Power Outage Detection:
A Social Media Driven Approach

General Assembly DSI (NYC)

Boom Devahastin Na Ayudhya
Chris Lee
Henry Blais

January 2019

The Problem

How to leverage user social media updates using NLP techniques to identify power outage hotspots?

Vision

- During disasters, residential areas experience power outages which can last for days
- Emergency response agencies need a timely and accurate way to respond to them.

Issue

- Traditional methods: live feeds from utility companies, satellite data of light emissions
- However, extremely expensive and difficult to maintain

Proposition

- Observed generation's ever growing dependence on social media over the past decade
- More cost-efficient path to pursue
- Using NLP tools, possible to leverage social media updates to identify power outage hotspots

The Problem

How to leverage user social media updates using NLP techniques to identify power outage hotspots?

The Approach

- **Social Media Platform Focus:** Twitter
- **Data Collection:** Webscrape Tweets using Twitter API; city/county geodata
- **Data Cleaning:** Filter out irrelevant fields; addressing missing data.
- **Exploratory Data Analysis:** Search for ways to help systematic construction of bag of words
- **Preprocessing**: Bag-of-Words construction, Tokenization, Vectorization
- **Classification Models**: Word2Vec, “Sentiment”-style analysis

Methodology: Data Collection

Determine which local areas to cover

Which neighborhoods/area to identify?

- New England: 6 East Coast states (MA, ME, VT, NH, CT, RI)

City/County Data Set

simplemaps Geographic Data Products

United States Census Bureau

USGS science for a changing world

What does this city/county data include?

- **State**
- **City**
- **County**
- **Latitude and Longitude (city center)**
- **Time Zone**
- **ID**
- **Zip code**
- **Population**
- **Density**

GENERAL ASSEMBLY

Determining social media options and scraping options

Social Media Options

Facebook	Twitter
difficult to navigate API	easy to navigate API
high restriction to access API	easier to get a permission using API

Twitter Scraping Options

	TwitterScraper	Twitter API
Pro	can scrape historical data	Geological data (but very limited)
Con	no geological data	API has limits of tweets scraping gives only the most recent 7 days data

Methodology: Data Cleaning

Scraping Tweets by TwitterScraper

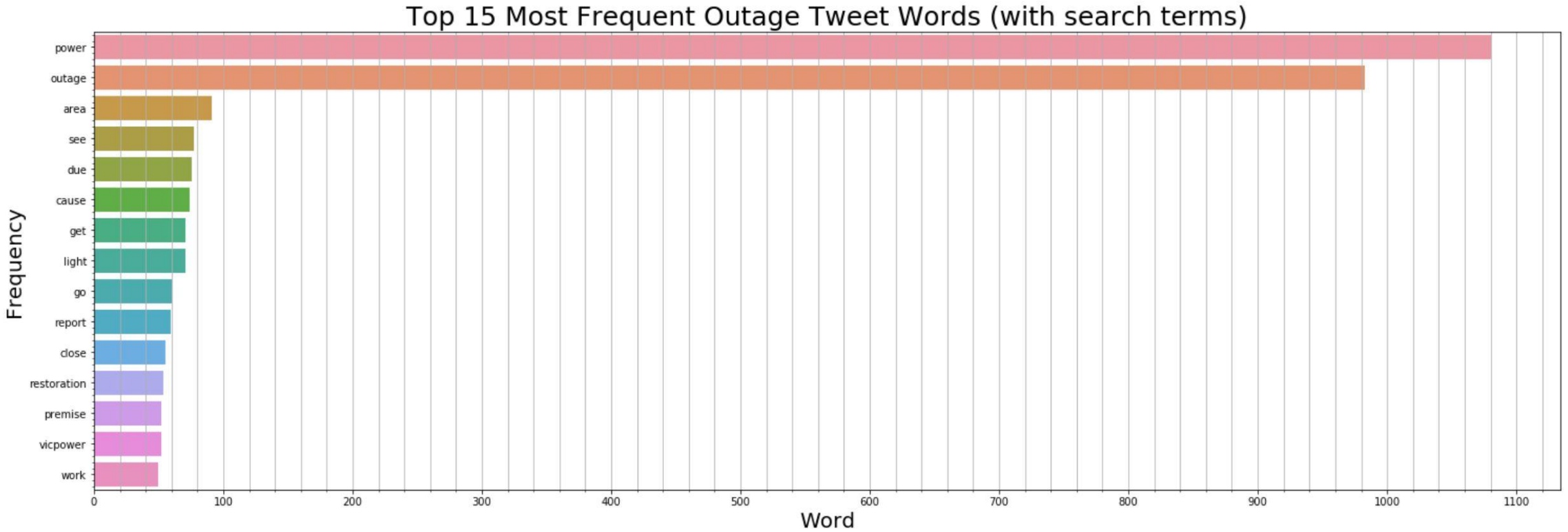
Problems Encountered (TwitterScraper)

