Are You A Hater?

Hate Speech on the Internet

Leon Zhou

"And the haters gonna hate, hate, hate, hate, hate…"
- Taylor Swift

Leon Zhou
lzhou95@gmail.com
5/16/18


Hate Speech *(n.)*:
speech that demeans on the basis of race, ethnicity, gender, religion, age, disability, or any other similar ground

The Problem

Hate Speech

```
['the', 'blacks', 'in', 'california', 'are', 'typical', 'n[redacted]']
```

Other Speech (offensive, non-offensive)

```
['using', 'y', 'mx', 'b', 'to', 'measure', 'the', 'slope', 'of', 'that', 'ass', 'girl']
```

```
['yal', 'hoes', 'dont', 'need', 'a', 'valentine', 'yal', 'need', 'jesus']
```

```
['told', 'that', 'hoe', 'she', 'special', 'like', 'the', 'mcrib', 'at', 'mcdonalds']
```

Workflow

Tweets → Normalization → Part of Speech Tagging
Naive Bayes Classifier
85.2% Accuracy

→ Transformation → Comment Classification
Multiple Classifiers
“It’s Complicated”% Accuracy
→ **Hate speech**
Offensive speech
Neither

RT @ComedyPics: If Kanye took Kim to McDonald’s then ya hoes don’t deserve $200 Dates http://t.co/2QK4I1fRT

```
RT @ComedyPics: If Kanye took Kim to McDonald&#8217;s then ya hoes
don&#8217;t deserve $200 Dates http://t.co/2QK4I1fRT
```

Retweets and Mentions

Emoji

Hyperlinks

Hashtags

Unicode Characters

```
RT @ComedyPics: If Kanye took Kim to McDonald&#8217;s then ya hoes don&#8217;t deserve $200 Dates http://t.co/2QK4I1fRT
```

Retweets and Mentions | **Emoji** | **Hyperlinks** | **Hashtags** | **Unicode Characters**

↓

```
if kanye took kim to mcdonalds then ya hoes dont deserve 200 dates
```

```
RT @ComedyPics: If Kanye took Kim to McDonald&#8217;s then ya hoes
don&#8217;t deserve $200 Dates http://t.co/2QK4I1fRT
```

Retweets and Mentions | **Emoji** | **Hyperlinks** | **Hashtags** | **Unicode Characters**

```
if kanye took kim to mcdonalds then ya hoes dont deserve 200 dates
```

[CS, PPSS, VBD, PPO, TO, NNS, WRB, NP, DOZ, NN, VB, CD, NNS]

(85.2% accuracy)

(theoretically)

`RT @ComedyPics: If Kanye took Kim to McDonald’s then ya hoes don’t deserve $200 Dates http://t.co/2QK4I1fRT`

Retweets and Mentions

Emoji

Hyperlinks

Hashtags

Unicode Characters

`if kanye took kim to mcdonalds then ya hoes dont deserve 200 dates`

[CS, PPSS, VBD, PPO, TO, NNS, WRB, NP, DOZ, NN, VB, CD, NNS]

(85.2% accuracy)

Unigrams

to	TO
mcdonalds	NNS
then	WRB
ya	NP
hoes	DOZ
dont	NN

Bigrams

to mcdonalds	TO NNS
mcdonalds then	NNS WRB
then ya	WRB NP
ya hoes	NP DOZ
hoes dont	DOZ NN

Trigrams

to mcdonalds then	TO NNS WRB
mcdonalds then ya	NNS WRB NP
then ya hoes	WRB NP DOZ
ya hoes dont	NP DOZ NN

```
RT @ComedyPics: If Kanye took Kim to McDonald&#8217;s then ya hoes don&#8217;t deserve $200 Dates http://t.co/2QK4I1fRT
```

Retweets and Mentions | **Emoji** | **Hyperlinks** | **Hashtags** | **Unicode Characters**

```
if kanye took kim to mcdonalds then ya hoes dont deserve 200 dates
```

[CS, PPSS, VBD, PPO, TO, NNS, WRB, NP, DOZ, NN, VB, CD, NNS]

