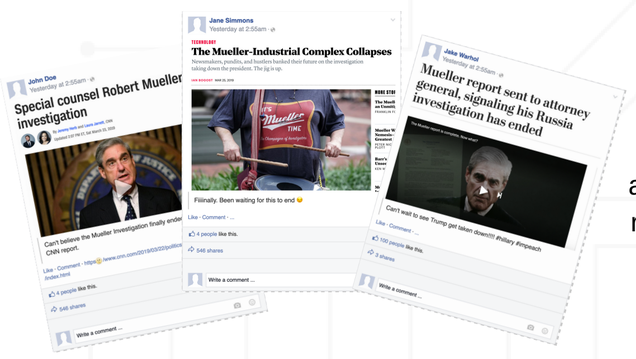
NEWSFEED ARTICLE CLUSTERING

Parker Glenn

Linguistics Undergrad



The Problem

The ability to publish has been democratized across various social media aggregators. What results is a messy newsfeed experience, full of repetition and ideological polarity.

The Solution

I aim to group together news articles covering identical events, presenting the user access to a tidy and politically diverse collection of shared news.

CUSTOM TOKENIZER → CUSTOM TFIDF → CLUSTERING → SUCCESS METRICS

```
sent = ["The United Nations ban pineapple on pizza, but Bill Gates intends to fights back."]
tokens = tokenize_and_stem_NER(sent)
print(tokens)
>>>> ['&the united nations', 'ban', 'pineappl', 'pizza', '*bill gates', 'intend', 'fight']
```

Feature (token)

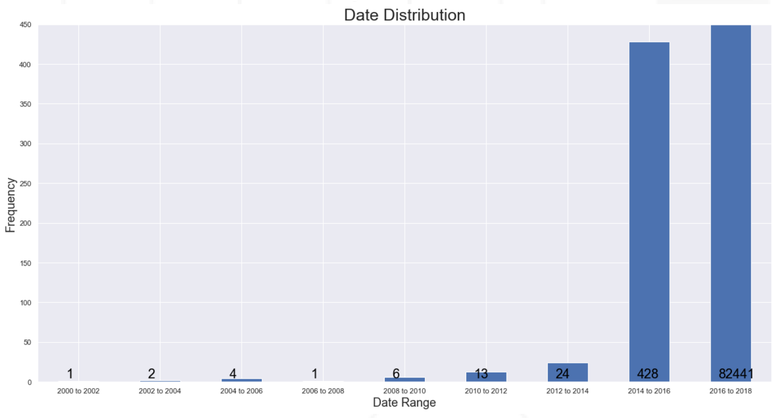
Document	0	1	2	3
0	0.975017	0	0	0
1	0	0.823339	0	0
2	0	0	0	0.704634
3	0	0	0.724984	0
4	0	0	0	0.548273


```
In [195]: from SuccessMetrics import success
     ...: y_pred = hac.labels_.tolist()
     ...: success(hac, y_pred, matrix)
     ...:
Working with 495 samples based on a spread of 520 clusters:

The F1 score for the model is 0.8545454545454545

The sillhouette score for the model is 0.09400594300211217
```

REFINE AND REPEAT

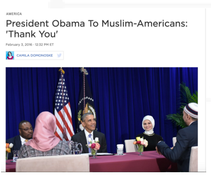
DATA PREPERATION

Utilizing a Kaggle dataset of aprox. 83,000 news articles, I noticed an extremely skewed distribution in dates.



I chose to take a sample of 1,000 documents from the right-most section, spanning from 2016 to 2018.

Each of the 1,000 articles were marked with a succinct event label to accurately describe the event it portrayed.



	Event	author	content	date	id	month	publi	title
271	Derek	obama_visits_baltimore_mosque	Camila Domo President Obama delivered remar	2/3/16	166916	2	NPR	Pres

TFIDF AND TOKENIZATION

To begin the extraction of meaningful data from the dataset of articles, I first tokenized and stemmed each article's content.

TF(x) = (# of times term x appears in document) / (Total number of terms in document for normalization)

