Abusive Chat Detection

Samuel P. Sears

Background

Domain: Chat Customer Service

No channel is completely free from Offensive, Hateful, Abusive customers

Abusive customers…

- Are difficult to de-escalate
- Decrease morale in the business area
- Are unacceptable

Business Problem

Goal

Prevent employee exposure to abusive customer messages

Need

Ability to detect abusive chats prior to being sent to customer service representatives

Challenges

Abusive language goes beyond screening for profanity

The Data

Modeling Data

Twitter Data from Kaggle

Hate Tweets
- 12,970 tweets
- 42% labeled as hate

Offensive Tweets
- 14,100 tweets
- 33% labeled as offensive

Profanity Data

bannedwordlist.com
- List of 78 profane words
- Would be provided by the business in real use case

Testing Data

Bitext Dataset
- Customer Support
- Over 20,000 messages

Preprocessing

Example: This is 1 example of a Tweet.

Remove Numbers → This is example of a Tweet.

Remove Punctuation → This is example of a Tweet

Convert to Lowercase → this is example of a tweet

TFIDF Vectorization

example	is	of	this	tweet
0.447	0.447	0.447	0.447	0.447

Notes:

- *Vectorization removes stop words*
- *This would be the result if only vectorizing one sentence*

Modeling

Data split 75% training | 25% testing

15 Models Tested for both Hate messages and Offensive message detection

CatBoost Classifier had best performance for both datasets on the withheld testing data

- Hate classifier AUC: 0.705
- Offensive classifier AUC: 0.665

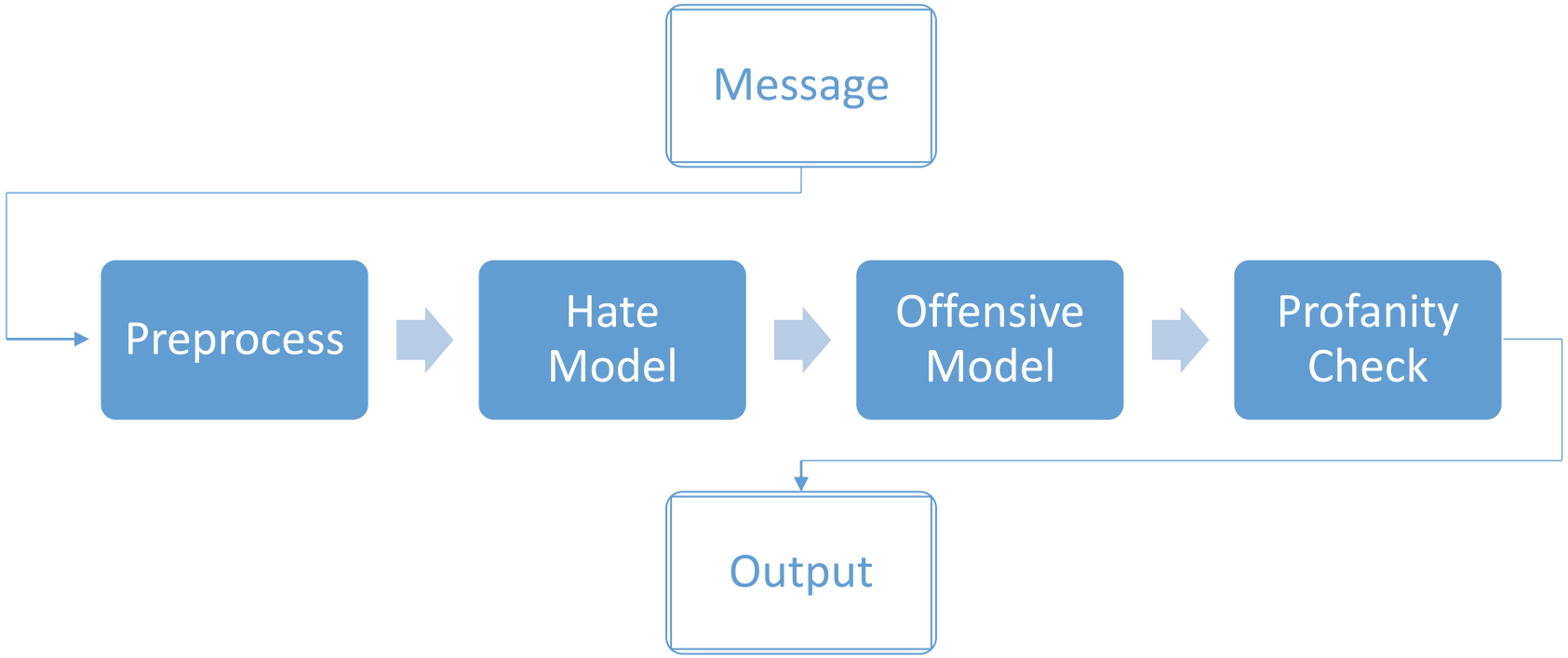
	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
et	Extra Trees Classifier	0.7213	0.7851	0.6388	0.6796	0.6584	0.4235	0.4243	9.8930
catboost	CatBoost Classifier	0.7185	0.7921	0.5620	0.7088	0.6265	0.4057	0.4129	39.3360
xgboost	Extreme Gradient Boosting	0.7162	0.7833	0.5868	0.6921	0.6347	0.4053	0.4093	18.5310
lightgbm	Light Gradient Boosting Machine	0.7143	0.7844	0.6116	0.6780	0.6428	0.4059	0.4075	0.7470
rf	Random Forest Classifier	0.7128	0.7878	0.6130	0.6752	0.6423	0.4034	0.4050	5.5850
lr	Logistic Regression	0.7091	0.7862	0.5592	0.6908	0.6176	0.3872	0.3931	1.0510
svm	SVM - Linear Kernel	0.7024	0.0000	0.5864	0.6660	0.6222	0.3788	0.3818	0.6170
ridge	Ridge Classifier	0.6981	0.0000	0.5980	0.6551	0.6249	0.3732	0.3746	0.4400
gbc	Gradient Boosting Classifier	0.6940	0.7634	0.4820	0.6974	0.5696	0.3445	0.3588	8.7450
ada	Ada Boost Classifier	0.6905	0.7416	0.5054	0.6774	0.5784	0.3421	0.3515	2.4160
lda	Linear								
dt	D								
qda	Quadratic								
knn									
nb									