(85.2% accuracy)

Unigrams

to	TO
mcdonalds	NNS
then	WRB
ya	NP
hoes	DOZ
dont	NN

Bigrams

to mcdonalds	TO NNS
mcdonalds then	NNS WRB
then ya	WRB NP
ya hoes	NP DOZ
hoes dont	DOZ NN

Trigrams

to mcdonalds then	TO NNS WRB
mcdonalds then ya	NNS WRB NP
then ya hoes	WRB NP DOZ
ya hoes dont	NP DOZ NN

[…, 0, 0, 0.001, 0, 1, 0, 0.561, 0, 0, 0.003, 0, 0, 0, 0, 0, 0, 0, 0.421, …]

The Magic of Teamwork

Class 0 (Hate Speech) F1 Score, 3 Classes - Stratified

	Param					
Model	Unigram	Uni-POS	Bigram	Bi-POS	Trigram	Tri-POS
Decision Tree	0.2650	0.0916	0.2519	0.0758	0.0918	0.2298
Gradient Boosted Trees	0.0338	0.2593	0.2473	0.0343	0.2393	0.0259
Logistic Regression	0.0000	0.2011	0.0000	0.2505	0.1953	0.0000
Naive Bayes	0.1184	0.1118	0.1122	0.1186	0.1127	0.1183
Stochastic Gradient Desc.	0.0707	0.0000	0.0664	0.0000	0.1924	0.0000

Avg. COF

0.0000 0.2650

The Magic of Teamwork

Class 0 (Hate Speech) F1 Score, 3 Classes - Stratified

Model	Unigram	Uni-POS	Bigram	Bi-POS	Trigram	Tri-POS
Decision Tree	0.2650	0.0916	0.2519	0.0758	0.0918	0.2298
Gradient Boosted Trees	0.0338	0.2593	0.2473	0.0343	0.2393	0.0259
Logistic Regression	0.0000	0.2011	0.0000	0.2505	0.1953	0.0000
Naive Bayes	0.1184	0.1118	0.1122	0.1186	0.1127	0.1183
Stochastic Gradient Desc.	0.0707	0.0000	0.0664	0.0000	0.1924	0.0000

Param

Avg. COF

0.0000 0.2650

Took majority opinion of five best models:

e.g. [0, 0, 0, 1, 2] → 0

0.21 < 0.25

The Magic of Teamwork

Class 0 (Hate Speech) F1 Score, 3 Classes - Stratified

	Param					
Model	Unigram	Uni-POS	Bigram	Bi-POS	Trigram	Tri-POS
Decision Tree	0.2650	0.0916	0.2519	0.0758	0.0918	0.2298
Gradient Boosted Trees	0.0338	0.2593	0.2473	0.0343	0.2393	0.0259
Logistic Regression	0.0000	0.2011	0.0000	0.2505	0.1953	0.0000
Naive Bayes	0.1184	0.1118	0.1122	0.1186	0.1127	0.1183
Stochastic Gradient Desc.	0.0707	0.0000	0.0664	0.0000	0.1924	0.0000

Avg. COF

0.0000 0.2650

Fit second layer classifier to make decision:

[0, 0, 0, 1, 2]

↓

e.g. **Naive Bayes Classifier**

0.31

> 0.25

Thank You

Appendix

Not Perfect

Some words critical to identification lost in processing

```
DEBUG:root:Incoming words:
DEBUG:root:['i', 'dont', 'trust', 'these', 'b[redacted]']
DEBUG:root:Outgoing words:
DEBUG:root:['i', 'dont', 'trust', 'these', 'birches']
```

Solution: manually added high-impact slang, profanity

```
DEBUG:root:Incoming words:
DEBUG:root:['yaya', 'ho', 'cut', 'avi', 'tho', 'rt', 'i', 'had', 'no', 'idea', 'she', 'was', 'sleep']
DEBUG:root:Outgoing words:
DEBUG:root:['may', 'o', 'cut', 'vi', 'the', 'i', 'i', 'had', 'no', 'idea', 'she', 'was', 'sleep']
```

