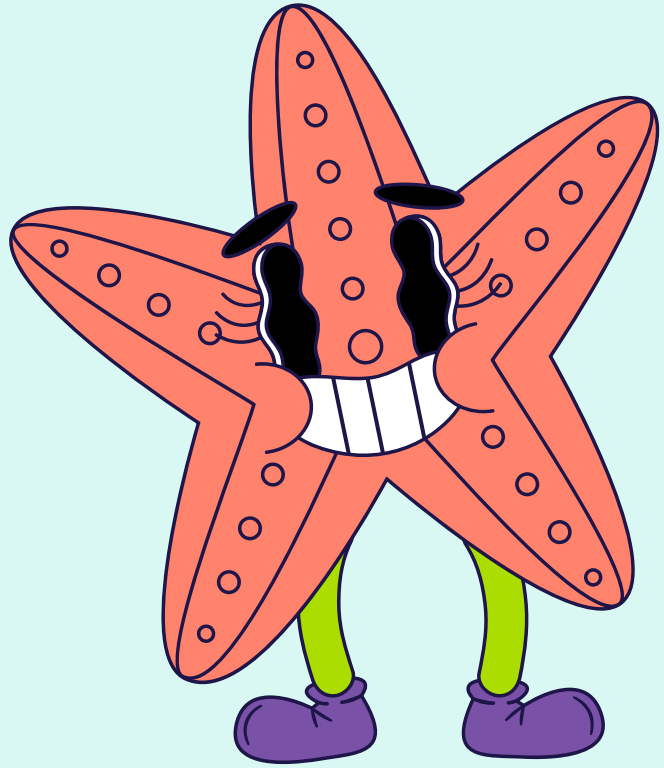

HUMOR DETECTION
GEOFF VOGT
FLATIRON DS
CAPSTONE PROJECT

BUSINESS UNDERSTANDING

- Business Objectives:
  - Develop a machine learning model to accurately detect humor in textual content
  - Allow stakeholders to measure the effectiveness of their content in eliciting humor by analyzing humorous/not-humorous texts against user engagement
  - Measure the impact of humorous ads on brand recall and engagement for advertisers

