
BBC NEWS
FOX NEWS channel
Money
uncertainty
fear
reality
PANICKING
Financial
economic
Corona
COVID-19 IN OKLAHOMA
NEW CASES
NIGHTLY NEWS WITH LESTER HOLT
BREAKING NEWS
HOSPITAL CRISIS
yahoo! news
r/news or
r/conspiracy

Our Data problem:

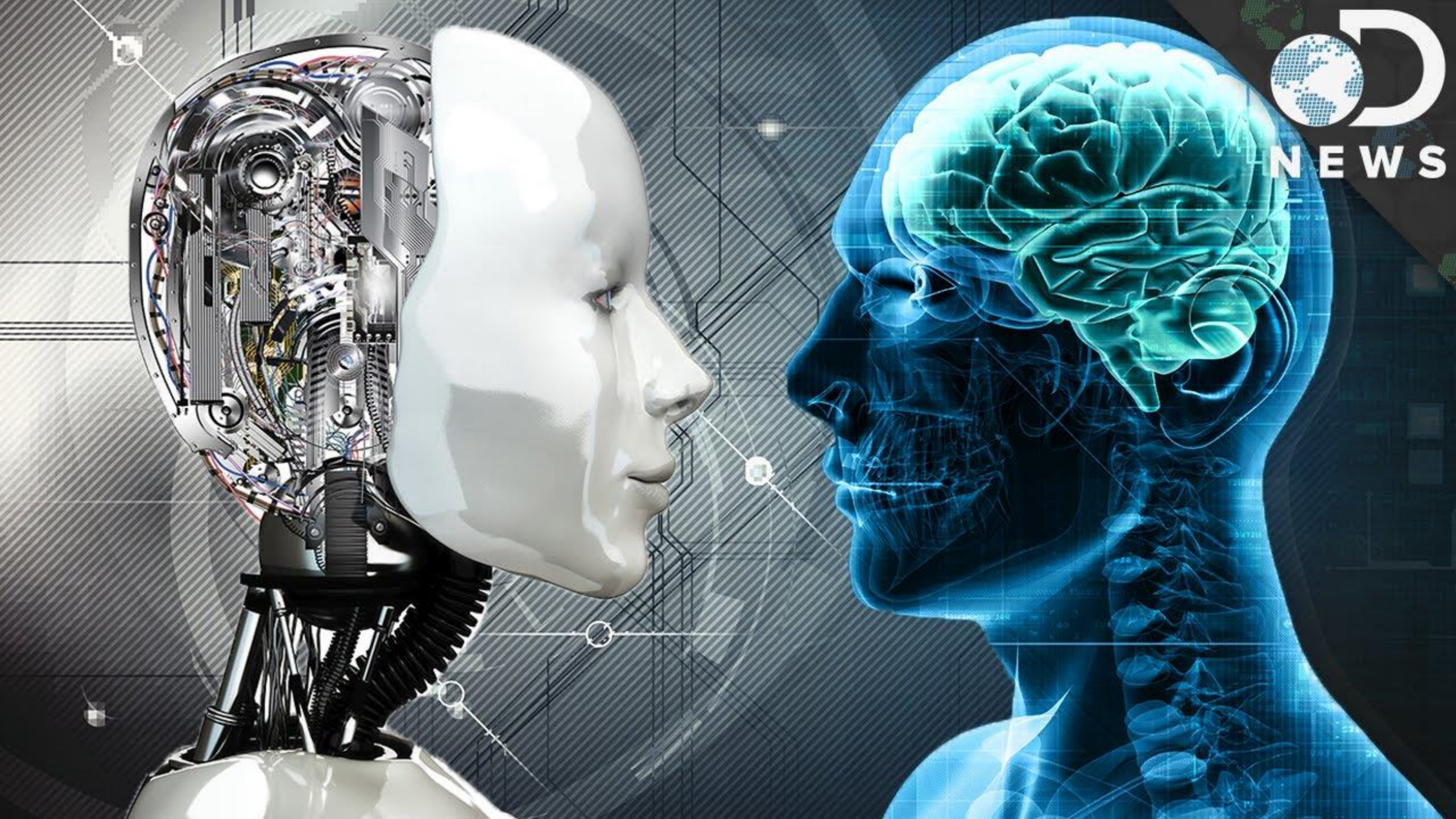
- r/conspiracy and r/news picked to see how hard it is to spot conspiracy from reported fact…or what is supposed to be at least
- Wanted a challenge: words “conspiracy” and “news” added to stop words
- Our question: How hard is it for machine learning to spot the difference between comments on the r/news and r/conspiracy subreddits well enough to accurately classify them?