Other words are just “acceptable losses”

Part of Speech Classification

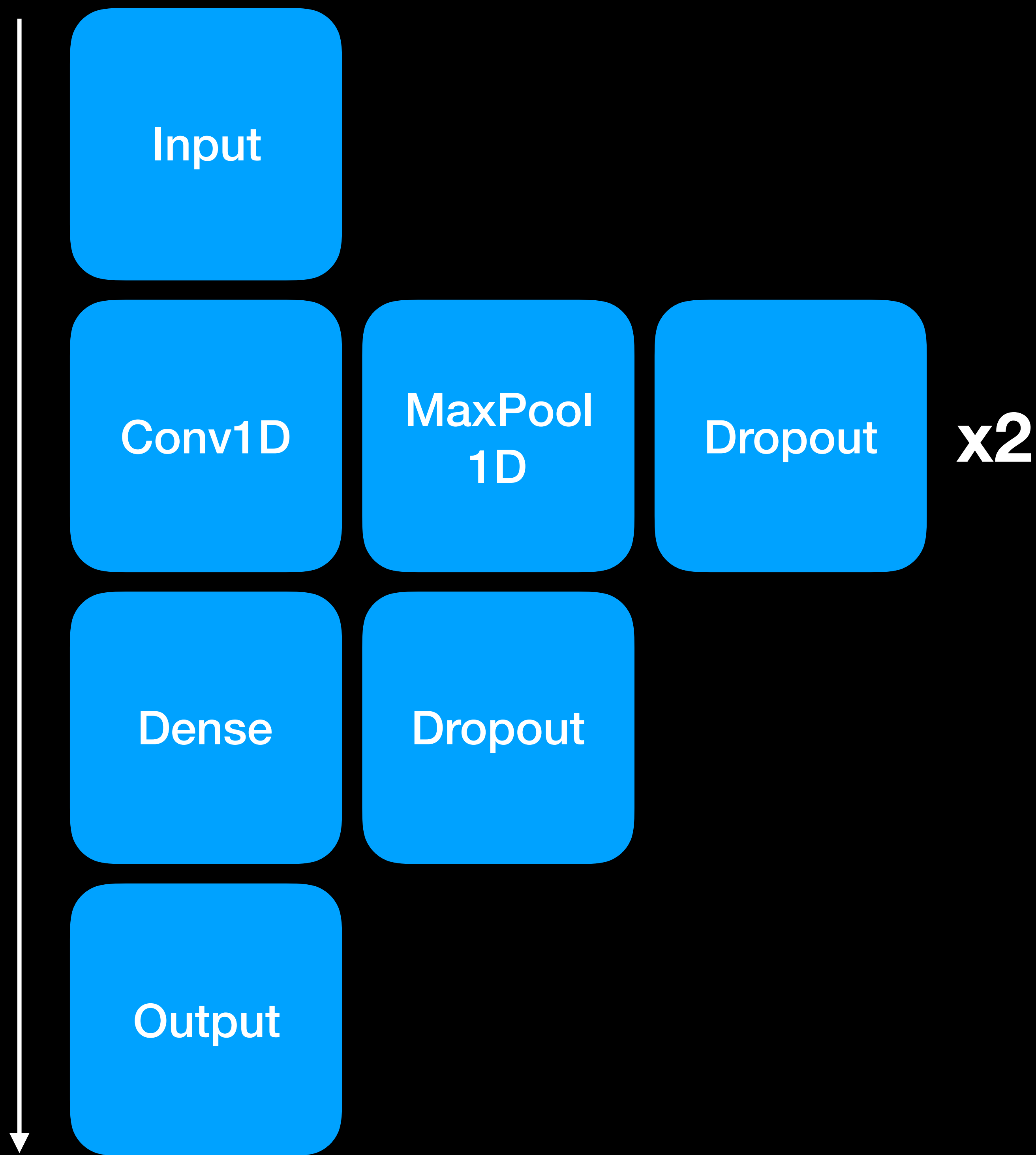
- Naive Bayes Classifier (**nltk.NaiveBayesClassifier**)
- Trained on Brown corpus (**nltk.corpus.Brown**)
- 85.2% Accuracy and F1-Score

