**Variable names:**

**Clustering.ipynb**

**nlp**: Holds the spaCy transformer pipeline

**dataframe** : Holds the original dataframe from the scraped data, also holds the data after we have the id’s specified

**article\_texts** : holds the content of the dataframe which is news articles

**tfidf\_vectorizer** : holds the TF-IDF vector matrix for the news articles

**kmeans** : executes the kmeans clustering and holds the kmeans cluster values

**cluster\_labels** : this variables holds the piece of information for kmeans cluster variables

**cluster\_centers**: holds the cluster center values from kmeans

**stage1\_cluster\_0\_points**: the index of clusters 0 from stage I clustering

**stage1\_cluster\_1\_points**: the index of clusters 1 from stage I clustering

**random\_10\_elements**: selecting random 10 elements from cluster 0 to check if the articles being removed are right

**filtered\_dataset\_1** : filtered news from the original dataset after clustering

**unwanted\_news\_stage\_1** : unwanted news which is a reminder from original dataset and filtered dataset

**unwanted\_article\_texts\_cleaned**: Handling the NaN values by replacing empty string

**unwanted\_tfidf\_vectorizer** : holds the TF-IDF vector matrix for the unwanted stage I news articles

**unwanted\_news :** stores the unwanted news articles

**example\_statement**: Provides the ideal required news article paragraph

**example\_statement\_embedding**: creates an embeddings for the example statement

**content\_embeddings:** creates the embeddings for content for unwanted\_news

**similarities:** creates a cosine similarity scores between the example statement and unwanted news content

**similar\_news:** similar news which is extracted which has score of similarity above 60%

**filtered\_dataset\_2** : filterned dataset after we have the similarity score of above 60% from the unwanted news

**filtered\_dataset**: the final filtered dataset with the combination of filtered\_dataset\_1 and filtered\_dataset\_2

**filtered\_dataset\_oneliner :** the filtered\_dataset\_oneliner holds the oneliner

**Function:**

**Clustering.ipynb**

**extract\_id(link) :** extract the ID from the URL(Link) of the news article

**preprocess\_text\_spacy(text) :** Function to preprocess the news article by removing the stop words, performing tokenization and lemmatization

**One\_Liners(df)** : Function which extracts the first sentence from the news article

**Files:**

**Clustering.ipynb**

**dataframe.csv** – Holds the data after the id’s are tagged to each news article

**tfidf\_matrix.pkl** – Holds the pickle file for the TF- IDF vector

**filtered\_dataset\_oneliner.csv** – final file from clustering notebook, which holds the fields – id, link,content, date, first\_line