Data description

- 42,881 comments
- 51.96% conspiracy
  - Tuesday, September 15, 2020 4:45:52 PM to Thursday, September 17, 2020 12:52:17 AM
- 48.04% news
  - Wednesday, September 16, 2020 3:41:27 AM to Thursday, September 17, 2020 12:53:19 AM
- 10385 news commenters
- 6208 conspiracy commenters
- 258 comment on both

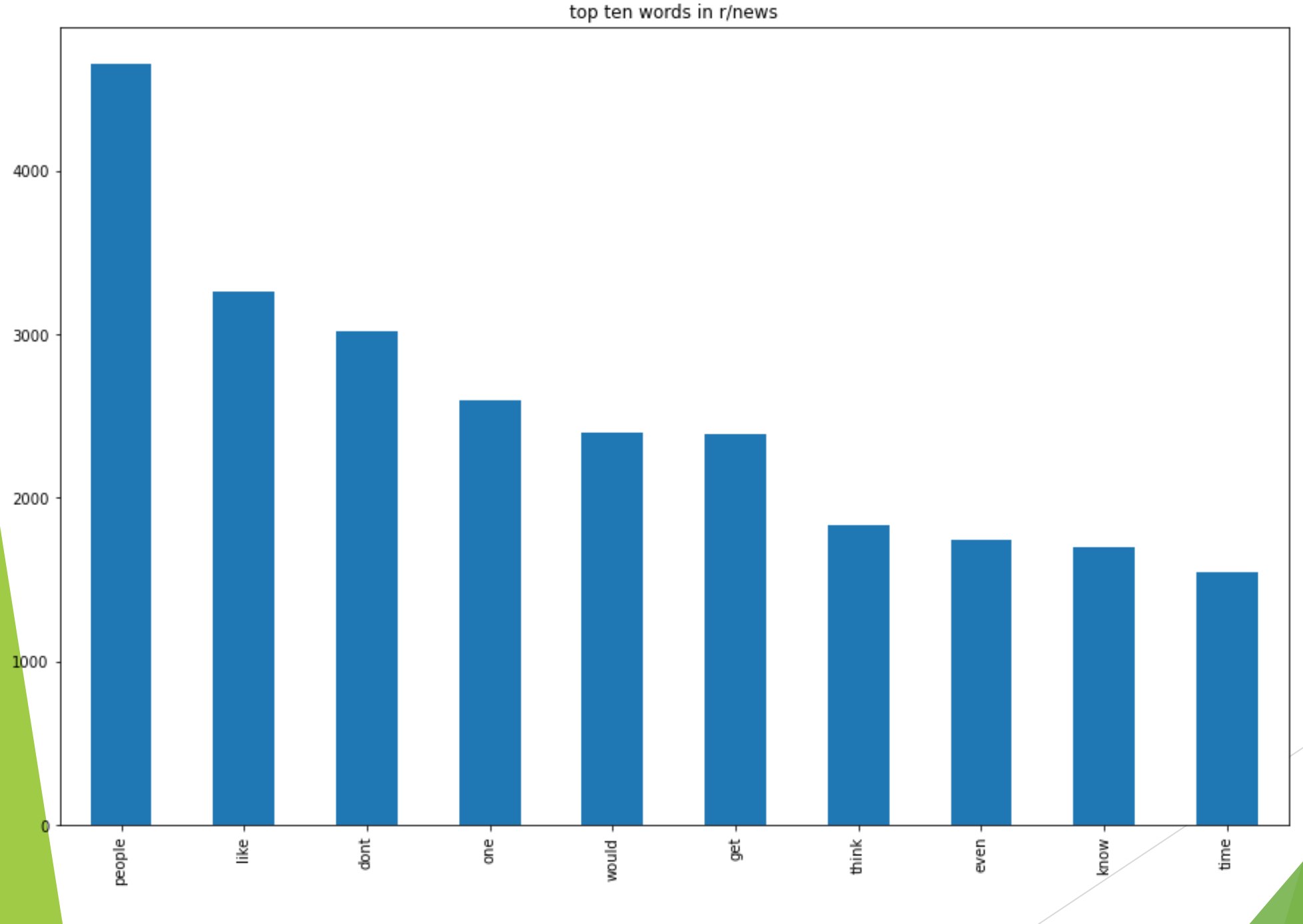
Data cleaning and preparation

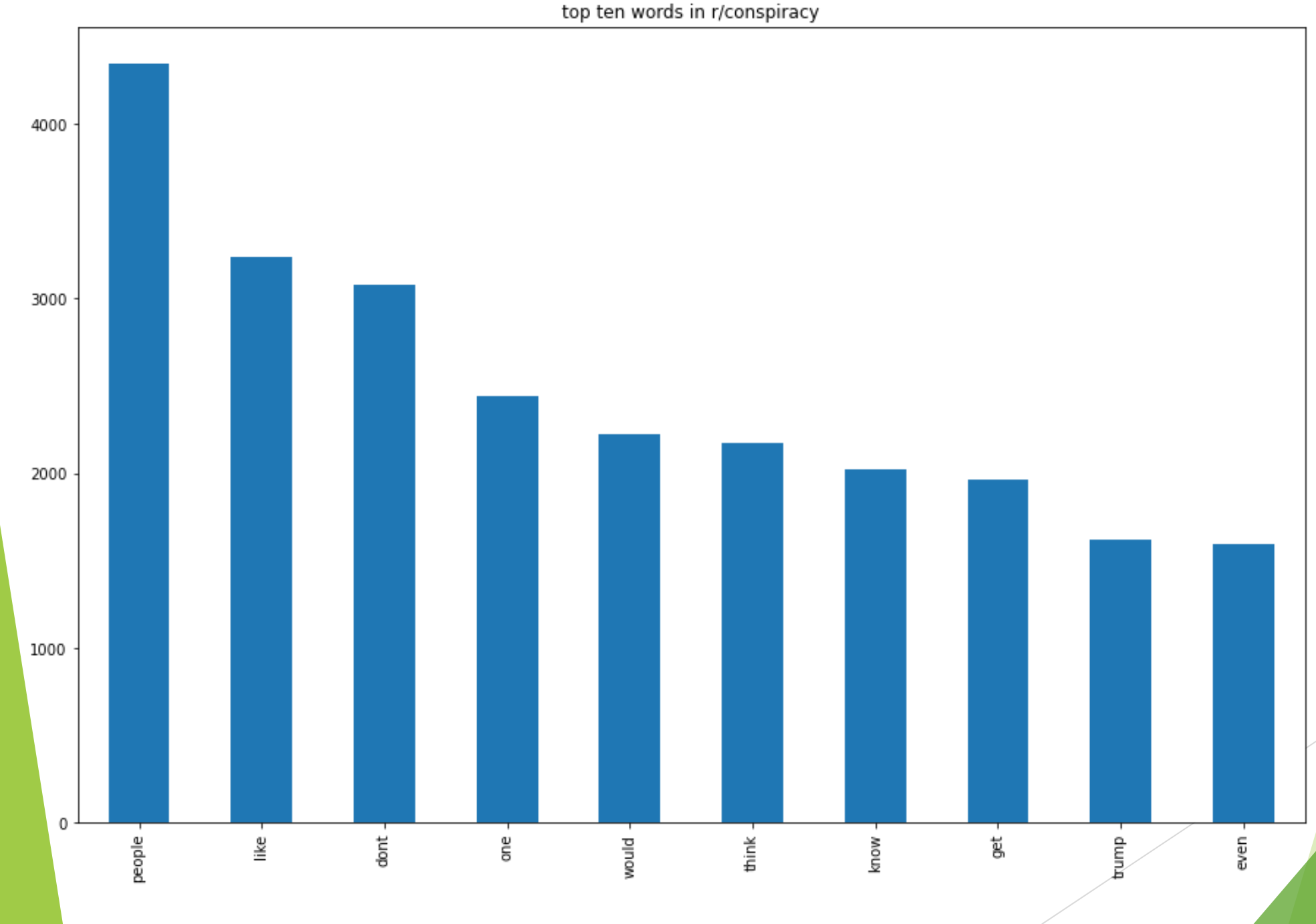
- Deleted comments, moderator comments removed (around 15% of content)
- Non-English character removed
- > < $, & | \ / [( ]) and other characters removed
- More than 3 letters in a row replaced ie. whaaaaaaaaaaat -> what
- URLs removed and counted
- Words lemmatized
- Sklearn stop words used, added more including “news” and “conspiracy”
- Tfidf vectorizer used to vectorize words into sparse matrix


