NLP Text Classification & Sentiment Analysis on Disaster Tweets

Nicole Hong

LHL Demo Day: April 28, 2022

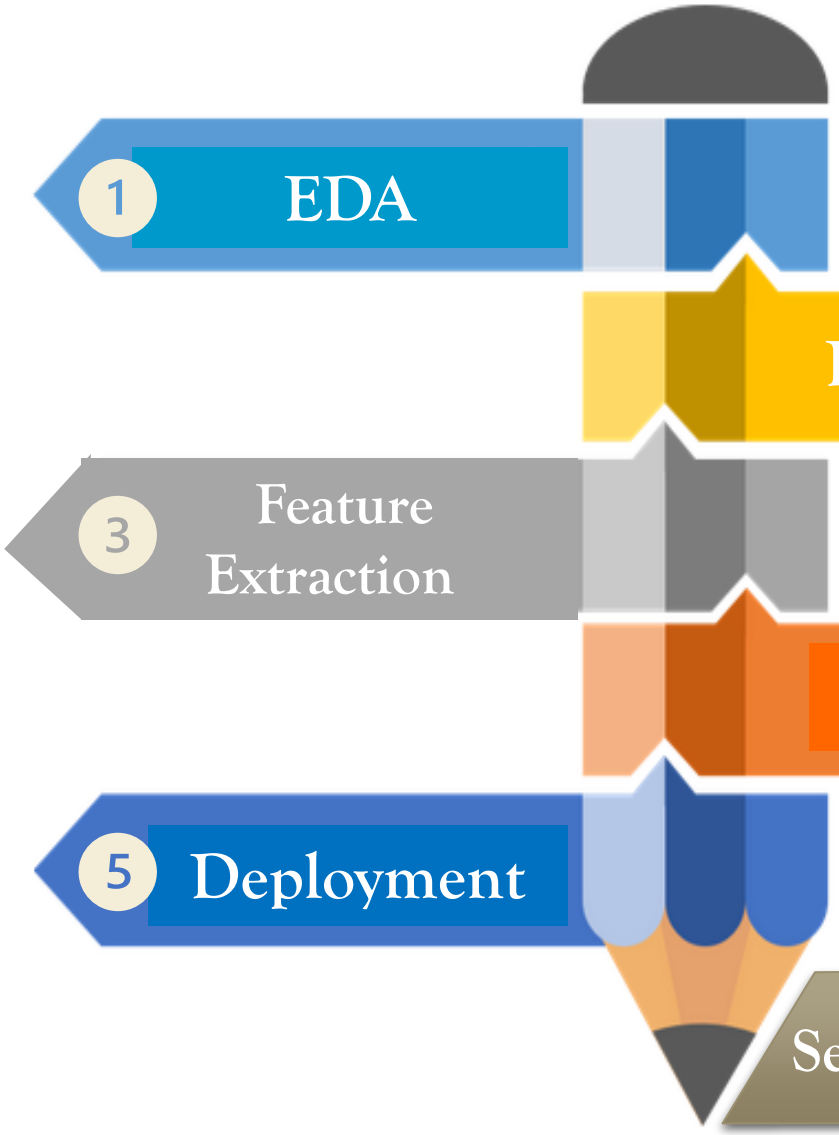
Why this Kaggle?

- Twitter as an important communication channel in times of emergency
- Smartphones enable announcement of an emergency in real-time
- More agencies interested in Twitter & Emergency Response Management

Objective

- Identify disaster related tweets with high accuracy (Classification)
- Determine level of crisis for better response to disasters (Sentiment Analysis)

Project Workflow



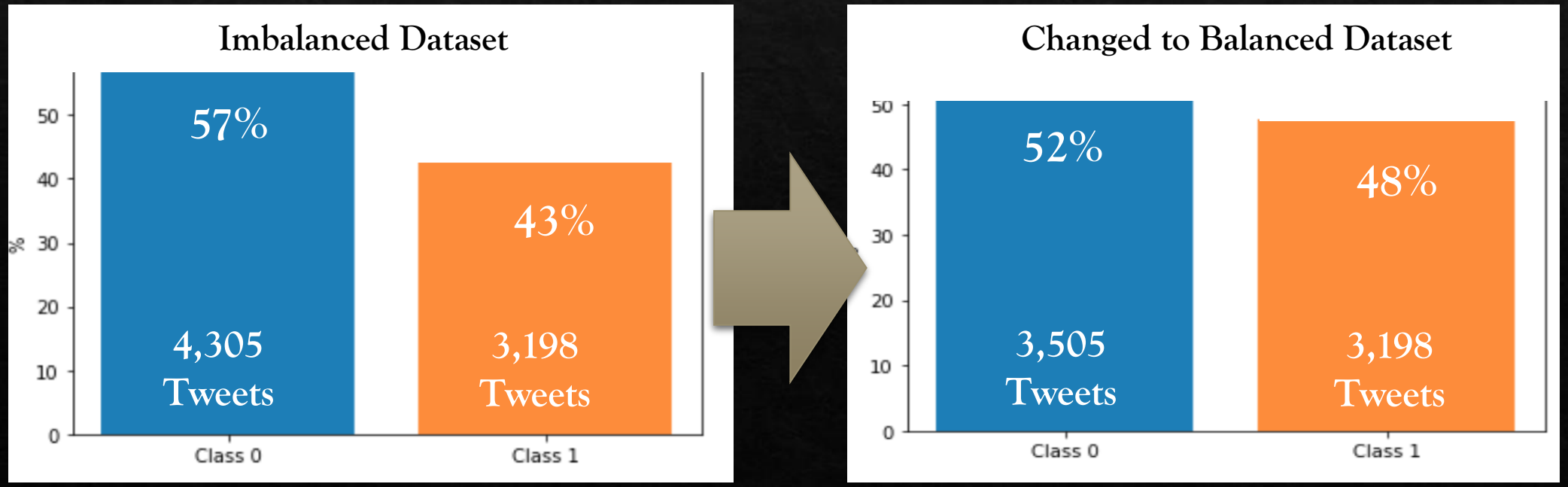
1. EDA — Text Patterns, Counts, Frequency & Visualizations
2. Preprocessing — Tokenization, URL/ Contraction Removal & Numbers /Special Characters Removal
3. Feature Extraction — N-Grams (i.e. Text Sequence), BOW & TFIDF
4. Modelling — Supervised Machine Learning, Deep Learning & Transformer Models
5. Deployment — Prediction & Model Evaluation
6. Sentiment Analysis — VADER Sentiment Lexicon & Sentiment Scores

About Kaggle Data

- Small Text Data:
  → Train Data = 7,613 entries
  → Test Data = 3,263 entries

- Variables with Keywords, Locations, Text and Target
- 28% of Missing Data in Keywords & Location
- Tweets in August 2015
- Not designed for Sentiment Analysis