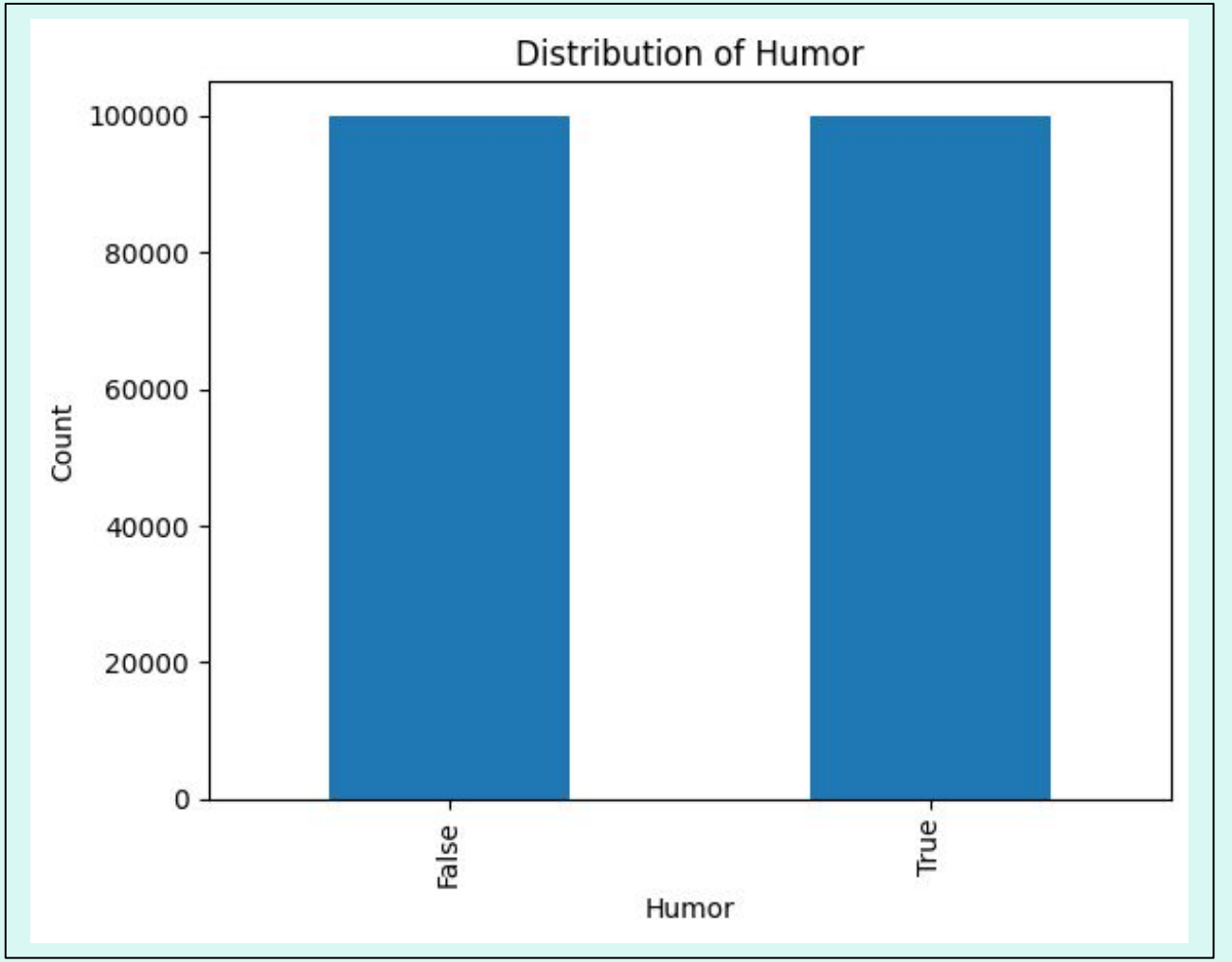


- Stakeholders:
  - Social media platforms
  - Advertisers
  - Content creators

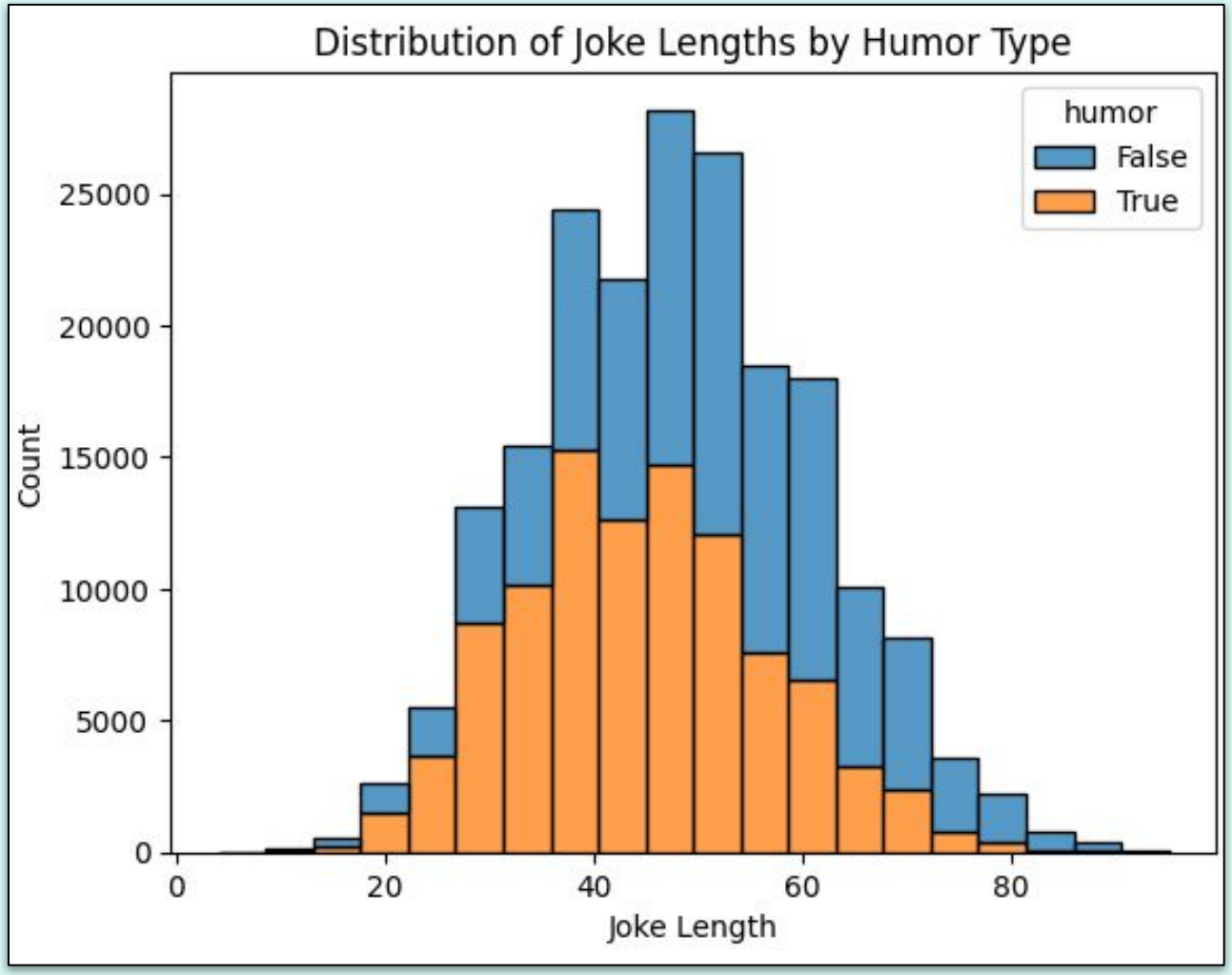
DATA UNDERSTANDING

- The dataset of 200k short texts labeled as ‘true’ or ‘false’ for binary humor target was provided in a 2020 paper by two NLP/linguistic professors
    - https://arxiv.org/abs/2004.12765
- Ex. "What is a pokemon master's favorite kind of pasta? Wartortellini!!" is labeled 'true' for 'humor'
- The non-humorous texts in the dataset are short headlines from online news sources.
- Ex: "Kim Kardashian baby name: reality star discusses the 'k' name possibility" is labeled 'false' for 'humor'