NEWS

Features engineered

1. **cl_count** - Number of capital letters appearing in each post
2. **exc_count** - Number of exclamation points appearing in each post
3. **qm_count** - Number of question marks points appearing in each post
4. **char_count** - Number of individual characters in each post
5. **word_count** - Number of words in each post
6. **url_count** - Number of urls in each post
7. **elongated_vowels** -
   Instances of three or more vowels together appearing in each post: ie "aaa", "iii", "uuu", etc
8. **percent_count** - Number of times % or 'percent' appears in a post
9. **q_word_count -**
   Number of occurences of question words "who," "what", "when", "why", "where", and "how"
10. **number_count** - Occurences of digits appearing in each post


top ten words in r/news
4000
3000
2000
1000
0
people
like
dont
one
would
get
think
even
know
time


top ten words in r/conspiracy
4000
3000
2000
1000
0
people
like
dont
one
would
think
know
get
trump
even

TF-IDF Vectorizer use

$$w_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right)$$

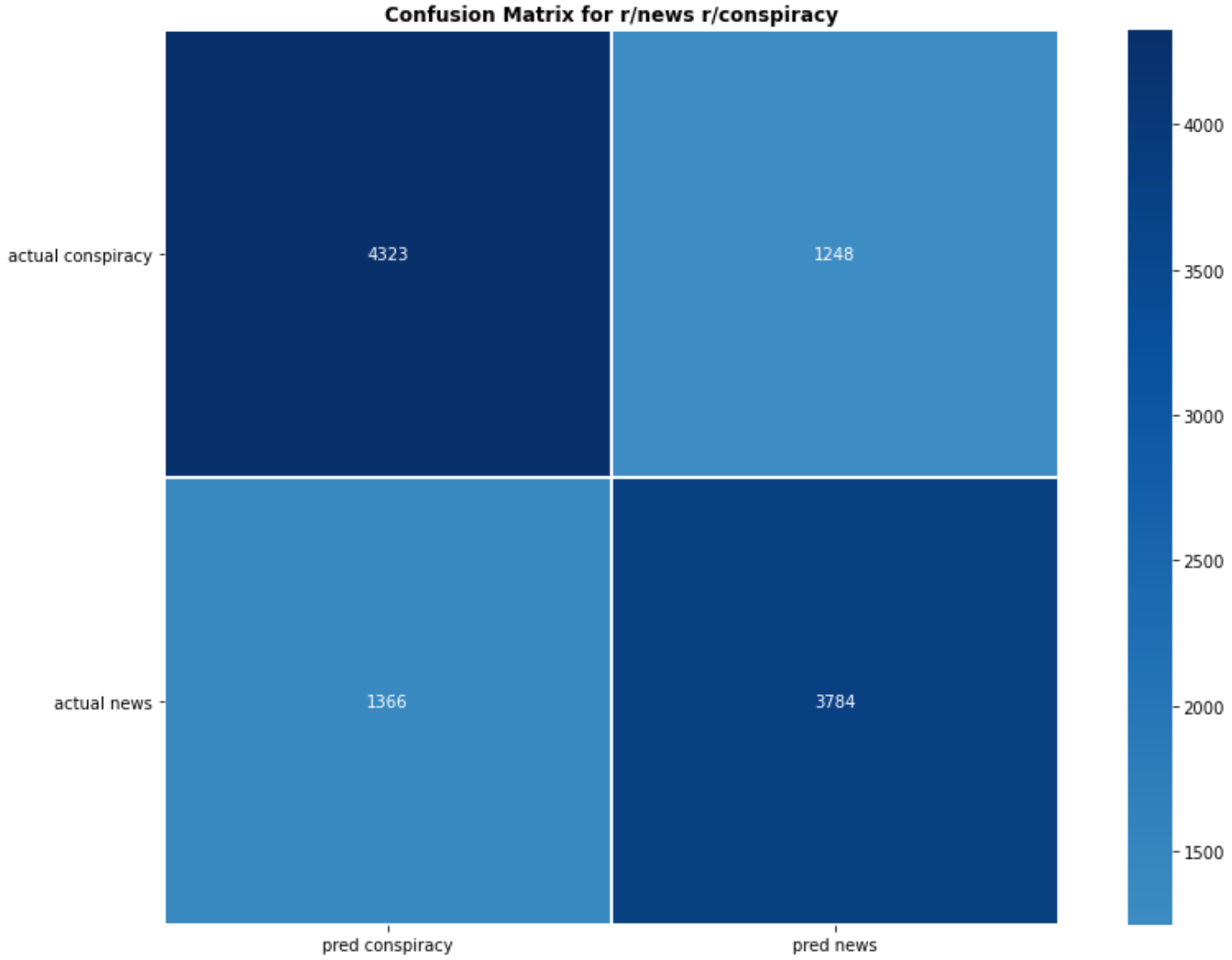
- Decided to use a TfidfVectorizer in combination with stop words to further filter word importance
  - Bag of words alone doesn't filter noise
- Uses (term frequency) x (inverse data frequency)
  - Term frequency: Number of times a word appears in a document divided by the total number of words in the document
  - Inverse data frequency: How common or not common a word is in a set of documents
    - The log of number of documents divided by number of documents that contain the word

