsanova. <https://t.co/FOLokA2lh4>
9/16/2016 15:02

[Result No.: 2] Tweet:
Russian Blogger Nosik Faces 2-Year Sentence for Syria Post: <https://t.co/NW49Um83lc>
9/19/2016 12:44

[Result No.: 3] Tweet:
Just posted a photo <https://t.co/YufgBaayk1>
9/19/2016 10:04

[Result No.: 4] Tweet:
#BlackLivesMatter St. Paul wants to charge cop who posted "Run them over" in advance of protest they held <https://t.co/Rt7cXzuJzf>
1/23/2016 5:18

[Result No.: 5] Tweet:
4 those who r out there in #Ferguson RN I love U U can make me happy by posting pic of u w/ a #4mysquad sign #IOIOIS #MillionManMarch
10/11/2015 0:06

This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRM Assignment 1 in Python.



(Result No.: 6) Tweet:
If your posting smoking pics or vids you better have one rolled when I see you.
10/19/2015 0:37

(Result No.: 7) Tweet:
Lit artwork via <https://t.co/Hs2hDvs0R8> #BlackLivesMatter
#Comics #Poster <https://t.co/GZqfVZCKn4>
10/20/2015 0:33

(Result No.: 8) Tweet:
.deray everytime you post this I really proud of us
#BlackLivesMatter
10/21/2015 14:13

(Result No.: 9) Tweet:
Florida deputies arrest 9 y.o. boy and post mugshot online, prompting criticism and cheers #BlackLivesMatter #scab
<https://t.co/hTRxeSjPyC>
10/21/2015 7:31

(Result No.: 10) Tweet:
I will never be tired to post it #PoliceBrutality #cops
#BlackLivesMatter <http://t.co/jpy44swFII>
10/4/2015 9:29

(Result No.: 11) Tweet:
Elizabeth Warren just gave the speech that #BlackLivesMatter activists have been waiting for - The Washington Post
#SmartNews #BlackTwitter
10/5/2015 16:41

(Result No.: 12) Tweet:
From Another Unoriginal poem about #PoliceBrutality Poster designed by Annie Banks #BlackLivesMatter #BlackTwitter
<http://t.co/Vd8kWwECNs>
10/8/2015 13:40

(Result No.: 13) Tweet:
@Xpressionable @menchthekid @AFRICaIsMankind @ReverantRevan @BluntStatement @F_M_D_B I just wanna post some pics <http://t.co/5CtJy3izP1>
10/8/2015 2:41

(Result No.: 14) Tweet:
I'll keep posting the photos of #BlackBeauty Her name is Analicia Chaves. Truly gorgeous. <https://t.co/U2LSZE7zz>
4/22/2016 1:00

(Result No.: 15) Tweet:
Miami police union targets woman for posting video of alleged police brutality... #BlackLivesMatter #Miami #Cops
<https://t.co/OiywXbjqjk5>
8/18/2015 11:49

(Result No.: 16) Tweet:
@NatvNewYorkr Oh now you telling me what to post about,'
8/19/2015 16:20

(Result No.: 17) Tweet:
'@ShaunKing there's no need to post dis video...All ppl know wat happen'd.
8/26/2015 16:35

(Result No.: 18) Tweet:
'@hogrider30 are ya gonna teach me ??? how to post or how to live?'
8/29/2015 11:03

(Result No.: 19) Tweet:
I will post more on this on my forum tonight, since this is genuinely interesting and something I think people should know. #ChurchPrivilege
9/18/2015 12:36

(Result No.: 20) Tweet:
'@PostThickyThick @SweetAs_Kai your all so so so perfect shaped, beautiful and happy on this pic, cant stop staring at it'
9/21/2015 21:21

(Result No.: 21) Tweet:
.@deray It's great.I am thankful that you post the most fresh notices about our Black Family movement.I so proud to be black.I proud of us
9/27/2015 11:57

(Result No.: 22) Tweet:
I posted a new video to Facebook <https://t.co/7cid0D2rfp>
10/18/2016 18:03

(Result No.: 23) Tweet:
I posted a new video to Facebook <https://t.co/9XNw4TzFv7>
10/18/2016 18:03

This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRM Assignment 1 in Python.



