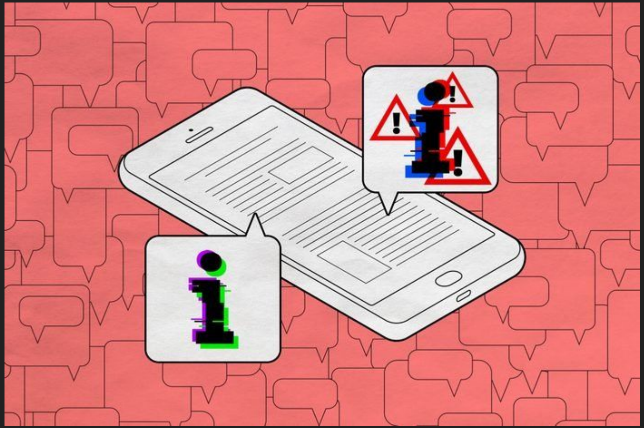
How to Recognize Fake News

Team Marshmallow:
Kyungjin Sohn, Soohyun Kim, Winnie Ren, Yang Xiao, Ziang Li

Motivation

- Covid-19 pandemic was accompanied by the circulation of misinformation, myths, and conspiracy theories about the disease.
- In 2021, nearly **eight in ten** adults believe or are unsure about at least one false claim related to COVID-19 (Kaiser Family Foundation, 2022)
- The circulation of fake news during periods of great political activity

Research Question



“Build a **fake news detection model**

using text characteristics”

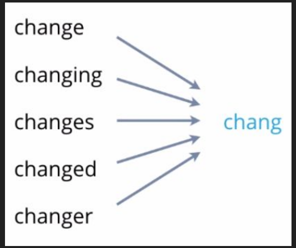
Data

- From Kaggle “**Fake and real news dataset**” (https://www.kaggle.com/clmentbisaillon/fake-and-real-news-dataset)
- A total of 44,898 observation
  - Fake: 23,481 and True: 21,417
- Data set includes: Title, Text, Subject of Article, Date

Title	Text	Subject	Date
Donald Trump Sends Out Embarrassing New Year’s Eve Message; This is Disturbing	Donald Trump just couldn t wish all Americans a Happy New Year and leave it at that. Instead, he had...	News	December 31, 2017
Drunk Bragging Trump Staffer Started Russian Collusion Investigation	House Intelligence Committee Chairman Devin Nunes is going to have a bad day. He s been under the as...	News	December 31, 2017

Data Cleaning

- Needed to remove the prefix "(Reuters)" in the true dataset
- To create corpus of our textual data,
  - removed numbers, punctuation, special characters, whitespace, stopwords
  - changed all text to lowercase
  - stemmed each word



Ex) Stemming

```
"WASHINGTON (Reuters) -
The head of a
conservative Republican
faction in the U.S.
Congress, who voted this
month for a huge
expansion of the national
debt to pay for tax cuts,
called himself a "fiscal
conservative" on Sunday
and urged budget
restraint in 2018."
```

ORIGINAL

```
The head of a
conservative Republican
faction in the U.S.
Congress, who voted this
month for a huge
expansion of the
national debt to pay for
tax cuts, called himself
a "fiscal conservative"
on Sunday and urged
budget restraint in
2018.
```

Prefix Removed

```
head conservative
republican faction
us congress voted
month huge
expansion national
debt pay tax cuts
called fiscal
conservative sunday
urged budget
restraint
```