Target Class Balance – Training Dataset



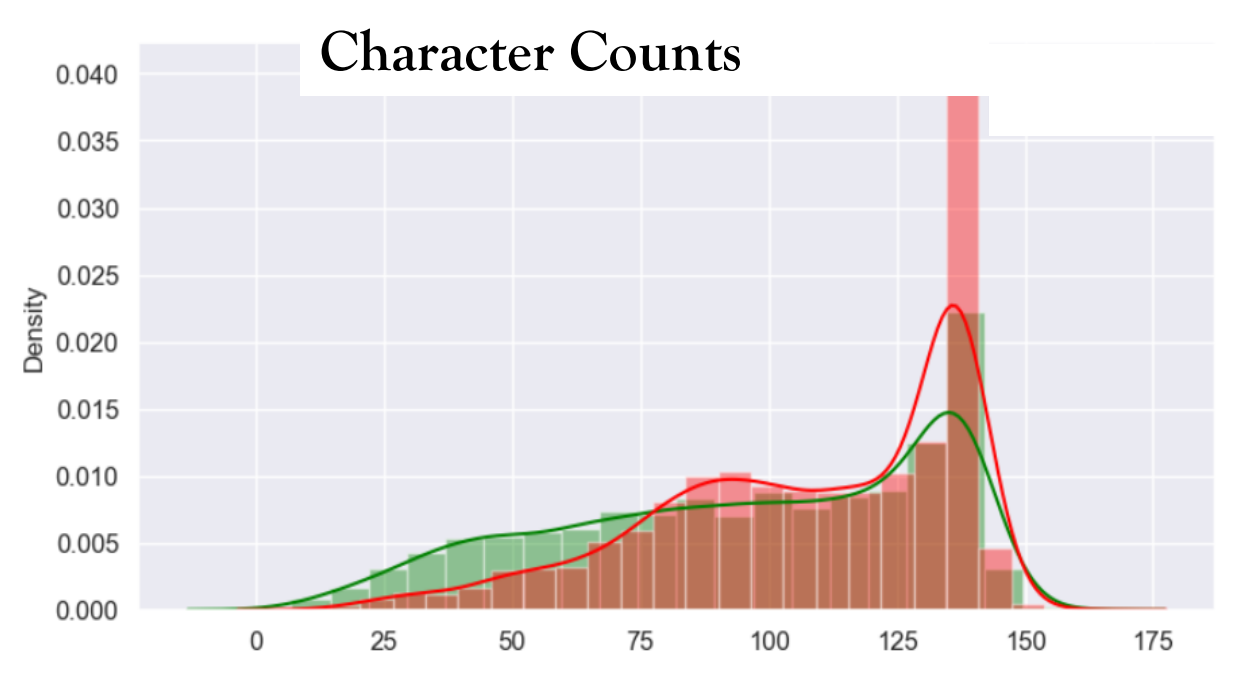
Class 0

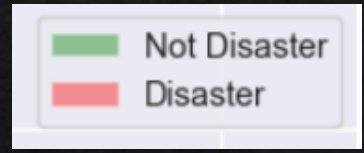
Non-disaster related Tweets

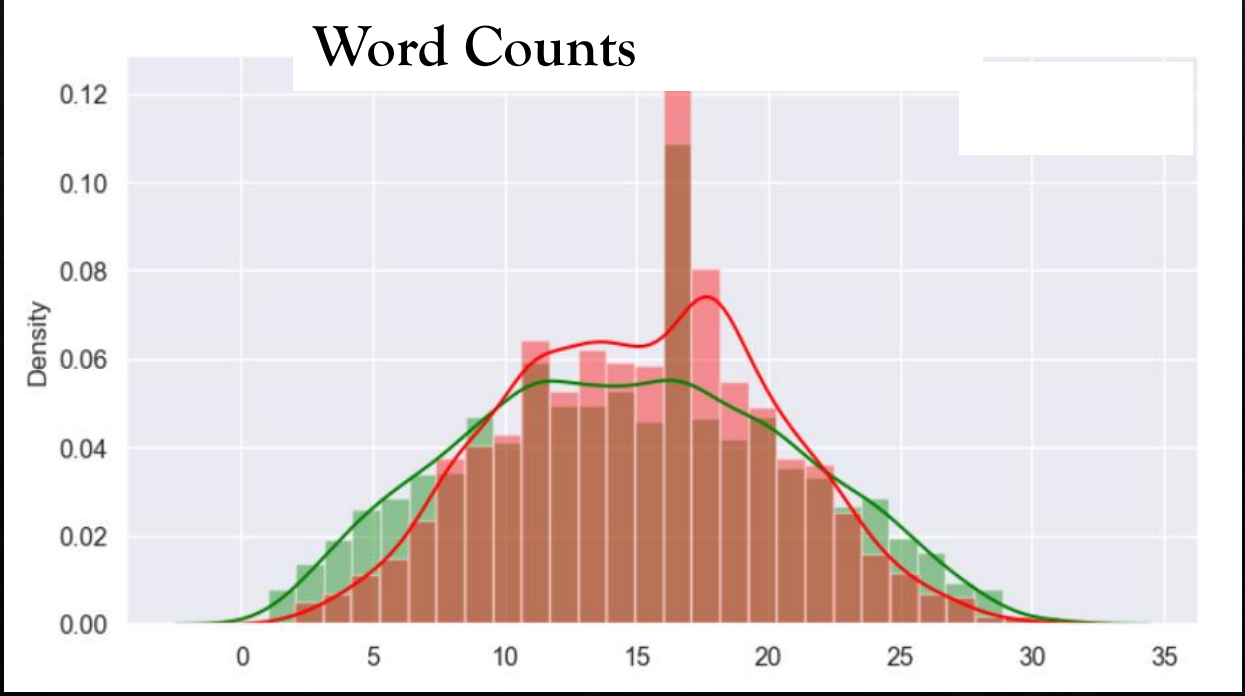
Class 1

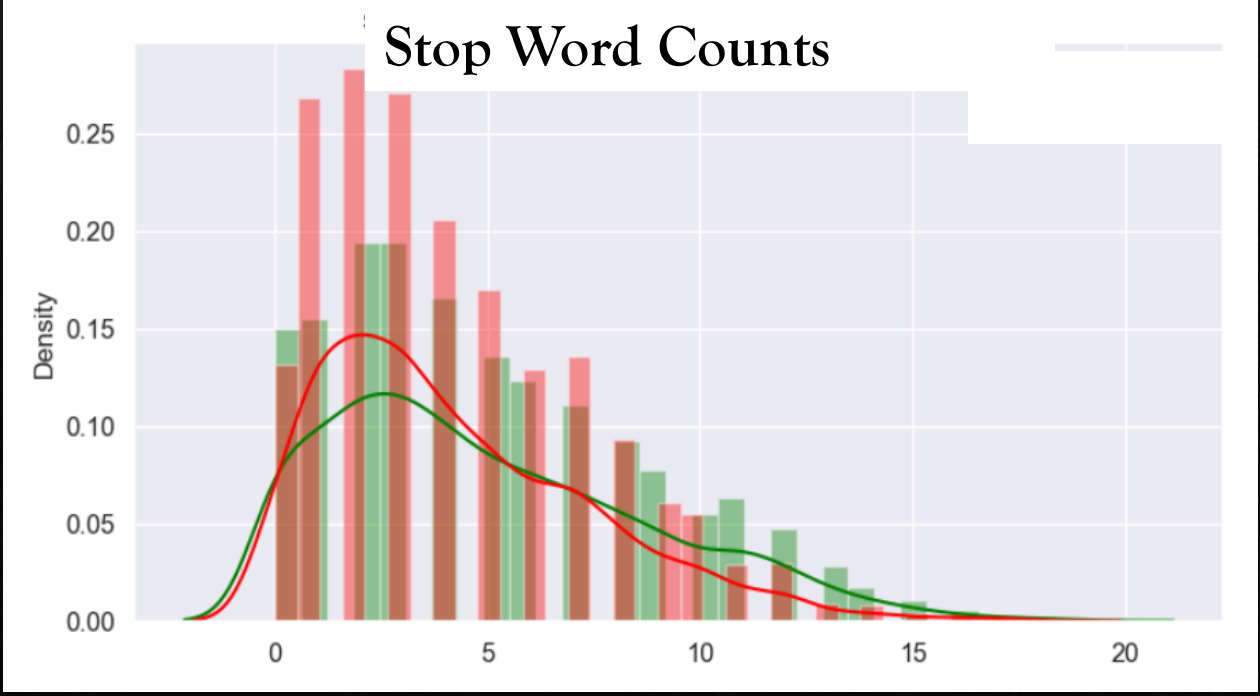
Disaster related Tweets

EDA: Distribution Plots

