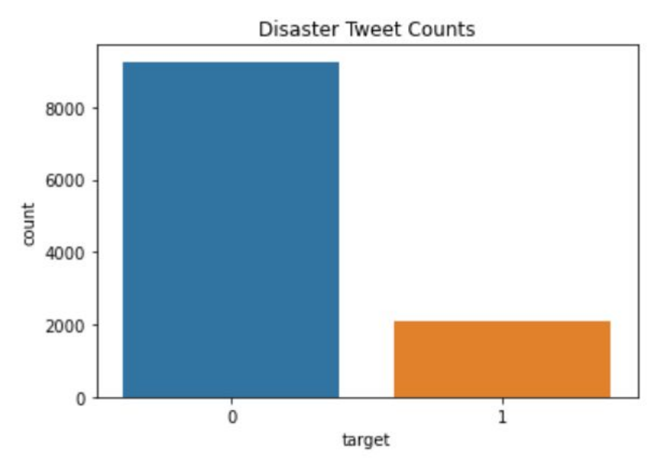
Disaster Tweet Detection

Justin Bell

How can we build an NLP model that is able to classify tweets as disasters with high accuracy?

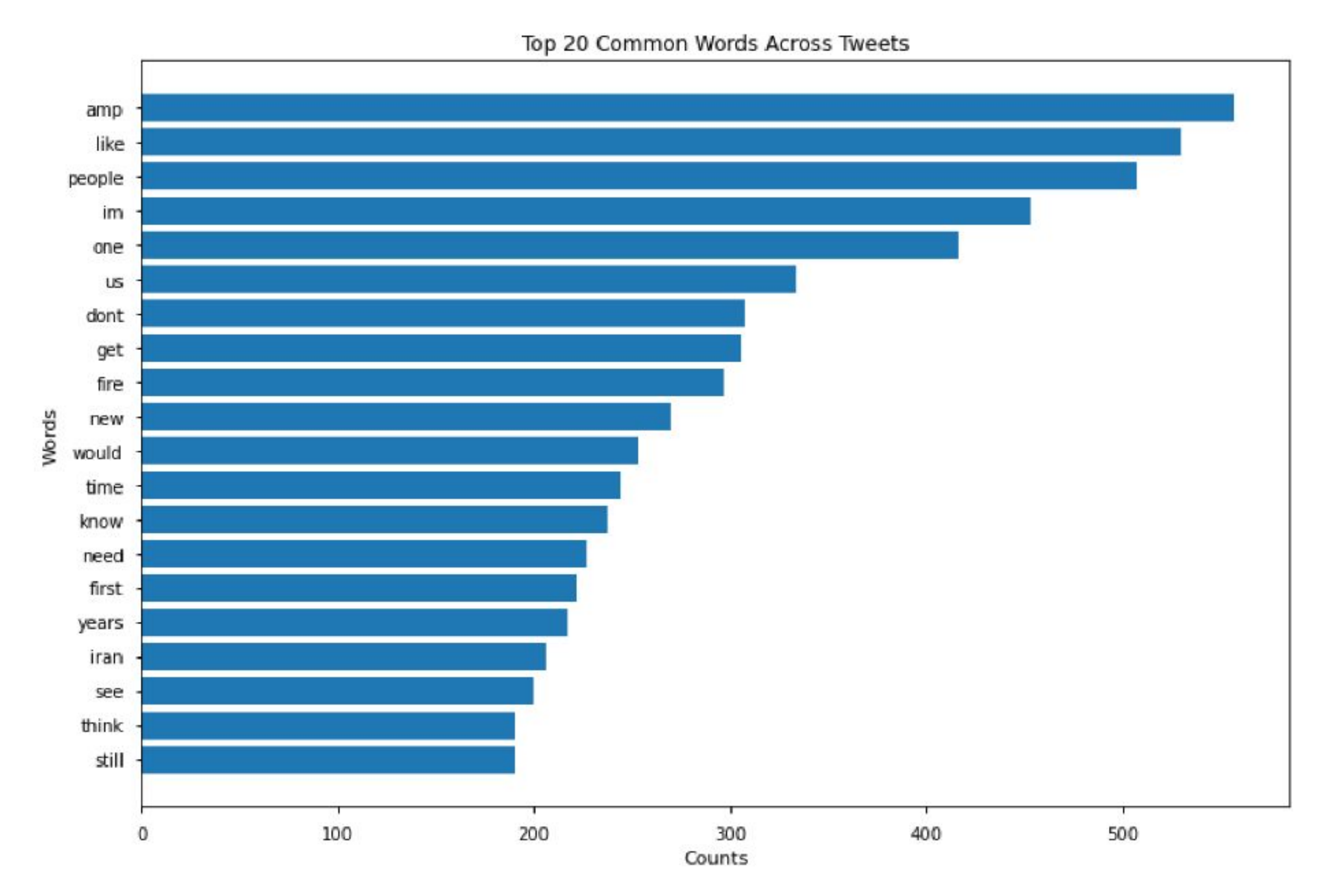
Data Wrangling