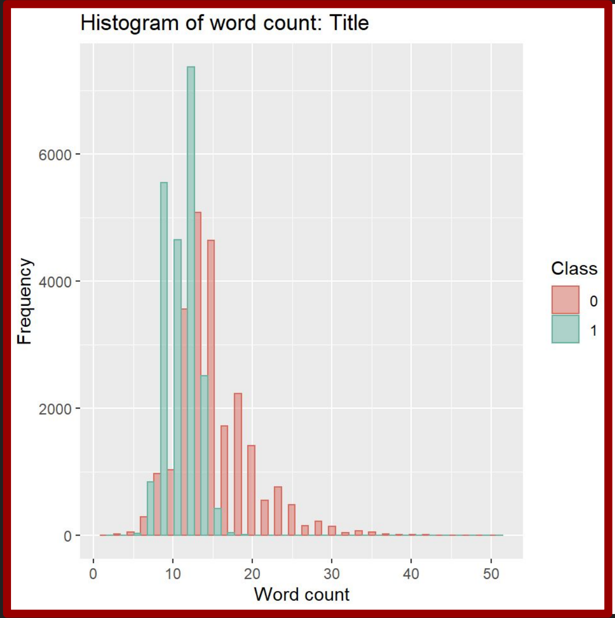
Corpus w/o Stemming

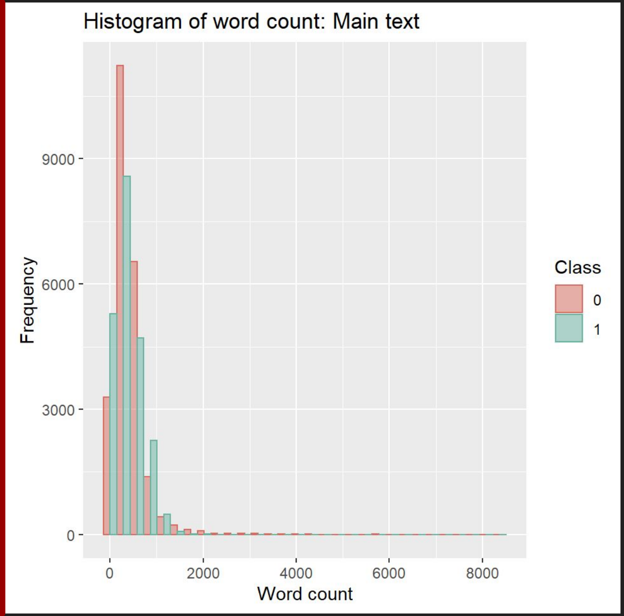
```
head conserv republican
faction us congress vote
month huge expans nation
debt pay tax cut call
fiscal conserv sunday
urg budget restraint
```

Corpus w/ Stemming

Exploratory Data Analysis

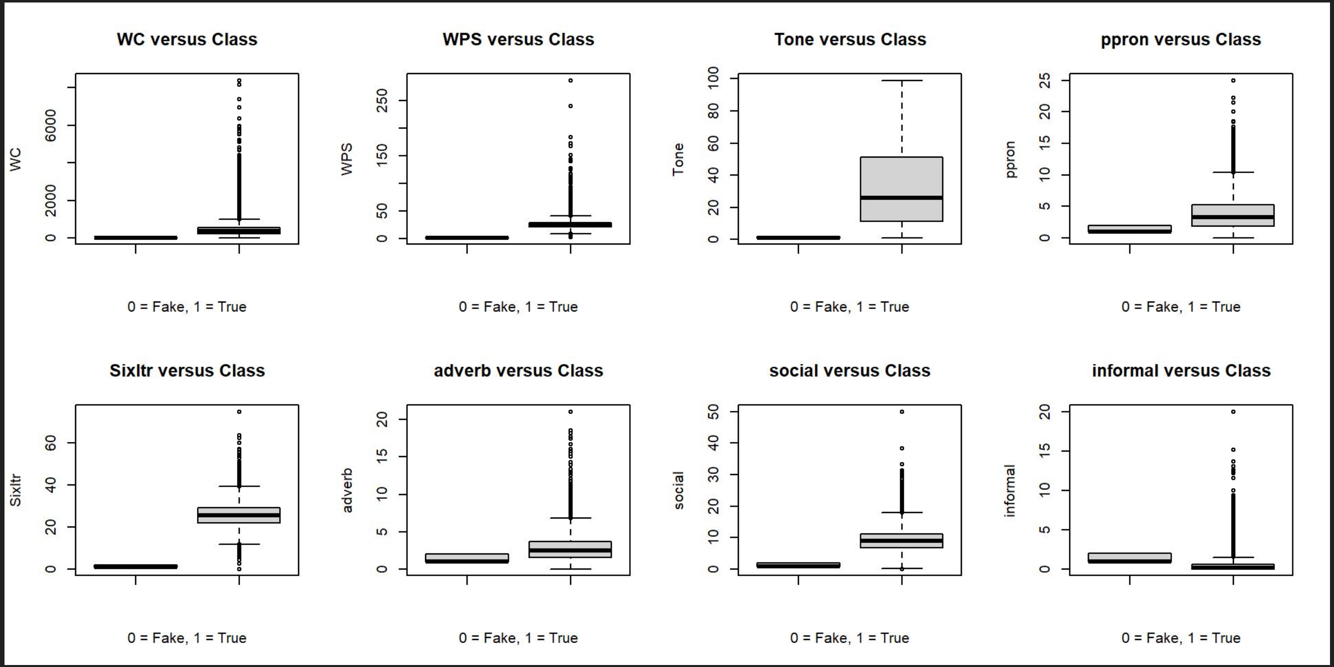
- Length comparison between Fake and True News
    - The spread of fake news is wider compared to the true news