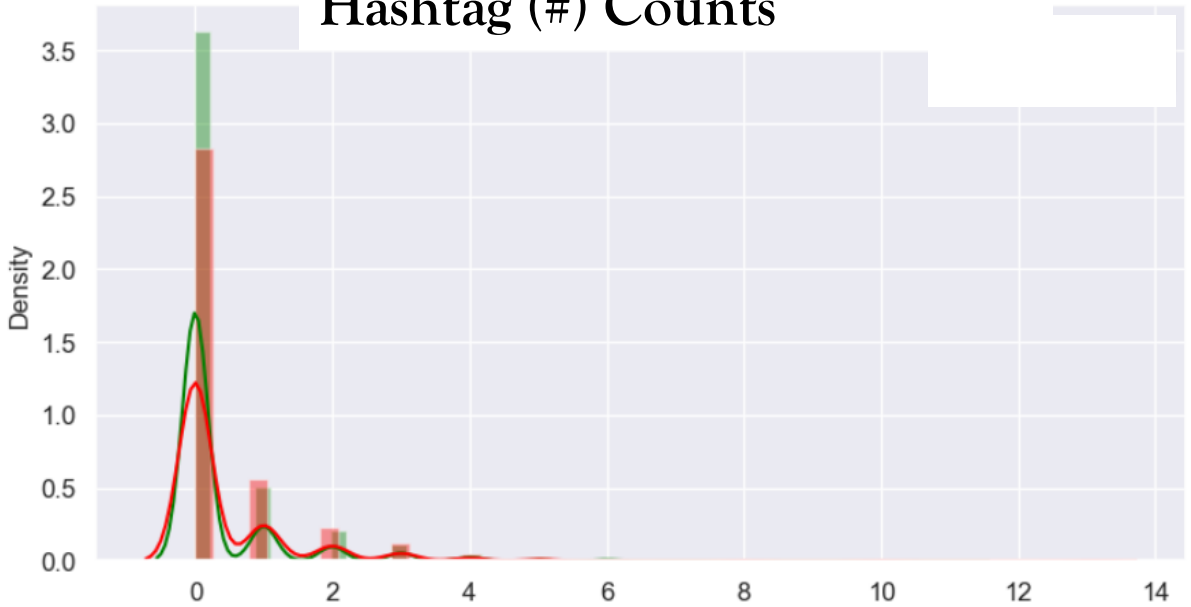




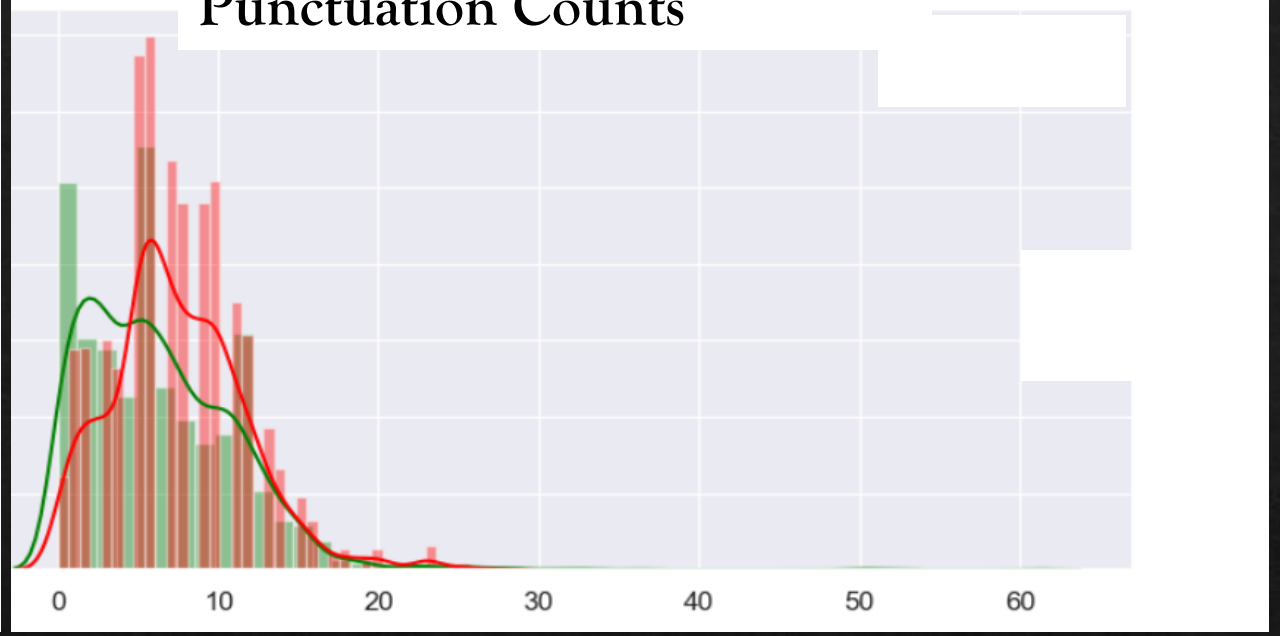




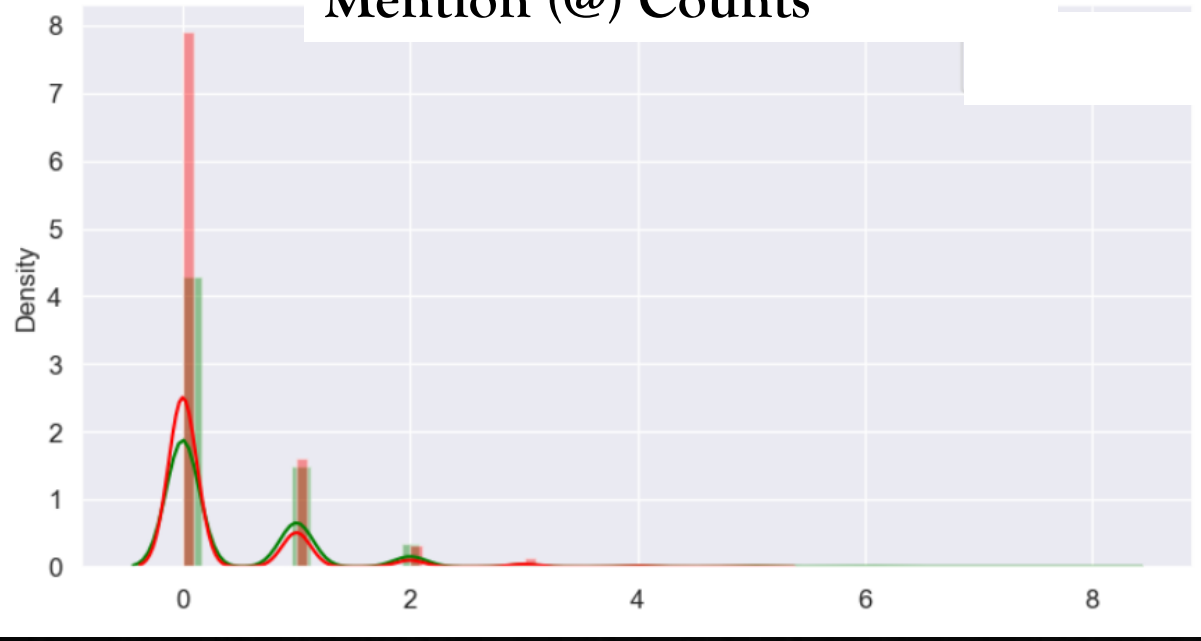
Hashtag (#) Counts



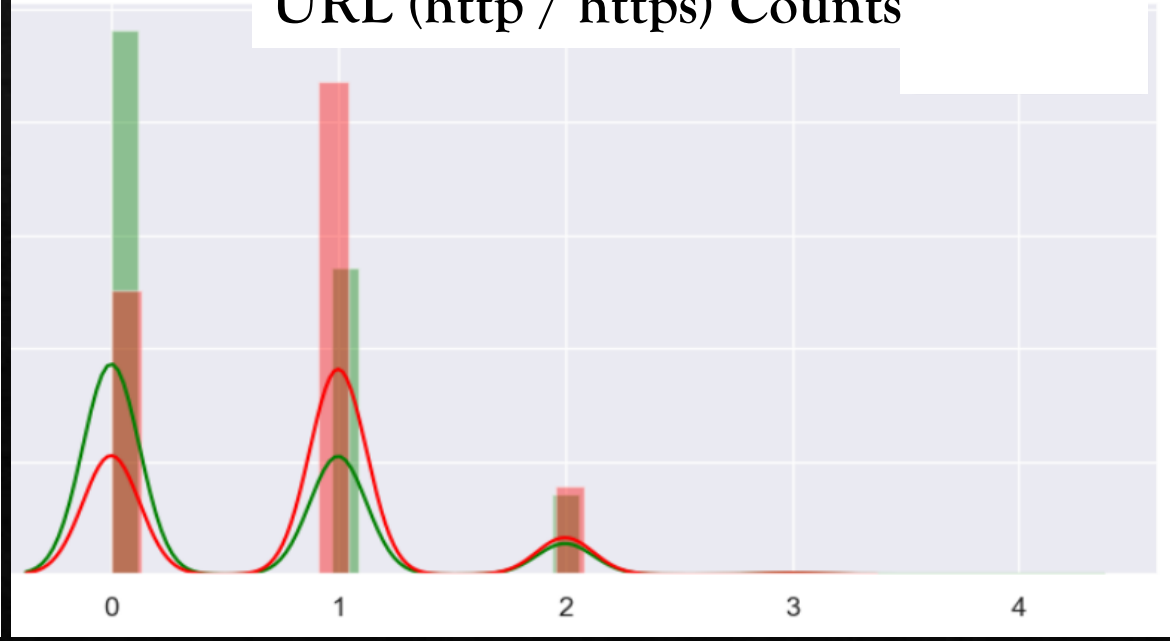
Punctuation Counts



Mention (@) Counts

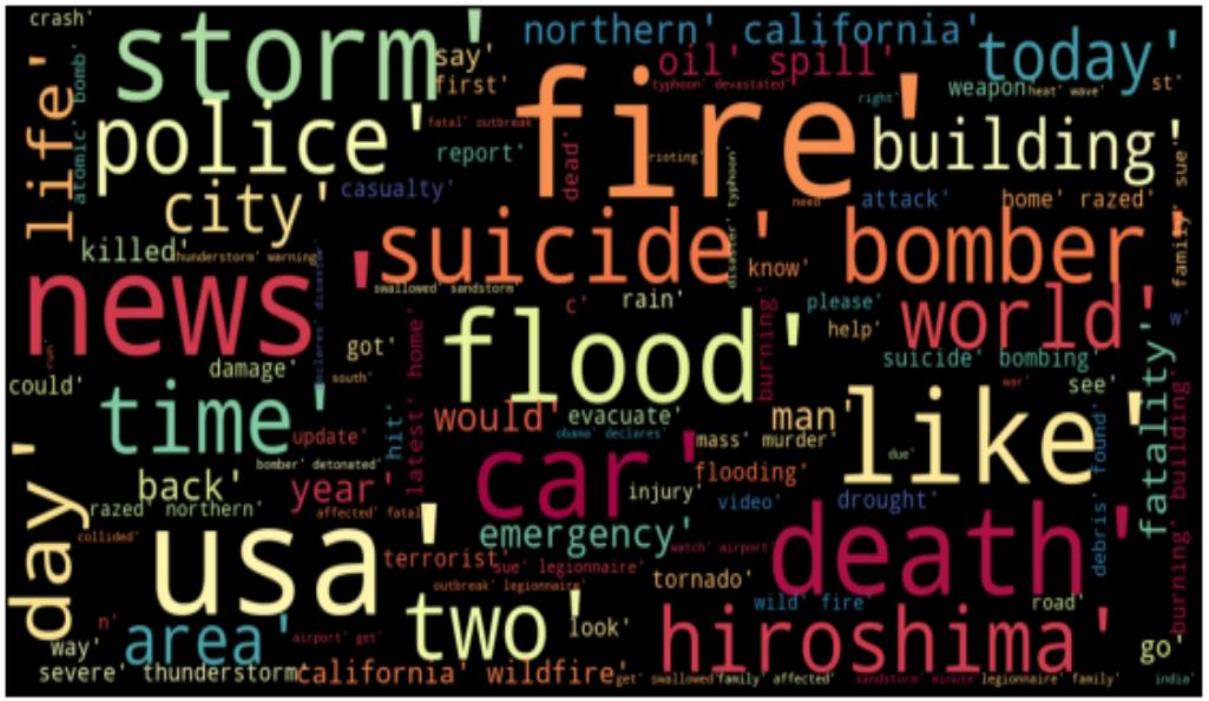


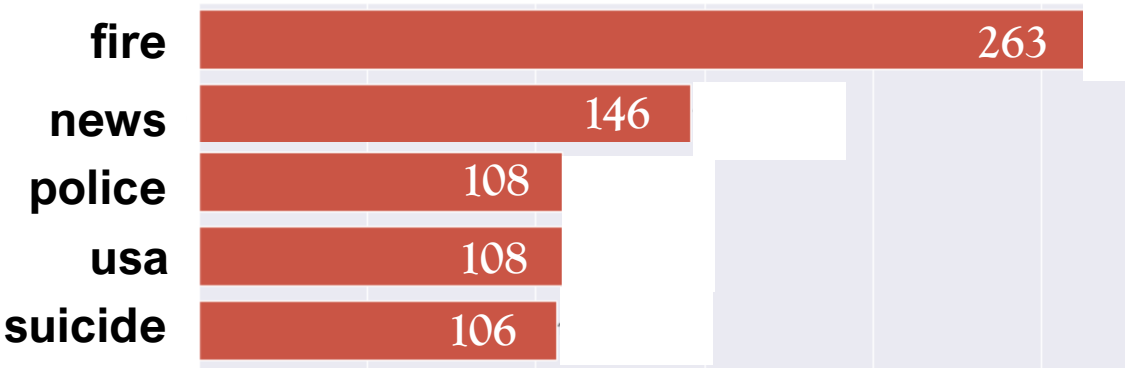
URL (http / https) Counts