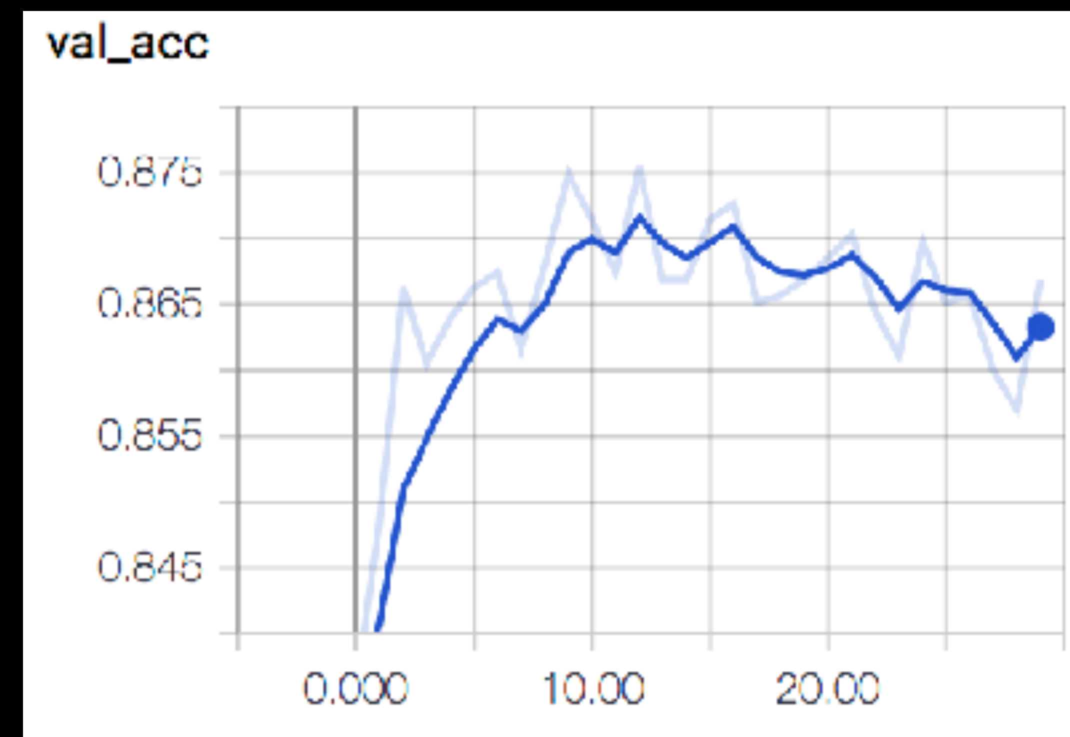
Contextual Features

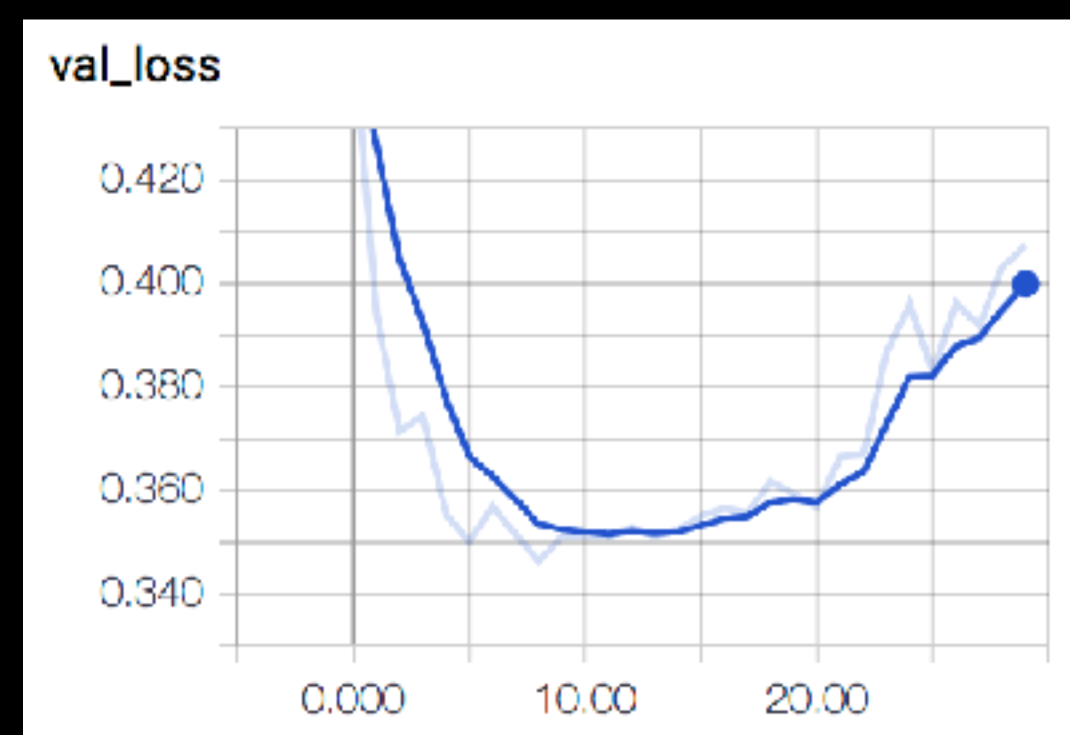
Characteristic Features

```python
def word_feature_extraction(sentence, index):
    word = sentence[index]
    previous_word = "<START>"
    next_word = "<END>"
    if index > 0:
        previous_word = sentence[index - 1]
    if index < len(sentence) - 1:
        next_word = sentence[index + 1]
    suffix1 = word[-1]
    suffix2 = word[-2:]
    suffix3 = word[-3:]
    numeric = True if word.isnumeric() else False
```