EXPLORATORY DATA ANALYSIS

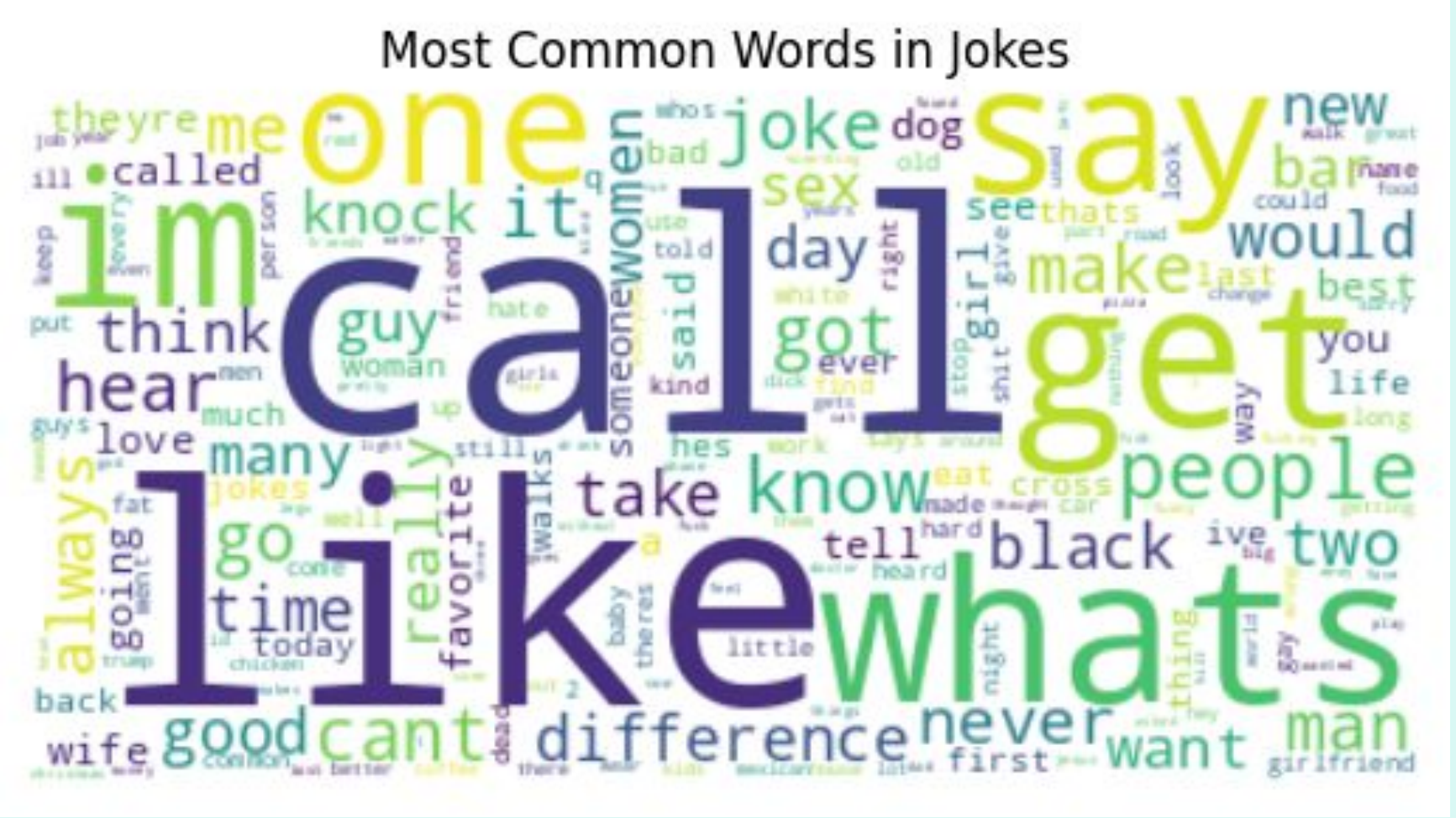


EXPLORATORY DATA ANALYSIS

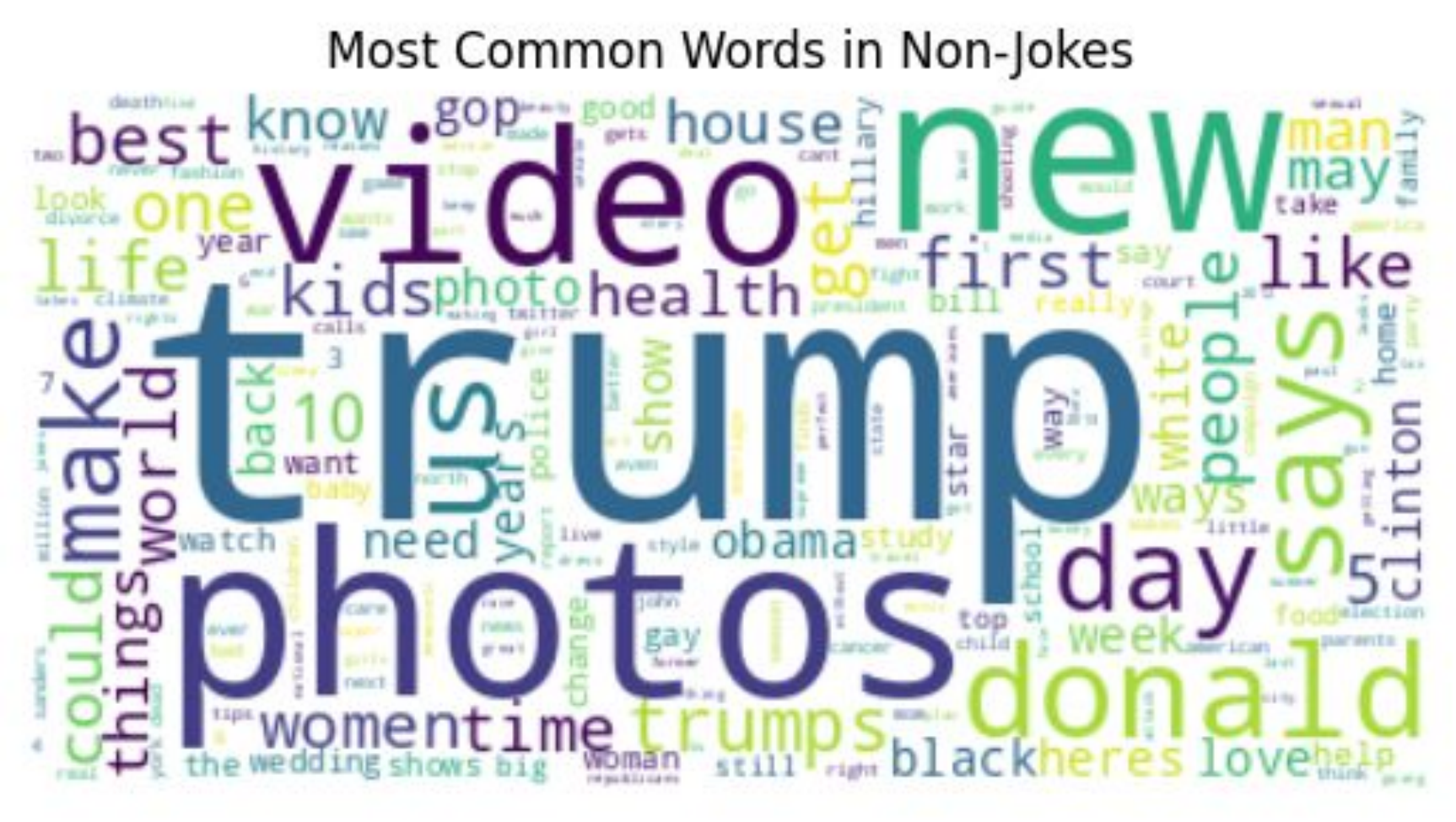


- Jokes and non-jokes follow the same normal distribution in terms of length of text
- 'text' column underwent removal of stop-words and punctuation
- Mean: 47.84 characters
- Minimum: 4 characters
- Maximum: 95 characters.

