Social Media Analysis of Twitter Data

Students:

Hanu Pathuri (hpathuri@eng.ucsd.edu)

Mohammed Tawashi (mtawashi@eng.ucsd.edu)

Amir Shirkhani (amshirkh@eng.ucsd.edu)

Naveed Mohammed (mnaveed@eng.ucsd.edu)

Vamsi Namuduri (rnamudur@eng.ucsd.edu)

Faculty Advisor:

Prof. Amarnath Gupta

Abstract

This project focuses on studying how Twitter images impact the narrative of hashtags. A lot of research in Twitter data has been focused on separate textual content without media attachments. Images attached to a tweet provide an additional dimension to help understand a tweet's context and the user's general opinion. The assumption is that the visual images reinforce the opinion that was presented in the text. The project aims at finding specific patterns in tweets where media files (images) are used to change the narrative of the corresponding hashtag and co-occurred hashtags. This is achieved by studying the topic of solo hashtag and co-occurred hashtags without or with associated media files. Media file as a visual channel is a powerful medium and can change the subject and original purpose of a hashtag for a given audience. A small number of media tweets (images) associated with a hashtag can have a higher influential impact on an observer/user than the same or even higher number of tweets without media. There are multiple benefactors to the findings from this project. 1- Election organizers as part of a campaign can study and detect endorsing and opposing trends and act by counter measures using similar techniques. 2- Social Media platforms and especially Twitter itself can detect patterns and potentially restrict the behavior. 3- Journalists can report to the general public on how a small group of influencers can sway a narrative and push various agendas.

Introduction:

In recent years, social media especially Twitter has a heavy impact on the public discourse and communication in the society. Twitter has the potential for increasing political participation and is an ideal platform for users to spread not only information in general but also political opinions publicly through their networks, political institutions (e.g., politicians, political parties, political foundations, etc.) for the purpose of encouraging more political discussions and influencing specific narratives and interests. There is an emerging need to continuously collect, analyze, and visualize politically relevant information from social media. This is definitely a challenging task due to the large amount and complexity of information and unstructured data.

Team Roles and Responsibilities:

Project Manager & Story Teller/Coordinator: Hanu Pathuri

Solution Architect & Project Coordinator: Vamsi Namuduri

Budget Manager & Business Analyst: Naveed Mohammed

Report Manager & Visualization Developer: Amir Shirkhani

Methods Expert & Data Engineer: Mohammed Tawashi

Data Acquisition:

The main data source for the project was the USCD Twitter database. Raw Twitter data is collected every day in a Postgres database. This database has only political tweets. Each day about 2.5 Million tweets are collected in the database. Apart from Tweets, the database also has User and Hashtag information and also links to the media that appeared in the tweets.

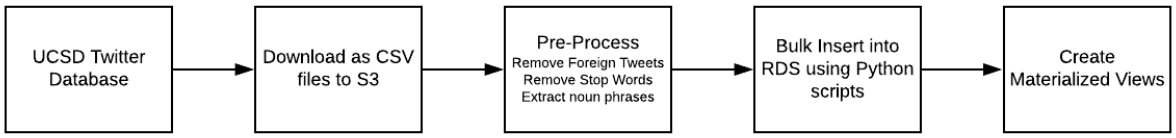
One of the preliminary questions that we tried to answer was related to image clustering and image object detection. Collecting the image features along with corresponding text helped in analysis of the images. It was used to determine if images were being used to influence the narrative of tweets. Text processing and image processing techniques were used to collect various features to feed into the models. The final analysis of the project was done on one month worth of data (Dec 2019).

Below is the volume of data we processed for one of the datasets:

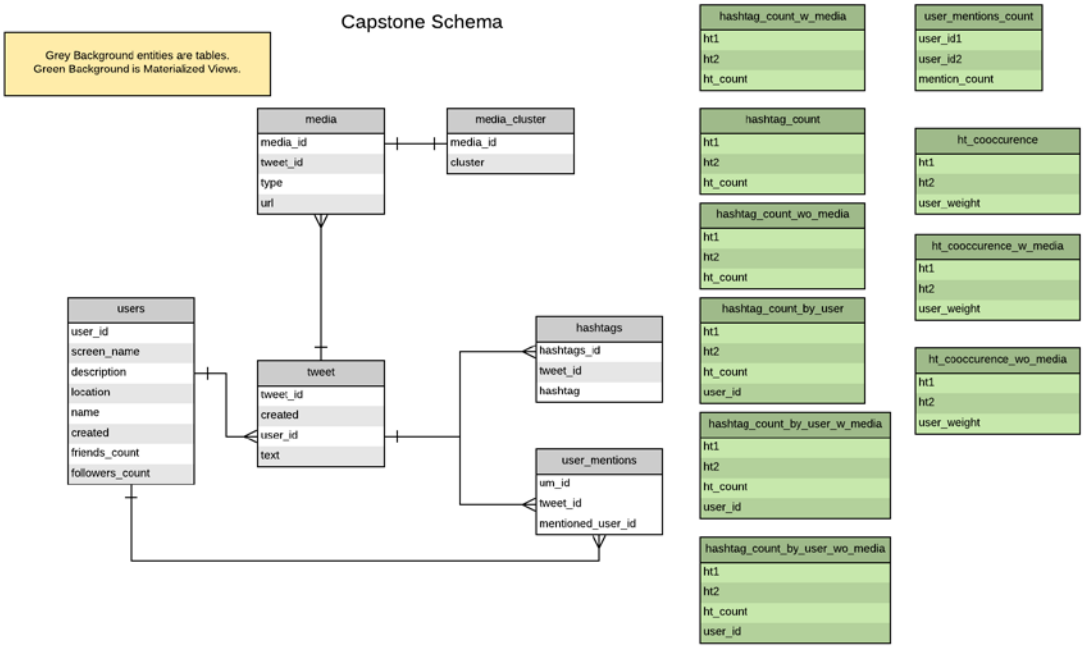
Users	2,473,910
Tweet	22,958,670
Media	664,932
Hashtags	2,308,751
User Mentions	31,520,555

Data from UCSD data center was moved into AWS using various python scripts. Postgres bulk COPY command was leveraged to export the data from UCSD database and files were moved into S3. Boto library was used to connect to AWS components like S3 to transfer data from S3 into RDS instances (Postgres).

