# Classifying Hate Speech on Twitter

Steve Donahue

# About Me

## Mathematician

Master of Arts

Twice-published

Programmer + Data Geek

## Educator

9 years teaching undergraduate courses

Rowan, Rutgers Universities

Camden and Cumberland CC’s

## Millenial

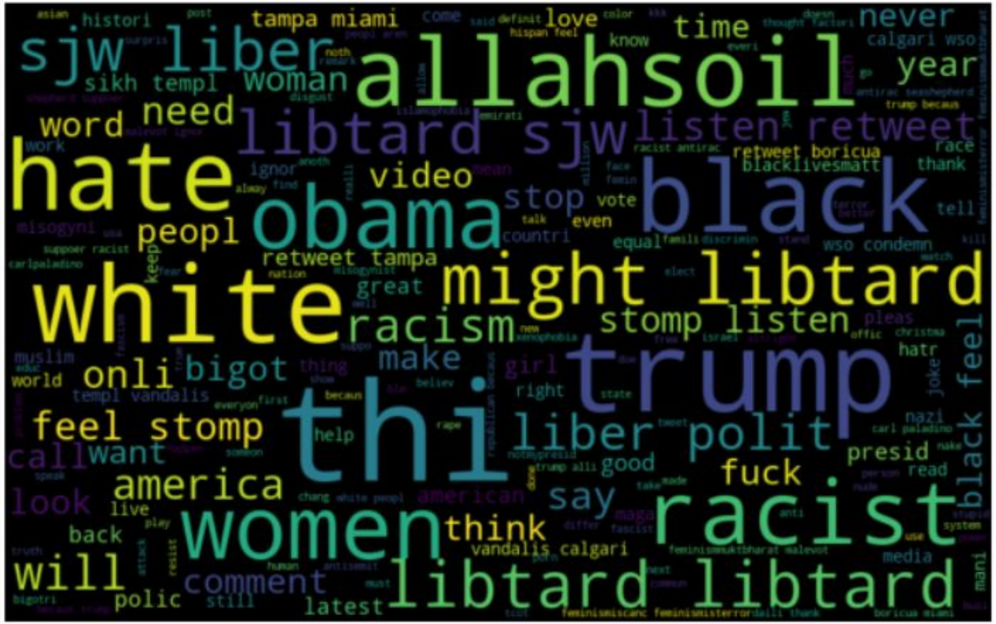
Musician

Athlete

World Traveler

# If you don’t have anything nice to say...

# If you don’t have anything nice to say...



# The Process

Step 1:

Clean the tweets into normal, standardized language so they are comparable.

Step 2:

Determine that hate speech measurably different from normal speech through EDA.

# The Process

Step 3:

Develop various features from the cleaned data set for training ML algorithms.

Step 4:

Train, test, tune, and stack ML models for best results.

# Cleaning Raw Data

1. Remove unhelpful characters

   ( % * & ! etc)