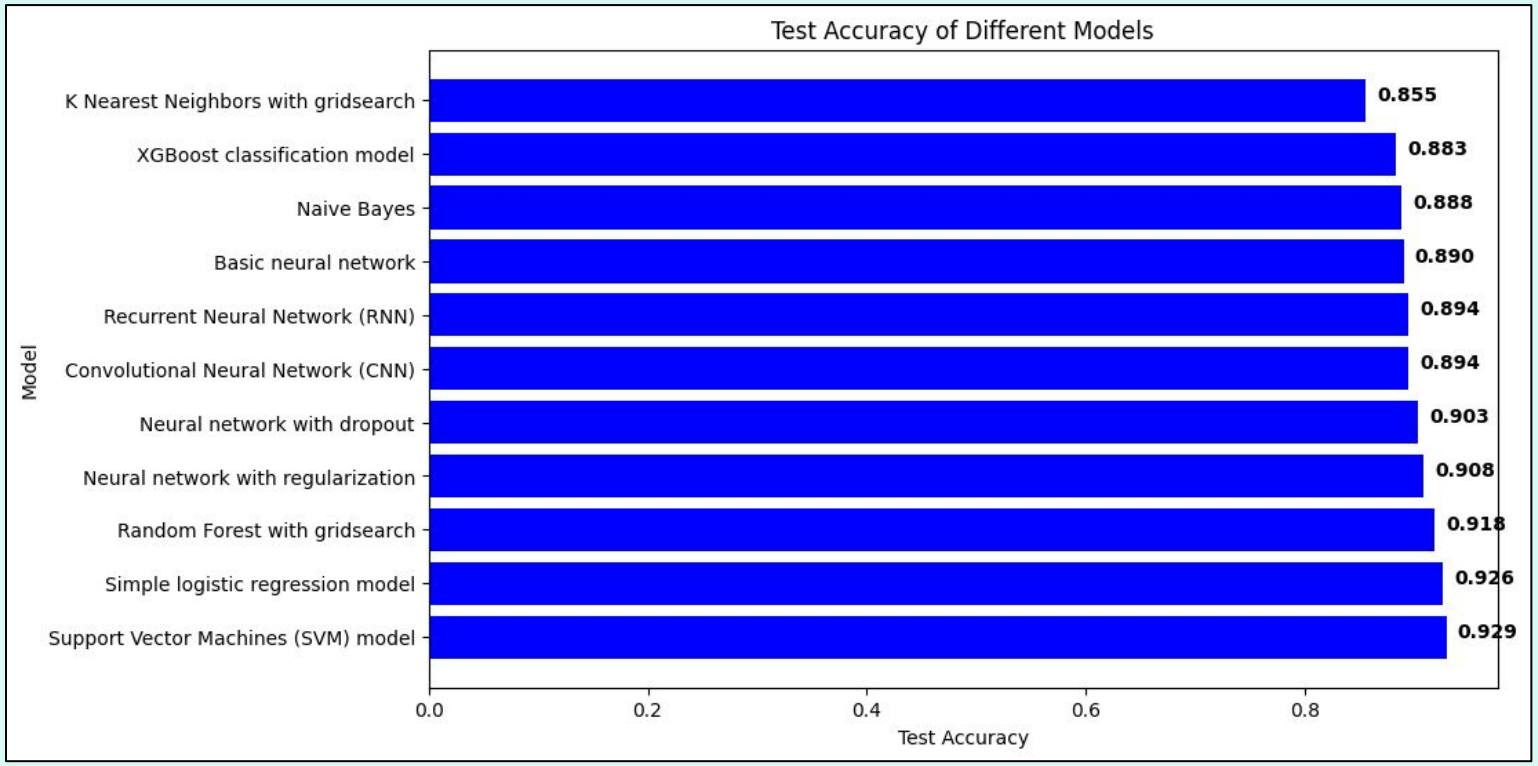
EXPLORATORY DATA ANALYSIS



EXPLORATORY DATA ANALYSIS



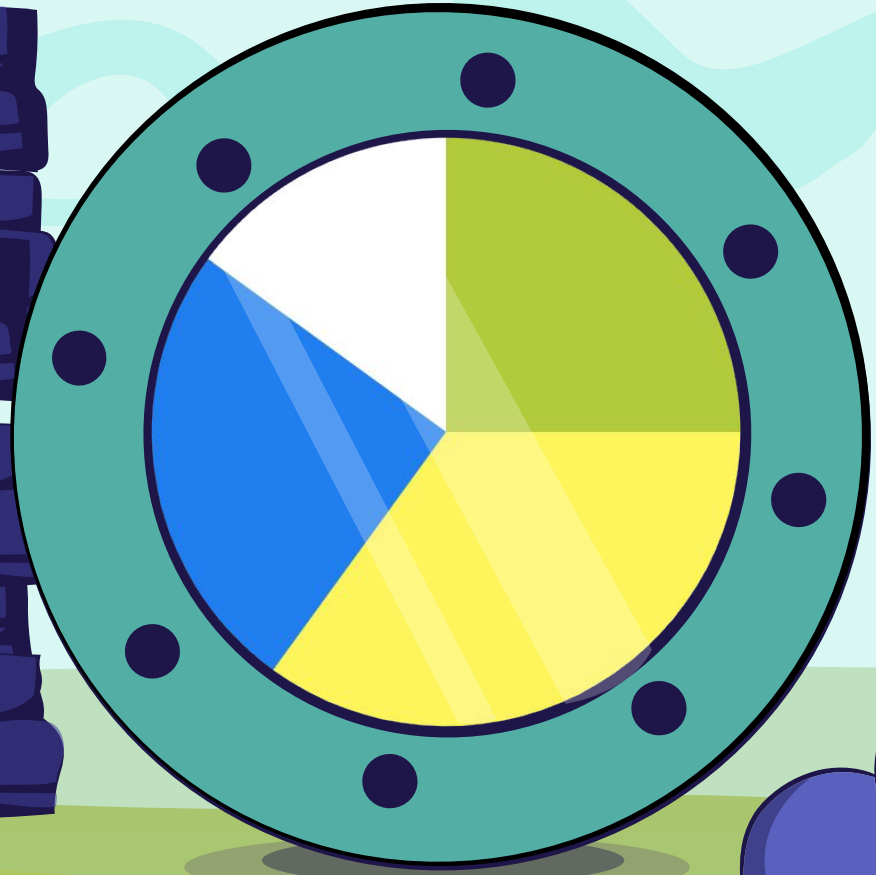
CLASSIFICATION MODELING



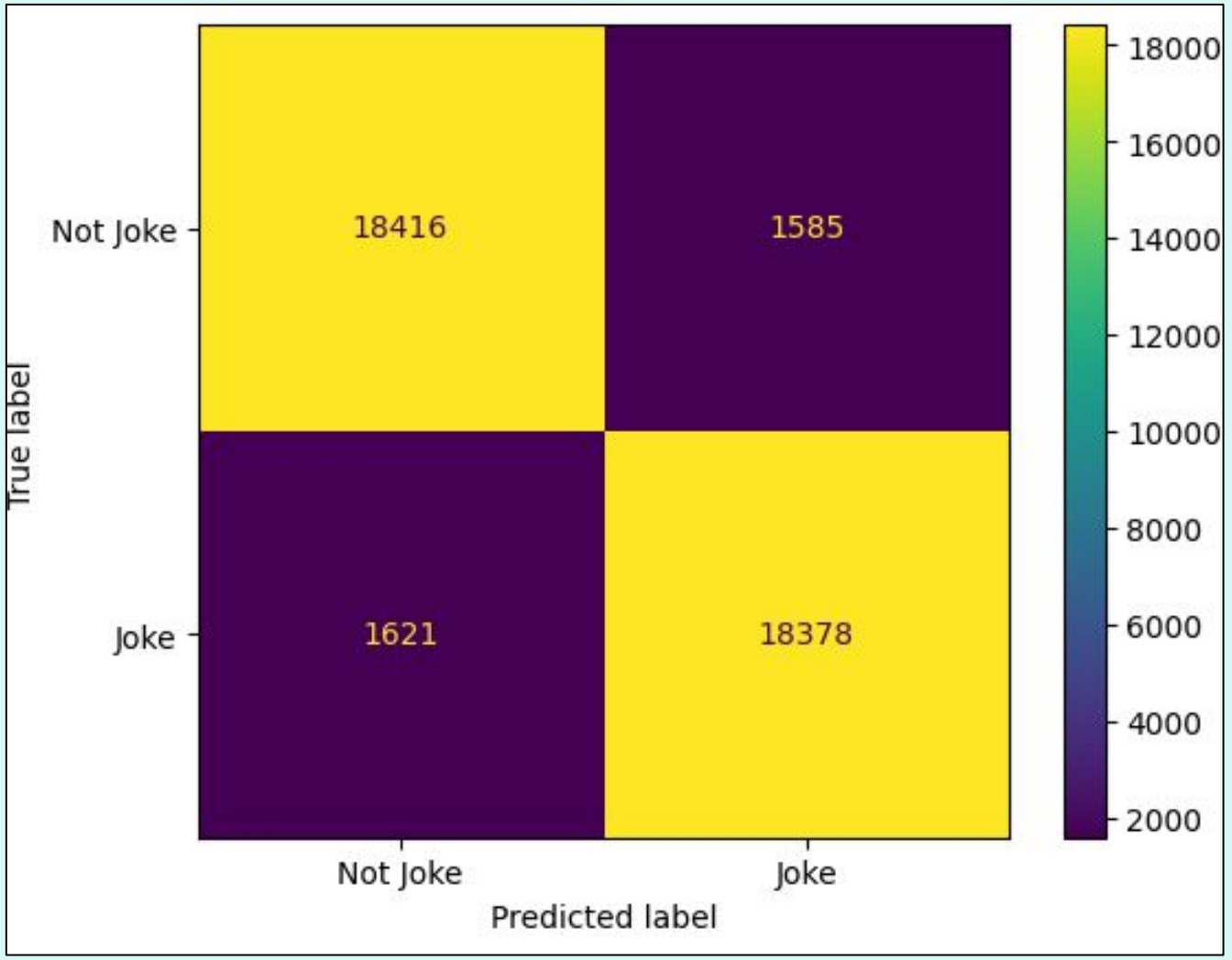
- 12 different classification models were tested using TF-IDF vectorization and selected highest performing train accuracy
- SVM had the highest test accuracy of 0.933, followed by random forest with .929
- Models like neural networks and XGBoost did not perform well for this problem
- Removing punctuation and/or Lemmatization decreased model accuracy
- Removing stop-words did not increase model accuracy

MODEL EVALUATION

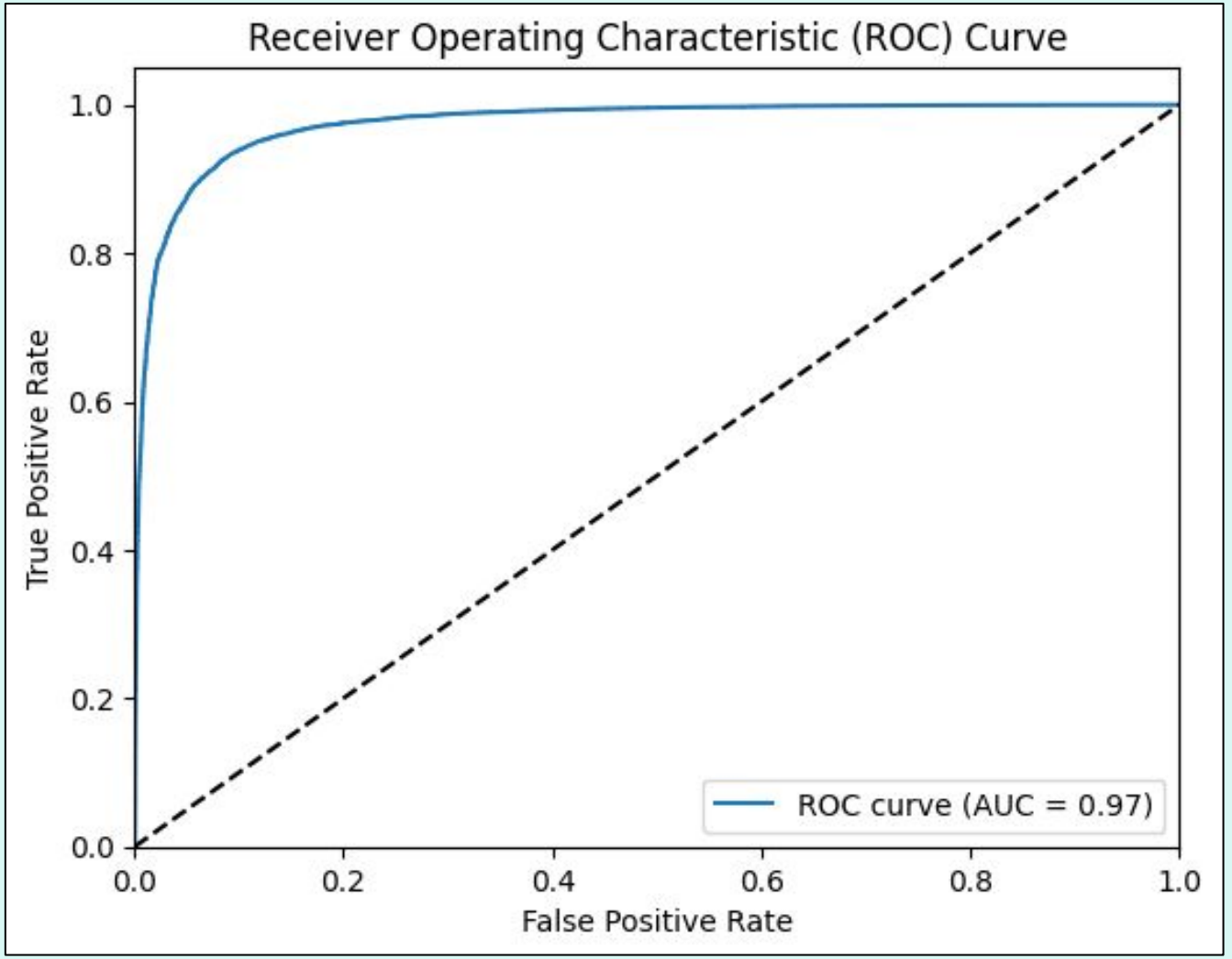
91% Accuracy	91% Precision
91% Recall	91% F1-Score