- Twitter locked us out of API partway through data collection
- Backup data - around 1000 tweets with no location
  - ➢ Workaround: randomly assign the city location data to backup data!

Back Up Data: `query_tweets` parameters

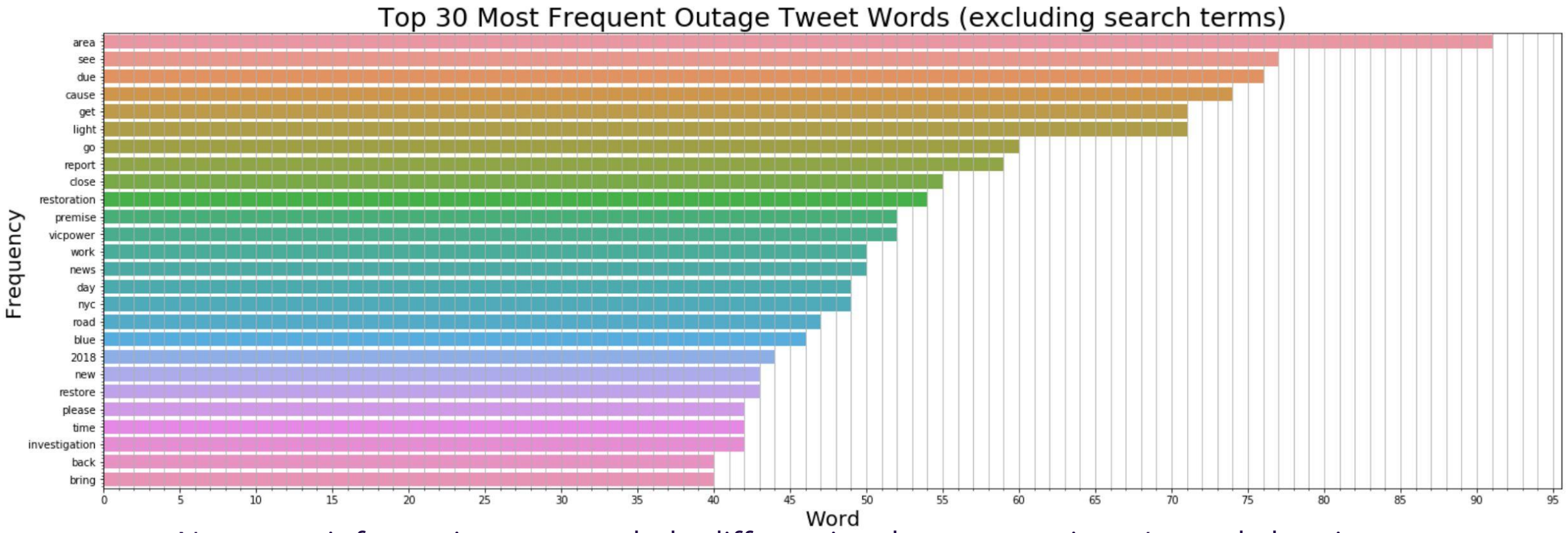
- Search term: 'blackout', 'blackouts', 'outage', 'outages', 'power outage'
- Periods: 01/01/2018 ~ 12/31/2018

Examining the Distribution of Top Words in Outage Tweets



- Huge differential b/w frequency of query search terms and other terms as expected
- “blackout” search term was not even in the top 30