Below is the data pipeline of how the data was processed along with image download and processing.



The data is stored in Postgres Database on AWS RDS. Below is the database schema:



Data Preparation:

Below are the high-level steps of data extraction and processing:

1. Connect to UCSD VPN and download the raw tweet data in a loop for each day in Dec 2019.
2. Since the Postgres database that was created in AWS has no access to UCSD VPN, the data had to be first locally downloaded to the local computer and then each file was uploaded to S3 Buckets on AWS.
3. Provisioned a compute optimized EC2 instance (c5.24xlarge) with 96 vCPUs and 192GB of memory.
4. Used Postgres Bulk Copy utility (COPY) to ingest the data in flat files into a staging table.
5. Data from the staging table is normalized into Users, Media, Tweet and Hashtag entities and populated.
6. Each user level info is extracted in the same way and username, screen name, follower count and friend count are populated.

Some of the data quality issues that were encountered during data preparation are:

1. The media links were broken and we had to filter those tweets.
2. URLs in the tweets.
3. Stop words and other common words in the domain interfering with topic modeling.
4. Break the sentences into parts of speech and retain noun phrases.

Without pre-processing the data, the model will not identify the topics correctly because of URLs and retweet strings and other Twitter related characters. The stop words and common words were also removed which helped in identifying the topics clearly. Noun phrases were extracted from each tweeting using Parts of Speech tagging with the help of Spacy library.

The main features that were selected for the analysis:

Tweet: created_at, in_reply_to_status_id, in_reply_to_user_id, source,retweet_count, retweeted, in_reply_to_screen_name,is_quote_status, favorite_count, id, text, place, lang,reply_count, userid, retweeted_id, hashtag, media,url, geo'
Image Features: tweet_id, user_id, created_at, text, retweet_count, type, url, hashtag, image_hash, is_person, is_bicycle, is_car, is_motorbike, is_aeroplane, is_bus, is_train, is_truck, is_boat, is_traffic_light, is_fire_hydrant, is_stop_sign, is_parking_meter, is_bench, is_bird, is_cat, is_dog, is_horse, is_sheep, is_cow, is_elephant, is_bear, is_zebra, is_giraffe, is_backpack, is_umbrella, is_handbag, is_tie

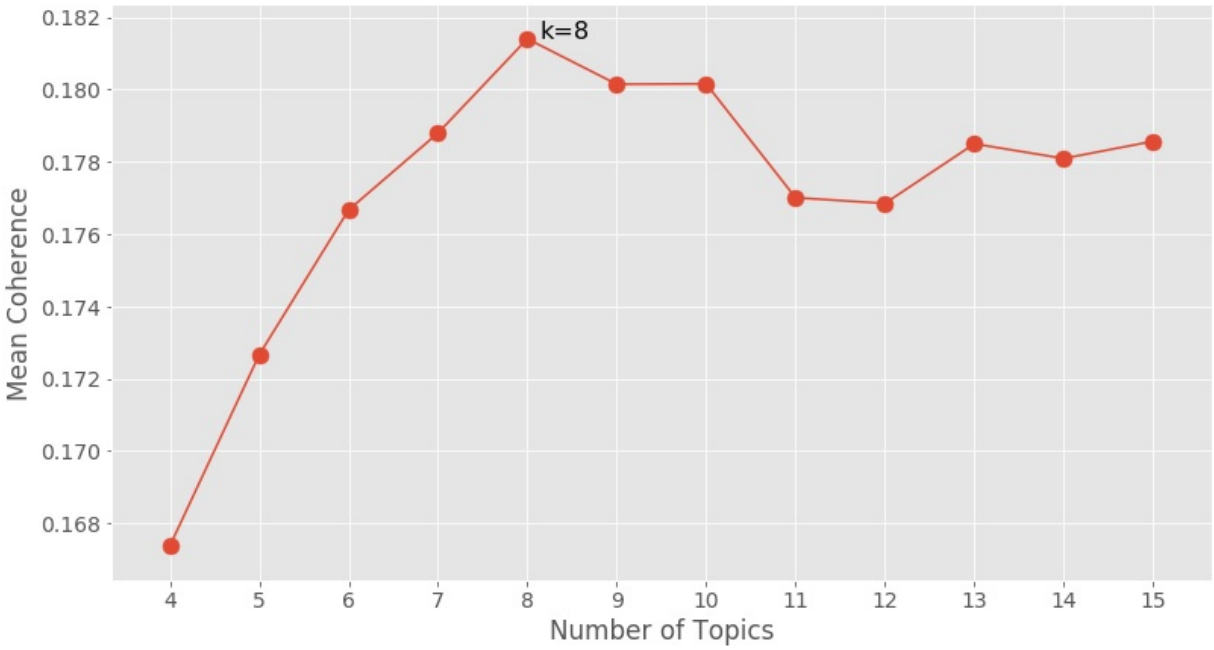
Text features: tweet_polarity, tweet_subjectivity , pos, neg, neu,, compound, wordcount , quest_mark, length,mentions

Data Analysis methods:

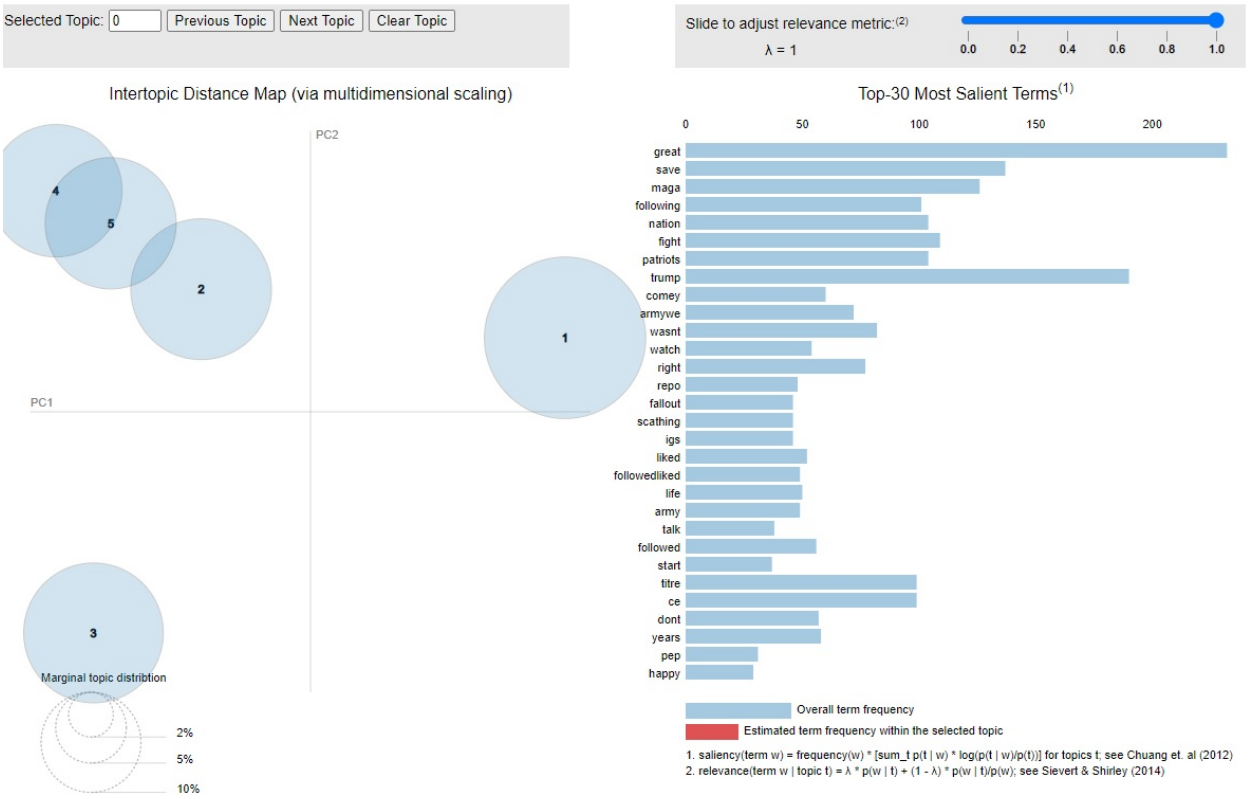
A. Topic Modeling and Tweet text analysis:

Below are the analysis methods performed on the tweet data:

1. Topic modelling on the tweet data using various methods - NMF, LDA, Guided LDA. "*Can each tweet be considered as a document in performing topic modelling?*" was something the team was not sure of during the course of the project, so performed topic modelling in various ways - considering each tweet as a document, aggregated tweet data per user, aggregated tweet data per hashtag and analyzed the results.
2. Clustering on images extracted from media tweets. Imagenet ResNet50 layer output features are extracted and a simple k-means clustering technique is applied on that.
3. Bigram analysis on each image cluster to understand prominent words and phrases per each image cluster.
4. Optimal number of topics analysis in topic modelling by calculating mean coherence (by building word2vec model and calculating similarity between topics).



5. Partitioning and community detection from cosine similarity of topic vectors (from topic modelling) networkx graph. Iterate through each image cluster.
6. Most common word analysis in each image cluster and Intertopic Distance Map analysis is also performed by using pyLDAvis library.



7. Text extraction from the images (in media tweets) is done. And object detection from the images is also performed to analyze patterns and anomalies in association with tweet text.

Image object detection features-

person	56849
tie	16613
chair	7180
umbrella	1760
car	988
cup	801
handbag	569
cell_phone	362
wine_glass	348
tvmonitor	293
laptop	288

Text extraction from the images -

hashtag	processed_image_text
ImpeachAndRemove	rate unemployment mam mb phone number rudy ...
impeached	lte i tweet ll you retweeed tommie sunshine ...
Impeached	mm um abwl menamw farmer cw thesenamrs mum a...
Impeached45	pm dec we need payback merrick garland need s...
impeachment	k k author cnn the good news trump republi...
ImpeachmentDay	ltwai text message today pm pres trump dems...
ImpeachmentEve	in all states will demand congress impealh...
IMPEACHMENTVOTE	bill clinton andrew johnson w hkh wmmmmwm k...
ImpeachTrump	days in office l false blaims lies i i if i ...

zoo

B. Hashtag and Co-occurrence relationship analysis

Multiple attempts and methods were used trying to find how media files are being used to influence hashtags. Preliminary exploration took place on a small data set (200K tweets) using our localized machines. Through the journey of this EDA we focused only on tweets with associated media files and tried to find any pattern through users/tweets distribution and temporal analysis. We were able to identify some temporal patterns for some tweets with associated media files. That was not enough to draw any insights. Therefore, we switched our focus on hashtags co-occurrences and how media files may influence their topics. We initially approached this problem by looking at the heavily used hashtags that are being used with media files associated. But then to understand the impact of media files we needed to understand the behavior when media files are not present. The idea further grew to visualize each hashtag as a node and the co-occurrence with other hashtags as an edge. Our database scheme was also modified to serve this purpose.

