InstaAsli

Clickbait Detector for Instagram

9:41 PM
Instagram
elizabethgilmore
Liked by andtrl, mfortress and 1,259 oth
elizabethgilmore Missing Su

Introduction

Clickbait is a frequently occurring problem on every social media platform today. For each platform, clickbait is a major issue for which they spend millions of dollars to address but are not able to. We, a team of six members, took up this problem and tried to solve it for Instagram. Using already existing datasets and datasets that we manually collected, we used Machine Learning techniques to classify a post on Instagram as a clickbait and classify the user who posted that post, as a spam user, or a user whose posts tends to be a clickbait.

Motivation

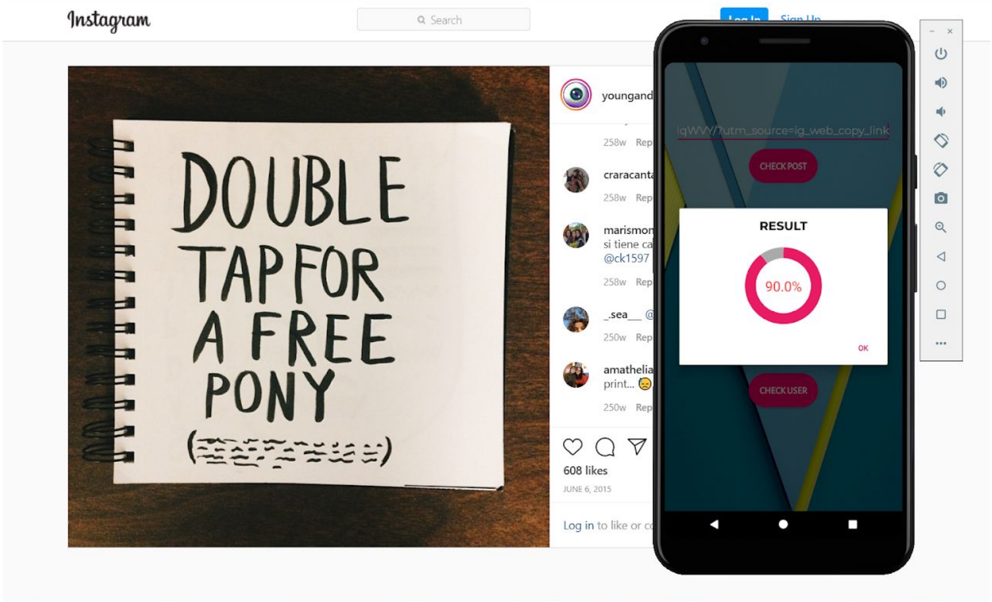
As the course delves into the importance of privacy and security while using social media, we decided to work on the authenticity of the information people consume during their time spent on Online Social Media. For other famous platforms (like YouTube, Facebook, etc.), there are already plenty of researchers working and releasing solutions but we felt that Instagram was a platform which is very famous yet there are not many solutions to detect clickbait on Instagram. Clickbait in itself is a major issue that needs to be worked upon.

What is InstaAsli?

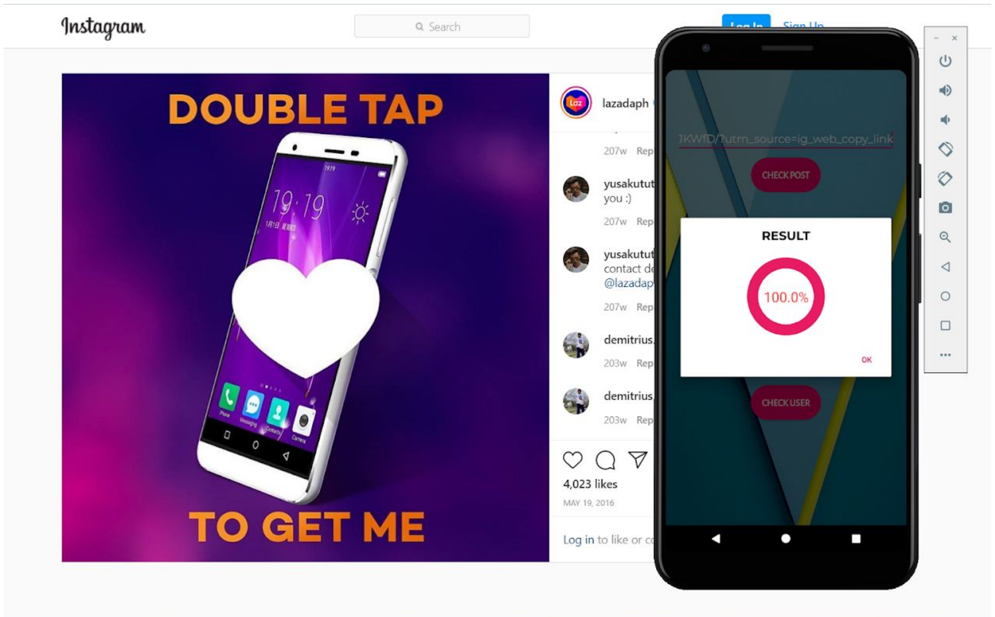
InstaAsli is an android application developed by our team to detect the authenticity of the post as well as the user account on Instagram, using extensive data analysis techniques.
InstaAsli directly opens from the Instagram app on a smartphone and asks for classifying a post. It displays the result soon after.