Examining the Distribution of Top Words in Outage Tweets



- Not many informative terms to help differentiate between serious / actual electric blackouts vs. irrelevant blackouts
- “light” as potential serious blackout term; “vicpower” as example of irrelevant term

Identifying Related Words: LexVec Word Embedding

General Principle

- Start with bag of words from power outage terminology article:

'blackout', 'power', 'outage', 'electric', 'electricity', 'electrical', 'transformer', 'watt', 'wattage', 'arc', 'circuit', 'breaker', 'cable', 'fault', 'conductor', 'fuse', 'riser', 'insulator', 'meter', 'interruption', 'maintenance', 'relay', 'grid', 'severe', 'weather', 'storm', 'substation', 'surge', 'switch', 'switchyard', 'station', 'transmission', 'system', 'lines', 'line', 'frequency', 'voltage'

- Compare each term to *every* word in English Wikipedia's 2015 bag of words
- Algorithm computes a similarity score from -1 to 1
- Take top 20 words, but filter out any with score < 0.40

Sample Case

Term from Starting Bag of Words	Top 20 Similar Words
'electricity'	'energy', 'megawatts', 'power', 'renewable', 'renewables', 'generators', 'utilities', 'hydropower', 'kilowatts', 'hydroelectricity', 'gas', 'kilowatt', 'baseload', 'kwh', 'heating', 'fuel', 'mwh', 'gigawatts', 'sewerage', 'biomass'

Tokenization

General Principle

- Process of melting a body of text down to constituent words (tokens).

- Create a nested list (i.e. a list of each tweets' list of tokens) to later iterate through:

Tweet	Inner List (of Clean Tokens)	Outer List (of Inner Lists)
"I bought a portable cell charger in 2018."	['bought', 'portable', 'cell', 'charger']	[['bought', 'portable', 'cell', 'charger'], ['massive', 'hits', 'southern', 'zim', 'twimbos', 'newsday', 'zimbabwe']]
"Massive power outage hits southern Zim http://bit.ly/2EW2Enj\xa0 #263Chat #Twimbos #NewsDay #Zimbabwe"'	['massive', 'hits', 'southern', 'zim', 'twimbos', 'newsday', 'zimbabwe']	

- Each tweet will have a list of cleaned tokens
  - ➢ List length = equal to number of tokens in that tweet.
- The outer list will contain these inner lists
  - ➢ List length = number of tweets in our scraped dataset

- Format is memory-efficient and makes life easier for next step

Word2Vec Vectorization – Brief Introduction

General Idea

- Model based on shallow neural networks that converts words into something computer can understand (e.g. 300 dimensional vector)

- Each word is assigned a vector positioned in the space such that words in similar contexts will be positioned closely together

How are we using it?