Since we focus on media files impact, only the common hashtags that were used with and without media files are processed. Below model details the step of our final methodology to detect any attempt to influence hashtags topics.

Hashtags topics incoherent model

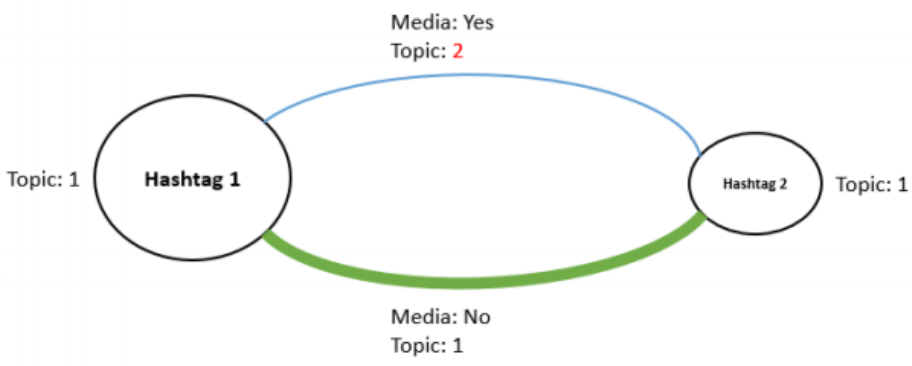
This detector model highlights any change of hashtags topics when they co-occurred together. It also checks if media files (images, videos) are used to achieve this change of narrative. The model works as follow:

1. Build and train a topic model (using NMF/LDA) based on aggregated tweets text per each hashtag (not per single tweet text). The aggregated text is cleaned before modeling by removing URLs, RT and cc,hashtags, mentions, double spacing, numbers, punctuations, and converting all to lowercase.

2. Apply topic modeling to generate a dominant topic per each single hashtag over its aggregated tweets text.
3. Calculate each hashtag weight based on the number of unique users that tweeted that hashtag
4. Aggregate all tweets texts of any 2 co-occurred hashtags in same tweets and that these tweets have associated media files. Then apply the topic modeling on the aggregated cleaned text.
5. Aggregate all tweets texts of any 2 co-occurred hashtags in same tweets and that these tweets don't have associated media files. Then apply the topic modeling on the aggregated cleaned text.
6. calculate the hashtags co-occurrence weight based on unique users who used these 2 hashtags jointly. This is done for both cases (media and no media).
7. Generate a new data frame for the common (interested) hashtags between the 2 cases of media and no media co-occurrences
8. Calculate the hashtags concentration factor for both co-occurrences edges (media/no media). The hashtags concentration factor is the ratio between number of tweets and number of unique users.

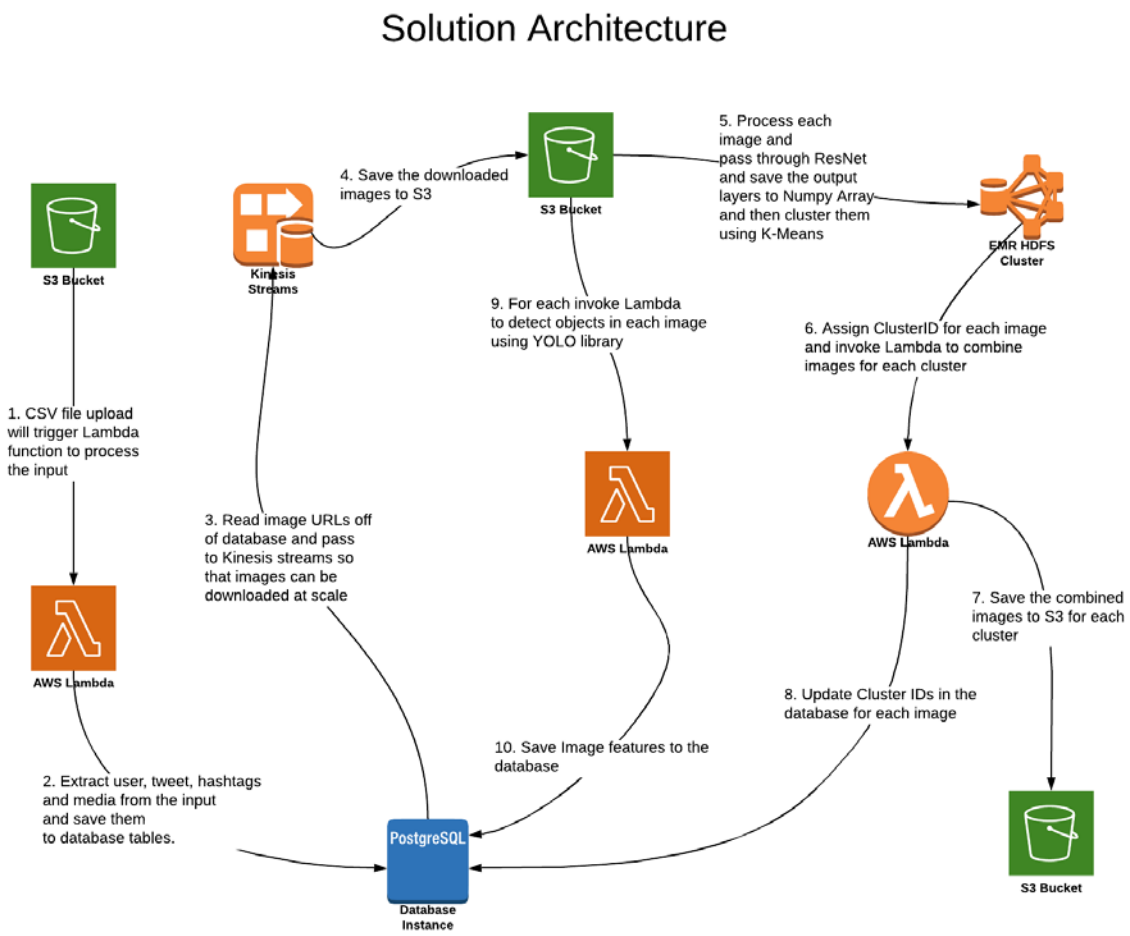
Any 2 co-occurred hashtags that sometimes media files are tagged with will have 4 different topics (a topic per each of the 2 hashtags, a topic when co-occurred with media, and a topic when co-occurred without media) in addition to the user concentration factor for the co-occurred relationships.