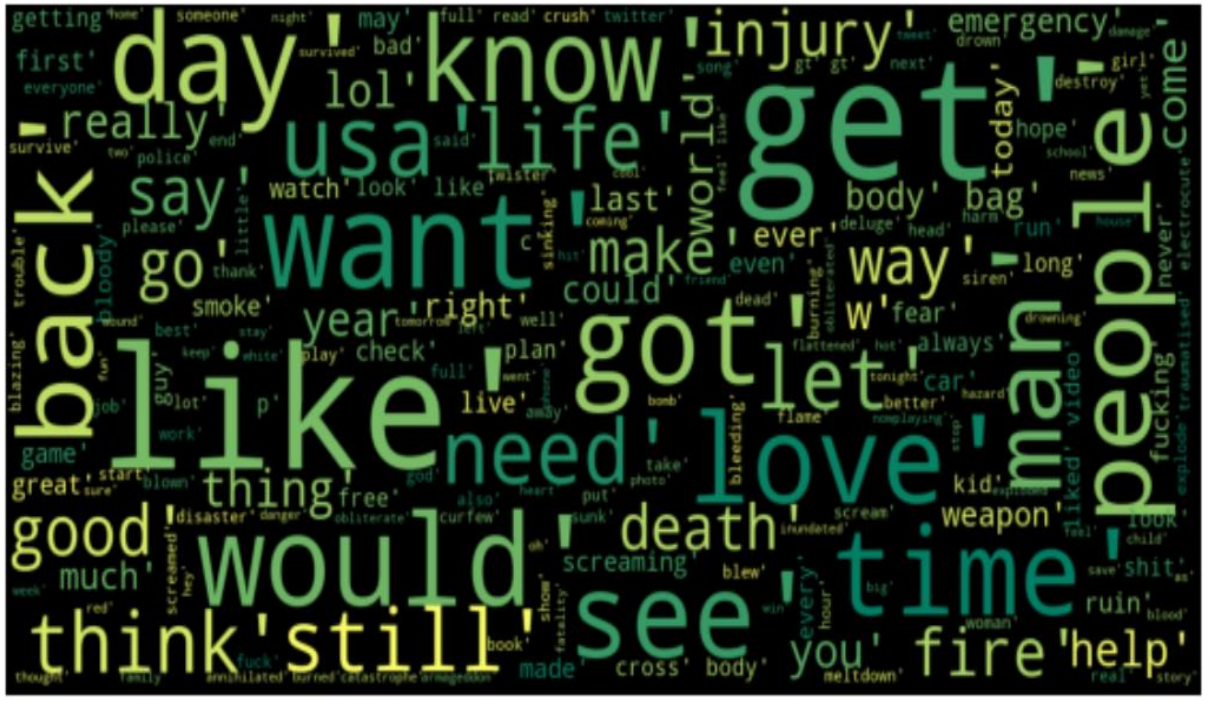
WordCloud: Commonly Used in Tweets

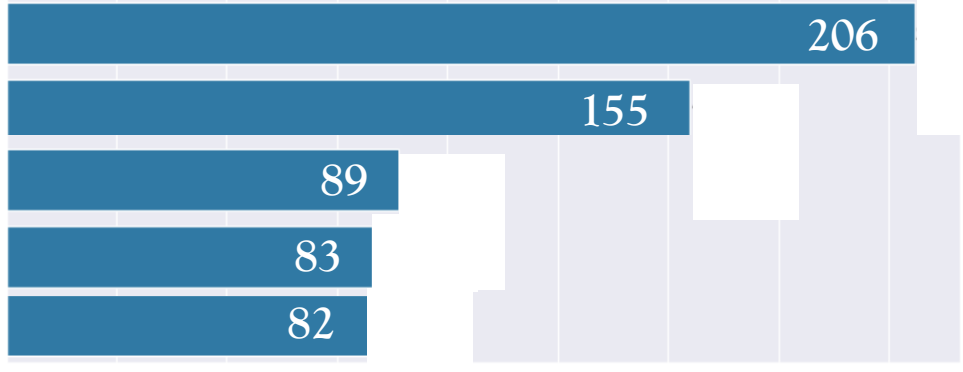
Disaster Tweets | Target = 1





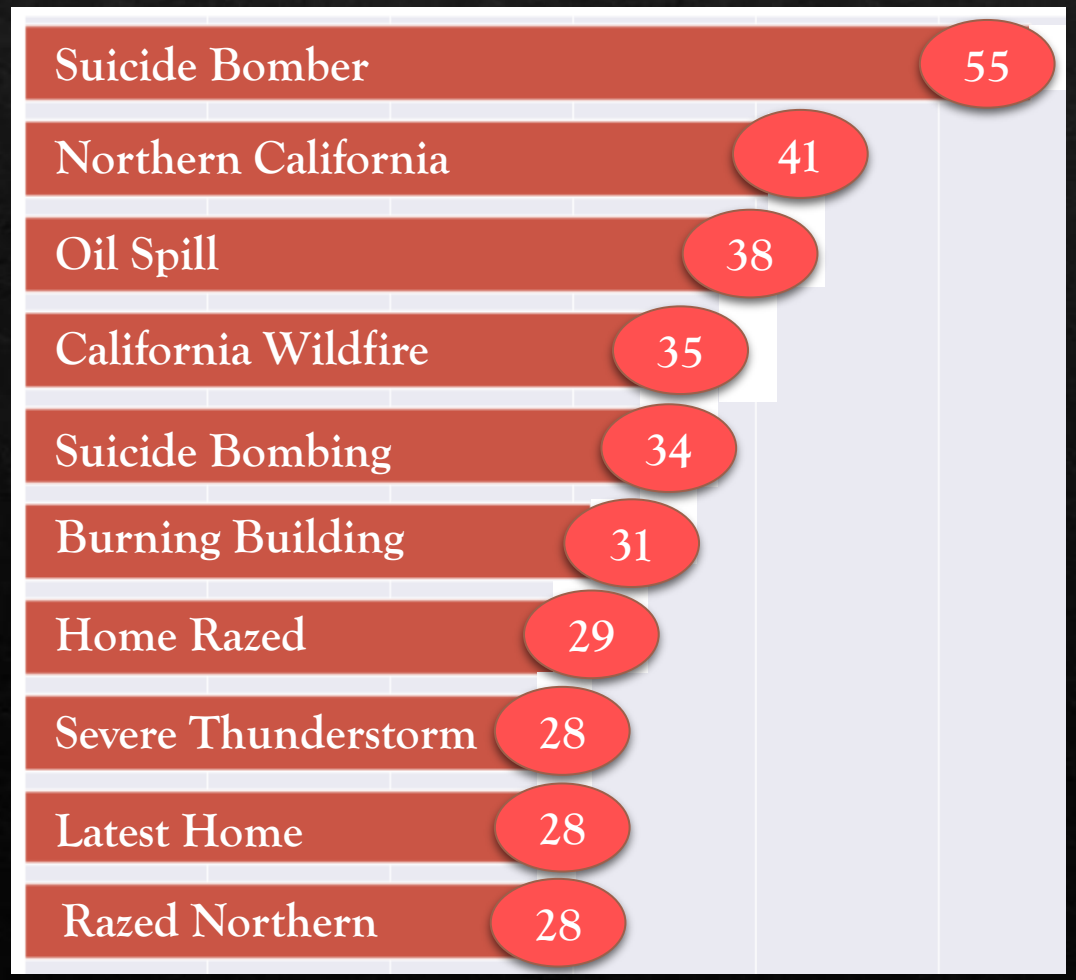
Non-Disaster Tweets | Target = 0



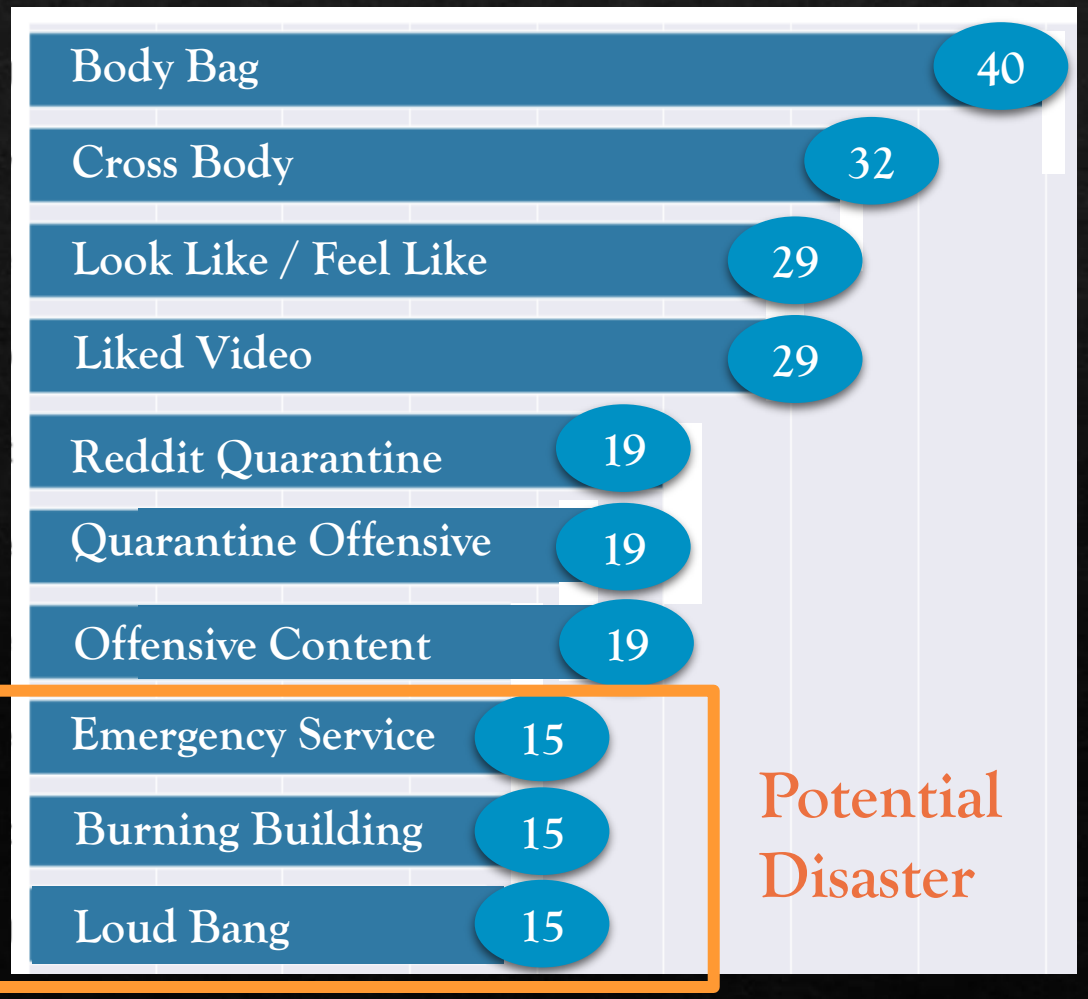


Representation: Text Sequence (N-Grams)