Examples

Clickbaits

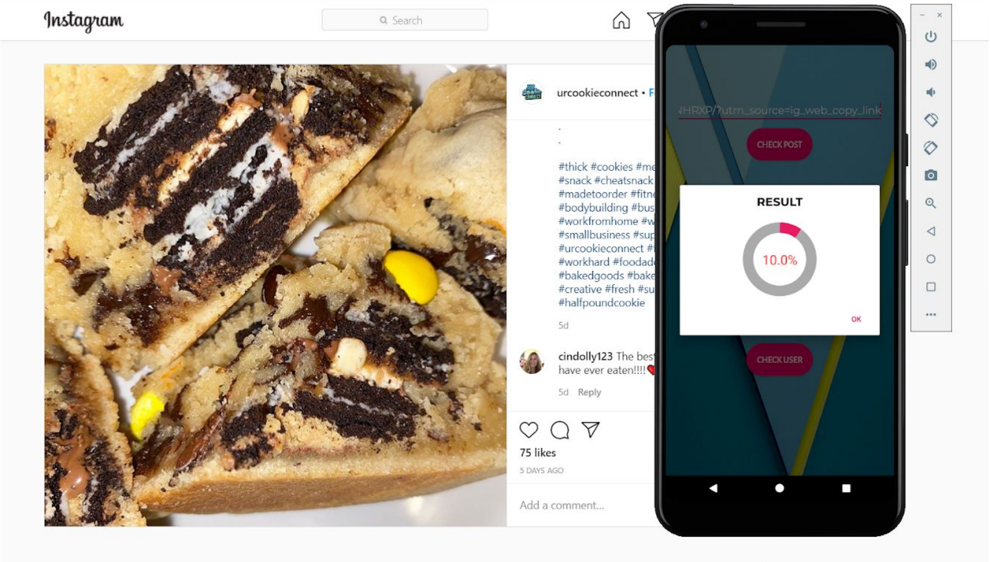


[Link]

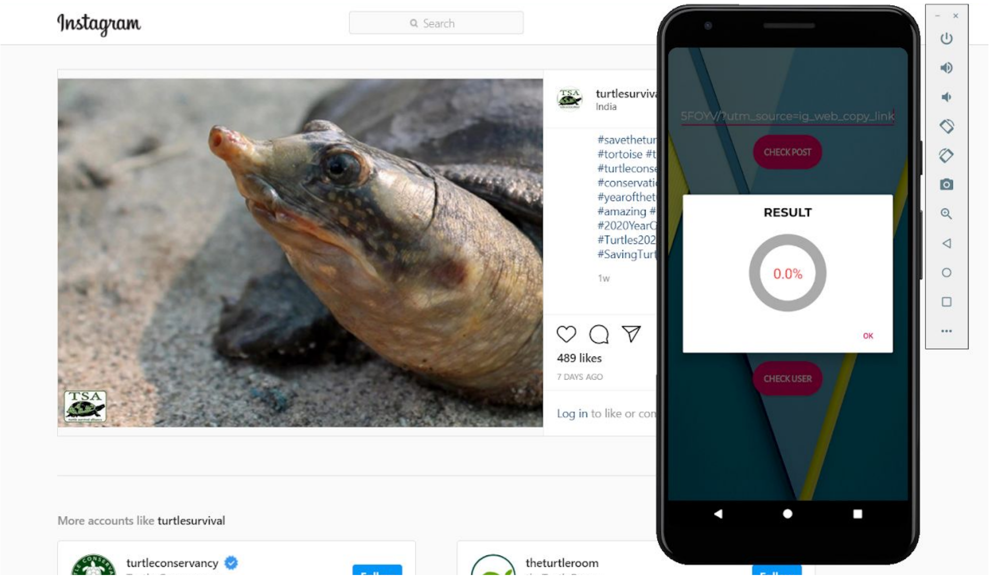


[Link]