Exploratory Data Analysis

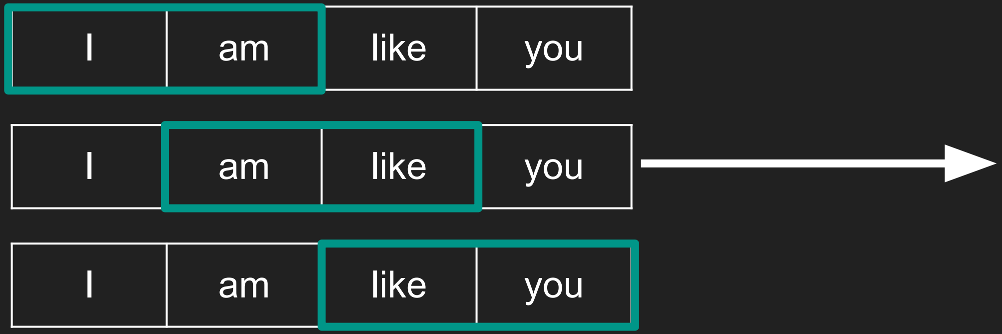
- Comparison of Text Characteristics using Linguistic Inquiry and Word Count (LIWC)



Feature Extraction - 1. Ngram Analysis

- Bag of n-gram (n=2)
  - “I like you”
  - “I am like you”
- Term Frequency (TF)

TF = (term count in the doc)/(total number of terms in the doc)



Term	Frequency	TF
I am	1	**1/3**
am like	1	**1/3**
Like you	1	**1/3**

Total number of terms in doc = 3

Feature Extraction - 1. Ngram Analysis

- TF-Inverse Document Frequency(TF-IDF)
  - The normalized version of Bag of n-gram

TF-IDF =TF * IDF

IDF = log{(number of docs)/(number of docs with this word)}

Term	Review 1	Review 2	Review 3	IDF	TF-IDF (Review 1)	TF-IDF (Review 2)	TF-IDF (Review 3)
This	1	1	1	0.00	0.000	0.000	0.000
movie	1	1	1	0.00	0.000	0.000	0.000
is	1	2	1	0.00	0.000	0.000	0.000
very	1	0	0	0.48	0.068	0.000	0.000
scary	1	1	0	0.18	0.025	0.022	0.000
and	1	1	1	0.00	0.000	0.000	0.000
long	1	0	0	0.48	0.068	0.000	0.000
not	0	1	0	0.48	0.000	0.060	0.000
slow	0	1	0	0.48	0.000	0.060	0.000
spooky	0	0	1	0.48	0.000	0.000	0.080
good	0	0	1	0.48	0.000	0.000	0.080

https://www.analyticsvidhya.com/blog/2020/02/quick-introduction-bag-of-words-bow-tf-idf/

Feature Extraction - 1. Ngram Analysis

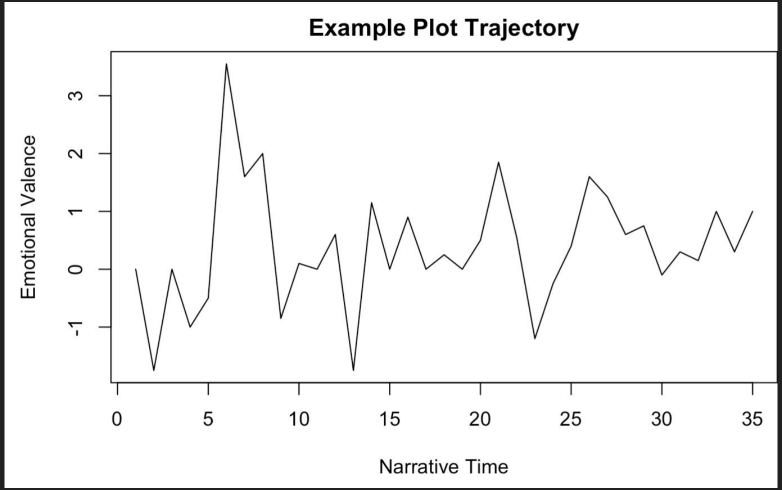
- Used the **'bind_tf_idf'** function in the **'janeaustenr'** library
- Bigrams: n=2
- Select the top 20 most frequently used terms for Fake and True news title and calculated the TF and TF-IDF
  - 40 TF + 40 TF-IDF
- Repeat for Text, discarding the overlapping terms
  - 33 TF + 33 TF-IDF