Disaster Tweets | Target = 1



Non-disaster Tweets | Target = 0

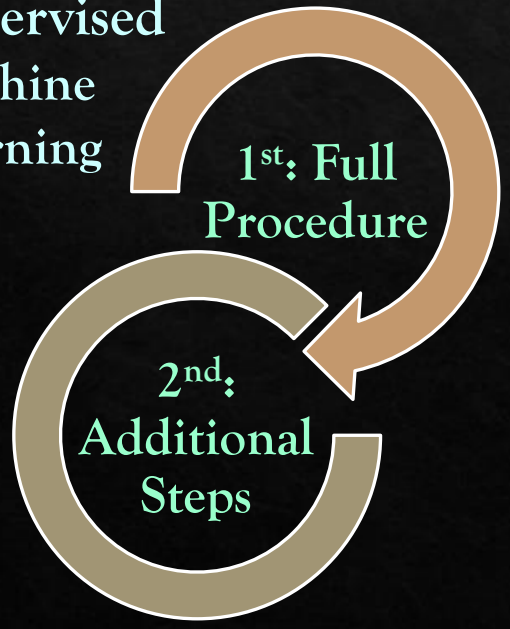


Basic Steps for Text Classification

1

Text Preprocessing

Supervised Machine Learning



Deep Learning / Transformer

None or TensorFlow / Keras Preprocessing Package

2

Representation

- Bi-Grams (n = 2)
- Tri-Grams (n = 3)