Finally, the model checks if these 4 topics are incoherent, especially the edge topics (co-occurred topics), in addition it also checks if the user concentration factor (tweets/user) on these edges is high. In such cases there is an attempt to change the narrative of these hashtags.



Solution Architecture, Performance and Evaluation:

Below is the scaled solution architecture to process the data and images on AWS using S3, RDS and EMR clusters.

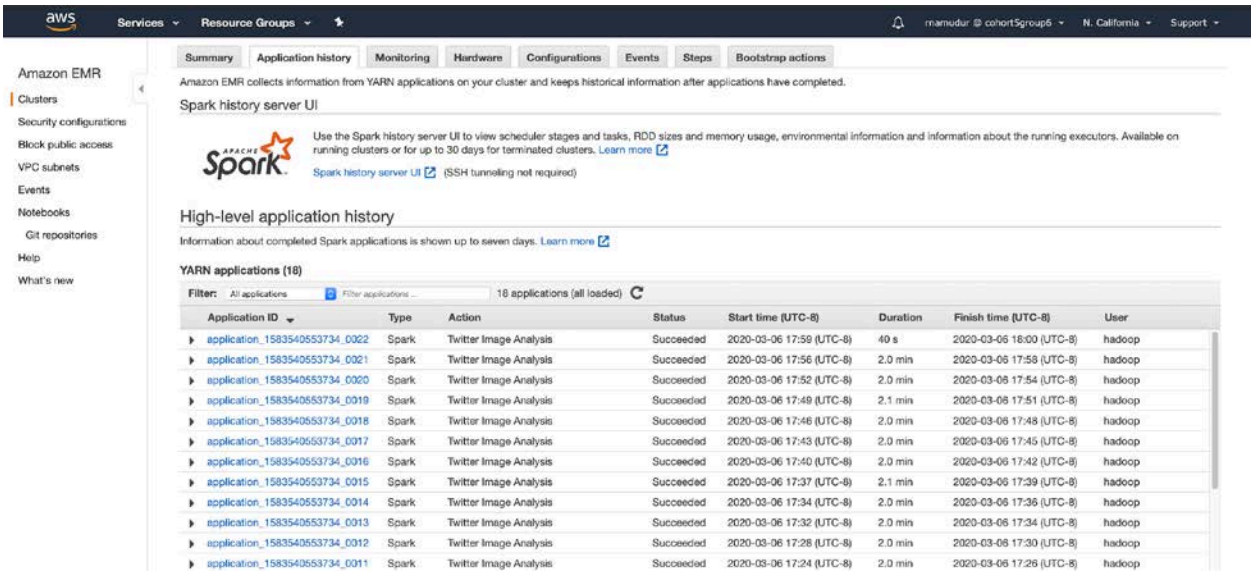


Python Multithreading along with PySpark was used to scale the pipeline to process millions of tweets and hundreds of thousands of Tweets. EMR cluster was configured with 5 slave nodes and dependent libraries were installed on the cluster. Spark session was created, and python scripts were implemented to process the data using Dataframe operations.

Below is the time taken by the python scripts using “Multiprocessing” module (joblib)

Dataset	**Batch Size**	**Time for end to end process (in seconds)**
10K Tweets	1K	23.04
10K Tweets	2.5K	15.52
10K Tweets	5K	15.44
100K Tweets	5K	101.6
100K Tweets	10K	95.25

Below are the runtimes for the EMR Spark jobs on AWS – The application extracts features from Images, clusters them based on K-Means and extracts text from Images.



The main services from AWS that were used are S3, RDS, EMR and Kinesis streams.