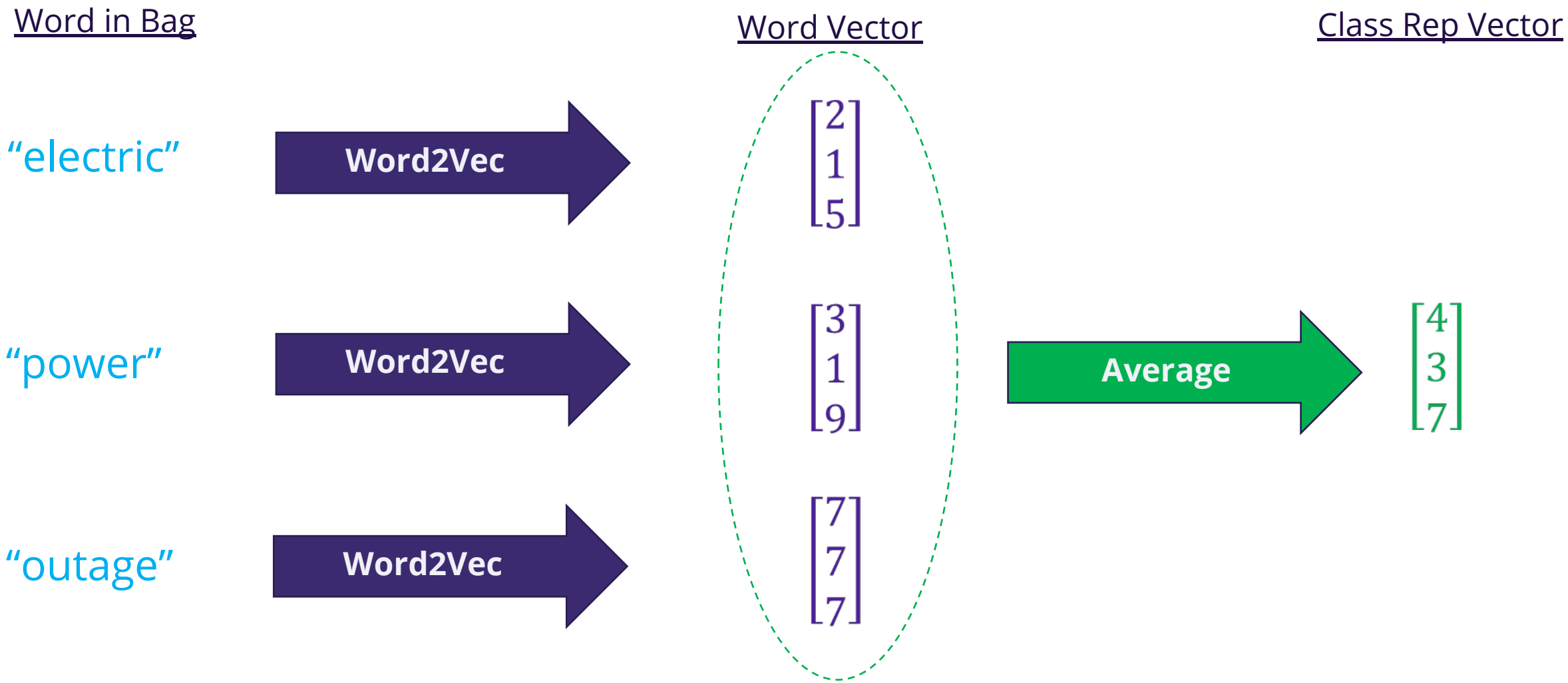
1. Train on Google News corpus (3M words)

2. Define bag of words for 2 classes:
   - ➢ Serious / actual electrical blackout
   - ➢ Non-serious / irrelevant blackouts (e.g. Netflix, drunken night)
3. Obtain **representative vector for each class** by vectorizing all words in bag, then average