10/8/2015 13:40

(Result No.: 15) Tweet:
Miami police union targets woman for posting video of alleged police brutality... #BlackLivesMatter #Miami #Cops
<https://t.co/OiywXbjqjk5>
8/18/2015 11:49

10/8/2015 2:41

(Result No.: 16) Tweet:
@NatvNewYorkr Oh now you telling me what to post about,'
8/19/2015 16:20

(Result No.: 17) Tweet:
'@ShaunKing there's no need to post dis video...All ppl know wat happen'd.
8/26/2015 16:35

(Result No.: 18) Tweet:
'@hogrider30 are ya gonna teach me ??? how to post or how to live?'
8/29/2015 11:03

(Result No.: 19) Tweet:
I will post more on this on my forum tonight, since this is genuinely interesting and something I think people should know. #ChurchPrivilege
9/18/2015 12:36

(Result No.: 20) Tweet:
'@PostThickyThick @SweetAs_Kai your all so so so perfect shaped, beautiful and happy on this pic, cant stop staring at it'
9/21/2015 21:21

(Result No.: 21) Tweet:
.@deray It's great.I am thankful that you post the most fresh notices about our Black Family movement.I so proud to be black.I proud of us
9/27/2015 11:57

(Result No.: 22) Tweet:
I posted a new video to Facebook <https://t.co/7cid0D2rfp>
10/18/2016 18:03

(Result No.: 23) Tweet:
I posted a new video to Facebook <https://t.co/9XNw4TzFv7>
10/18/2016 18:03

Cosine:

This screenshot shows the Python Tweets Search Engine interface. The search query "Cosine" has been entered. The results are as follows:

- (Result No.: 0) Tweet:
Russia Russia Russia Russia Russia Russia Russia Russia
Russia Russia Russia Russia Russia Russia Russia Russia
Russia Russia! <https://t.co/TIPUVKowFW>
2/28/2017 8:53
- (Result No.: 1) Tweet:
Russia Russia Russia Russia Russia Russia Russia Russia
Russia Russia Russia Russia Russia Russia Russia Russia
Russia Russia! <https://t.co/RVQymMXUv>
2/28/2017 8:53
- (Result No.: 2) Tweet:
#Russia ??? <https://t.co/dCbPyhPB0y>
8/2/2017 7:57
- (Result No.: 3) Tweet:
Russia Russia Russia <https://t.co/VXoxPmUomZ>
12/15/2016 10:02
- (Result No.: 4) Tweet:
Russia Owns Trump?
12/29/2017 17:42
- (Result No.: 5) Tweet:
Russia! Why Russia? Why now? <https://t.co/USiZdy1Tz7>
3/19/2017 7:50

This screenshot shows the Python Tweets Search Engine interface. The search query "hare krishna" has been entered. The results are as follows:

- (Result No.: 0) Tweet:
These are called Arctic Hares <https://t.co/aZIecCirQK>
4/9/2017 7:46
- (Result No.: 1) Tweet:
Growing Hare Krishna congregation to open \$3.6 million
temple in Baltimore County <https://t.co/cT0k8sn7s6>
<https://t.co/pHEBRVcvYf>
4/28/2017 11:13
- (Result No.: 2) Tweet:
Harry's thighs are fat and ugly he needs lose weight
@yonceswaneoel @MacDaddyAllii @Krishna_TheSage
@alysssamariah <https://t.co/ITLkEWf7fM>
7/3/2015 3:33
- (Result No.: 3) Tweet:
This herbalife madness all up and down my timeline is really
annoying Lose weight in a Son! @Krishna_TheSage @wry214
<https://t.co/j2CNc5YnLO>
7/3/2015 3:34
- (Result No.: 4) Tweet:
Speakers today included: Quaker, Muslim, Catholic, Interfaith,
Unitarian Universalist, Zen, Hare Krishna, Native & SE Pacific
indigenous ppl <https://t.co/2pqccg9P7>
11/3/2016 10:26
- (Result No.: 5) Tweet:
When I re enter my body upon waking, the first two words out
my mouth are mother goddess (Hare) & Krishna.
10/16/2016 10:40

This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.