Non - Clickbaits

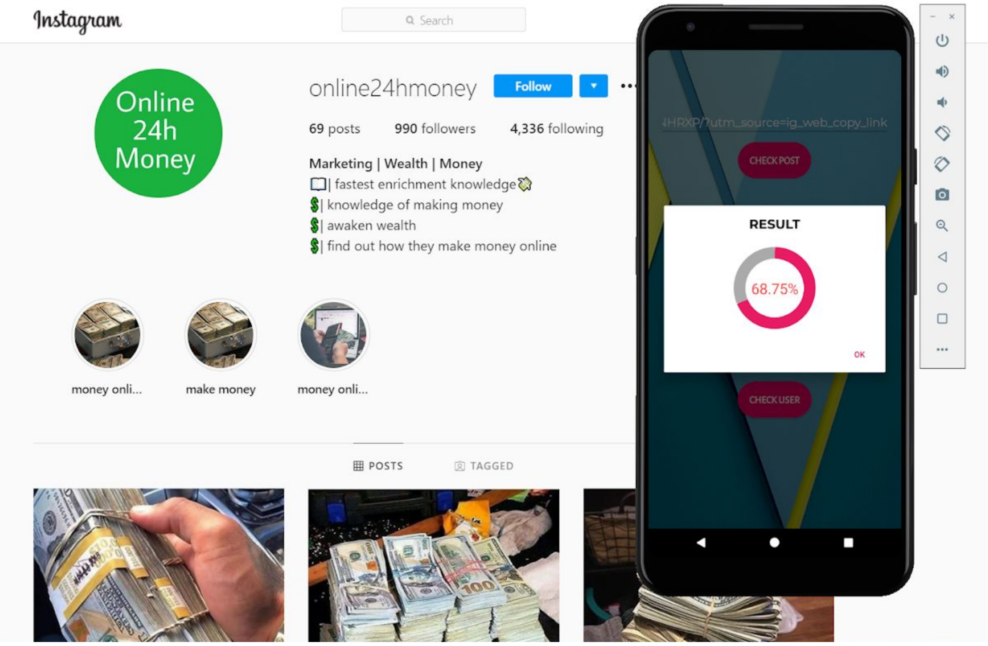


[Link]



[Link]