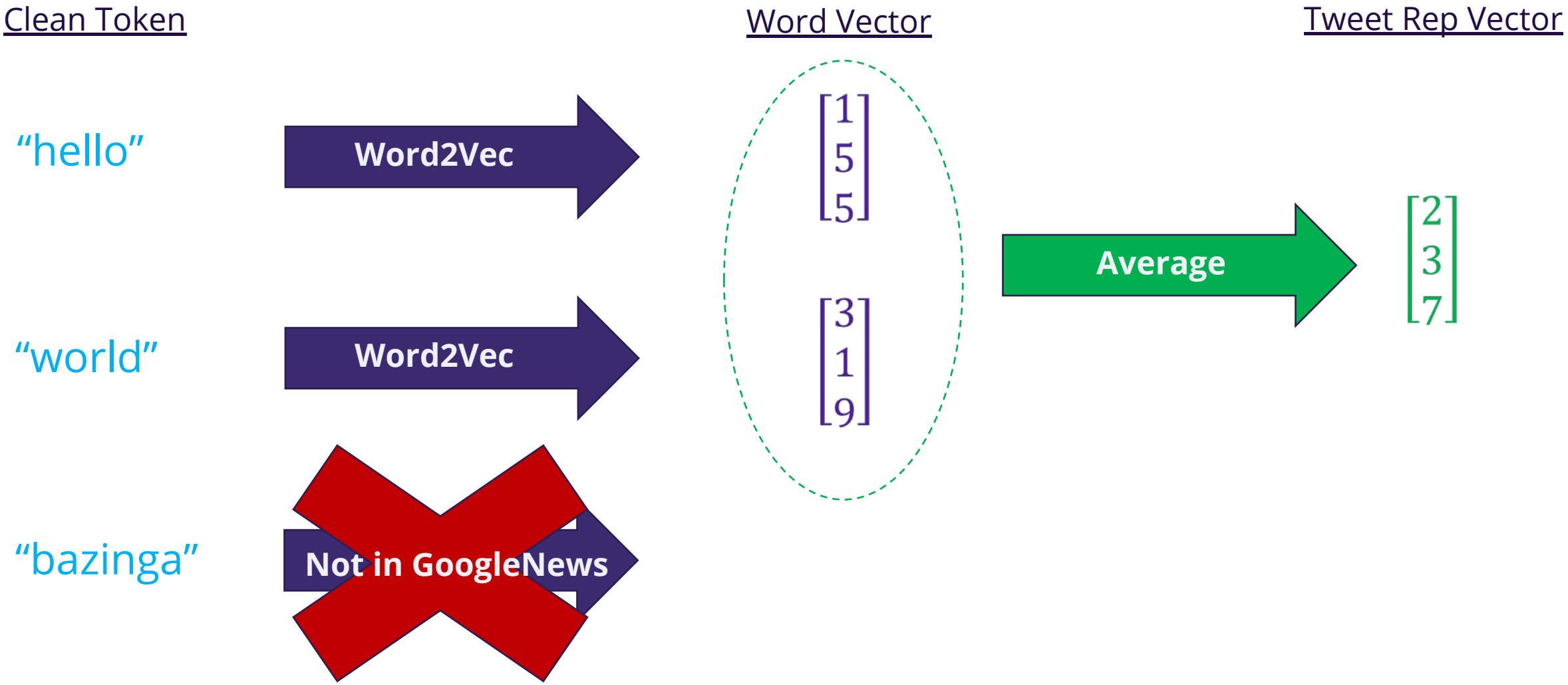
Defining Bag of Words for Classes

Class	Bag of Words
Serious	'electricity', 'electrical', 'conedison', 'con edison', 'generator', 'generators', ‘failure', 'malfunction', 'fuse', 'blow', 'explode', 'power grid', 'breaker', 'loss', 'dark', 'darkness', 'pitch black', 'blind', 'massive', 'major', 'serious', 'inclement', 'surge', 'storm', 'solar flare', 'alert', 'light', 'lights'
Irrelevant	'netflix', 'hulu', 'time warner', 'twc', 'at&t', 'att', 'verizon', 'tmobile', 't-mobile', 'phone', 'internet', 'wireless' 'conference', 'prepare', 'meeting', 'strength', 'training', 'discipline', 'vicpowers', 'baseball'

Example – class_bag_of_words = ["electric", "power", "outage"]



Example – Tweet: "Hello World! #Bazinga"



Classification Modeling using "Sentiment" Analysis

Rationale

- Dealing with an unsupervised learning problem since no true target variable for class
- Nevertheless, need to assign classes to each tweet