(Result No.: 6) Tweet:
Me And My VP Are On A Mission From God, Or Mohammed, Or Krishna ... Or Oprah. #RejectedPrezCampaignSlogans
<http://t.co/WMLlo32Shd>

6/23/2015 14:58

(Result No.: 7) Tweet:
Someone please motivate me to get out of bed and workout @krishna_mercy @Cassery_Rock @HarleyQuinn_Lon @MotherHen8
<http://t.co/0wRALJjI8E>

7/12/2015 13:12

(Result No.: 8) Tweet:
It must be as good as a workout @Melvinstaiberg @Krishna_Davida @Shanistaaa @_Howard94 @greensitt_z @MKforgetAshley
<http://t.co/yvCoVfayP1>

7/29/2015 16:11

(Result No.: 9) Tweet:
@svsnerendipity Jaenne @JoesSweetpea Krishna @YesEquation The @Ray_Floressss Flores @thatricokid rico
<http://t.co/FasormaXrC>

8/15/2015 2:04

(Result No.: 10) Tweet:
@Krishnamromo Krishna @jecoolest810 Joe
@ElizabethMoreen Elizabeth @lauren_borko Mr._MrsBorko @radcanine @Rowi
<http://t.co/rQqg4Djfx>

8/17/2015 2:06

(Result No.: 11) Tweet:
@catt_repher kathy @Minutemaniac #MentalHealthGunGrab @YungMurphz Murphy @krishnamromo Krishna
<http://t.co/1nW5eQbV>

8/17/2015 2:09

(Result No.: 12) Tweet:
@NiamhFitness Niamh @audryx_ondxi Arya @JoesSweetpea Krishna @AttitudeLih_imurBDooKAY @_GlamorlushLife DickBreaker
<http://t.co/tyOk1ZWyVm>

8/18/2015 1:45

(Result No.: 13) Tweet:
@D1Blyden JuiceMan @blurryfacegrace grace
@JoesSweetpea Krishna @shiftingthinkng Rachel @damnitnichi nich
<http://t.co/C3lFgZ2jyk>

8/19/2015 2:19

(Result No.: 14) Tweet:
@SimpsonArnold1 SimpsonArnold @JoesSweetpea Krishna @Fanaureylian Sofeece @bowiescheckbone bourgeoisophie
<http://t.co/YeBHNaD4dp>

8/21/2015 1:56

(Result No.: 15) Tweet:
@fitness_guru Fitness @JoesSweetpea Krishna @PlanetDr Sarah @ACTISIVAN Maria @PuglistProdigy PuglistProdigy
<http://t.co/K2zbhVQKXY>

8/22/2015 3:26

(Result No.: 16) Tweet:
@VSgyrlVee @JoesSweetpea Krishna @poetic swift selina
@Hectorgb03 Hector @vwqtpie tarina @kittehboi Mr
<http://t.co/jqfCnGEELY>

8/28/2015 3:54

(Result No.: 17) Tweet:
@dynamulice Stacie @Joliet1979 TKarwacka @altimelbelber Elena @JoesSweetpea Krishna
http://t.co/presume_quiet38 Tina

8/29/2015 1:38

Cosine with relevance feedback:

This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.



Type of query:
Relevance

Search tweets by boolean, phrase, wildcard or cosine

Go

Enter relevant docs as a list

Click to see datframe view

(Result No.: 0) Tweet:
Vladimir Putin says he will not expel US diplomats from Russia
<https://t.co/PKxeah5dR8>

12/30/2016 20:10

(Result No.: 1) Tweet:
Vladimir Putin
<https://t.co/HbXyTEBUX>

4/23/2017 19:25

(Result No.: 2) Tweet:
The many identities of Russian President Vladimir Putin.
<https://t.co/TveoXSmnrz>

12/2/2016 7:43

(Result No.: 3) Tweet:
Your new friend Vladimir Putin, GOP...
<https://t.co/KeaTUokSSH>

1/17/2016 10:56

(Result No.: 4) Tweet:
Vladimir Putin meets with John Kerry #SochiTalks
<https://t.co/1m1kTfOOGU>

1/17/2016 10:56

(Result No.: 5) Tweet:
Vladimir #Putin gave an interview to Rossiya-1 television channel presenter Vladimir Solovyov
<http://t.co/8tC48mtvju>

1/17/2016 10:56

This screenshot shows a web browser window with multiple tabs open. The active tab displays a search results page for 'Vladimir Putin' on a local host server at port 8501. The results are presented in a grid format, each row containing a tweet summary, its author, timestamp, and a link to the full tweet.