Clickbait Posting Users

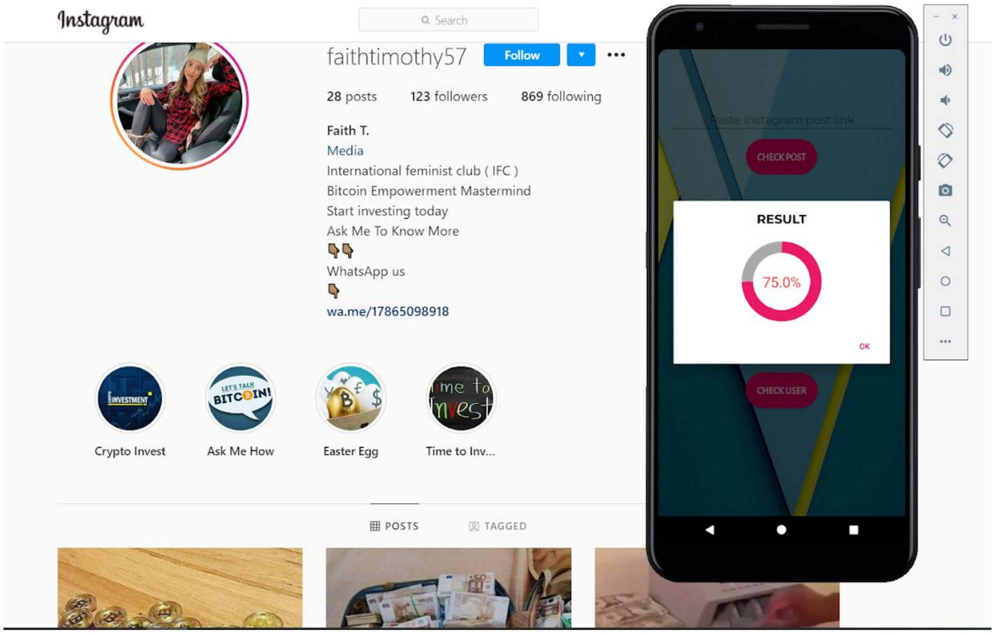


online24hmoney

69 posts 990 followers 4,336 following

Marketing | Wealth | Money
🖵| fastest enrichment knowledge
💲| knowledge of making money
💲| awaken wealth
💲| find out how they make money online

[Link]



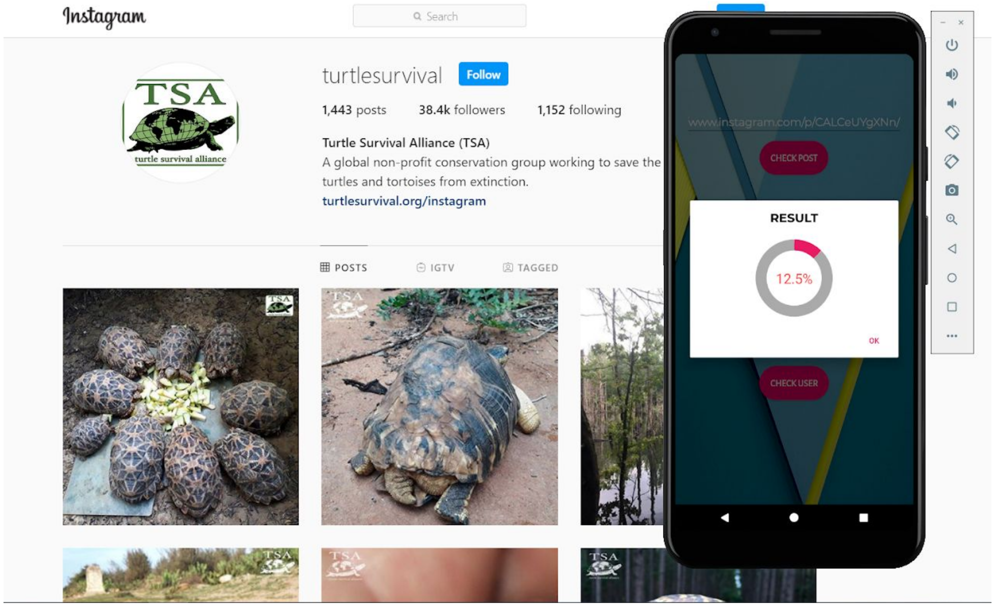
faithtimothy57

28 posts 123 followers 869 following

Faith T.
Media
International feminist club (IFC)
Bitcoin Empowerment Mastermind
Start investing today
Ask Me To Know More
👇👇
WhatsApp us
👇
wa.me/17865098918

[Link]