MODEL EVALUATION



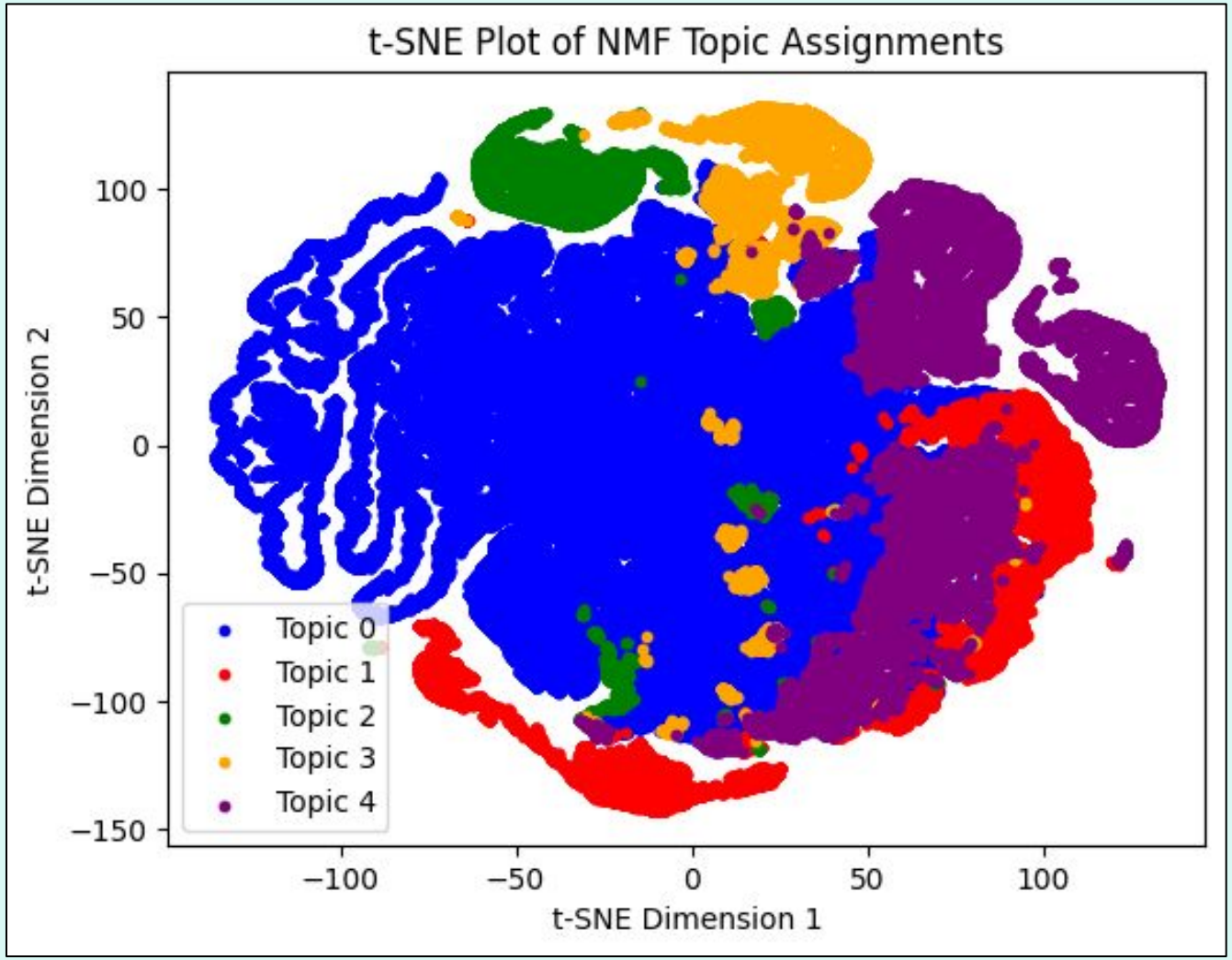
MODEL EVALUATION



TOPIC MODELING

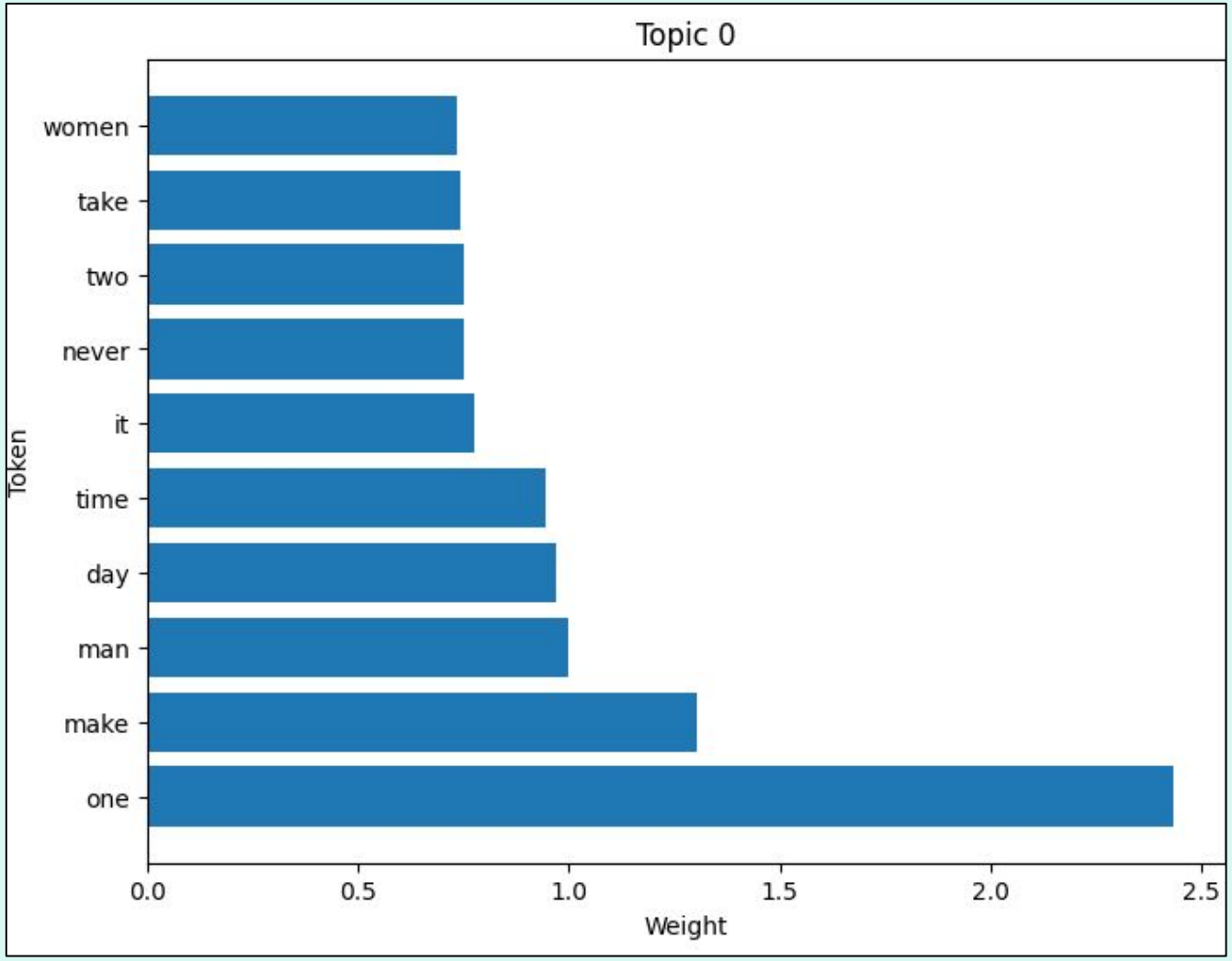
- LDA and NMF were two topic modeling techniques used to discover hidden topics in the documents for humorous texts
- Top tokens were extracted to determine the most important words for each defined topic
- t-SNE is applied to the topic distributions of the documents to reduce dimensionality visualize the topic distributions
- A similar t-SNE plot was created to represent 2D space for humorous vs. non-humorous texts
- NMF is chosen as the technique for topic modeling due to clearest results of topic distributions