Result No.: 9 Tweet:	Result No.: 10 Tweet:	Result No.: 11 Tweet:
Vladimir Putin A look at Russia's president https://t.co/talHPyygkG #news https://t.co/3d0zFLVg2D https://t.co/TfIVjGdxAt	President Obama says he told Vladimir Putin to "cut it out." https://t.co/ZibHb8Tou9 https://t.co/07aaSLPLeA	Vladimir Lenin - What is Soviet power? https://t.co/2VvFVC9U42
8/17/2016 14:33	12/25/2016 03:03	9/29/2016 21:26
Result No.: 12 Tweet: A.M.G. - Go Hard Like Vladimir Putin https://t.co/nL60WuhC1K	Result No.: 13 Tweet: President Trump Meets Russian President Vladimir Putin https://t.co/ZibHb8Tou9 https://t.co/07aaSLPLeA	Result No.: 14 Tweet: Trump on Vladimir Putin: 'We've had some very, very good talks' https://t.co/EnTavPffrA https://t.co/UVTO7yweXw
6/30/2017 15:35	7/7/2017 11:09	7/7/2017 16:10
Result No.: 15 Tweet: Vladimir Putin — not THAT Vladimir Putin — nabbed in Florida https://t.co/PzU5tXC1HD	Result No.: 16 Tweet: Only Vladimir Putin and white people can vote. https://t.co/PzU5tXC1HD	Result No.: 17 Tweet: Vladimir Putin Won't Meet With Rex Tillerson After Syria Strikes https://t.co/9lk2PYK2TV https://t.co/1hnQnNnfMy
8/30/2016 23:59	1/25/2017 23:20	4/10/2017 20:48

(After feedback)

This screenshot shows a web browser window with multiple tabs open. The active tab displays a search interface for 'Vladimir Putin' on a local host server at port 8501. The interface includes a search bar, a relevance dropdown, and a list of results.

Relevance

Search tweets by boolean, phrase, wildcard or cosine

vladimir putin

Go

Enter relevant docs as a list

0,1,2,3,4,5

Feedback submitted! New results are:

Click to see dataframes view

Result No.: 0 Tweet:	Result No.: 1 Tweet:	Result No.: 2 Tweet:
Obama: 'Vladimir Putin is not on our team' via @CNNPolitics https://t.co/DG6HgiAlym	Vladimir Putin. https://t.co/lHbxTEBUX	The many identities of Russian President Vladimir Putin. https://t.co/TvseXSrnrt
1/7/2017 1:03	4/23/2017 19:25	12/2/2016 7:43
Result No.: 3 Tweet:	Result No.: 4 Tweet:	Result No.: 5 Tweet:
Your new friend Vladimir Putin, GOP... https://t.co/KeaTUxXSSH	10 critics of Vladimir Putin who wound up dead https://t.co/wshZMowUKY https://t.co/D6emitsOsUf	Vladimir Putin must be saying, "Come back Barack" https://t.co/TEXmgxfuS
12/17/2016 23:08	3/25/2017 13:42	11/20/2017 7:46
Result No.: 6 Tweet:	Result No.: 7 Tweet:	Result No.: 8 Tweet:

This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.

(Result No.: 6) Tweet:
#TrumpsFavoriteHeadline Vladimir Putin Congratulated Trump on His Presidency
https://t.co/6hHyygk5o #news | https://t.co/3d02FUvg2D
https://t.co/TFyjvG0xAt

8/17/2016 14:18

(Result No.: 7) Tweet:
Why is Vladimir Putin his nose in US elections?
#RejectedDebateTopics

10/19/2016 14:57

(Result No.: 8) Tweet:
Russian dissident: Vladimir Putin will use hacking tactic again
https://t.co/qg07xaVNH #Russia #Putin

12/20/2016 5:32

(Result No.: 9) Tweet:
Vladimir Putin: I look at Russia's president
https://t.co/nhHyygk5o #news | https://t.co/3d02FUvg2D
https://t.co/TFyjvG0xAt

9/17/2016 15:18

(Result No.: 10) Tweet:
President Obama says he told Vladimir Putin to "cut it out"
https://t.co/V87nWba5Bj

12/25/2016 6:03

(Result No.: 11) Tweet:
Trump gives Putin an 'A' on leadership Why DonaldTrump is praising Vladimir #Putin? http://t.co/M3n5CD0lO

10/1/2015 13:08

(Result No.: 12) Tweet:
Vladimir Putin arrested on trespassing charges #BREAKING
https://t.co/V87nWba5Bj

8/30/2016 13:45

(Result No.: 13) Tweet:
The many identities of Russian President Vladimir Putin.
https://t.co/TvseXSmrz