3

Feature Extraction

Supervised Machine Learning

- Bag of Word (BOW) Vector
- TFIDF Vector

Deep Learning / Transformer

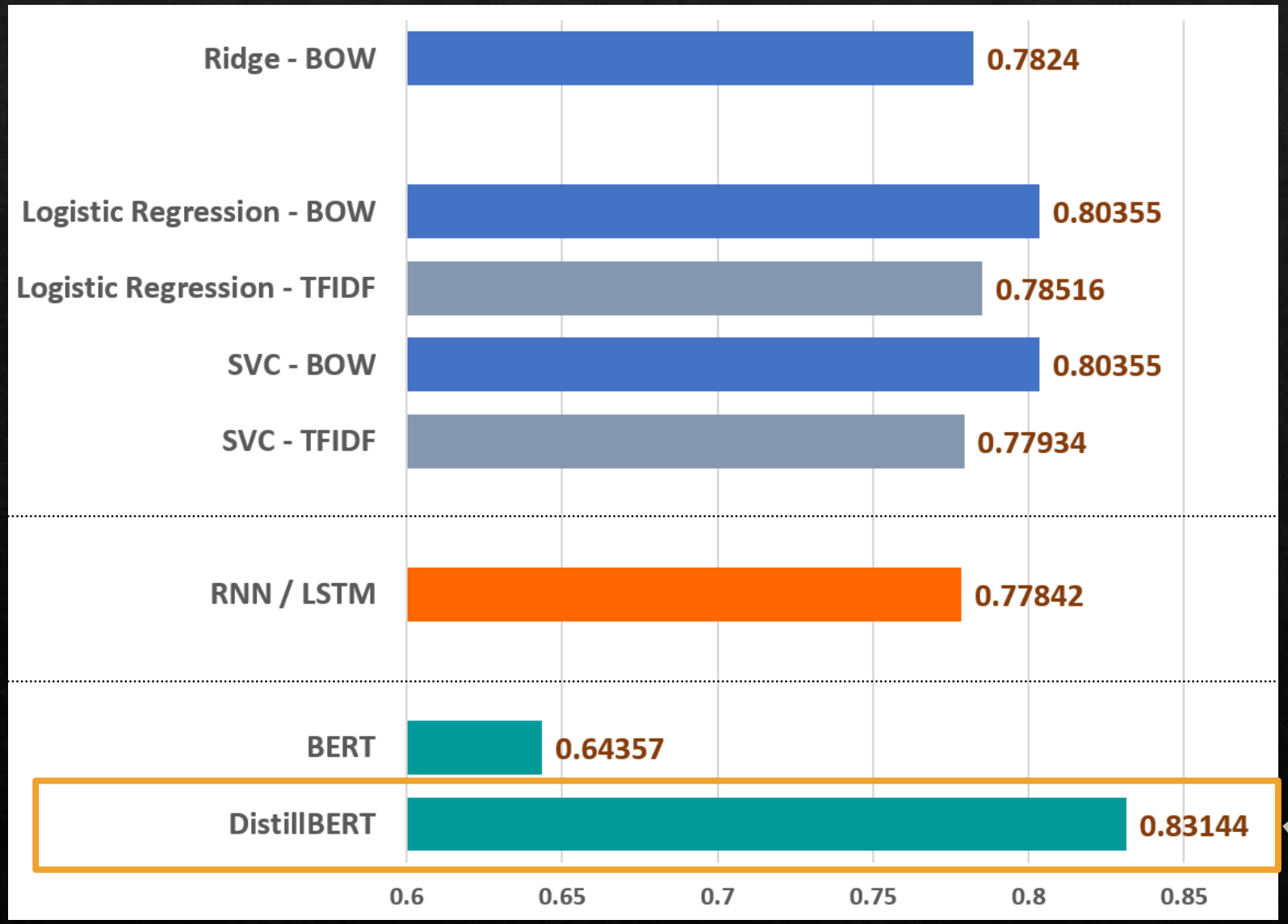
- Layers of Neural Nets
- Pre-trained Features

Text Preprocessing

1st Preprocessing:

- Remove Duplicate Tweets
- Replace Abbreviated/ Special Text to Standard Text
- Remove URL
- Remove Contractions
- Tokenize
- Remove Numbers & Unicode
- Remove Punctuation
- Normalize
- Clean Repeated Characters in Text
- Remove Stop Words
- Lemmatize
- Remove Contractions
- Join Tokenized Words to Sentence

Machine Learning – Classification Performance



Kaggle – Best F1 Scores

Supervised Machine Learning

Deep Learning

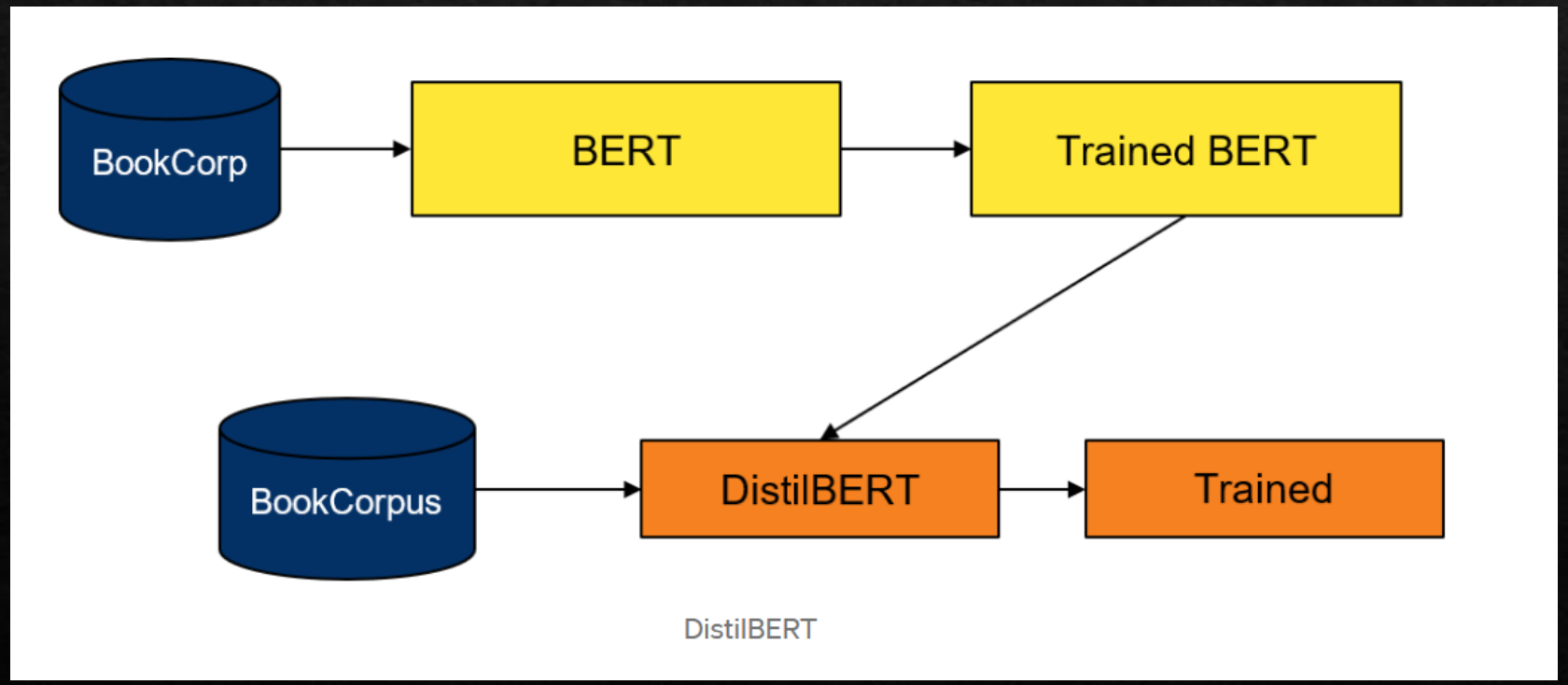
Transformer

Best Performing Model

Transformer - DistilBERT

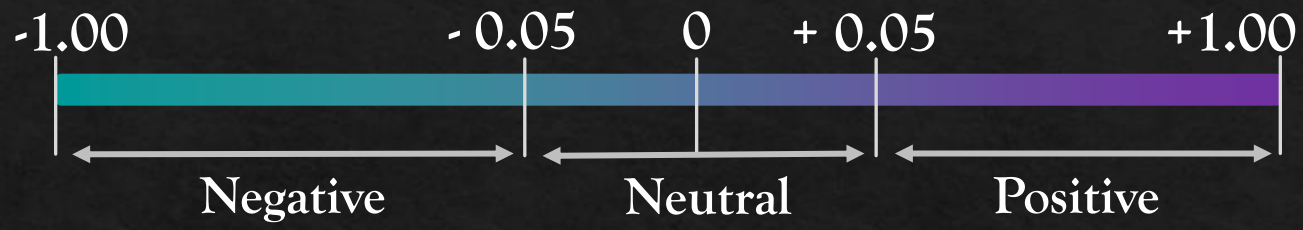
Hugging Face

Smaller, lighter and faster version of BERT developed and open-sourced by the team at HuggingFace