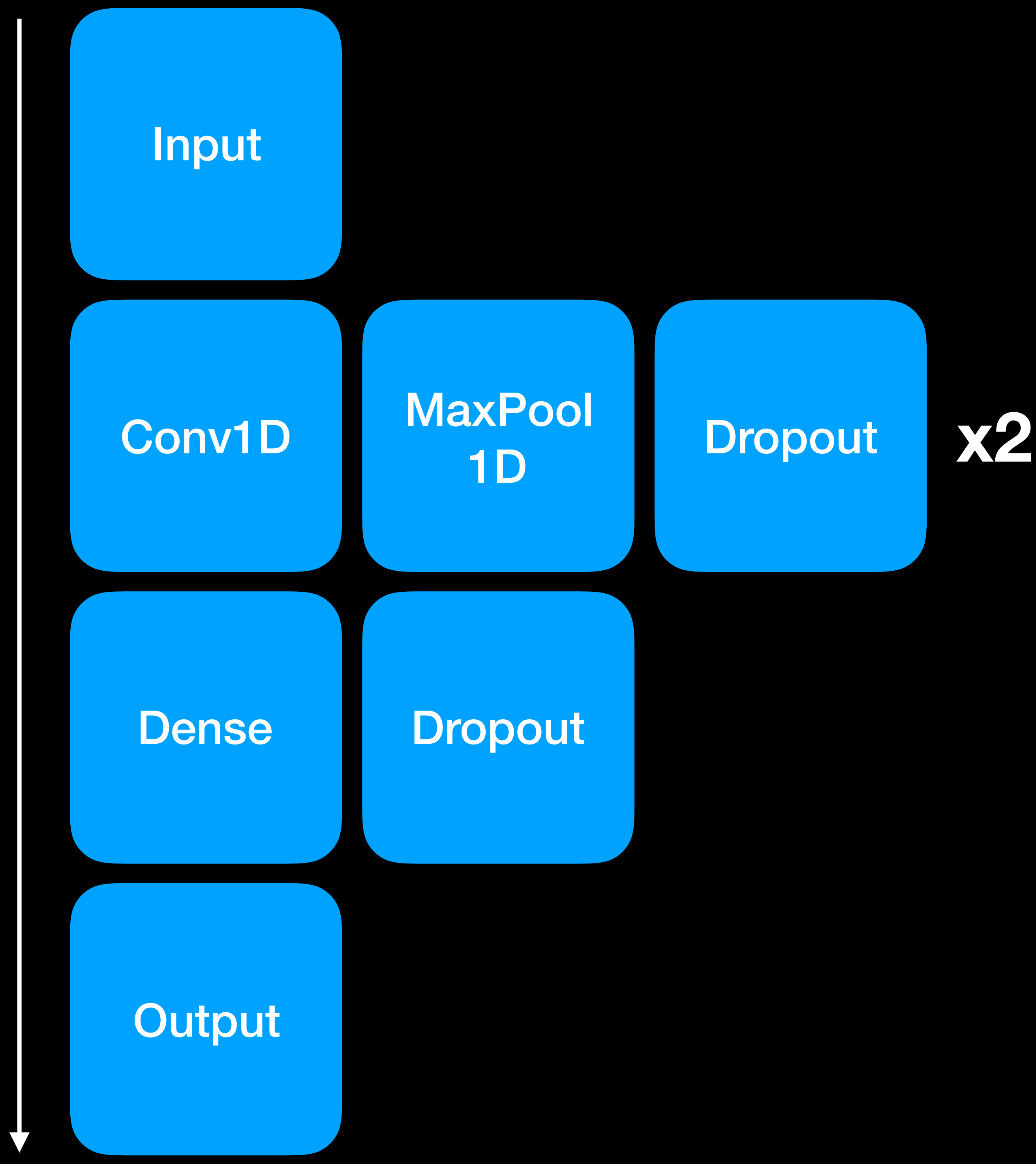
Diving Deep


Input
Conv1D
MaxPool 1D
Dropout
x2
Dense
Dropout
Output


val_acc
0.875
0.865
0.855
0.845
0.000
10.00
20.00


val_loss
0.420
0.400
0.380
0.360
0.340
0.000
10.00
20.00

Diving Deep



Epoch 11
Avg. F1-Score: 0.84
Class 0 F1-Score: 0.00

Epoch 12
Avg. F1-Score: 0.85
Class 0 F1-Score: 0.16

Convolutional Neural Network

```
batch_size = 50
num_epochs = 30
kernel_size1 = 50
kernel_size2 = 20
pool_size1 = 10
pool_size2 = 10
conv_depth1 = 32
conv_depth2 = 64
drop_prob1 = 0.25
drop_prob2 = 0.4
hidden_size = 512
```

```
inp = Input(shape=(length, 1))
conv1 = Conv1D(filters=conv_depth1, kernel_size=kernel_size1,
               strides=kernel_size1, padding='same', activation='relu')(inp)
pool1 = MaxPool1D(pool_size=pool_size1, strides=pool_size1, padding='same')(conv1)
drop1 = Dropout(rate=drop_prob1)(pool1)
conv2 = Conv1D(filters=conv_depth2, kernel_size=kernel_size2,
               strides=kernel_size2, padding='same', activation='relu')(drop1)
pool2 = MaxPool1D(pool_size=pool_size2, strides=pool_size2, padding='same')(conv2)
drop2 = Dropout(rate=drop_prob1)(pool2)

flat = Flatten()(drop2)
hidden = Dense(units=hidden_size, activation='relu')(flat)
drop3 = Dropout(rate=drop_prob2)(hidden)
out = Dense(units=3, activation='softmax')(drop3)
```

```
model = Model(inputs=inp, outputs=out)
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])

model.fit(Xtr, Ytr, batch_size=batch_size, epochs=num_epochs, verbose=1, validation_split=0.1,
          callbacks=[TensorBoard(log_dir=os.path.expanduser('~/TensorBoard/')), metrics])
```

```
Test Loss after Epoch 12 : 0.336706623265
Test Accuracy after Epoch 12:  0.877605917988
```

