

A Supplementary Materials

A.1 Training Language Model

We first describe the language model used to generate the adversarial dataset. The language model is trained by the standard perplexity objective function, i.e., for a tweet $T = [e_1, e_2 \dots e_l]$, we want the model to predict the next token e_i based on the previous tokens $[e_1, e_2 \dots e_{i-1}]$. We train a language model for the forward and backward direction, respectively.

As for the model architecture, we embed each token to create a real vector representation $[w_1, w_2 \dots w_l]$; then we stack three layers of single direction LSTM with hidden dimension 128 on top of the embedded token representation. Then the hidden representation from the last LSTM layer is fed the final classification layer with a softmax activation, the output of which is the probability of each token in the entire vocabulary.

We randomly sampled 80% of the tweets from the unlabeled corpus (~ 1 million tweets) used in Chang et al. (2018) as our training data, and use the rest as validation data. We apply early stopping on our training by calculating the validation loss on the validation set.

A.2 Results for Other Unigrams

We ran a simple ℓ_1 logistics linear regression with unigram features on the aggression label to obtain a list of unigrams that are highly correlated with the *aggression* label. This results in ~ 300 unigrams with positive weights, and we manually selected a subset of them based on two criteria: 1) possible to insert them into a significant proportion of tweets 2) the experts do not use them to determine the label. Here we list the unigrams and their corresponding weight, and use bold front for selected unigrams. We report the adversarial flip results for other unigrams we have selected in table 12.





Unigram	Corresponding Weight
#lldv	5.755
jj	5.283
disrespect	5.218
nobody's	5.060
irritated	4.968
keta	4.864
delete	4.791
boot	4.700
snitchk	4.698
saturday	4.615
glock	4.560
opps	4.535
buses	4.427
snitching	4.418
sticks	4.396
cpdk	4.394
	4.118
,bitch	4.110
street	4.057
situation	3.881
guns	3.675
snitchkkkkkkkk	3.662
ls	3.659
rat	3.581
nuskigang	3.578
haters	3.517
kill	3.470
600	3.362
	3.362
#playcrazygang	3.324
thinkin	3.313
	3.290
police	3.284
mtg	3.065
piss	3.032
dumbass	3.023
gatta	2.972
pimp	2.942
gunn	2.843
	2.773
mitch	2.772
bust	2.748
poled	2.690
dicks	2.662
dot	2.660
cype	2.589
rayband	2.463

Table 6: Unigrams and their corresponding weights sorted in decreasing weight order

Unigram	Corresponding Weight
.m	2.448
boy	2.364
blowing	2.340
extorted	2.292
gutta	2.260
wacked	2.244
guys	2.238
wen	2.230
lame	2.164
blackin	2.145
mfks	2.138
👊	2.134
killas	2.080
😡	2.069
dead	2.043
🚫	2.029
fuck	2.022
30	1.963
n***as	1.952
👊	1.919
bih	1.910
land	1.900
thoe	1.875
dthang	1.852
800	1.851
bout	1.810
strapped	1.807
repping	1.786
turn	1.767
shoutout	1.693
lucky	1.680
basically	1.674
ill	1.669
mfs	1.667
😏	1.632
shut	1.631
crease	1.615
page	1.603
shirt	1.583
punching	1.561
boa	1.549
niqqas	1.529
fight	1.517
👊	1.466
n***az	1.463
riot	1.453
ebt	1.444

Table 7: Unigrams and their corresponding weights sorted in decreasing weight order (continued)

Unigram	Corresponding Weight
fee	1.438
blk	1.437
pole	1.405
o	1.377
👊	1.374
...lol	1.369
hoes	1.362
doer	1.348
pussy	1.347
wanna	1.337
gotta	1.331
another	1.315
let	1.303
mean	1.297
watchem	1.274
odee	1.202
ready	1.196
ass	1.191
into	1.175
@	1.113
put	1.106
learn	1.105
line	1.100
yea	1.087
👊	1.074
kids	1.040
📌	1.037
tell	1.001
frontstreetsavagesquad	0.981
got	0.970
fucked	0.962
run	0.959
screaming	0.944
changed	0.940
delusional	0.940
#tyquanworld	0.910
any	0.909
talking	0.878
wassup	0.866
✳️	0.854
on	0.846
😏	0.845
thats	0.844
name	0.826
life	0.811
hate	0.810
lazy	0.803