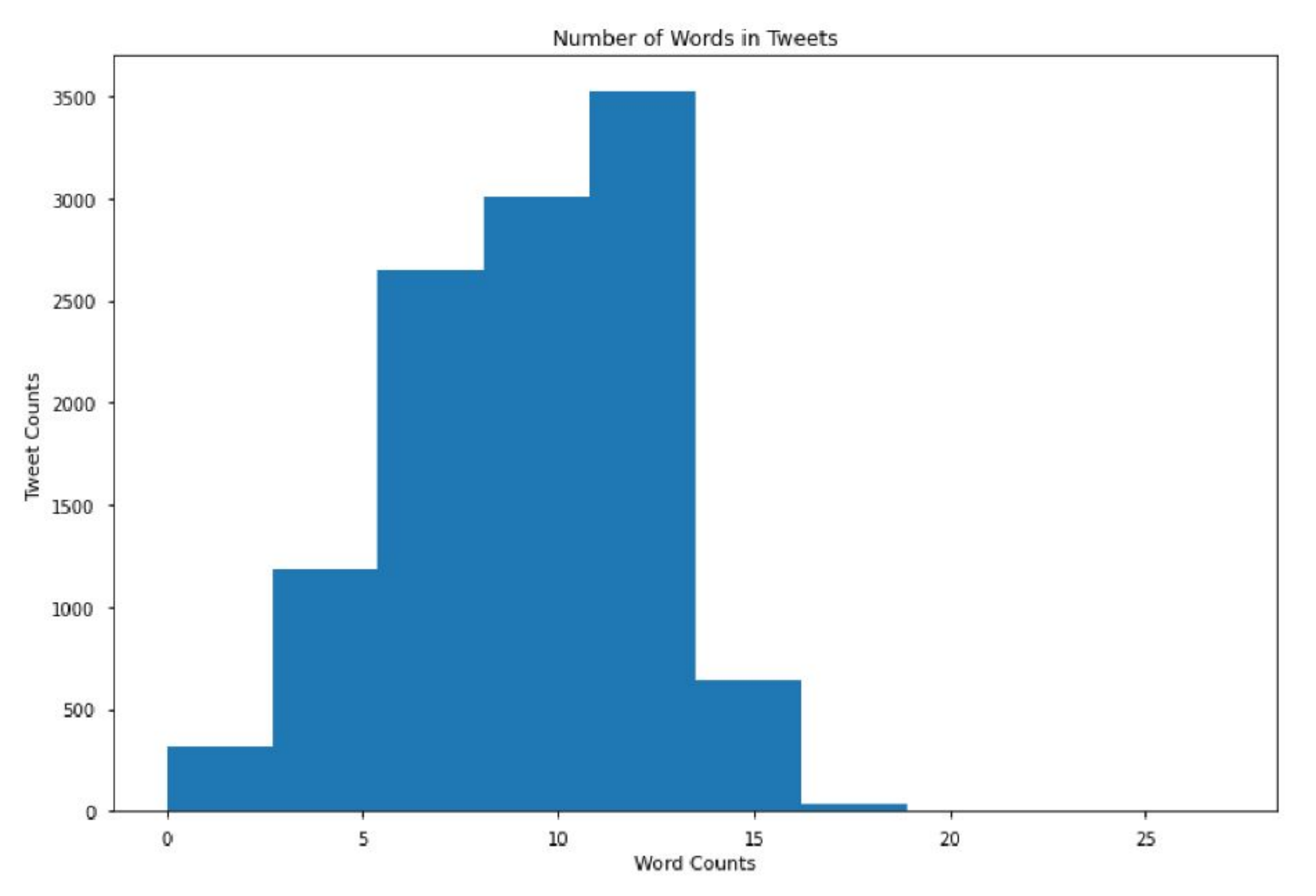
- Import data
- Check for balanced target classes
- Data cleaning
  - Converting text to lowercase
  - Regular expression string removal
  - Tokenized text
  - Remove stop words