Results

Each model only trained on **one** type of feature

Run with stratified sampling

Each square corresponds to a distinct trained model

Average F1 Score, 3 Classes - Stratified Sampling

	Param					
Model	Unigram	Uni-POS	Bigram	Bi-POS	Trigram	Tri-POS
Decision Tree	0.8626	0.6746	0.8605	0.6707	0.6663	0.8546
Gradient Boosted Trees	0.7109	0.8878	0.8761	0.7092	0.8836	0.7081
Logistic Regression	0.6981	0.8660	0.7002	0.8763	0.8558	0.6985
Naive Bayes	0.3677	0.1172	0.1210	0.3725	0.1248	0.3716
Stochastic Gradient Desc.	0.7791	0.6845	0.7876	0.6886	0.8072	0.6850

Avg. Avgf

0.1172 0.8878

The Catch

Class Imbalance

	Precision	Recall	F1 Score	Support
Hate Speech	0.07	0.44	0.12	427
Offensive	0.78	0.27	0.40	5747
Neither	0.27	0.56	0.36	1261
Avg. / Total	0.65	0.33	0.37	7435

Naive Bayes - Unigram

Each square corresponds to a distinct trained model

Average F1 Score, 3 Classes - Stratified Sampling

	Param					
Model	Unigram	Uni-POS	Bigram	Bi-POS	Trigram	Tri-POS
Decision Tree	0.8626	0.6746	0.8605	0.6707	0.6663	0.8546
Gradient Boosted Trees	0.7109	0.8878	0.8761	0.7092	0.8836	0.7081
Logistic Regression	0.6981	0.8660	0.7002	0.8763	0.8558	0.6985
Naive Bayes	0.3677	0.1172	0.1210	0.3725	0.1248	0.3716
Stochastic Gradient Desc.	0.7791	0.6845	0.7876	0.6886	0.8072	0.6850

Avg. Avgf

0.1172 0.8878

The Catch

Class Imbalance

	Precision	Recall	F1 Score	Support
Hate Speech	0.07	0.44	0.12	427
Offensive	0.78	0.27	0.40	5747
Neither	0.27	0.56	0.36	1261
Avg. / Total	0.65	0.33	0.37	7435

Naive Bayes - Unigram

Each square corresponds to a distinct trained model

Class 0 (Hate Speech) F1 Score, 3 Classes - Stratified

	Param					
Model	Unigram	Uni-POS	Bigram	Bi-POS	Trigram	Tri-POS
Decision Tree	0.2650	0.0916	0.2519	0.0758	0.0918	0.2298
Gradient Boosted Trees	0.0338	0.2593	0.2473	0.0343	0.2393	0.0259
Logistic Regression	0.0000	0.2011	0.0000	0.2505	0.1953	0.0000
Naive Bayes	0.1184	0.1118	0.1122	0.1186	0.1127	0.1183
Stochastic Gradient Desc.	0.0707	0.0000	0.0664	0.0000	0.1924	0.0000

Avg. COF

0.0000 0.2650

The Magic of Teamwork

If one model can't do the job, why not five?

Model	Unigram	Uni-POS	Bigram	Bi-POS	Trigram	Tri-POS
Decision Tree	0.2650	0.0916	0.2519	0.0758	0.0918	0.2298
Gradient Boosted Trees	0.0338	0.2593	0.2473	0.0343	0.2393	0.0259
Logistic Regression	0.0000	0.2011	0.0000	0.2505	0.1953	0.0000
Naive Bayes	0.1184	0.1118	0.1122	0.1186	0.1127	0.1183
Stochastic Gradient Desc.	0.0707	0.0000	0.0664	0.0000	0.1924	0.0000

Took the majority opinion of model outputs as decision

[0, 0, 0, 1, 2]

↓

0

	Precision	Recall	F1 Score	Support
Hate Speech	0.42	0.14	**0.21**	429
Offensive	0.89	0.97	0.93	5747
Neither	0.85	0.71	0.77	1249
Avg. / Total	0.86	0.88	0.86	7435

The Magic of Teamwork

If one model can’t do the job, why not five?