Final model: Multinomial Naïve Bayes

- Dropped the extra features: they didn't help
- Multinomial Naïve Bayes
  - Alpha: 0.4
  - Fit_prior: false
- TfidfVectorizer
  - analyzer: "word“
  - ngram_range = (1,3)
  - stop_words = 'english’
  - max_df = 0.15
  - L2 (Ridge) regularization
- Training accuracy: 0.986
- Testing accuracy: 0.756

Misclassification

Accuracy on test data: 75.6%
Sensitivity: 73%
Specificity: 77.6%



Misclassified as news

- “thank you for this very informative historical insight. ” 0.952741
- “this is just another story.. thanks for clearing that up. ” 0.652555
- “it’s getting outta hand. smfh” 0.627631

Misclassified as conspiracy

- “this is definitely not the time to have that traditional wedding.” 0.537267
- “geez louise dude, i’m sorry. ” 0.520081
- “that doesn't sound pretty nice at all” 0.866516
- “this is true. it depends what kind of restaurant it is. my mom owns an asian take-out restaurant so since we were already optimized for take-out, covid didn’t really affect us much. all we had to do was close down seating. it’s constantly busy and we’ve been making almost double what we used to after we opened again. ” 0.860726

Runner-up model: Linear SVC

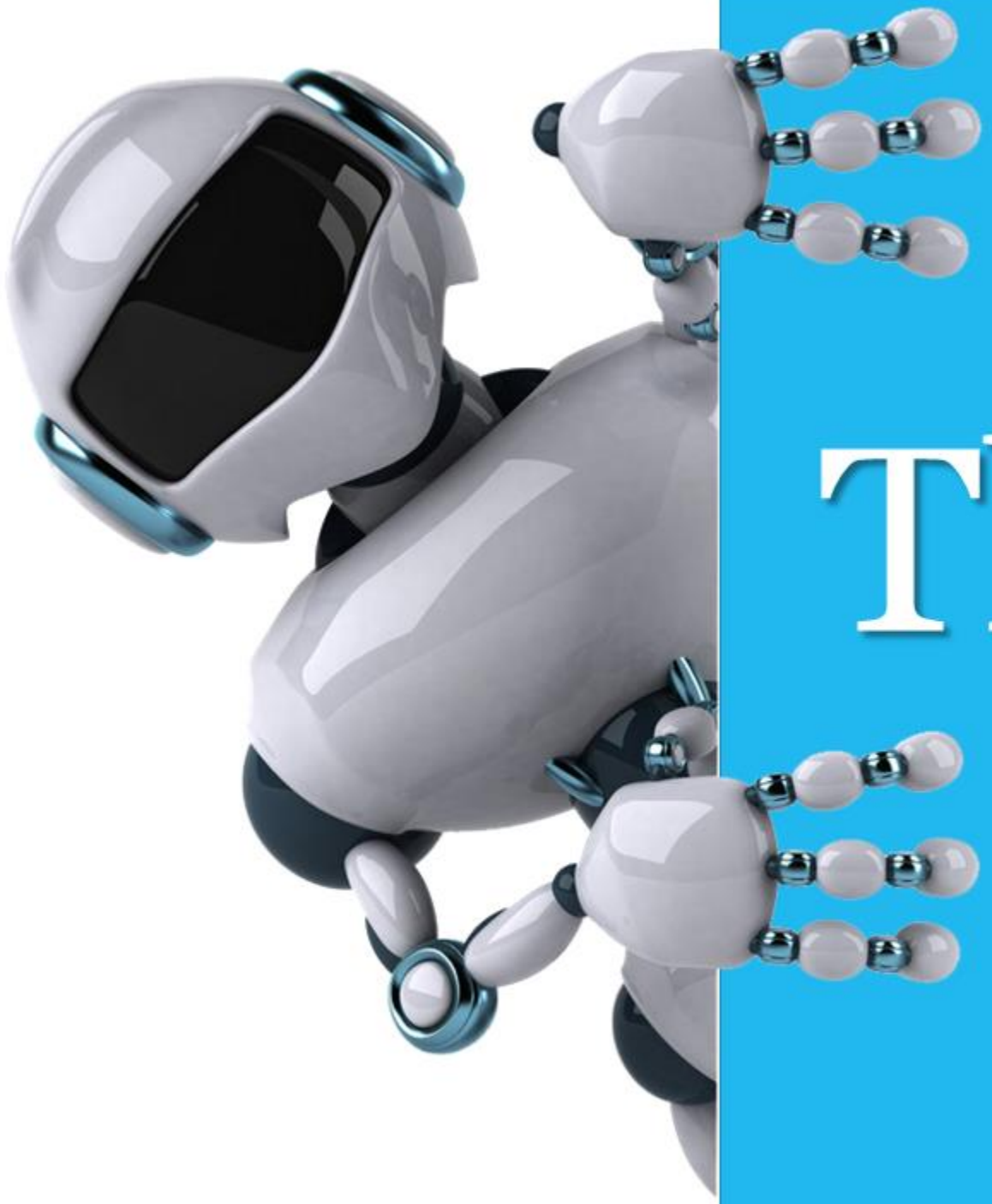
- Linear Support Vector Classifier
  - Other kernel shapes attempted with not as great results