Non - Clickbait Posting Users



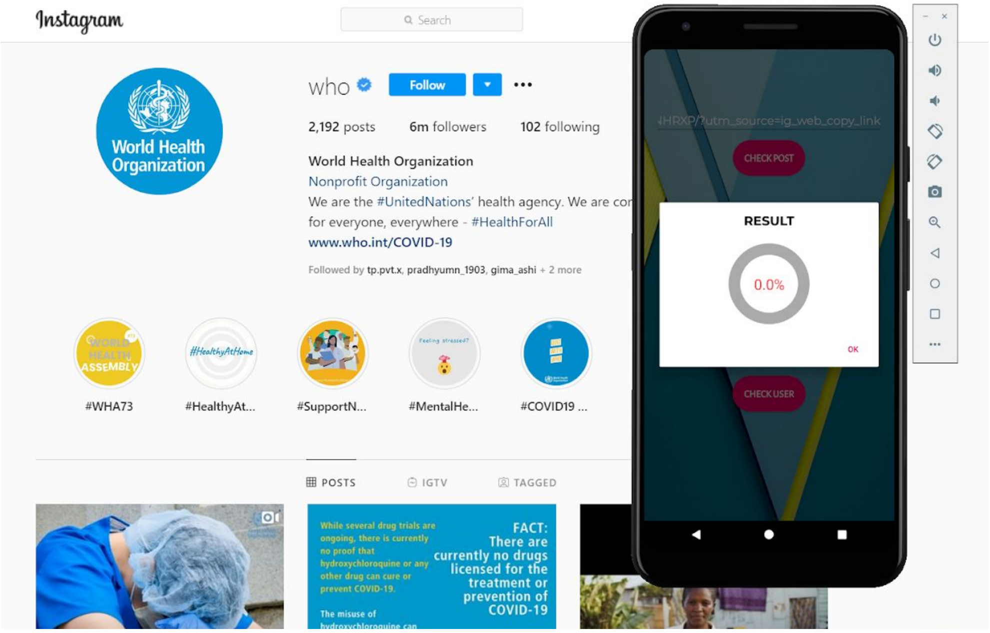
turtlesurvival

1,443 posts 38.4k followers 1,152 following

Turtle Survival Alliance (TSA)
A global non-profit conservation group working to save the turtles and tortoises from extinction.
turtlesurvival.org/instagram

www.instagram.com/p/CALCeUYgXNn/

[Link]


Instagram
Search
Follow
World Health Organization
#WHA73
#HealthyAt...
#SupportN...
#MentalHe...
#COVID19 ...
POSTS
IGTV
TAGGED
While several drug trials are ongoing, there is currently no proof that hydroxychloroquine or any other drug can cure or prevent COVID-19.
FACT: There are currently no drugs licensed for the treatment or prevention of COVID-19
The misuse of hydroxychloroquine can
CHECK POST
RESULT
0.0%
OK
CHECK USER

who

2,192 posts 6m followers 102 following

World Health Organization
Nonprofit Organization
We are the #UnitedNations' health agency. We are cor
for everyone, everywhere - #HealthForAll
www.who.int/COVID-19
Followed by tp.pvt.x, pradhyumn_1903, gima_ashi + 2 more

/HRXP/?utm_source=ig_web_copy_link

[Link]

Methodology

The app runs Python code on the backend. This code takes the link of the post as input and classifies it into a clickbait after running two sets of classifiers for classifying the post and one set of classifiers for classifying the user of that post. We trained many classifiers but, in the end, we chose the ones which are the lightest and most accurate.

The Python code uses the *scikit-learn* library's classifiers that are already trained and stored in a pickle file (obtained using the *pickle* library of Python). It uses the libraries *InstagramAPI* (not official API) [Link] and *instalooter* [Link] to scrape information about the input post from its link and also the information about the user that posted that post. After that, the text of the image posted in the post is extracted using the library *pyinstaller* [Link]. The text on the image and the text in the caption are processed afterwards. Stopwords are removed, emojis are removed, links are removed, non-ASCII characters are removed and finally the punctuation is removed. After that, the classifiers are run and the final clickbait percentage is returned.

Analysis & Performance

We began our analysis with the pre-existing and manually collected databases that we had.

A. Database 1[Link]

This database had the following statistics.