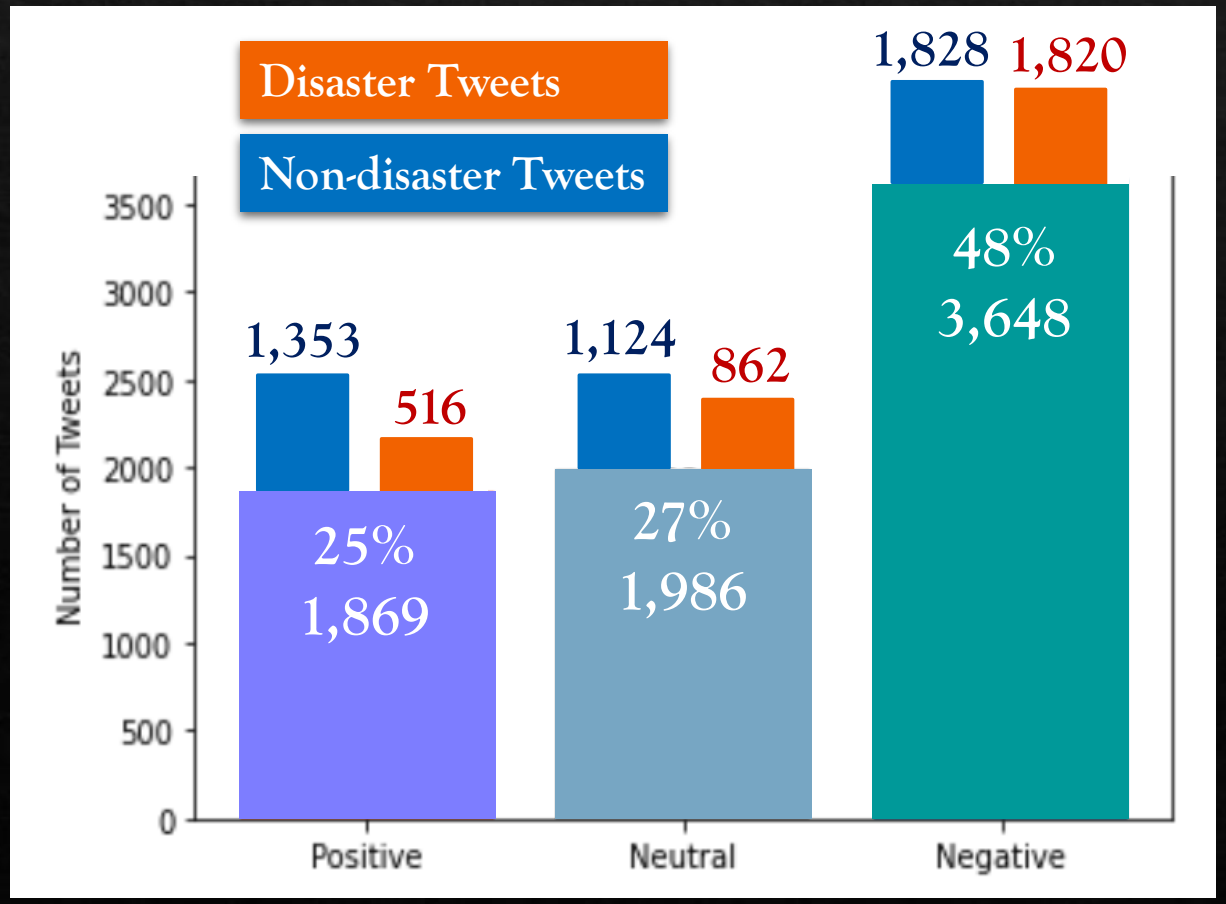


- 40% less parameters than BERT-base-uncased, runs 60% faster, preserving over 95% of BERT's performances
- Implementation on TensorFlow/Keras, Trained DistilBERT used to generate sentence embedding
- Basic NN Architecture (with Dense and Dropout layers)

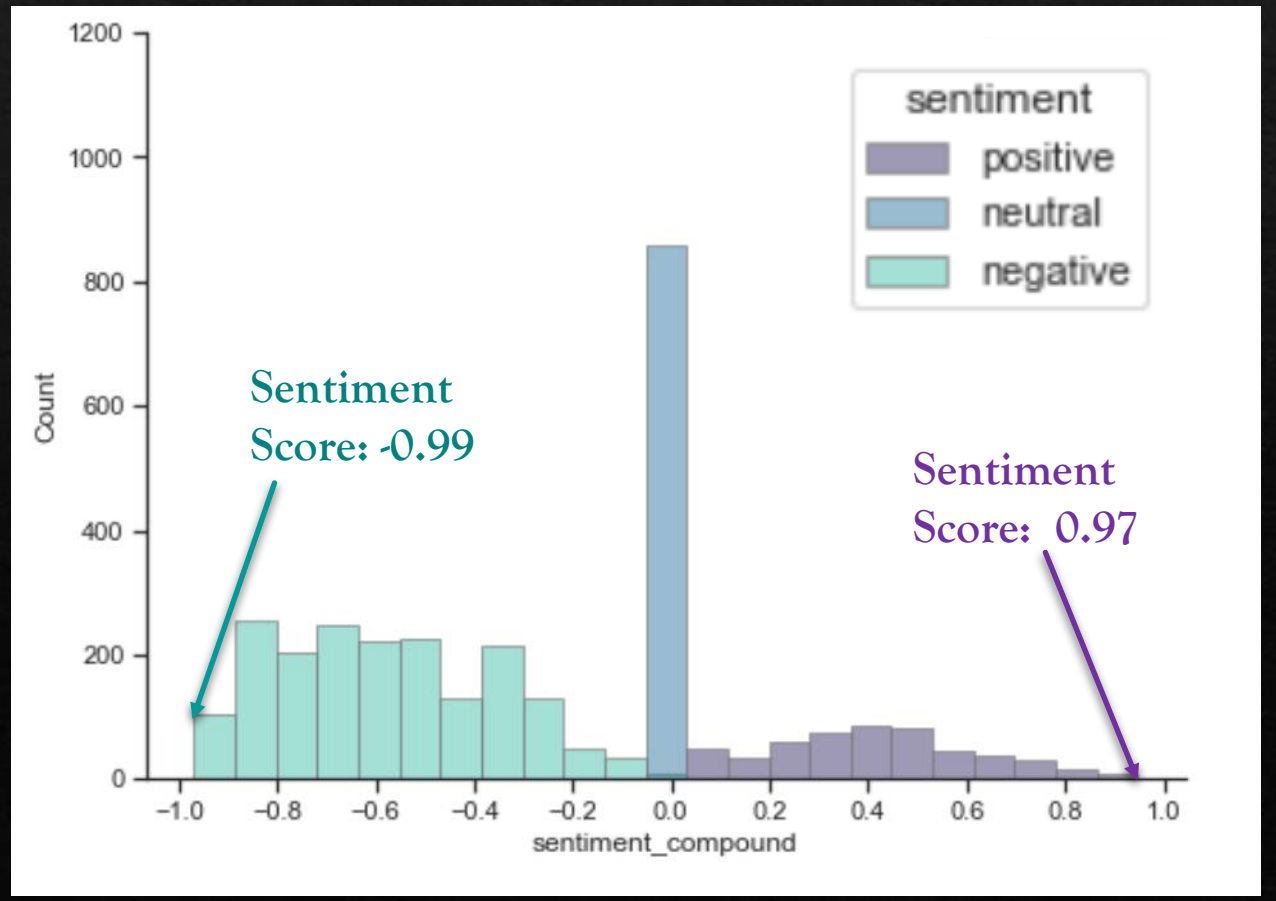
Sentiment Analysis



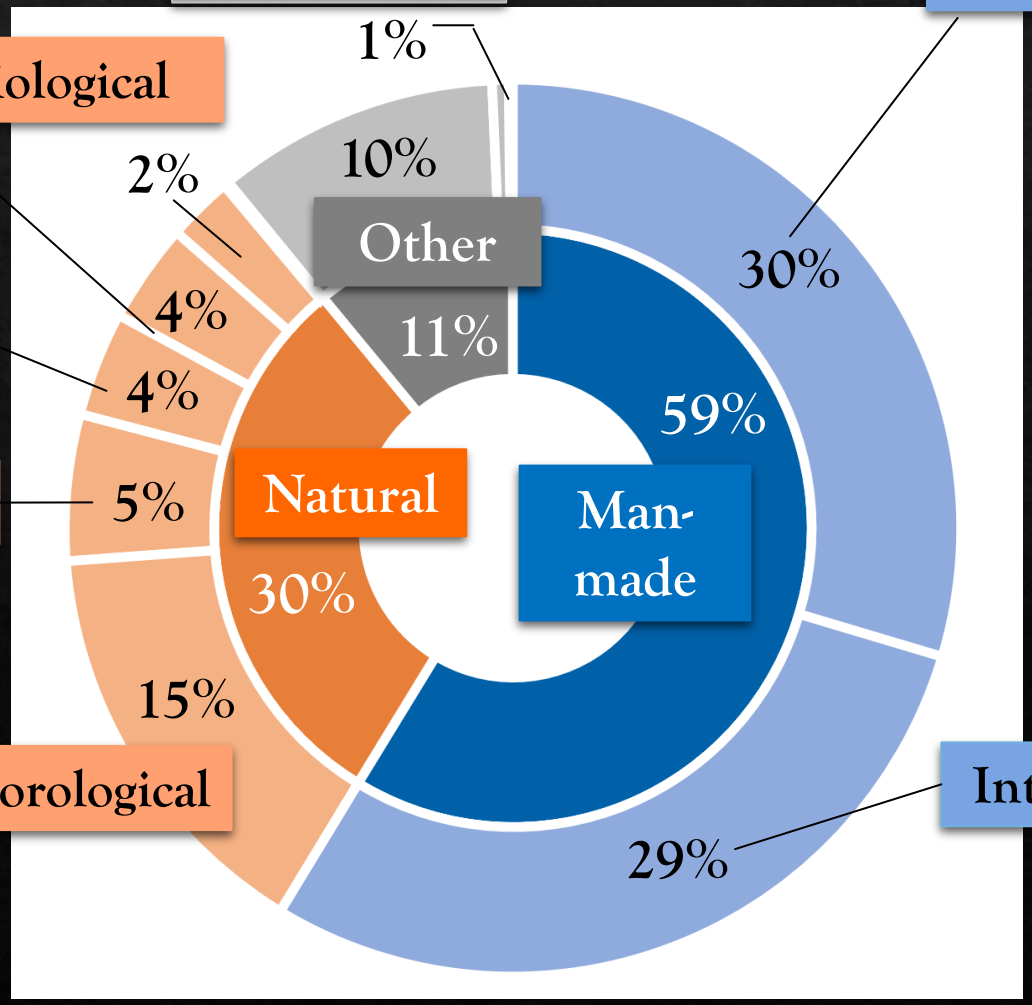
Counts of Sentiment Classes in All Tweets