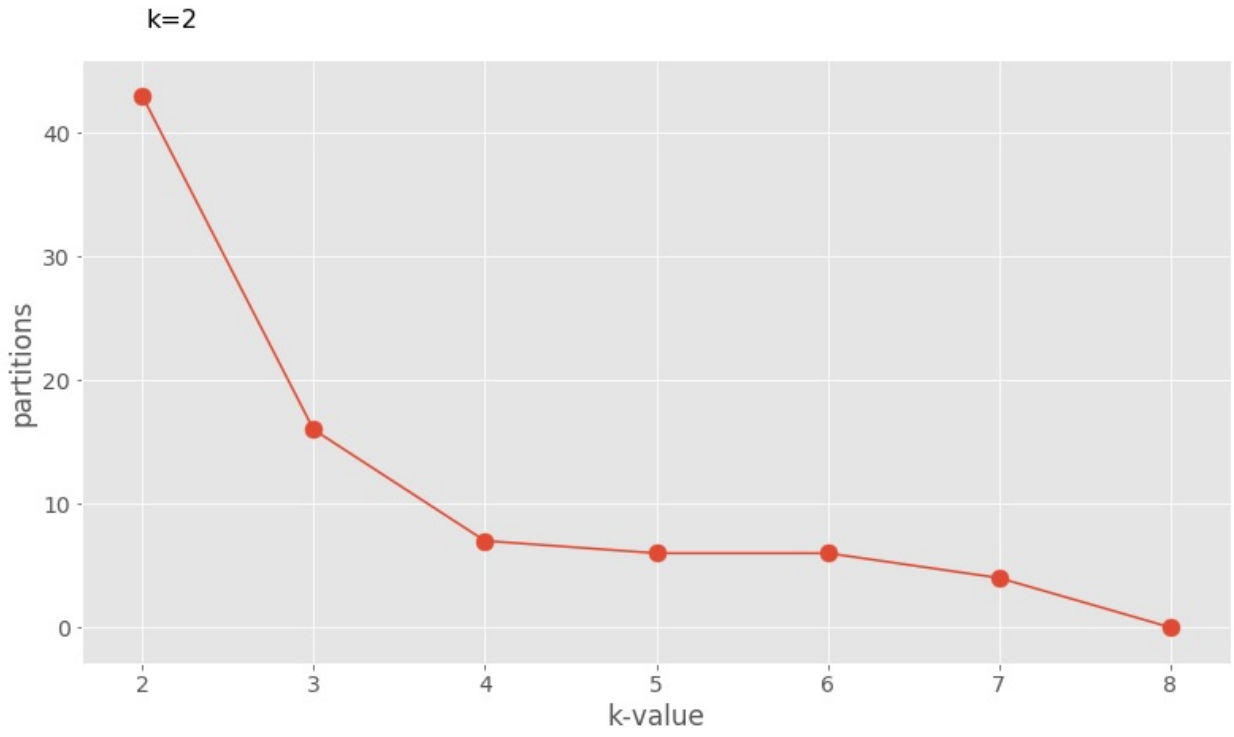
Some of the budget was managed using the following techniques:

1. For the most part the files were zipped to save the storage on S3.
2. Reduce the network traffic in and out of EC2 instances by keeping all the data in S3 (local to the region).
3. Compute optimized EC2 instances were used for Image processing instead of GPU. Spot instances could be potentially used but due to the nature of spot instances terminating abruptly, we couldn't use them.
4. The database storage was downsized after the dataset was completely loaded.

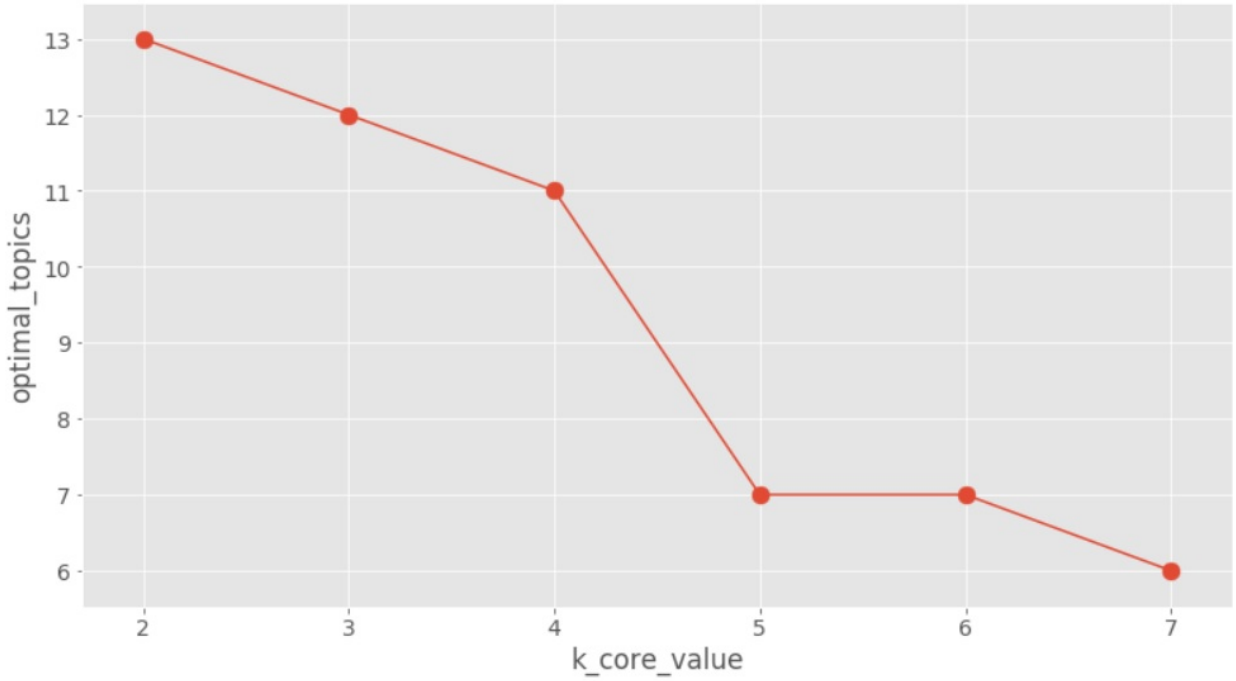
Findings and Reporting:

A. Topic Modeling and Tweet text findings:

1. As we sweep through the k-core value from 2 to some higher number (say 30), we notice that the number of partitions within the sub graph gradually reduces and eventually becomes 0 at some k-core value meaning there is no subgraph with in+out degree >= that k-core value.



2. As we sweep through the k-core value from 2 to some higher number (say 30), we notice that the optimal number of topics within the sub graphs gradually reduces until it reaches some minimum.