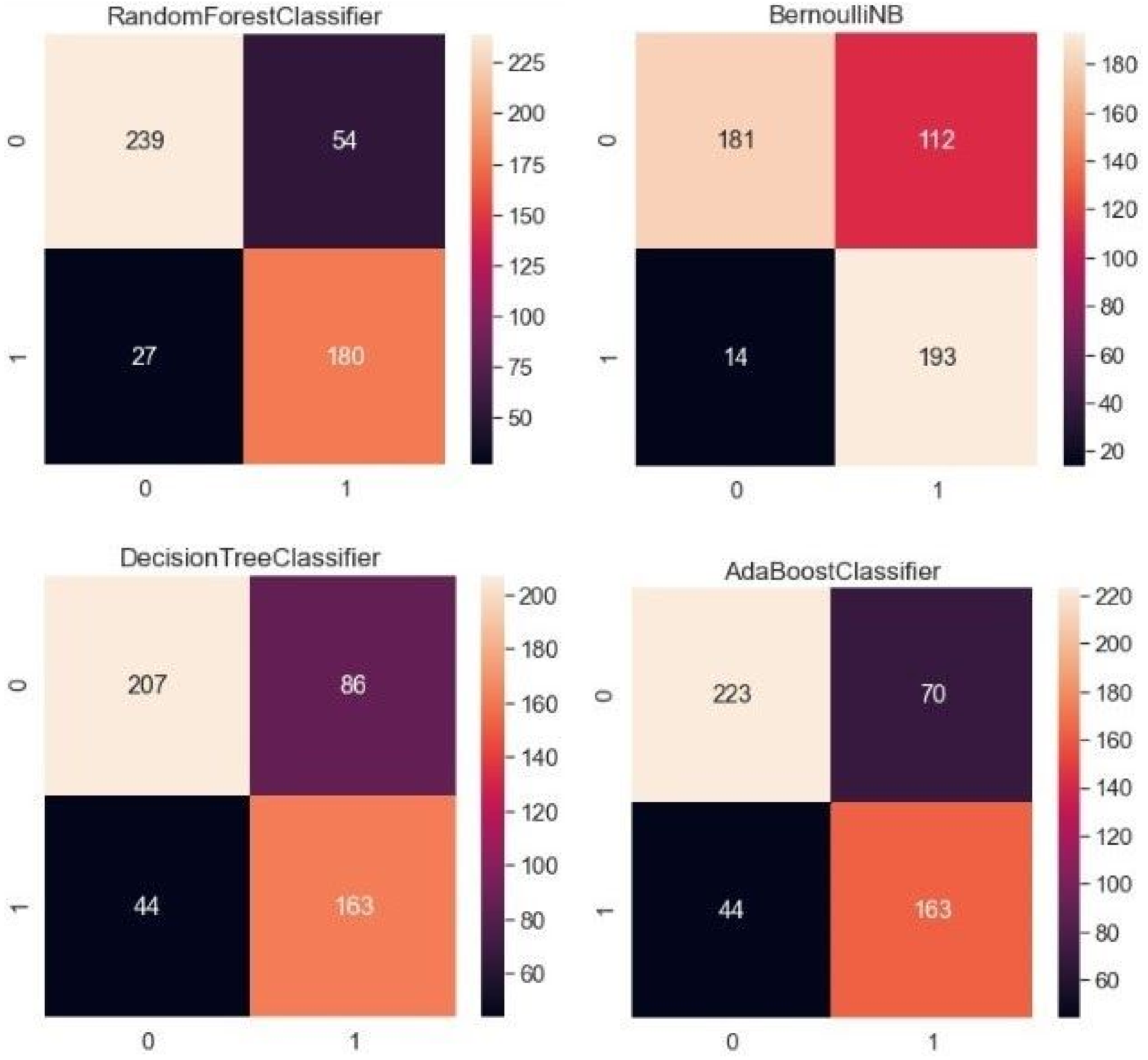
Train data:	**Test data:**	**Total data:**
Count of 0s: 3967 => 54.57%	Count of 0s: 293 => 58.60%	Count of 0s: 4260 => 54.83%
Count of 1s: 3302 => 45.43%	Count of 1s: 207 => 41.40%	Count of 1s: 3509 => 45.17%
Total: 7269	Total: 500	Total: 7769