Sentiment Score Distribution – Disaster Tweets



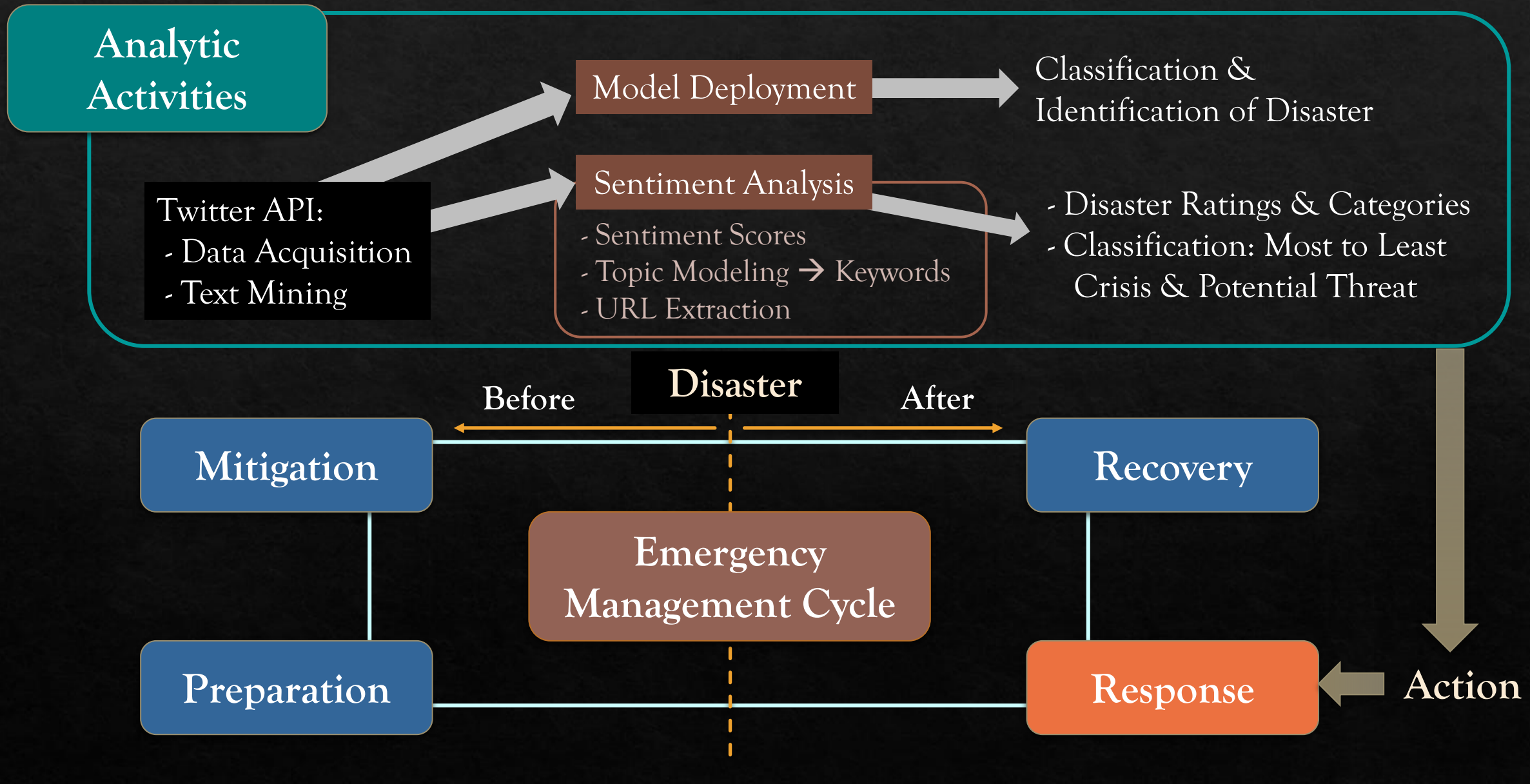
Disaster Tweets

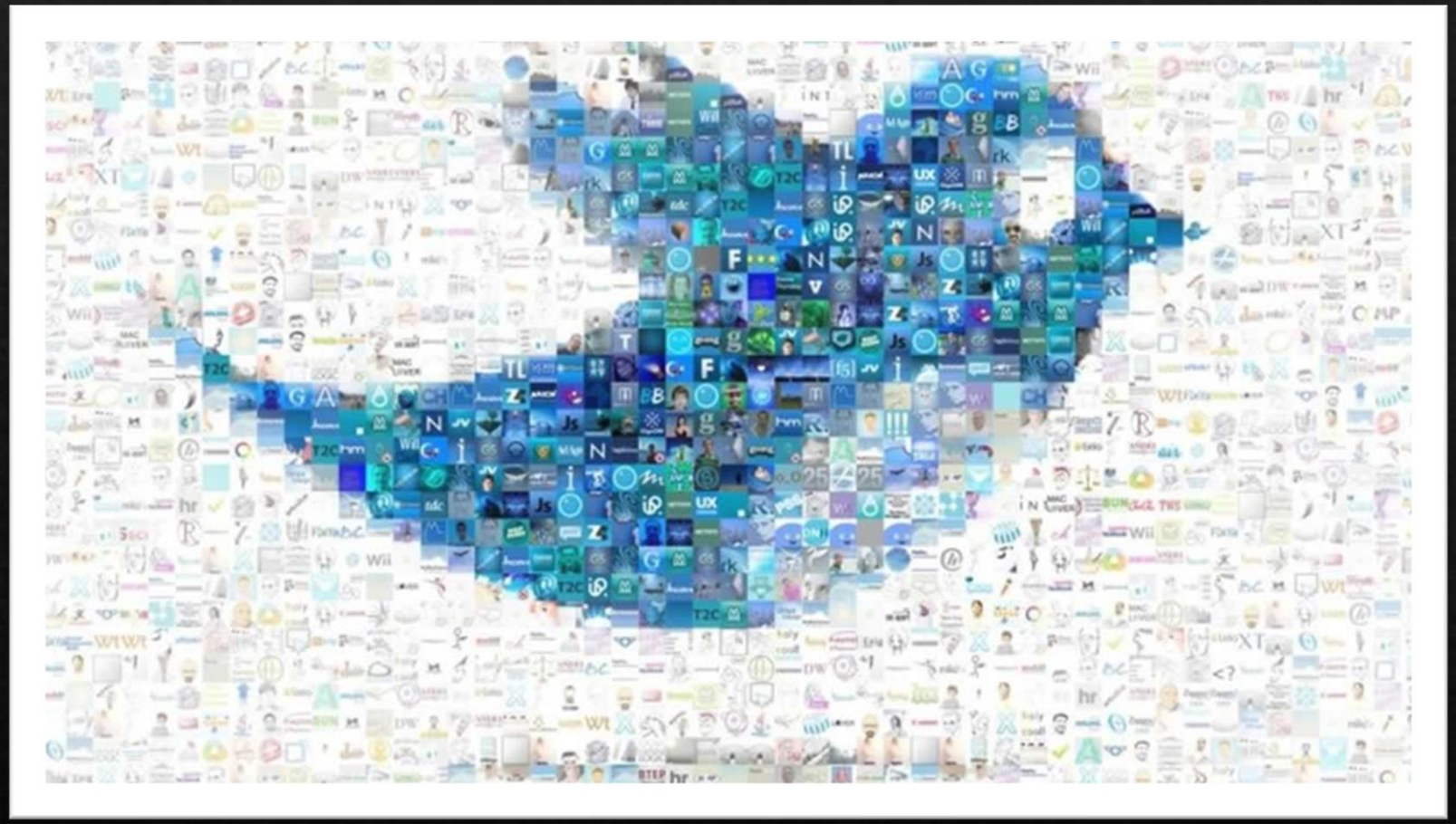


Disaster Class & Categories

Class	Sub-class
Natural	Meteorological Hydrological Geophysical Climatological Biological
Man-made	Intentional Accidental
Other	Non-disaster related

Emergency Response Management





Thank You!

n.hong@queensu.ca

www.linkedin.com/in/nicolehhong

https://github.com/Nicole-Hong