# Clustering Analysis of Political Tweets

**CLOUD COMPUTING | GROUP 4**

Brian DelRocini, Rizwan Chowdhury, Kinjal Patel

# Objective

**TOPIC**

Analyze themes prevalent in politicians' Twitter feeds (Joe Biden & Donald Trump) though K-Means clustering

## → Purpose

- Analyze glimpse of political discourse in our nation
- Understand how social media is used by different politicians

## → Technology Used

- PySpark RDDs (map, sortByKey, groupByKey)
- SQL SparkSession (dataframes, hashing) TF, IDF
- ML (KMeans)
- AWS (EMR, S3)
- Twitter (API)

KEY WORD

# Clustering

→ **Definition**

Grouping unlabeled data

→ **What**

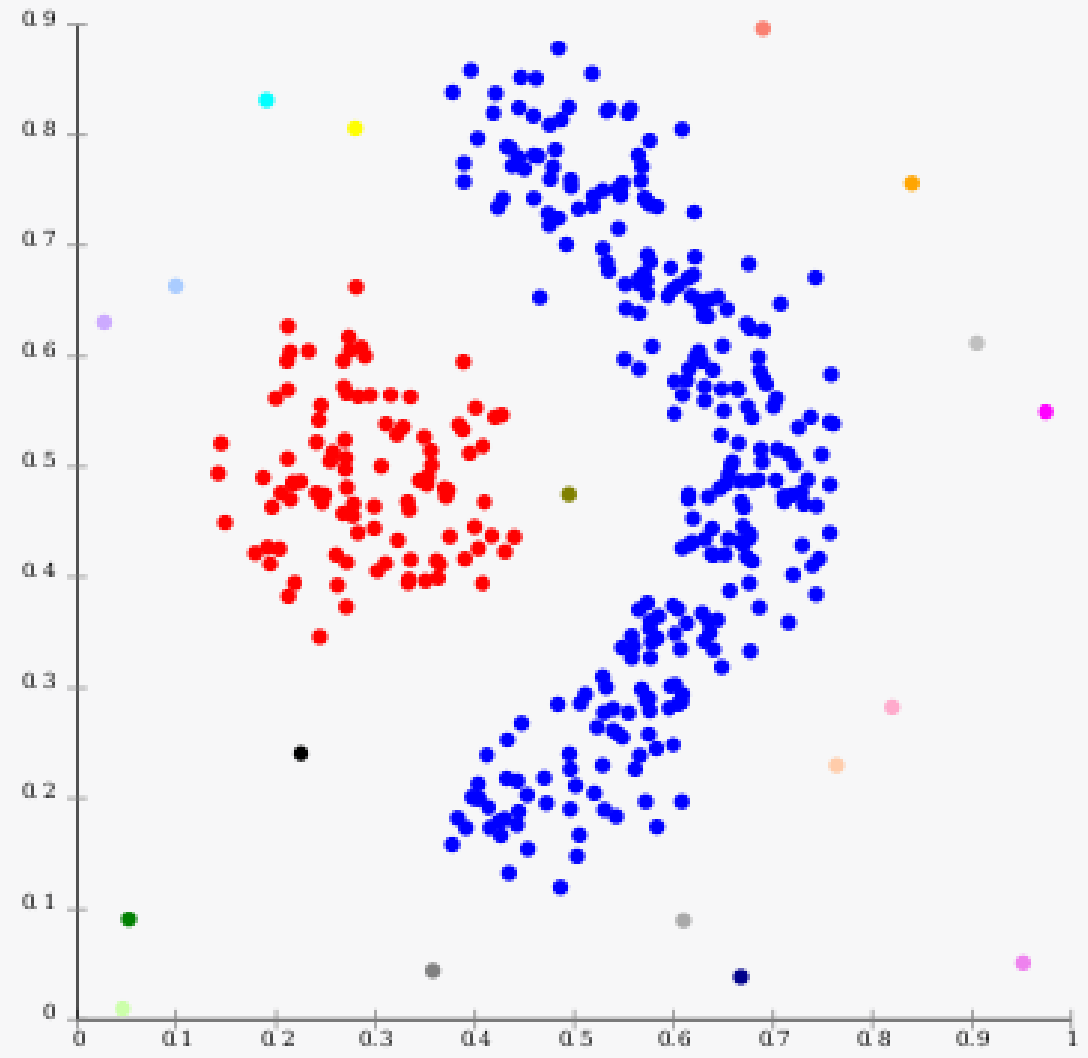
Used to find relations between data

→ **Purpose**

Find patterns or themes by grouping similar data

→ **Note**

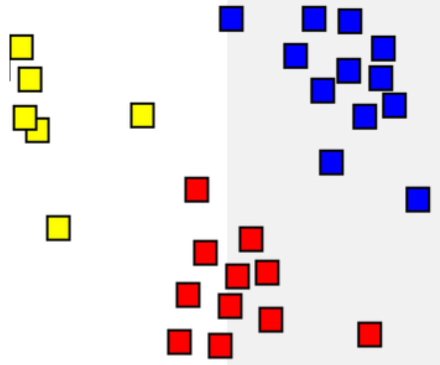
'Labeled clustering' is simply classification



# Procedure

## 1 Retrieve Tweets

- Get data to do cluster analysis on by retrieving datasets
- Target politicians who use Twitter



## 2 Machine Learning

- Utilize Pyspark map-reduce & its ML libraries (K-means clustering) to build a pipeline to carry out analysis

## 3 Amazon EMR Comparison

- Process Pyspark analysis on cloud
- Compare running time and computational resources