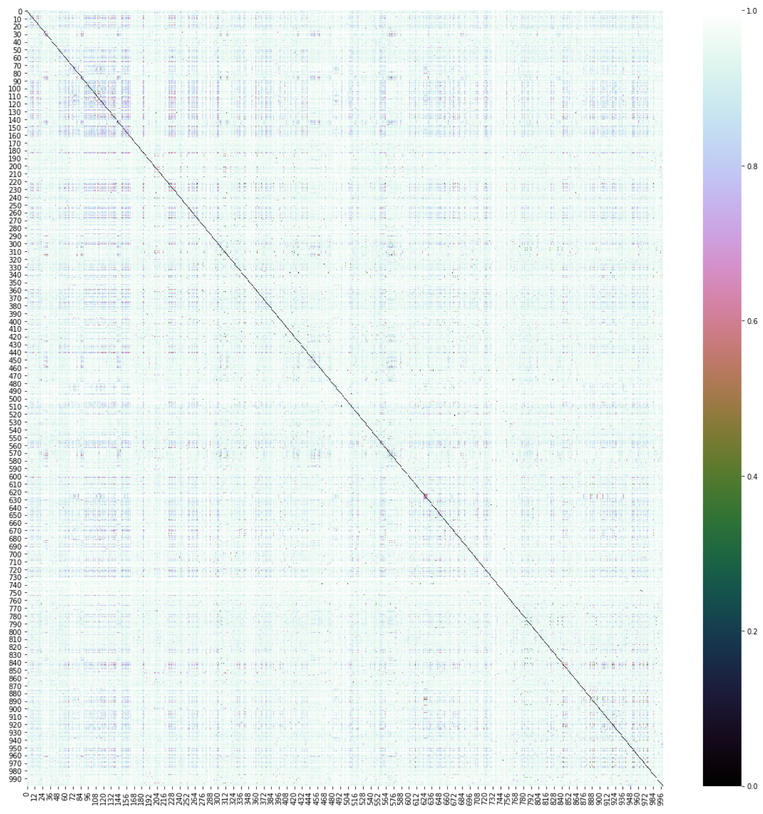
IDF(x) = log(total # number of documents / # of documents with term x in them)

TFIDF score = TF(x)IDF(x)

Feature (token)

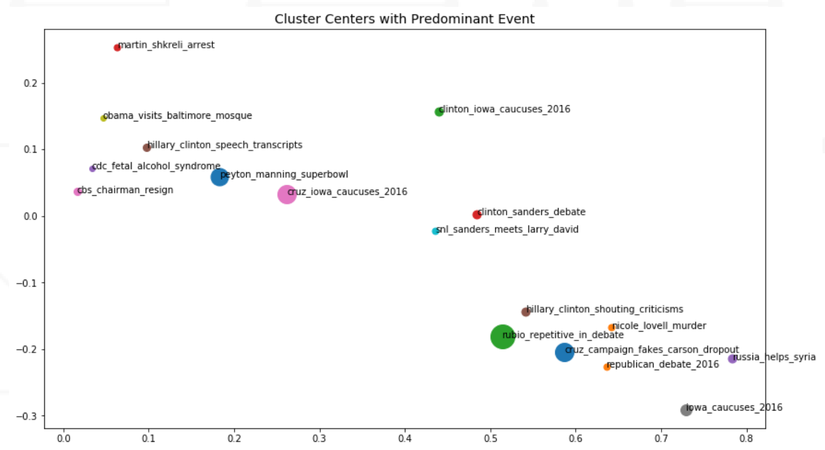
Document	0	1	2	3
0	0.975017	0	0	0
1	0	0.823339	0	0
2	0	0	0	0.704634
3	0	0	0.724984	0
4	0	0	0	0.548273

TFIDF DISTANCE BETWEEN ARTICLES



CLUSTERING

3 total clustering algorithims were tested: Kmeans, Hierarchical Agglomerative Clustering (HAC), and Birch.



DEFINING SUCCESS

SILHOUETTE SCORE

Silhouette score defines success based on the intrinsic nature of clusters.

The Sillhouette Coefficient of one data point is defined as such:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}, \text{ if } |C_i| > 1$$

Where a(i) is the mean intra-cluster distance for the sample, and b(i) is the mean nearest-cluster distance

F1-SCORE

At its core, F1-Score is the harmonic mean of precision and recall.

I defined a cluster's "predominant" event as that which occured most frequently in a cluster. If a tie occurred, their ratios were compared, where:

ratio(event) = (# of times event is used in the cluster) / (# of total occurrences of event across the whole dataset)

y_true was constructed through the following steps:

1. Find the event which occurs most frequently within a cluster.

2. If there is a tie, invoke the ratio function to choose one event.

3. After steps 1 & 2 have been iterated over the entire dataset, if two or more clusters share the same predominant event, invoke the ratio function as needed to decide on a one-to-one event-to-cluster pairing.

SUCCESS (NO ENTITY WEIGHTING)

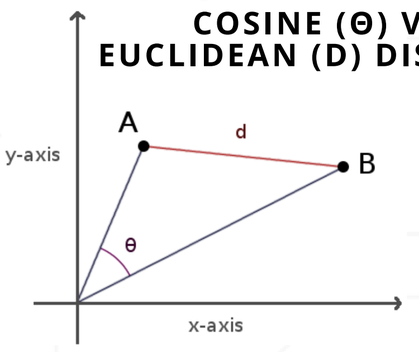
Clustering	F1 Score	S-Score
Kmeans	.905	.066
HAC	.903	.087
Birch	.920	.081

ENTITY WEIGHTING

After utilizing the custom tokenizer to divide entities into Person and Non-Persons, I began to experiment with weightings.