TOPIC MODELING

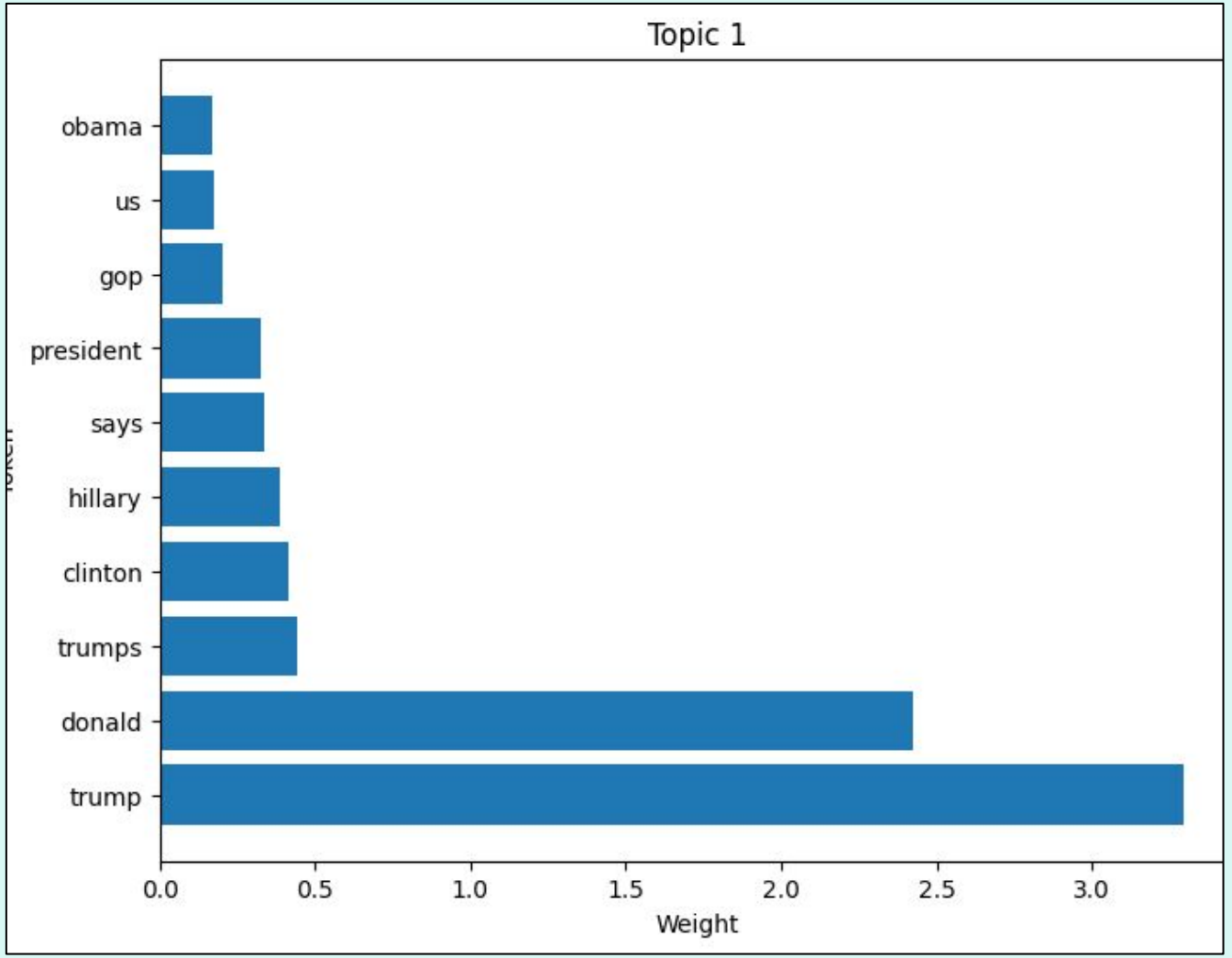


- Topic 0: one, make, man, day, time, it, never, two, take, women
- Topic 1: trump, donald, trumps, clinton, hillary, says, president, gop, us, obama
- Topic 2: get, cross, road, chicken, side, it, married, christmas, cant, pregnant
- Topic 3: people, black, world, white, cant, hate, types, 10, think, racist
- Topic 4: new, photos, york, years, video, week, year, fashion, best, 10