# Retrieve Tweets

PART 1

# Tweet Datasets

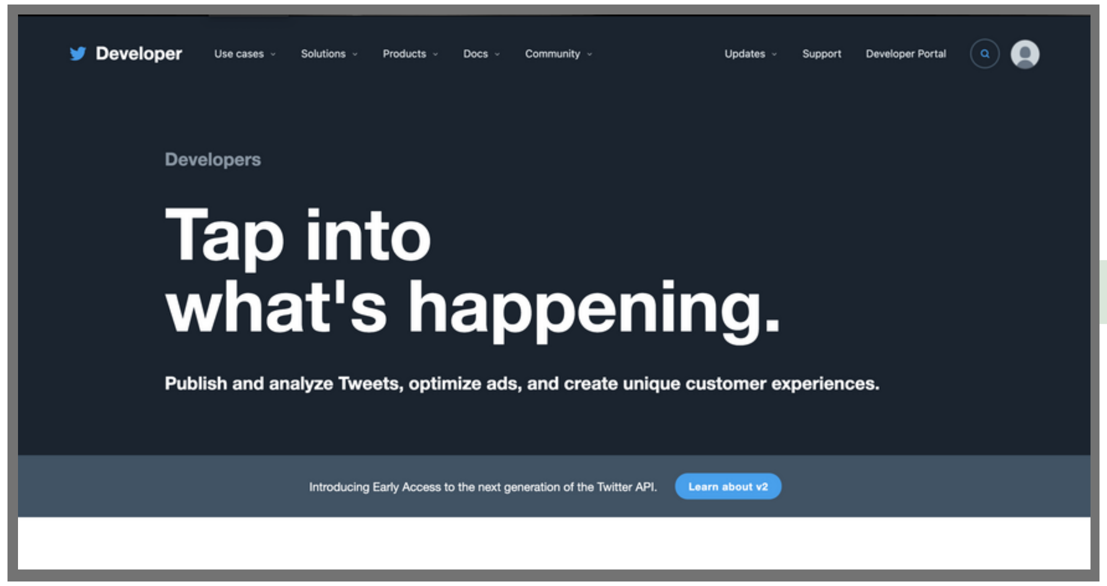
**Donald J. Trump**
@realDonaldTrump

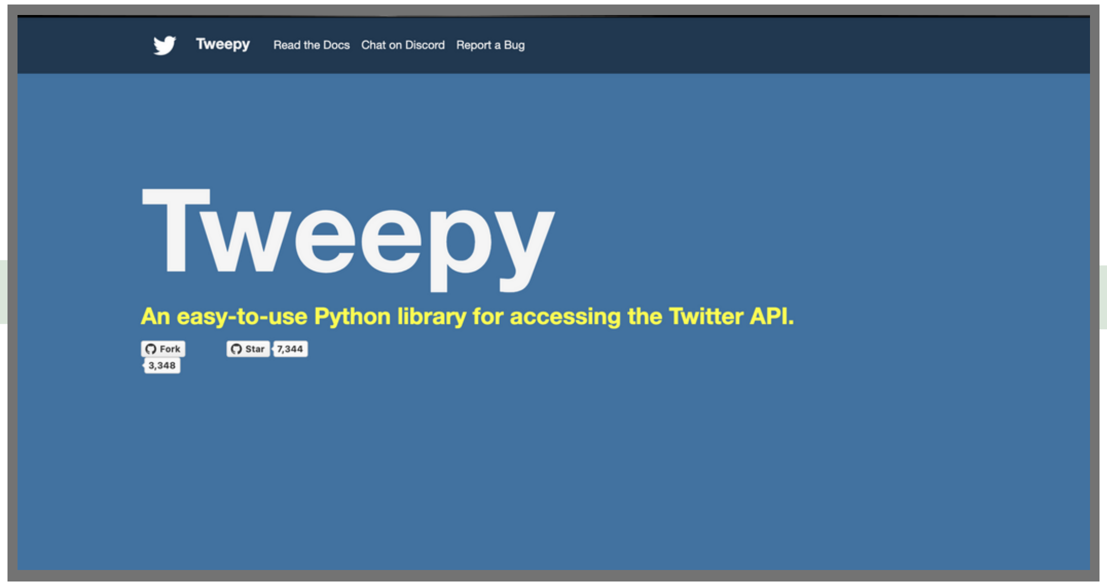
**Joe Biden**
@JoeBiden

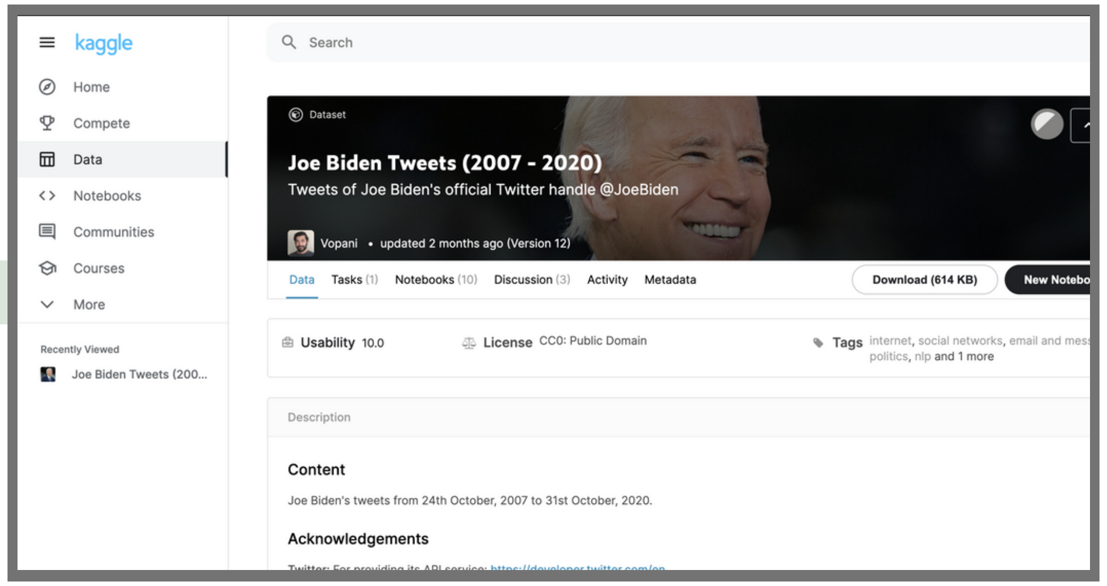
~23,750 tweets

~9,000 tweets

# Data Retrieval




1 Access Twitter API & get credentials

2 Use Tweepy API to access tweets (~3k max)

3 Increase set by combi-ning with older data

# Machine Learning

PART 2

# ML Pipeline

**GOAL**

- Generate clusters centers from the given data
- Clusters are sets of words relevant to topics
- Draw conclusions on the tweets by politicians