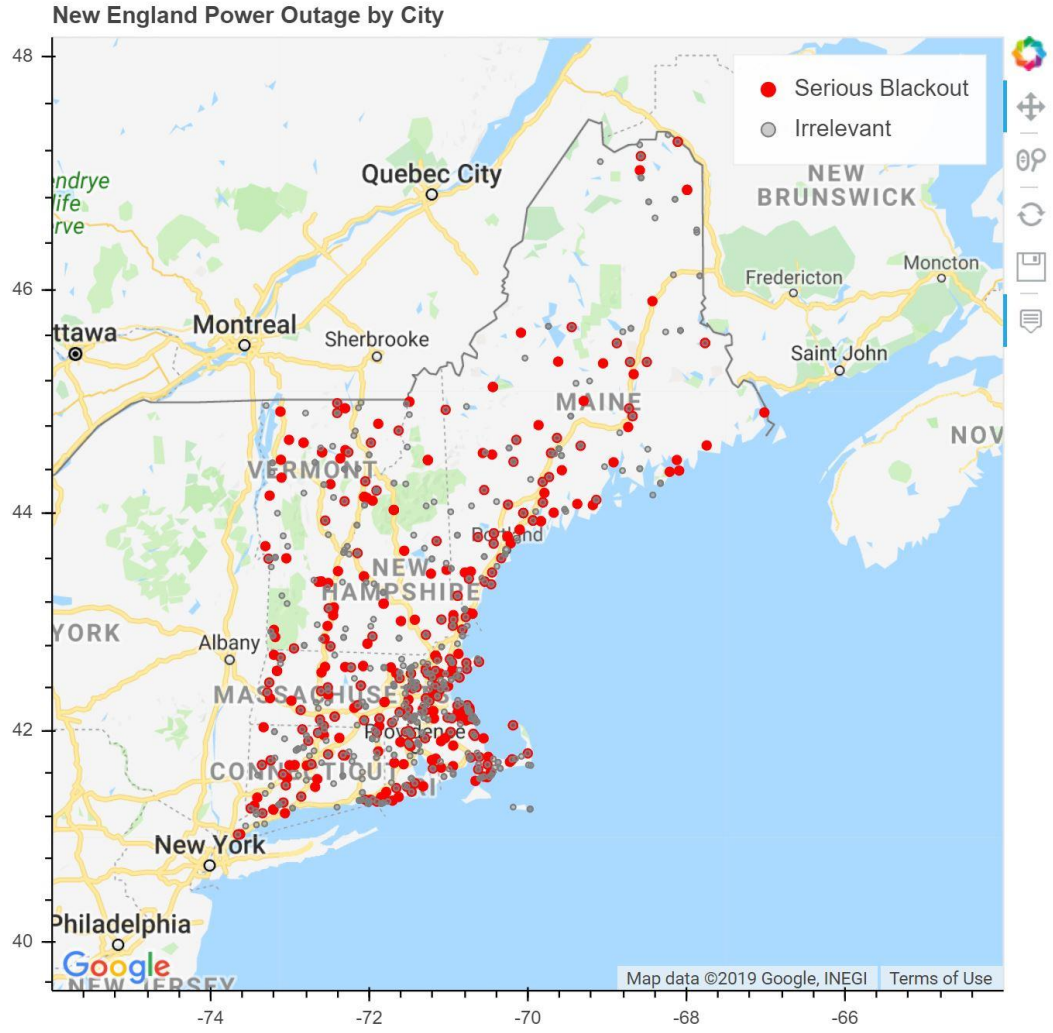
General Idea

- Use cosine similarity on representative vectors (a sentiment analysis based approach)
- For each tweet,
  1. Compute cosine similarity between **tweet rep vector** vs. **serious rep vector**.
  2. Compute cosine similarity between **tweet rep vector** vs. **irrelevant rep vector**.
  3. Classify tweet based on relative magnitude of cosine similarity:
     - ➢ If cosine similarity to **serious rep vector** is higher, then assign serious_blackout=1
     - ➢ If cosine similarity to **irrelevant rep vector** is higher, then assign serious_blackout=0

- Then combine with (latitude, longitude) data to visualize hotspots:

Results: Visualization

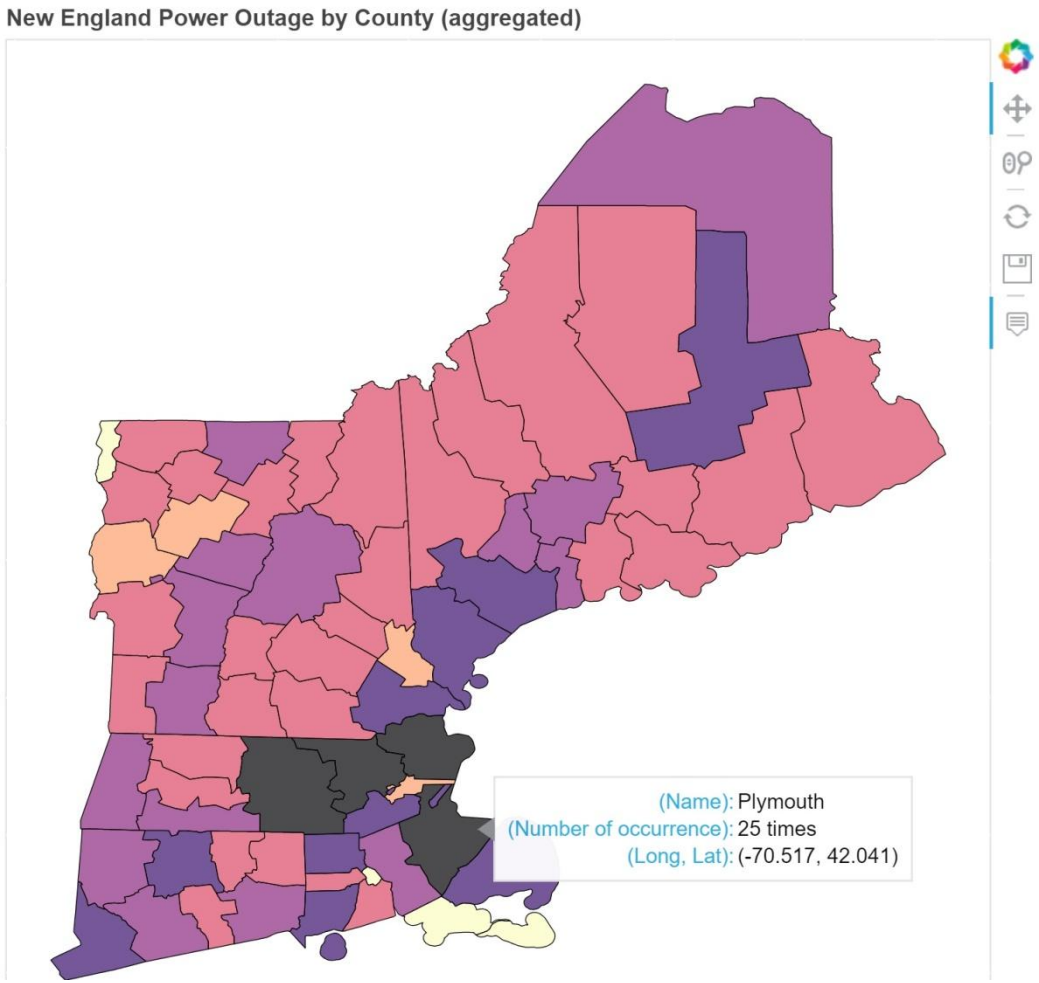
Bokeh Hotspot Map of New England Tweet Outages by Category (City Basis)



	state	name	lat	lng	blackout
0	MA	Middlesex	42.4186	-71.1638	1.0
1	ME	Aroostook	46.4990	-67.8688	0.0
2	CT	New London	41.3145	-72.0087	0.0
3	VT	Chittenden	44.3759	-73.2265	0.0
4	NH	Grafton	44.0364	-71.6895	1.0

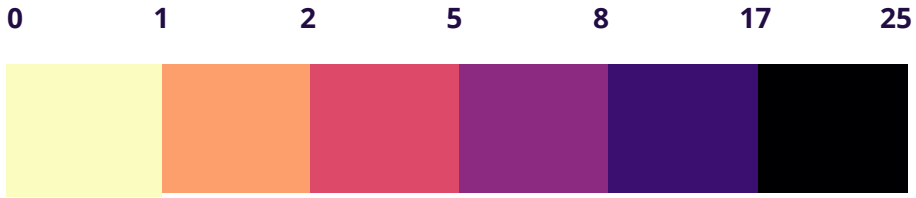
Results: Visualization

Bokeh Heat Map of New England Power Outages (County Basis)



	num_occurrence
MAPlymouth	25
MAMiddlesex	23
MAWorcester	21
MAEssex	17
MABarnstable	14

Frequency Color Key



Key Takeaways and Improvements

- Class representative vectors for serious vs. irrelevant had cosine similarity of 0.3
  - ➢ Implication: Somewhat unrelated, but not necessarily perfectly orthogonal.
  - ➢ Refine class corpora to reduce cosine similarity of **serious blackout** vs **irrelevant**

- While this is a **proof of concept** model on *historical* data, but results are realistic.
  - ➢ Ideal Final Product: Tool utilizing Twitter's live API stream
  - ➢ Requirements from Client: paid subscription to Twitter's API

- Tackled as unsupervised learning problem due to absence of true classifications to train on.
  - ➢ Alternative Route: S**upervised learning problem**
  - ➢ Requirements from Client: data entry to manually categorize a sample of tweets and provide us that data

- Infer locations by including list of cities/neighborhoods in query search terms.