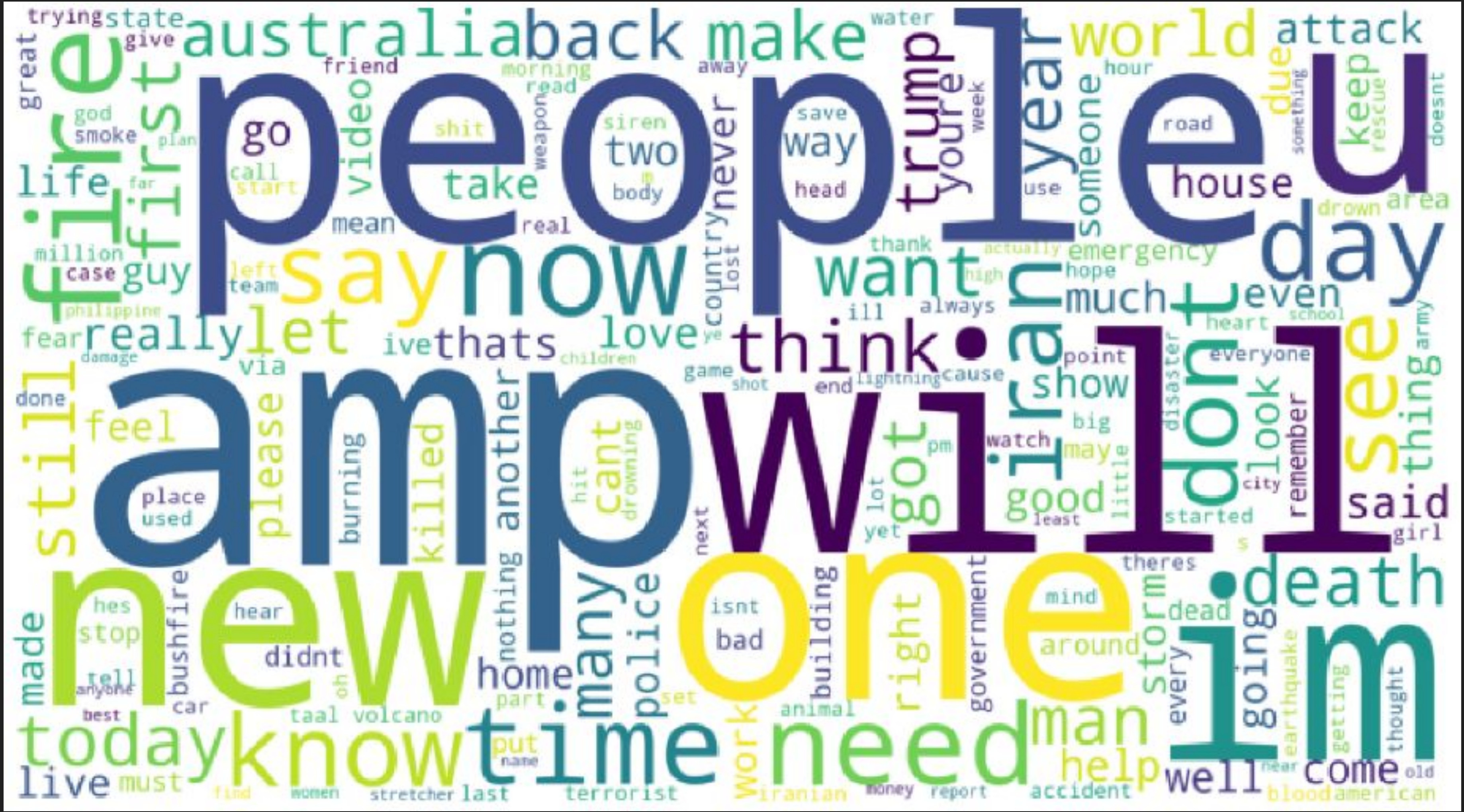


EDA

- Create some plots:
  - Top 20 Words → hbar plot
  - Words per Tweet → histogram
  - Word Cloud