1 Gather Data → 2 Pre-Process Data → 3 Feature Extraction → 4 ML Algorithm → 5 Analyze Results

1

## 2 Pre-Process Data

3

4

5

### → Clean Words

- Non-alphabetical characters
- Common words (stopwords)
- Links

### → Stem Words

- Find word's root

### → Purpose

- Helps normalize text data.
- Normalized (better for ML algo)

### → Algorithm

- PySpark map, groupByKey, & sortByKey
- NLTK library functions to clean tweets & get stopwords

PART 2 | ML

3

# Feature Extraction



## → Term Frequency (TF)

- Does bag-of-words on each document, counts # of times each term occurs in data
- Stored in array v, where index *i* corresponds to a term, & *v[i]* corresponds to the count
- Used hashingTF, so collisions do have the possibility to occur

## → Inverse Document Frequency (IDF)

- Diminishes weight of unimportant words, such as “the”, “and”, etc.
- Ensures better focus on important words

## → Notes

- Data from TF-IDF can then be put into a clustering analysis algorithm
- Finds groups of prevalent words/topics in documents/speeches

1

2

3

4

## ML Algorithm

5

→ **K-Means Clustering**

- A type of clustering analysis where the goal is to divide all n observations into a user defined k number of partitions

- There are k cluster centers, where each observation is grouped with the cluster center with the nearest mean

- The number k is determined by what creates the best groups, therefore is varied per data

# Trump Results



```
Cluster 1 :
[['saw', 'thank'], ['look', 'great', 'would'], ['report', 'trump', 'donald', 'one'], ['america', 'safe', 'media', 'r
ebuild'], ['make', 'insid', 'time'], ['take', 'pour', 'peopl'], ['new', 'trump', 'soon']]

Cluster 2 :
[['candid', 'even', 'way', 'signatur', 'famili'], ['drove', 'democrat', 'excit'], ['take', 'pour', 'peopl'], ['eas
i', 'countri', 'know'], ['congress', 'border', 'disgrac'], ['fake', 'news', 'hillari'], ['america', 'mexico', 'trum
p', 'real']]

Cluster 3 :
[['disrespect', 'berra'], ['keep'], ['hemorrhag'], ['go', 'knew'], ['increas', 'want'], ['wont'], ['anthem']]

Cluster 4 :
[['justic', 'send'], ['spike'], ['mayor', 'wow'], ['polit'], ['game', 'fine'], ['month', 'fine'], ['even']]

Cluster 5 :
[['parti', 'nation', 'rove'], ['focus', 'infrastructur'], ['whether', 'dem', 'job'], ['virginia', 'terribl'], ['wa
r', 'guy'], ['made', 'member'], ['good', 'bias']]

{'FeatureExtraction': 14.03867220878601, 'PreProcessing': 2.4080276489257812e-05, 'MLAlgo': 5.051386117935181, 'Tota
lDuration': 19.09008240699768}
```

1

2

3

4

5

# Analysis (Trump)

## → Cluster 1

- Uses twitter to further his campaign
- Evident from terms “trump”, “great”, “america”, “make”, & “rebuild”

```
[['saw', 'thank'], ['look', 'great', 'would'], ['report', 'trump', 'donald', 'one'], ['america', 'safe', 'media', 'r
ebuild'], ['make', 'insid', 'time'], ['take', 'pour', 'peopl'], ['new', 'trump', 'soon']]
```