donald.trump_tf_title <dbl>	donald.trump_tf_idf_title <dbl>
0.1250000	0.4973433
0.0000000	0.0000000
0.0000000	0.0000000
0.0000000	0.0000000
0.1428571	0.5683924
0.0000000	0.0000000

year.old_tf_text <dbl>	year.old_tf_idf_text <dbl>
0.007407407	0.018052051
0.000000000	0.000000000
0.003937008	0.009594594
0.000000000	0.000000000
0.000000000	0.000000000
0.000000000	0.000000000

Feature Extraction - 2. Sentiment Analysis

"Syuzhet" Package



- get_sentences()
  - openNLP sentence tokenizer
  - parsing a text into a vector of sentences
- get_sentiment()
  - Assess sentiment of each sentences
  - Syuzhet is default method
  - "bing", "afinn", "nrc", and "stanford"
- get_nrc_sentiment()
  - Saif Mohammad's NRC Emotion lexicon with eight emotions (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) and two sentiments (negative and positive)

Feature Extraction - 2. Sentiment Analysis

- Different methods return slightly different results since each methods use slightly different scale

```
bing_vector <- get_sentiment(poa_v, method = "bing")
head(bing_vector)
```

```
## [1]  1  0 -1 -1  0  0
```

```
afinn_vector <- get_sentiment(poa_v, method = "afinn")
head(afinn_vector)
```

```
## [1] 3 0 0 1 0 0
```

```
nrc_vector <- get_sentiment(poa_v, method = "nrc", lang = "english")
head(nrc_vector)
```

```
## [1]  1  1 -1  0  0  0
```

Feature Extraction - 2. Sentiment Analysis

Features created for Title and main Text:

1) Minimum, 1st Quartile, Median, Mean, 3rd Quartile, Maximum (1+6)
2) Anger, Anticipation, Disgust, Fear, Joy, Sadness, Surprise,Trust (16)

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-1.7500	0.0000	0.3000	0.3714	0.9500	3.5500

anger	anticipation	disgust	fear	joy	sadness	surprise	trust
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
6	5	6	3	7	5	5	9

Classification Models

- Train : Test data split = 75 : 25
- Methods
  1) LDA
  2) QDA: 39 variables
  3) KNN: k = 1, k = 184
  4) SVC(linear): cost = 5
  5) SVM(polynomial): cost = 1, degree = 1
  6) Decision Trees: 11 variables
  7) Bagging: m = 169
  8) Random Forest: m = 13
  9) Logistic
  10) Neural Networks: learning rate = 0.01

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_2 (Dense)             (None, 128)               21760