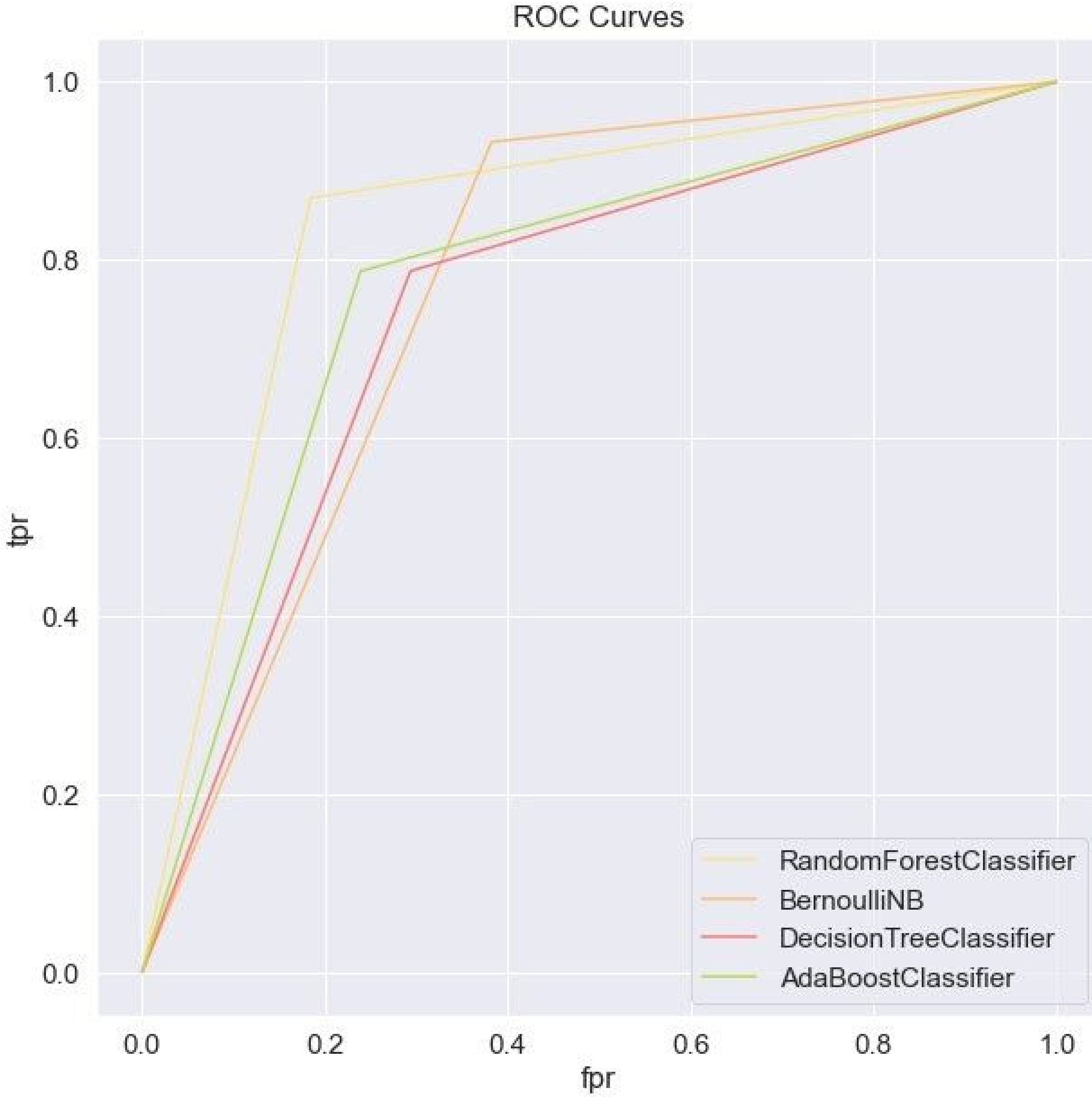
(0 ⇒ Not a Clickbait Post, 1 ⇒ Clickbait Post)

We trained an ensemble of classifiers and chose the best ones as follows:

Classifier Type	**Train Accuracy**	**Train F1 Score**	**Test Accuracy**	**Test F1 Score**
RandomForestClassifier	0.993	0.992	0.838	0.816
BernoulliNB	0.894	0.892	0.748	0.754
DecisionTreeClassifier	0.743	0.729	0.740	0.715
AdaBoostClassifier	0.812	0.793	0.772	0.741

The confusion matrices of the corresponding classifiers and their ROC curves are as follows:


RandomForestClassifier
0
1
239
54
27
180
0
1
225
200
175
150
125
100
75
50
BernoulliNB
0
1
181
112
14
193
0
1
180
160
140
120
100
80
60
40
20
DecisionTreeClassifier
0
1
207
86
44
163
0
1
200
180
160
140
120
100
80
60
AdaBoostClassifier
0
1
223
70
44
163
0
1
220
200
180
160
140
120
100
80
60


ROC Curves
tpr
fpr
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
RandomForestClassifier
BernoulliNB
DecisionTreeClassifier
AdaBoostClassifier

B. Database 2 [Link]

This database had the following statistics.