## → Cluster 2

- Uses Twitter to vent against those he deems his opponents
- Evident from terms “hillary”, “fake”, “news”, “democrat”

```
[['candid', 'even', 'way', 'signatur', 'famili'], ['drove', 'democrat', 'excit'], ['take', 'pour', 'peopl'], ['eas
i', 'countri', 'know'], ['congress', 'border', 'disgrac'], ['fake', 'news', 'hillari'], ['america', 'mexico', 'trum
p', 'real']]
```

## → Cluster 3

- Focused immensely on Colin Kaepernick & his protest
- Evident from terms such as “anthem” and “disrespect”

```
[['disrespect', 'berra'], ['keep'], ['hemorrhag'], ['go', 'knew'], ['increas', 'want'], ['wont'], ['anthem']]
```

# Biden Results

```
Cluster 1 :
[['presid', 'give'], ['trump', 'unit'], ['retir', 'biden', "vp'"], ['safe', 'vp'], ['love', 'tax'], ['mark', 'speak'], ['peopl', 'back', 'young']]

Cluster 2 :
[['well'], ['care', 'rescu'], ['led'], ['barack', 'charact'], ['retir', 'biden', "vp'"], ['key', 'show', 'keep'], ['rememb', 'choic']]

Cluster 3 :
[['background'], ['administr'], ['close'], ['protect'], ['incompet', 'profoundli'], ['everi'], ['secur']]

Cluster 4 :
[['donald', 'vote', 'campaign'], ['worker', 'make', 'rule'], ['backlog', 'everi', 'almost'], ['instead', 'vote', 'vp', 'support'], ['long', 'refurbish'], ['fight', 'stay'], ['vote', 'today']]

Cluster 5 :
[['restor', 'solv'], ['attempt', 'sunday'], ['urgent'], ['never'], ['treatment', 'veteran'], ['war', 'sacr'], ['retir', 'biden', "vp'"]]

{'PreProcessing': 0.00043773651123046875, 'FeatureExtraction': 10.655813694000244, 'MLAlgo': 8.119296312332153, 'TotalDuration': 18.775547742843628}
```

1

2

3

4

5

# Analysis (Biden)

## → Overall

- Clusters appear to be more tame than Donald Trump's
- Trump's clusters more negative (terms such as "fake news", "disgrace", "disrespect", & "terrible")
- Only negative term in Biden's is "incompetent"

## → Cluster 3

- Uses Twitter to reassure Americans that he wants to protect
- Evident from terms "protect" & "secure"

```
[['background'], ['administr'], ['close'], ['protect'], ['incompet', 'profoundli'], ['everi'], ['secur']]
```

## → Cluster 4

- Uses Twitter to urge people to vote
- Evident from terms "vote", "campaign", "support", & "fight"

```
[['donald', 'vote', 'campaign'], ['worker', 'make', 'rule'], ['backlog', 'everi', 'almost'],
['instead', 'vote', 'vp', 'support'], ['long', 'refurbish'], ['fight', 'stay'], ['vote', 'today']]
```

# Amazon EMR

PART 3

# Technical Analysis

## Colab vs. Amazon EMR

**DATA**

- 23,766 Donald Trump tweets
- 3.33 MB file
- 5 Clusters (KMeans) with 30 Iterations

| | **Google Colaboratory** | **Cluster: Amazon EMR Cluster** |
|---|---|---|
| **Hardware** | - AMD EPYC 7B12<br>- 2 CPU's<br>- 2.25 GHz | - Intel Xeon® Platinum 8175M<br>- 4 vCPUs<br>- 3.1 GHz |
| **Results - Time (s)** | Pre-Processing: 3.2e-04<br>Feature Extraction: 28.76<br>KMeans Algorithm: 9.37<br>Total Duration: 38.13 | Pre-Processing: 2.4e-05<br>Feature Extraction: 14.04<br>KMeans Algorithm: 5.05<br>Total Duration: 19.09 |

# EMR/S3 Demo

# Improvements

# Thank You!