Model	Unigram	Uni-POS	Bigram	Bi-POS	Trigram	Tri-POS
Decision Tree	0.2650	0.0916	0.2519	0.0758	0.0918	0.2298
Gradient Boosted Trees	0.0338	0.2593	0.2473	0.0343	0.2393	0.0259
Logistic Regression	0.0000	0.2011	0.0000	0.2505	0.1953	0.0000
Naive Bayes	0.1184	0.1118	0.1122	0.1186	0.1127	0.1183
Stochastic Gradient Desc.	0.0707	0.0000	0.0664	0.0000	0.1924	0.0000

Fit a second layer classifier to make decision

[0, 0, 0, 1, 2]

↓

Naive Bayes Classifier

↓

2

	Precision	**Recall**	**F1 Score**	**Support**
Hate Speech	0.32	0.31	**0.31**	429
Offensive	0.93	0.92	0.92	5747
Neither	0.81	0.85	0.83	1249
Avg. / Total	0.86	0.88	0.86	7435

Binary Classification

Not interesting - just target the curse words

Offensive category already largest - lumping it into hateful or other category skews

Hateful + Offensive Classification F1 Score (Binary) - Stratified

Model	Unigram	Uni-POS	Bigram	Bi-POS	Trigram	Tri-POS
Decision Tree	0.9594	0.8555	0.9609	0.8545	0.8520	0.9575
Gradient Boosted Trees	0.9051	0.9689	0.9632	0.9056	0.9646	0.9061
Logistic Regression	0.9087	0.9630	0.9085	0.9664	0.9574	0.9086
Naive Bayes	0.4713	0.1666	0.1822	0.4802	0.1754	0.4751
Stochastic Gradient Desc.	0.9328	0.9076	0.9274	0.9089	0.9330	0.9091

COF

0.1666 0.9689

PCA

Class 0 (Hate Speech) F1 Score, 3 Classes - Stratified, PCA to 50 Elements

Model	Unigram	Uni-POS	Bigram	Bi-POS	Trigram	Tri-POS
Decision Tree	0.1099	0.0799	0.1584	0.0995	0.0798	0.1455
Gradient Boosted Trees	0.0308	0.1706	0.0521	0.0397	0.0693	0.0393
Logistic Regression	0.0000	0.0565	0.0000	0.1487	0.0687	0.0000
Naive Bayes	0.2681	0.0815	0.0890	0.1741	0.0769	0.1519
Stochastic Gradient Desc.	0.0314	0.0000	0.0702	0.0000	0.1389	0.0000

Param

COF 0.0000 – 0.2681

Class 0 (Hate Speech) F1 Score, 3 Classes - Stratified, PCA to 250 Elements

Model	Unigram	Uni-POS	Bigram	Bi-POS	Trigram	Tri-POS
Decision Tree	0.1296	0.0707	0.1622	0.0852	0.0824	0.0876
Gradient Boosted Trees	0.0394	0.1434	0.0568	0.0352	0.0731	0.0308
Logistic Regression	0.0000	0.0596	0.0000	0.1532	0.0696	0.0000
Naive Bayes	0.2653	0.0815	0.0890	0.1707	0.0769	0.1610
Stochastic Gradient Desc.	0.1923	0.0136	0.0047	0.0000	0.1412	0.0181

Param

COF 0.0000 – 0.2653

Hate Speech ***(n.)*****:**

speech that demeans on the basis of race, ethnicity, gender, religion, age, disability, or any other similar ground

Takeaways and Improvements

- It's a **HARD** problem
  - Intricacies and subtleties in language use
- Do the EDA
  - My definition of offensive is not same as readers'
    - Very conservative, any profanity
- Design with class imbalance in mind from the start
- Relationship between tweets; many near duplicates due to twitter's reply system
- Improvements to POS may reduce errors
- Models that take in multiple groups of features

This will be summarized and made much less wordy