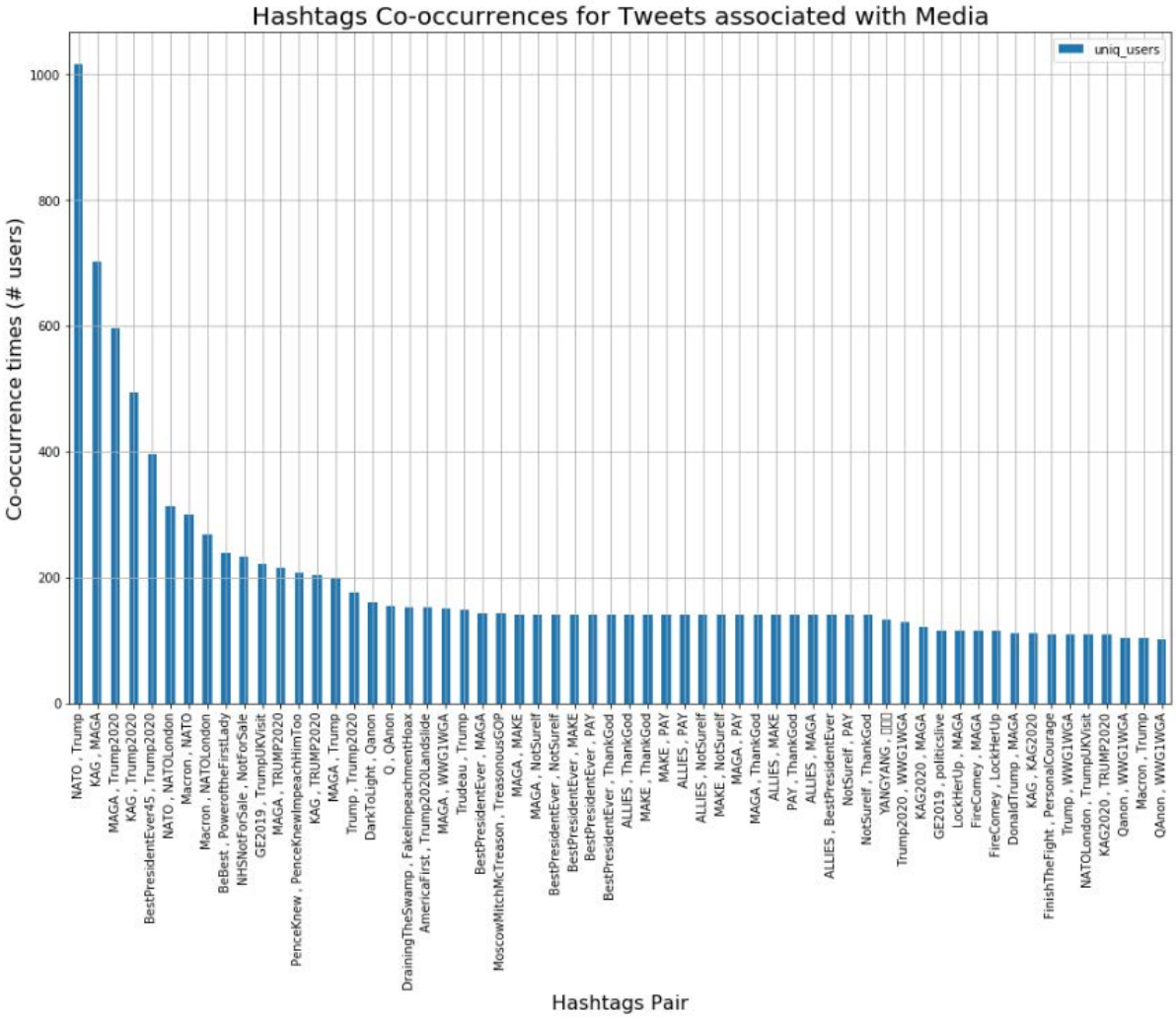


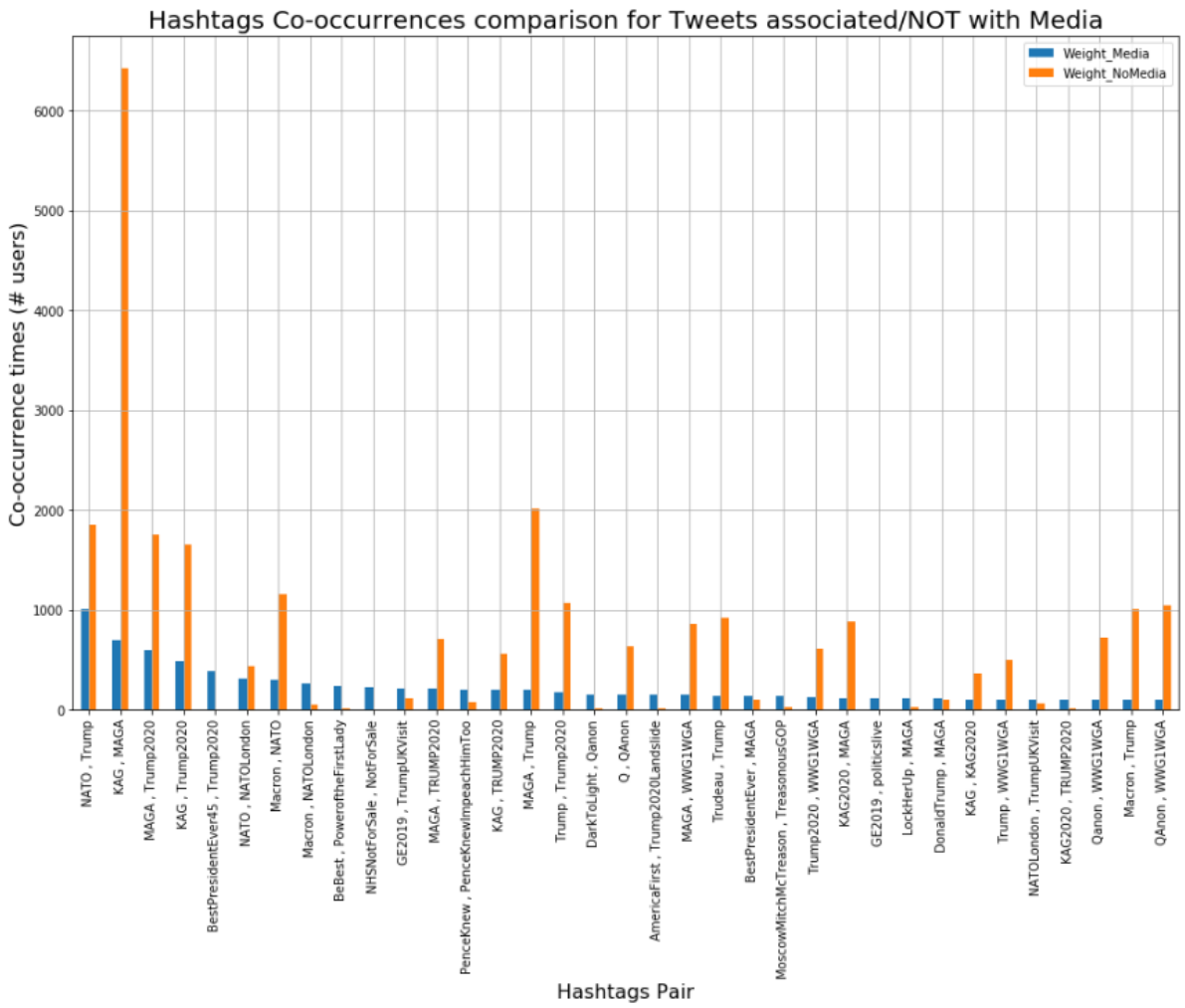
3. Topic modelling on tweet texts is performed using LDA, Guided LDA and NMF techniques. We notice that NMF results make more sense compared to LDA results.
4. We extracted the objects information from all the images available in the media tweets. we couldn't extract any major findings from image object features that could augment our tweet analysis in sense (Note: because the tweet data is predominantly political content most of the images had persons/ties/chairs).
5. The topic associated with hashtags tweet text in a partition of maximum k-core value (subgraph with maximum possible degree in hashtag occurrence w/ media graph) seems to be same as the topic associated with the images used in that particular subset of tweets. It basically implies images are used in conjunction with tweet text to augment the similar narrative.

B. Hashtags topics incoherent model findings:

1. Bar charts below show the heavily co-occurred hashtags and the interested ones between co-occurrence with and without media files associated.
2. These results are from 1-week tweets data.
3. We found a case in the 200K data set where a different set of users used some hashtags jointly with associated media files.

4. This model couldn't be run on the 1-month tweets dataset due to running out of AWS budget and that can't be run locally.
5. An interactive visualization is built to demonstrate this model





Conclusions:

1. Text extraction from media files associated with tweets are, in most cases, supplementing the narrative of the tweet's text.

2. Media files are being used as a powerful tool to contaminate the original narrative of single/co-occurred hashtag pairs.

3. Object detection from images did not result in any additional insights.

References/Appendices