2. Drop unhelpful words

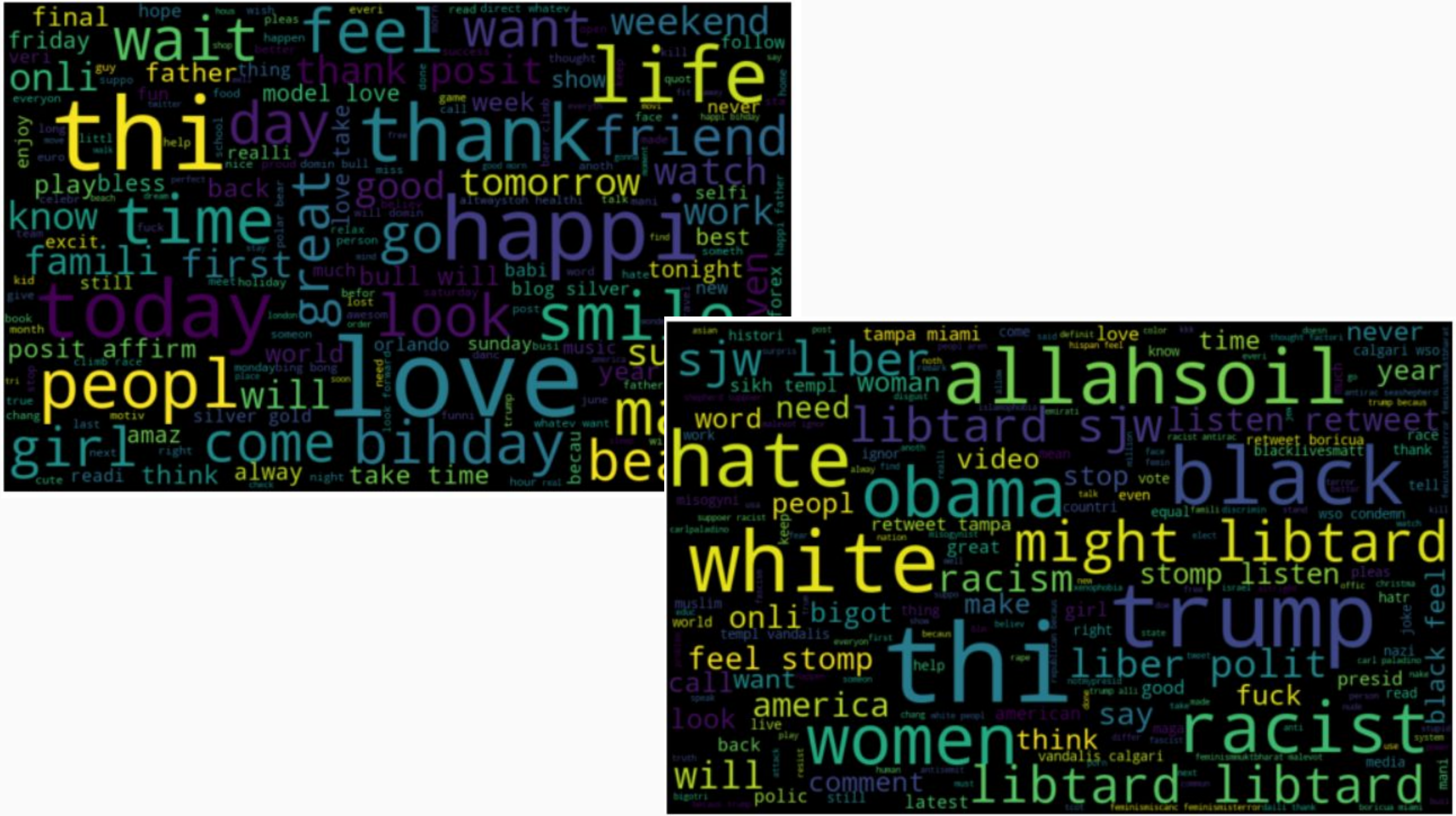
   (as the his etc)

3. Tokenize remaining words for standard language across all tweets.

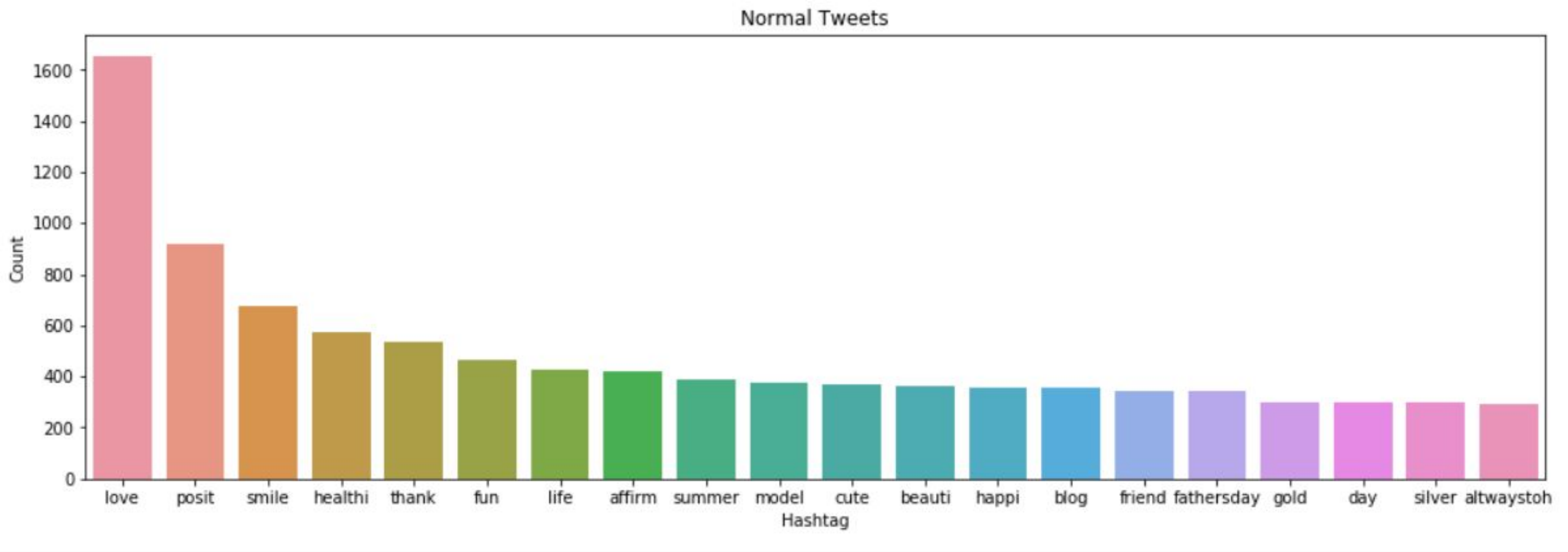
   (loving -> love)