The TF score of certain entities were manipulated to grant them greater moving power in the TFIDF matrix.

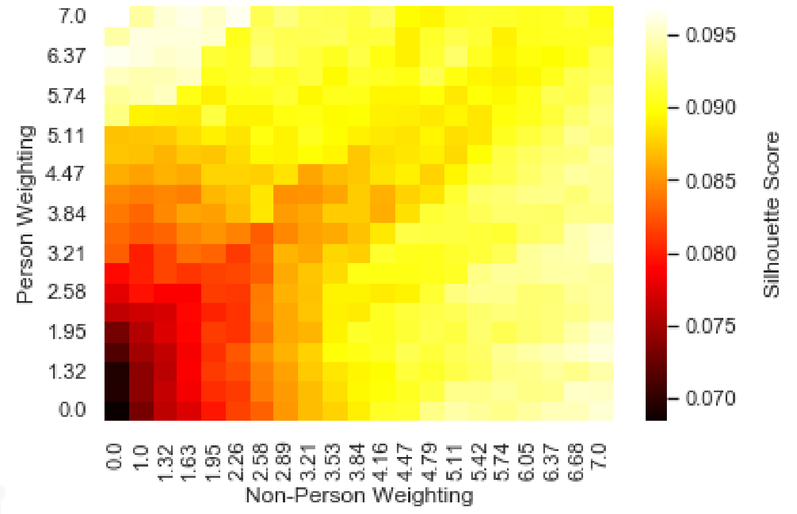
COSINE (Θ) VS. EUCLIDEAN (D) DISTANCE



EXPLORING ENTITIES

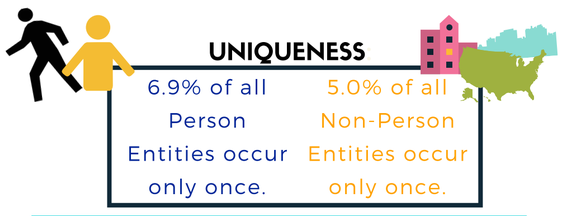
GPE: Countries, cities, states
ORG: Organizations
LOC: Non-GPE Locations
EVENT: Named hurricanes, sports events, etc.
FAC: Buildings, highways, etc.
PERSON: People, including fictional

ENTITY WEIGHTING AND SILHOUETTE SCORE

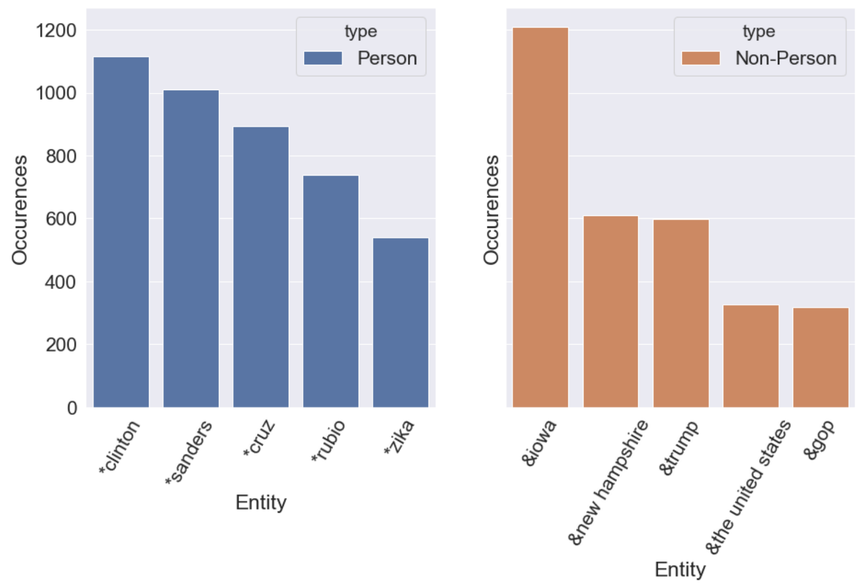


Person Entities occurred 25,901 times

Non-Person Entities occurred 32,470 times



TOP 5 MOST POPULAR ENTITIES



After various rounds of hyperparameter testing with F1 Score and Sillhouette Score, a **Person Weight of 2.27** and a **Non-Person Weight of 6.36** yielded the best success rates.

SUCCESS (WITH ENTITY WEIGHTING)

Clustering	F1 Score	S-Score
Kmeans	.850	.075
HAC	.855	.094
Birch	.869	.086