1. https://link.springer.com/chapter/10.1007/978-3-030-34980-6_31
2. https://towardsdatascience.com/tweet-analytics-using-nlp-f83b9f7f7349
3. https://www.kaggle.com/chadalee/text-analytics-on-russian-troll-tweets-part-1
4. https://osome.iuni.iu.edu/tools/botslayer/
5. https://link.springer.com/article/10.1007/s10115-019-01429-z?shared-article-renderer
6. https://www.freecodecamp.org/news/how-we-changed-unsupervised-lda-to-semi-supervised-guidedlda-e36a95f3a164/
7. https://medium.com/capital-one-tech/learning-to-read-computer-vision-methods-for-extracting-text-from-images-2ffcdae11594
8. https://www.pyimagesearch.com/2018/09/17/opencv-ocr-and-text-recognition-with-tesseract/
9. https://machinelearningmastery.com/object-recognition-with-deep-learning/
10. https://www.pulsarplatform.com/blog/2018/challenges-analyzing-social-media-images-scale-research-sage-publishing/
11. https://blog.unmetric.com/identify-dominant-objects-and-colors-in-brands-images
12. https://buffer.com/resources/the-power-of-twitters-new-expanded-images-and-how-to-make-the-most-of-it
13. https://towardsdatascience.com/spectral-clustering-82d3cff3d3b7
14. https://dominicatkinson.com/post/trump-part-1/
15. https://github.com/cbaziotis/ekphrasis
16. https://github.com/eriklindernoren/PyTorch-YOLOv3
17. https://machinelearningmastery.com/divergence-between-probability-distributions/
18. http://proceedings.mlr.press/v32/steeg14.pdf
19. http://www.eng.biu.ac.il/~goldbej/papers/dagm02.pdf
20. http://openaccess.thecvf.com/content_ICCV_2019/papers/Ji_Invariant_Information_Clustering_for_Unsupervised_Image_Classification_and_Segmentation_ICCV_2019_paper.pdf
21. http://www.cs.utexas.edu/users/inderjit/public_papers/kdd_cocluster.pdf
22. https://homes.cs.washington.edu/~bboots/files/GuerinBMVC18.pdf
23. https://arxiv.org/pdf/1511.06335.pdf
24. https://paperswithcode.com/task/fake-image-detection