Table 8: Unigrams and their corresponding weights sorted in decreasing weight order (continued)

Unigram	Corresponding Weight
much	0.799
mino	0.791
!url	0.788
!	0.785
knew	0.747
tire	0.698
bro	0.691
dnt	0.688
🤔	0.723
ae_	0.687
🔴	0.678
\$	0.658
wid	0.657
pictures	0.654
game	0.651
ever	0.642
st	0.635
n***a	0.625
world	0.616
went	0.616
who	0.614
tw	0.613
😭	0.601
anotha	0.578
a	0.571
👏	0.565
frm	0.563
die	0.562
mufuka	0.535
irrelevant	0.525
👊	0.523
yall	0.512
bet	0.511
out	0.508
snoop	0.491
betta	0.491
girl	0.480
@user	0.478
that	0.460
🤔	0.458
ain't	0.447
watch	0.434
torrance	0.424
beef	0.415
mouth	0.407
mom	0.406
👑	0.400

Table 9: Unigrams and their corresponding weights sorted in decreasing weight order (continued)

Unigram	Corresponding Weight
o'hare	0.394
he	0.391
's	0.388
yo	0.387
🤔	0.385
forever	0.381
#nolacking	0.375
👊	0.373
snitched	0.360
gave	0.355
if	0.354
talk	0.352
keep	0.329
as	0.327
internet	0.325
da	0.323
producer	0.323
gang	0.322
opp	0.314
chica	0.301
homie	0.297
onna	0.289
beat	0.288
bitch	0.287
😂	0.283
casper	0.283
#loony	0.269
hoe	0.260
whoever	0.255
shootin	0.252
#mob	0.197
lady	0.183
rose	0.183
fake	0.181
we	0.178
gun	0.177
be	0.176
smoked	0.175
#8tre	0.170
caught	0.162
jarocity	0.155
!	0.153
🤔	0.137
shordty	0.126
just	0.104
everyday	0.090
nail	0.076

Table 10: Unigrams and their corresponding weights sorted in decreasing weight order (continued)

Unigram	Corresponding Weight
snitch	0.073
bumming	0.071
of	0.070
ona	0.067
u	0.058
steady	0.052
like	0.050
talm	0.048
ways	0.047
#wuggaworld	0.044
soft	0.042
#rt	0.041
yellin	0.041
do	0.037
in	0.035
#cmb	0.034
#cvg	0.029
👎	0.027
no	0.025
#051ym	0.025
tutugang	0.018
#6775	0.010
👤	0.008
#jar	0.005
goof	0.005
dime	0.003
#bricksquad	0.003
👤	0.001

Table 11: Unigrams and their corresponding weights sorted in decreasing weight order (continued)

Models \ Unigrams	in	be	do	any	u	out
Blevins et al. (2016)	3.68	0.0*	0.0*	39.92 [!]	48.08 [!]	25.64 [!]
Chang et al. (2018)	7.56 [!]	6.56 [!]	0.4	9.2	6.84*	3.6
CNN + Twitter	1.88*	1.48	0.84	1.16*	13.88	1.8
LSTM	4.24	2.72	1.08 [!]	8.36	16.12	2.48
LSTM + Rationale	3.96	1.12	0.08	3.64	12.76	1.4*



Models \ Unigrams	basically	yea	ever			we
Blevins et al. (2016)	206.08 [!]	67.84 [!]	47.64 [!]	106.36 [!]	49.84 [!]	69.6 [!]
Chang et al. (2018)	3.64*	2.96	2.04	3.56	2.28*	16.04
CNN + Twitter	5.4	1.52*	1.04	3.08	3.16	10.52*
LSTM	7.44	13.6	2.28	1.64	9.68	16.92
LSTM + Rationale	4.16	8.0	0.16*	0.48*	3.92	11.96

Table 12: The number of model’s prediction flip from non-aggressive to aggressive, out of the 800 attacking tweets generated by inserting a specified “neutral” unigram. To obtain a stable estimate, the count of number of flips of each model is averaged across 5 model runs and models trained on 5 folds - thus leading to non-integer results. For each column, the worst performing entry is marked with “!” and best with “*”.