  - Rbf kernel gave similar results to linear SVC
- Features including engineered features
  - Log of features scaled
- TfidfVectorizer
  - analyzer: “word”
  - ngram_range = (1,3)
  - max_df = 0.15
  - stop_words = None
- Training score: 0.994
- Testing score: 0.742

Feature importance

- 'coordination work', 2.809
- 'germ theory modern', 2.677
- 'pole simply', 2.644
- 'ket us bad', 2.623
- 'german war', 2.581
- 'genocidal law', 2.315
- 'republic process', 2.183
- 'stylistic', -3.814
- 'pore mask fucking', -2.794
- 'vaccination similarly glaxosmithklines', -2.775
- 'pedestrian failure deliberate', -2.768
- 'earnings report', -2.611
- 'possibly used', -2.505
- 'pedophelia right', -2.479

Key Takeaways

- Despite similar content, word occurrence and frequency is a powerful basis on which to classify comments to news and conspiracy subreddits with around 75% accuracy.
- I would like to further explore feature engineering and its uses in NLP. What are new ways that I can add meaning and predictive power to words from which computers can learn?
- Use beyond this project: While this project specifically likely does not answer any substantial world problem, similar projects could be designed for fields such as marketing where certain words may indicate a tendency to gravitate towards certain topics.



Thank you