Top 20 Common Words Across Tweets
Words
amp
like
people
im
one
us
dont
get
fire
new
would
time
know
need
first
years
iran
see
think
still
0
100
200
300
400
500
Counts

Number of Words in Tweets
Tweet Counts
3500
3000
2500
2000
1500
1000
500
0
0
5
10
15
20
25
Word Counts


people
amp
will
u
one
new
im
fire
first
say
now
australia
back
make
world
attack
trump
year
iran
day
want
think
dont
see
death
know
time
need
today
man
really
let
thats
love
go
two
way
take
video
never
youre
house
much
even
someone
keep
due
life
guy
still
feel
please
killed
another
many
police
cant
got
good
show
thing
look
said
storm
going
well
come
help
live
must
made
home
work
right
bad
building
burning
nothing
around
dead
earthquake
government
every
bushfire
stop
hes
tell
hear
didnt
isnt
mind
theres
remember
started
girl
least
little
city
may
big
watch
pm
cause
point
everyone
disaster
army
school
heart
hope
emergency
area
drown
rescue
doesnt
something
road
hour
week
use
head
save
water
away
morning
read
friend
siren
shit
weapon
body
call
start
plan
far
smoke
god
give
state
trying
great
million
case
left
team
philippine
mean
real
country
lost
thank
actually
high
always
ill
ye
fear
damage
children
ive
via
game
shot
end
lightning
done
place
used
hit
drowning
lot
yet
next
s
car
anyone
best
part
set
taal volcano
animal
put
name
last
terrorist
iranian
money
report
accident
blood
american
stretcher
women
find
near
old
getting
thought
oh
m

Preprocessing

- Remove unused columns
- Remove one null row
- Split data into train/test sets
  - 80/20
- Create vectorizers
  - Count
  - TFIDF
- Fit with training features (tweets)
- Transformed training and test features

Modeling

- Tried 5 different classification models
    - Logistic Regression
    - Random Forest
    - KNN
    - SVM
    - Gradient Boosting

Model Hyperparameters

Logistic Regression:

- penalty = l2
- max_iter = 500
- C = 0.1
- solver = lbfgs

Random Forest:

- n_estimators = 500
- n_jobs = -1

KNN:

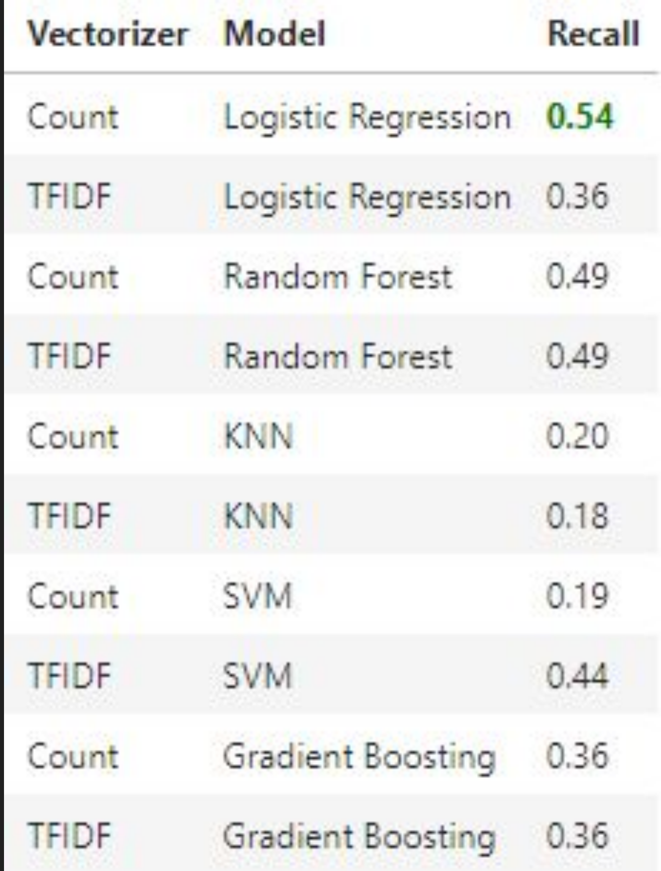
- n_neighbors = 6
- n_jobs = -1

SVM:

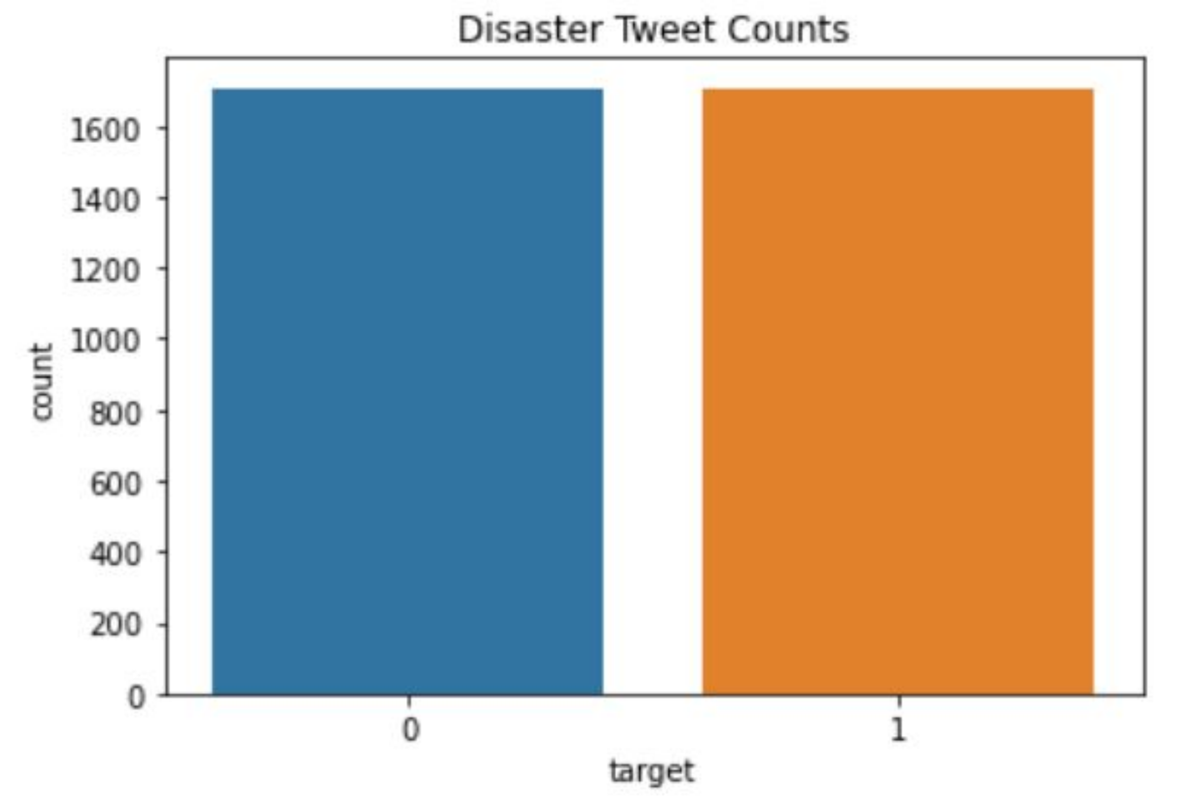
- C= 1
- gamma= 1

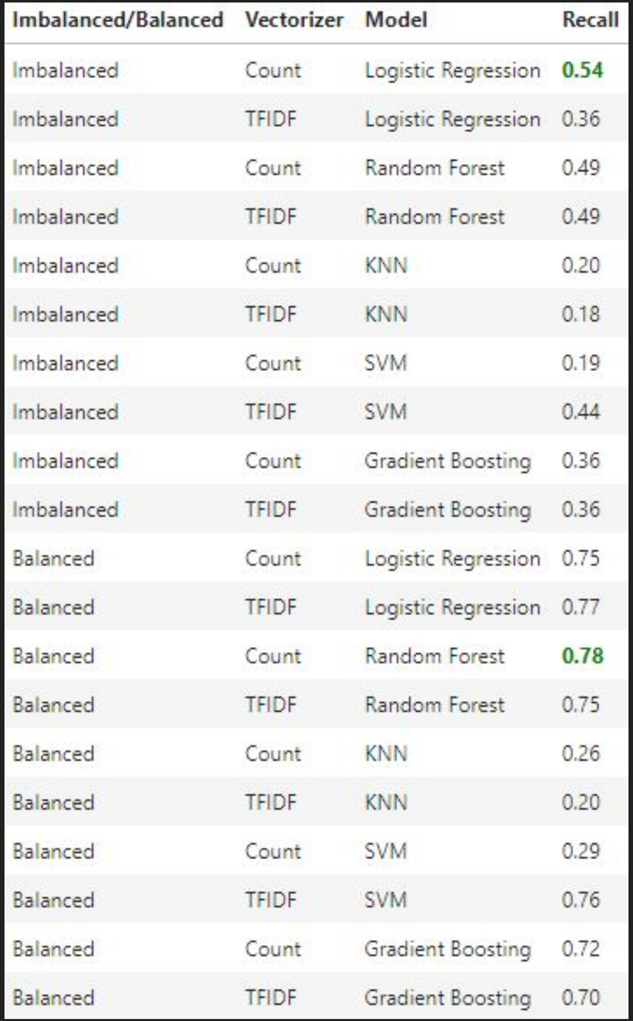
Gradient Boosting:

- n_estimators = 100
- max_depth = 10
- max_features = auto
- learning_rate =0.1

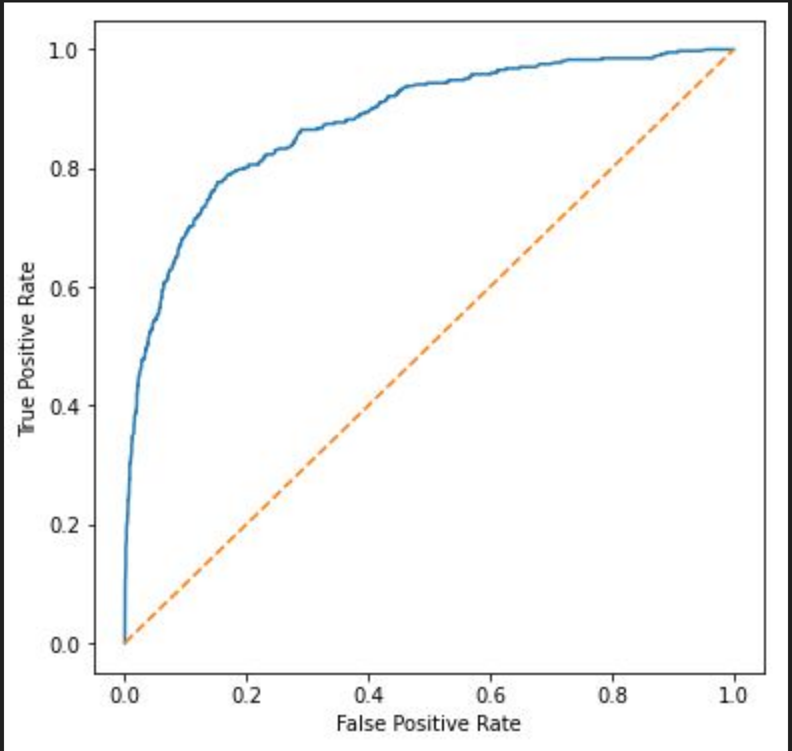


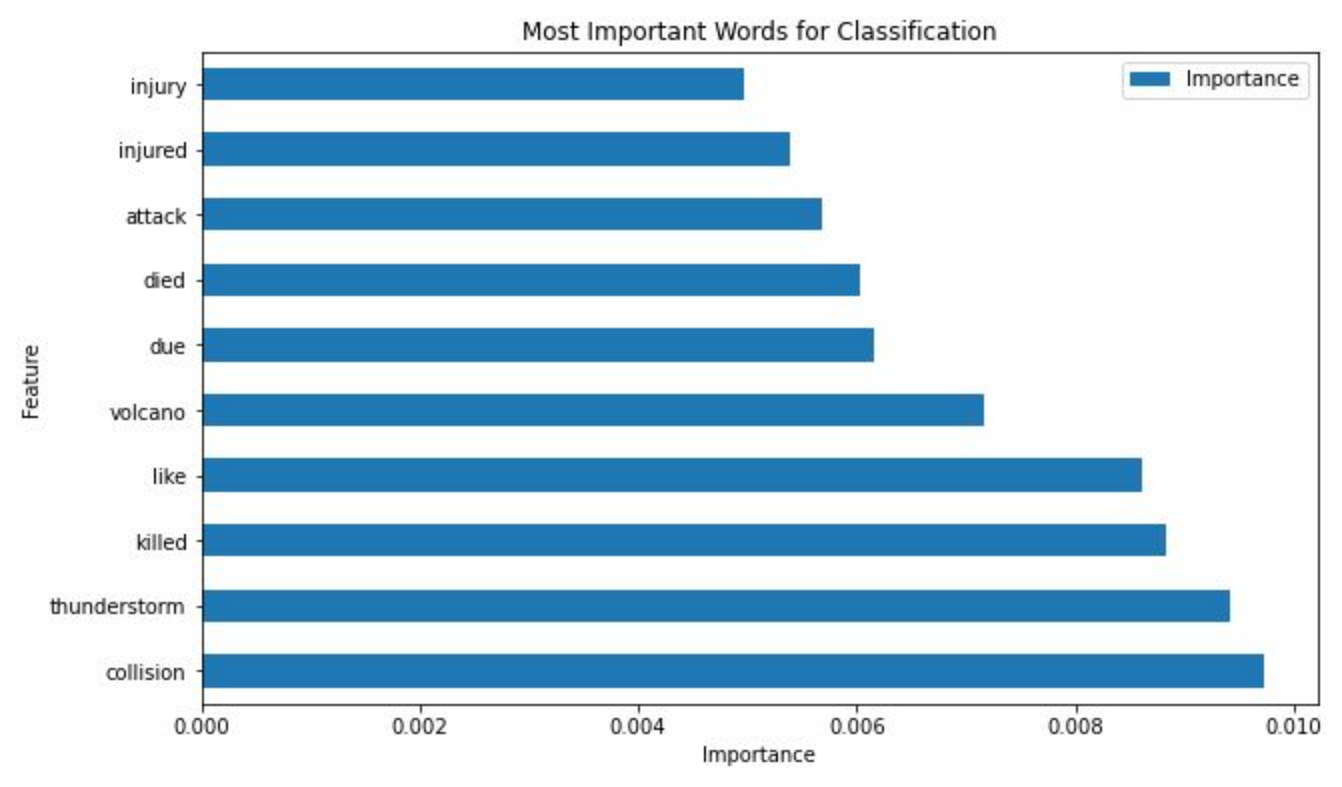
Vectorizer	Model	Recall
Count	Logistic Regression	**0.54**
TFIDF	Logistic Regression	0.36
Count	Random Forest	0.49
TFIDF	Random Forest	0.49
Count	KNN	0.20
TFIDF	KNN	0.18
Count	SVM	0.19
TFIDF	SVM	0.44
Count	Gradient Boosting	0.36
TFIDF	Gradient Boosting	0.36


Disaster Tweet Counts
count
0
200
400
600
800
1000
1200
1400
1600
0
1
target



Imbalanced/Balanced	Vectorizer	Model	Recall
Imbalanced	Count	Logistic Regression	**0.54**
Imbalanced	TFIDF	Logistic Regression	0.36
Imbalanced	Count	Random Forest	0.49
Imbalanced	TFIDF	Random Forest	0.49
Imbalanced	Count	KNN	0.20
Imbalanced	TFIDF	KNN	0.18
Imbalanced	Count	SVM	0.19
Imbalanced	TFIDF	SVM	0.44
Imbalanced	Count	Gradient Boosting	0.36
Imbalanced	TFIDF	Gradient Boosting	0.36
Balanced	Count	Logistic Regression	0.75
Balanced	TFIDF	Logistic Regression	0.77
Balanced	Count	Random Forest	**0.78**
Balanced	TFIDF	Random Forest	0.75
Balanced	Count	KNN	0.26
Balanced	TFIDF	KNN	0.20
Balanced	Count	SVM	0.29
Balanced	TFIDF	SVM	0.76
Balanced	Count	Gradient Boosting	0.72
Balanced	TFIDF	Gradient Boosting	0.70


1.0
0.8
0.6
0.4
0.2
0.0
True Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate


Most Important Words for Classification
Importance
injury
injured
attack
died
due
volcano
like
killed
thunderstorm
collision
Feature
0.000
0.002
0.004
0.006
0.008
0.010
Importance

Future Scope

- Best model can be used by disaster relief organizations
- Explore more models
  - Deep Learning Neural Networks
- Plot word clouds for subsetted data (TP, TN, FP, FN)
- Similar projects