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
et	Extra Trees Classifier	0.7624	0.7645	0.4943	0.6981	0.5782	0.4197	0.4320	16.5900
svm	SVM - Linear Kernel	0.7609	0.0000	0.4579	0.7170	0.5574	0.4049	0.4246	0.8490
ada	Ada Boost Classifier	0.7584	0.7092	0.3736	0.7807	0.5049	0.3707	0.4149	2.3690
rf	Random Forest Classifier	0.7583	0.7628	0.4064	0.7468	0.5260	0.3824	0.4144	10.9070
catboost	CatBoost Classifier	0.7579	0.7694	0.3598	0.7959	0.4950	0.3646	0.4144	35.4010
lr	Logistic Regression	0.7556	0.7690	0.3626	0.7797	0.4947	0.3610	0.4070	1.5690
xgboost	Extreme Gradient Boosting	0.7540	0.7516	0.3859	0.7477	0.5084	0.3662	0.4021	18.9290
ridge	Ridge Classifier	0.7534	0.0000	0.4665	0.6876	0.5554	0.3934	0.4079	0.5290
gbc	Gradient Boosting Classifier	0.7426	0.7366	0.2792	0.8279	0.4171	0.3009	0.3743	8.8440
lightgbm	Light Gradient Boosting Machine	0.7364	0.7201	0.3749	0.6844	0.4840	0.3270	0.3538	0.6850
lda	Linear Discriminant Analysis	0.7222	0.7240	0.5257	0.5900	0.5555	0.3546	0.3562	7.8580
dt	Decision Tree Classifier	0.6908	0.6341	0.4881	0.5359	0.5107	0.2854	0.2862	5.7990
knn	K Neighbors Classifier	0.6845	0.5347	0.1059	0.6608	0.1818	0.0966	0.1624	14.9170
qda	Quadratic Discriminant Analysis	0.6759	0.5145	0.0384	0.6572	0.0722	0.0379	0.0971	10.0030
nb	Naive Bayes	0.4873	0.5608	0.7764	0.3691	0.5003	0.0947	0.1235	0.3100

See full print out in Github

Full Pipeline

Ex. How do I change payment methods?



Hate Score: 24%
Offensive Score: 23%
Profane Words: []

Test on Customer Support Messages

Risks

- Models were not trained or tested on customer service domain
- Test data was not unbalanced like real word solution
- Pipeline may incorrectly classify non-abusive customers as abusive

Solution

Test pipeline on random labeled sample from BiText Free Customer Support Dataset

- Sample size – 299
- Hate messages – 0 (0%)
- Offensive messages – 13 (4%)
- Messages with profanity – 13 (4%)

Results

- 100% accuracy in classifying hate messages
- 100% accuracy in classifying offensive messages
- 100% accuracy in detecting profanity

Further Analysis

- Train / Test models on real company messages
- Tune model thresholds according to company’s level of comfort
- Provide company defined profanity list
- Analyze current chat architecture to see where this model could be integrate
- Design response and handling for when a chat is classified as abusive

Project Repository

https://github.com/ssears219/Abusive-Chat-Detection