[2/2] huge fan fare and big talking before they leave. chaos and pay disputes when they get there. #allshowandnogo

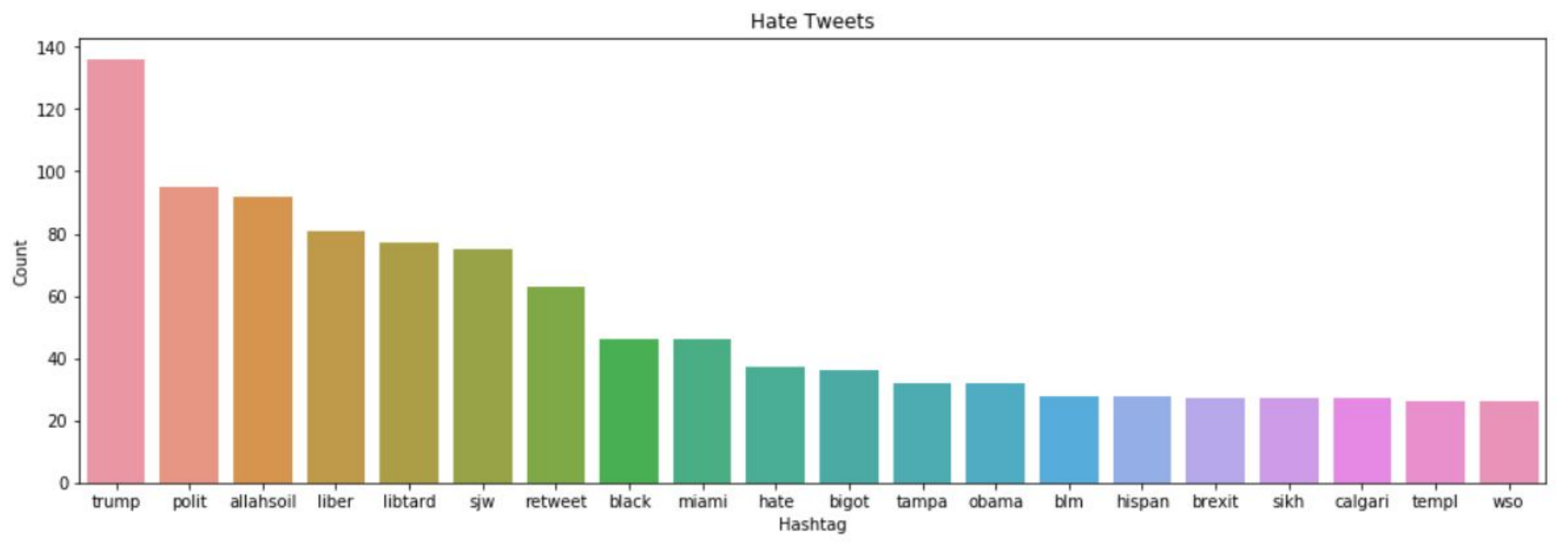
huge fare talk befor they leav chao disput when they there #allshowandnogo