 dropout_1 (Dropout)         (None, 128)               0

 dense_1 (Dense)             (None, 64)                8256

 dropout (Dropout)           (None, 64)                0

 dense (Dense)               (None, 1)                 65

=================================================================
Total params: 30,081
Trainable params: 30,081
Non-trainable params: 0
_________________________________________________________________
```

Method	Accuracy	Sensitivity	Specificity	False Positive	Test Error
LDA	0.864	0.856	0.871	0.129	0.136
QDA	0.800	0.874	0.734	0.266	0.200
KNN (K=1)	0.891	0.869	0.911	0.089	0.109
KNN (K=184)	0.868	0.905	0.835	0.165	0.132
SVC	0.858	0.82	0.892	0.108	0.142
SVM	0.732	0.76	0.707	0.293	0.268
Decision Tree	0.788	0.616	0.943	0.057	0.212
Bagging	0.923	0.902	0.941	0.059	0.077
Random Forest	**0.931**	**0.914**	**0.947**	**0.053**	**0.069**
Logistic	0.888	0.869	0.905	0.095	0.112
Neural Network	0.887	0.890	0.884	0.116	0.113

Classification Models

- Test-error simulation
  1) Randomly extracted 30% of the data from the test set
  2) Computed the test error
  3) Repeated step 1 and 2 50 times

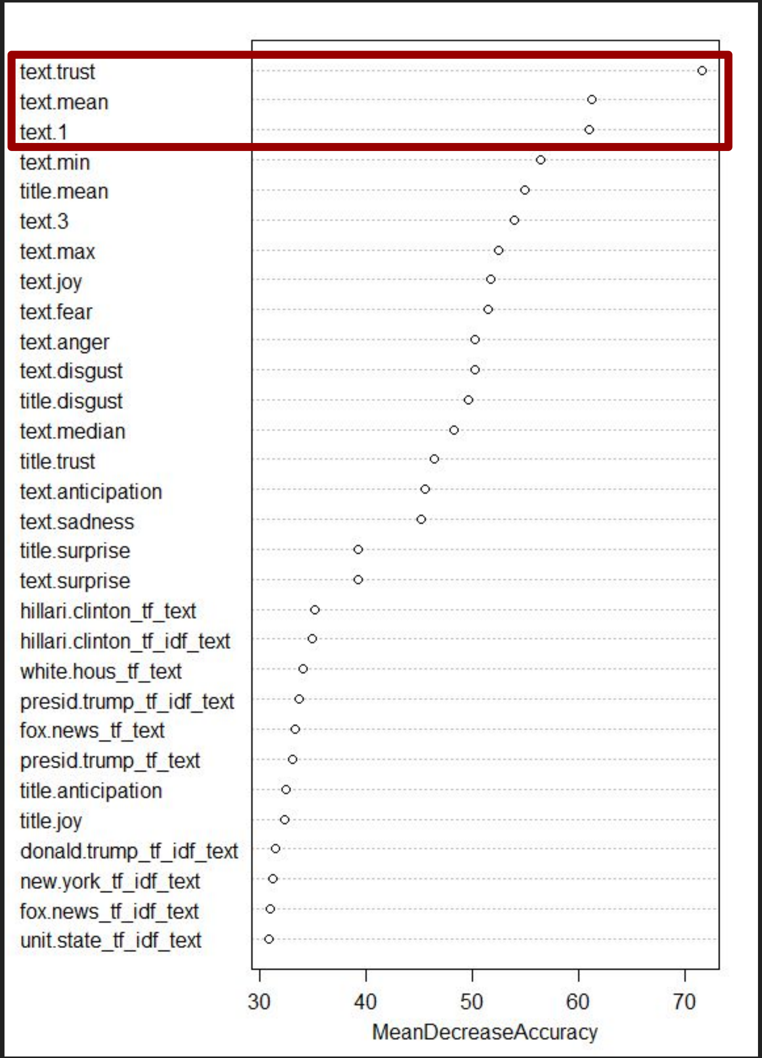
Comparison of error rates between different classifications
Error rate
0.25
0.20
0.15
0.10
0.05
LDA
QDA
KNN..K.1.
KNN..K.184.
SVC
SVM
Decision.Tree
Bagging
Random.Forest
Logistic
Neural.Network
Methods

Classification Models

- Best model: **Random Forest**
- **Mean Decrease Accuracy**

↑ MeanDecreaseAccuracy = ↑ Variable Importance

1) (Emotion = trust) of text data
2) Mean sentiment of the text
3) Lower quartile of text sentiment



Conclusion

- Based on our model, we can clearly identify fake news using text characteristics.
- Applying this, we hope to help people to be able to clearly distinguish fake news and prevent negative ramifications from fake news.

Future Work

- Besides n-gram and sentiment analysis, we can study other feature extraction methods.
- Despite the fact that we have achieved an accuracy of over 90% using the Random Tree Model, other models such as RNN might be able to increase the accuracy further
- In the future, it would be interesting to see if our model is applied to different datasets and compare the accuracy of this model

Reference

[1] Misinformation vs. Disinformation: How to Tell the Difference (https://www.rd.com/article/misinformation-vs-disinformation/)
[2] Fake and real news dataset (https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset)
[3] Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques (https://www.researchgate.net/publication/320300831_Detection_of_Online_Fake_News_Using_N-Gram_Analysis_and_Machine_Learning_Techniques)
[4] Introduction to the Syuzhet Package (https://cran.r-project.org/web/packages/syuzhet/vignettes/syuzhet-vignette.html)
[5] Quick Introduction to Bag-of-Words (BoW) and TF-IDF for Creating Features from Text (https://www.analyticsvidhya.com/blog/2020/02/quick-introduction-bag-of-words-bow-tf-idf/)
[6] Psychological Determinants of the Susceptibility to Fake News amidst the COVID-19 Pandemic (http://www.clinicalbrain.org/publication/2020_deception-covid-fakenews/)

Thank you!