12/2/2016 7:14

(Result No.: 14) Tweet:
Vladimir Putin calendar 2016: You're welcome
https://t.co/KT68kxBUX3

12/29/2015 8:00

(Result No.: 15) Tweet:
Watch Live: Coverage of President Trump's meeting with Vladimir Putin https://t.co/wDkOcCP29i
https://t.co/T4ldJLQ12

7/7/2017 13:45

(Result No.: 16) Tweet:
A.M.G. - Go Hard Like Vladimir Putin https://t.co/nEDWuhC1K

6/30/2017 15:35

(Result No.: 17) Tweet:
#TrumpsFavoriteHeadline Vladimir Putin Congratulated Trump on His Presidency
https://t.co/6hHyygk5o

8/17/2016 15:05

(Result No.: 18) Tweet:
Vladimir Putin arrested on trespassing charges #BREAKING
https://t.co/V87nWba5Bj

8/30/2016 13:45

(Result No.: 19) Tweet:
The many identities of Russian President Vladimir Putin.
https://t.co/TvseXSmrz

12/2/2016 7:14

(Result No.: 20) Tweet:
Vladimir Putin calendar 2016: You're welcome
https://t.co/KT68kxBUX3

12/29/2015 8:00

(Result No.: 21) Tweet:
Only Vladimir Putin and white people can vote.
https://t.co/PzUshXC1HD

1/25/2017 23:20

(Result No.: 22) Tweet:
Russia bans picture of Vladimir Putin in drag
https://t.co/V1dPmklLy https://t.co/XGrQ46mtQ

4/6/2017 10:41

(Result No.: 23) Tweet:
#TrumpsFavoriteHeadline Vladimir Putin Congratulated Trump on His Presidency
https://t.co/6hHyygk5o

8/17/2016 14:20

This app is to serve as a front-end for the tweets dataset search engine system implemented for AIRW Assignment 1 in Python.

(Result No.: 15) Tweet:
Vladimir Putin arrested on trespassing charges #BREAKING
https://t.co/V87nWba5Bj

8/30/2016 13:45

(Result No.: 16) Tweet:
The many identities of Russian President Vladimir Putin.
https://t.co/TvseXSmrz

12/2/2016 7:14

(Result No.: 17) Tweet:
Vladimir Putin calendar 2016: You're welcome
https://t.co/KT68kxBUX3

12/29/2015 8:00

(Result No.: 18) Tweet:
Watch Live: Coverage of President Trump's meeting with Vladimir Putin https://t.co/wDkOcCP29i
https://t.co/T4ldJLQ12

7/7/2017 13:45

(Result No.: 19) Tweet:
A.M.G. - Go Hard Like Vladimir Putin https://t.co/nEDWuhC1K

6/30/2017 15:35

(Result No.: 20) Tweet:
#TrumpsFavoriteHeadline Vladimir Putin Congratulated Trump on His Presidency
https://t.co/6hHyygk5o

8/17/2016 15:05

(Result No.: 21) Tweet:
Vladimir Putin arrested on trespassing charges #BREAKING
https://t.co/V87nWba5Bj

8/30/2016 13:45

(Result No.: 22) Tweet:
The many identities of Russian President Vladimir Putin.
https://t.co/TvseXSmrz

12/2/2016 7:14

(Result No.: 23) Tweet:
Vladimir Putin calendar 2016: You're welcome
https://t.co/KT68kxBUX3

12/29/2015 8:00

(Result No.: 24) Tweet:
Only Vladimir Putin and white people can vote.
https://t.co/PzUshXC1HD

1/25/2017 23:20

(Result No.: 25) Tweet:
Russia bans picture of Vladimir Putin in drag
https://t.co/V1dPmklLy https://t.co/XGrQ46mtQ

4/6/2017 10:41

(Result No.: 26) Tweet:
#TrumpsFavoriteHeadline Vladimir Putin Congratulated Trump on His Presidency
https://t.co/6hHyygk5o

8/17/2016 14:20

CONCLUSION :

The implemented information retrieval system using the Boolean Model, wildcard queries, and phrase queries can effectively retrieve relevant documents from a corpus based on user queries. The system also includes a relevance feedback mechanism that can improve the retrieval results based on user feedback. The system can be further improved by incorporating more advanced ranking algorithms and query expansion techniques.

THANK YOU