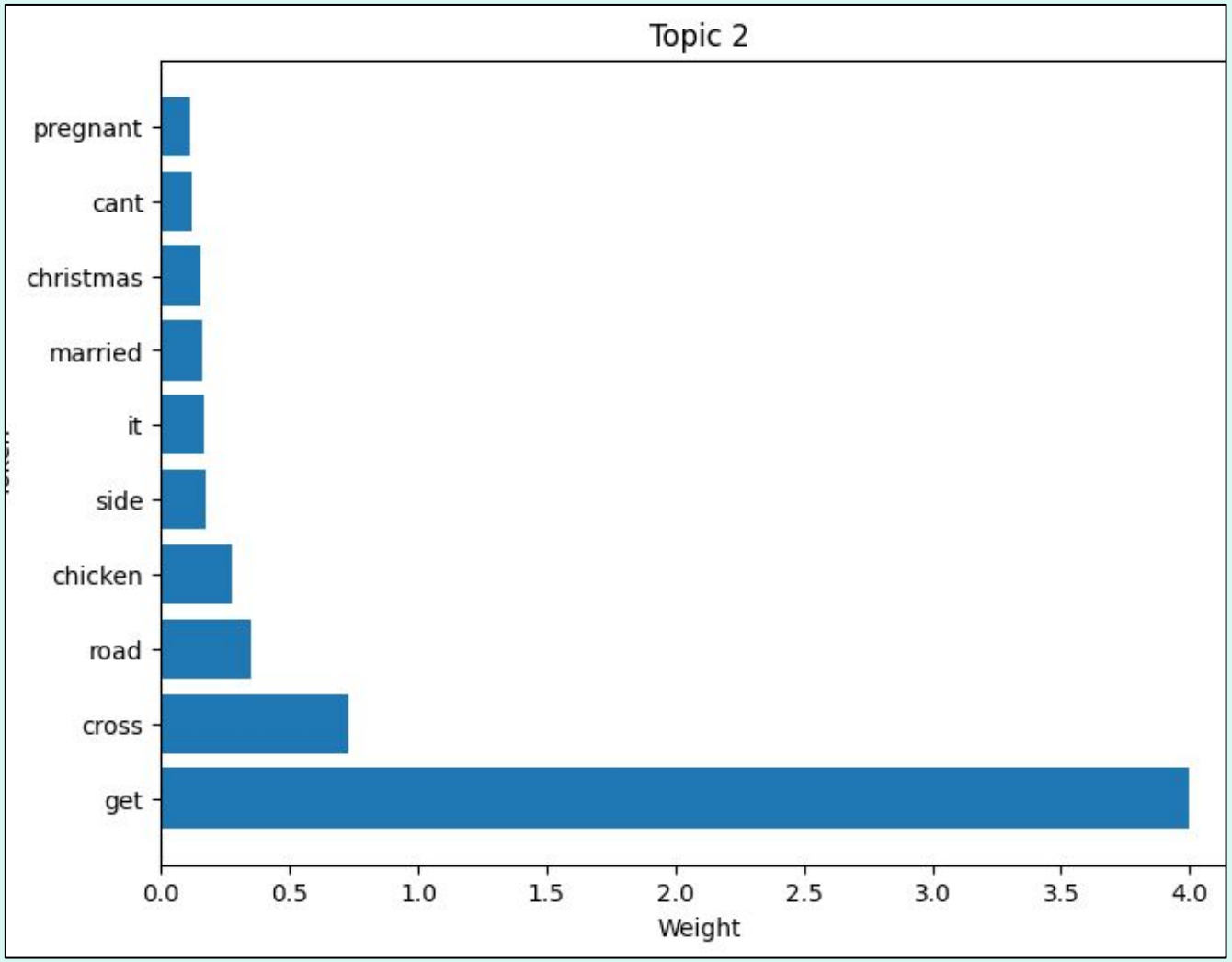
TOPIC 0: GENERAL JOKES



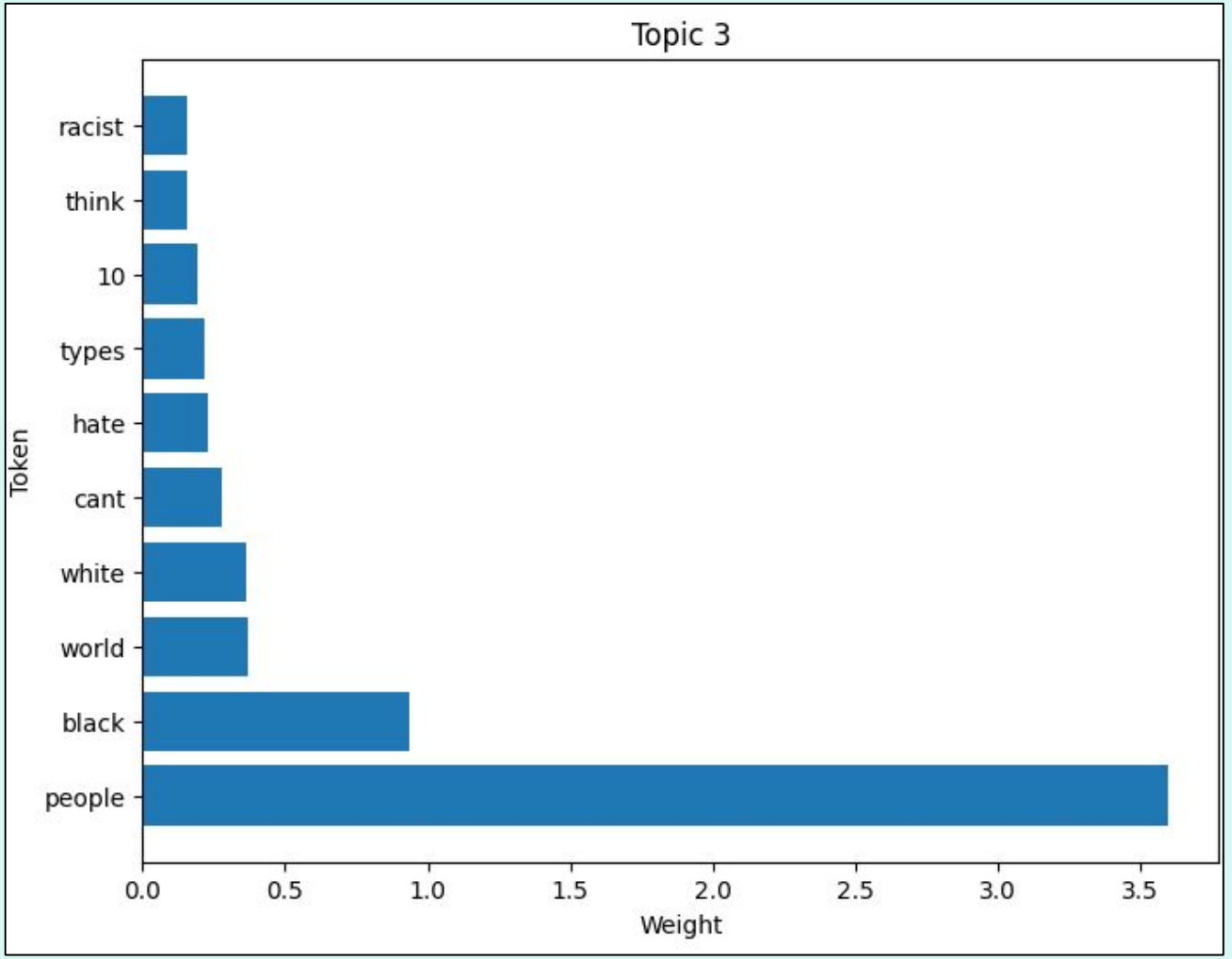
TOPIC 1: POLITICAL JOKES



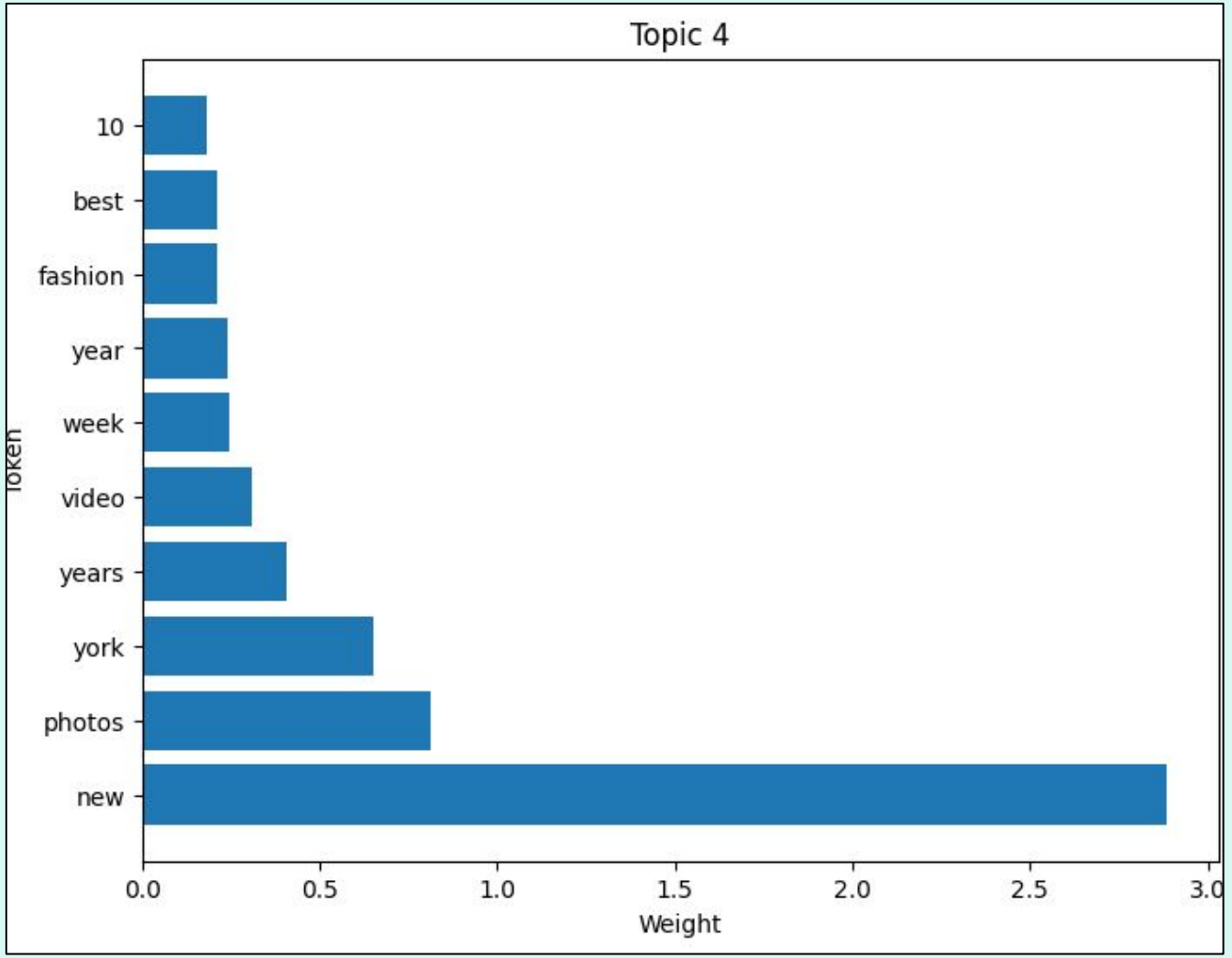
TOPIC 2: "CHICKEN CROSS THE ROAD"

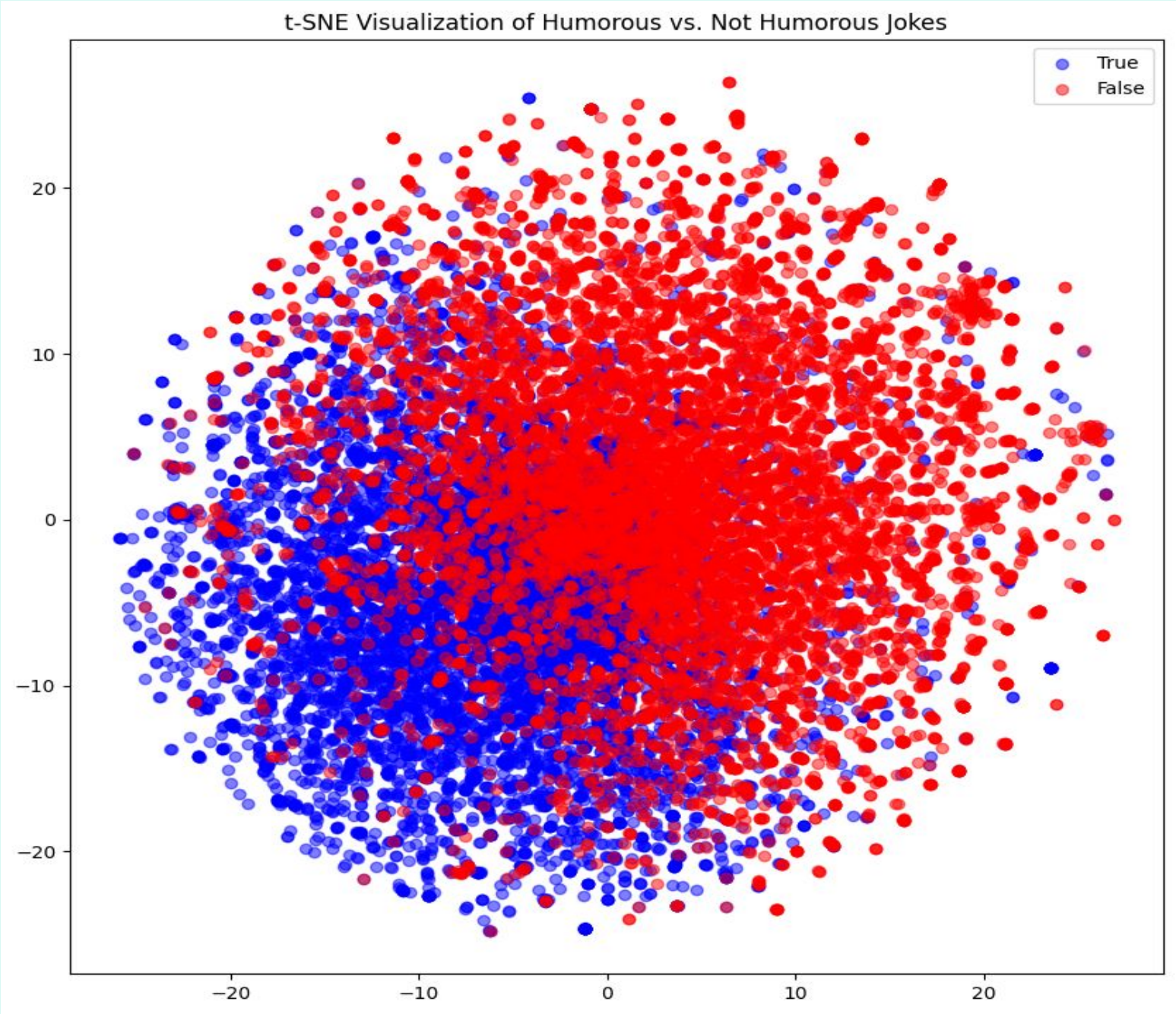


TOPIC 3: RACIAL/RACIST JOKES



TOPIC 4: WHAT'S TRENDING




t-SNE Visualization of Humorous vs. Not Humorous Jokes
True
False
20
10
0
−10
−20
−20
−10
0
10
20

UNIVERSAL SENTENCE ENCODER

- Pre-trained Universal Sentence Encoder model from TensorFlow is used to encode jokes into vectors
- Encoded vectors are stored in a list called joke_vectors_list and a for loop is used to encode the jokes in batches due to dataset size limitations
- Cosine similarity is calculated between all pairs of joke vectors
- Similarity matrix contains cosine similarity values for all pairs of joke vectors
- User inputs a string to encode and assess cosine similarity to the list of jokes with top jokes returned

UNIVERSAL SENTENCE ENCODER

- Example: I enter the phrase "Dr. Seuss cat in the hat" and enter "5" for number of results to show

1.
```
user_input = input("Enter a sentence: ")
```

2.
```
num_jokes = int(input("How many jokes would you like to see? "))
```

3.
```
Top 5 jokes:
1. What did dr. seuss call the book he wrote about star wars? the cat in the at-at
2. What was schrodinger's favorite childhood book? the cat in the box by dr. seuss
3. What is dr. seuss' favorite play? green eggs and hamlet
4. Did you read dr seuss as a kid because green eggs and damn
5. What do you call a magician in a dr. seuss book? who-dini
```

CONCLUSION + RECOMMENDATIONS

Recommendations

- Machine learning models can effectively classify texts as humorous or not
- Use model to tag short-texts as humorous to analyze against user engagement and tailor the experience
- Tag humorous texts with topic models to further analyze user cohorts against engagement

Future work

- Collecting more data and expanding the dataset can improve model generalizability
- Classification of different kinds of jokes (satire, puns etc.) for advertising analysis