final
hope
friday
wait
feel
want
weekend
follow
onli
father
thing
thank
posit
show
life
thi
day
thank
friend
model love
week
never
realli
watch
playbless
back
good
tomorrow
selfi
know
time
great
go
happi
work
excit
best
famili
first
bull will
tonight
still
blog silver
today
look
smile
posit affirm
orlando
sunday
world
peopl
will
love
girl
amaz
come
bihday
alway
think
take time
readi
historí
tampa miami
love
time
never
sjw liber
allahsoil
sikh templ
woman
year
need
word
libtard sjw
listen
retweet
hate
video
black
obama
stop
retweet boricua
blacklivesmatt
peopl
retweet tampa
white
might libtard
racism
stomp listen
onli
bigot
make
trump
thi
feel stomp
liber polit
want
presid
fuck
america
say
women
think
racist
will
comment
libtard libtard
polic
latest

# Most popular hashtags in normal tweets



# Most popular hashtags in  tweets



# Feature Selection:

Bag of Words : select 1000 of the most common words for classification

TFIDF : Like Bag of Words, but with trained weights on key words

Word to Vector : Select 200 words, whose combinations create context
Trained using a neural network, and vectorized

Document to Vector : Like Words to Vector, but with the entire tweet.

# Stacked Machine Learning:

# The Twice-Cooked Technique

# Stacked Machine Learning:

## The Level 1

Start with the best ingredients:

| Classifier | F1 Score |
|---|---|
| LogReg (TFIDF) | .544 |
| Random Forest (TFIDF) | .562 |
| SVM (Word to Vector) | .571 |
| LogReg (Word to Vector) | .533 |
| Light GBM (Word to Vector) | .622 |

# Best Single Model Score

Light GBM (Tuned):

64.7% accuracy

Analytics Vidhya
Learn everything about analytics

# Stacked Machine Learning:

## Level 2

Feed predictions from Level 1 to Level 2:

| Classifier | Competition Score |
|---|---|
| Log Reg | 70.7% accuracy |
| Light GBM | **73.5% accuracy** |

As of 03/31/19, this result places in the top 25% of 650 competitors.

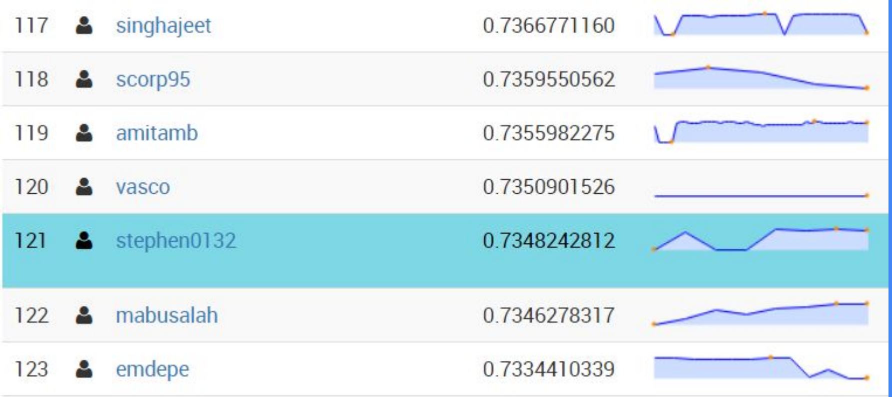
# Next Steps

Continued stacking

Feature Engineering

Data Set Balancing

## Score to beat:  85.8%



| | | | |
|---|---|---|---|
| 117 | 👤 | singhajeet | 0.7366771160 |
| 118 | 👤 | scorp95 | 0.7359550562 |
| 119 | 👤 | amitamb | 0.7355982275 |
| 120 | 👤 | vasco | 0.7350901526 |
| 121 | 👤 | stephen0132 | 0.7348242812 |
| 122 | 👤 | mabusalah | 0.7346278317 |
| 123 | 👤 | emdepe | 0.7334410339 |