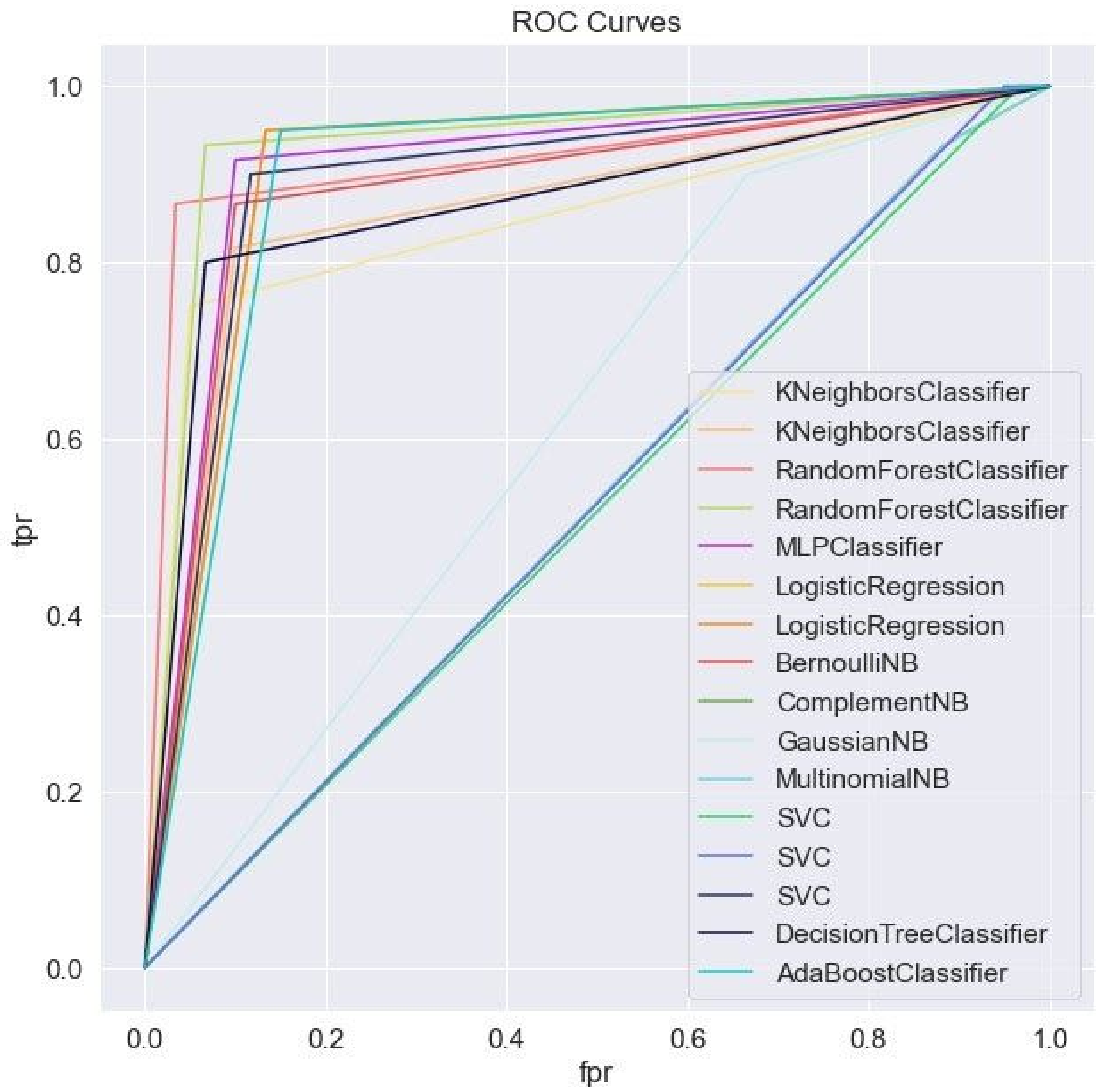
Train data:	**Test data:**	**Total data:**
Count of 0s: 288 => 50.00% Count of 1s: 288 => 50.00% Total: 576	Count of 0s: 60 => 50.00% Count of 1s: 60 => 50.00% Total: 120	Count of 0s: 348 => 50.00% Count of 1s: 348 => 50.00% Total: 696

(0 ⇒ Not a Clickbait Posting User, 1 ⇒ Clickbait Posting User)

We trained an ensemble of classifiers and chose the best ones as follows:

Classifier Type	**Train Accuracy**	**Train F1 Score**	**Test Accuracy**	**Test F1 Score**
KNeighborsClassifier	0.994	0.941	0.850	0.833
KNeighborsClassifier	0.943	0.943	0.858	0.852
RandomForestClassifier	0.990	0.989	0.917	0.912
RandomForestClassifier	1.000	1.000	0.993	0.933
MLPClassifier	0.938	0.939	0.908	0.909
LogisticRegression	0.915	0.915	0.908	0.912
LogisticRegression	0.915	0.915	0.908	0.912
BernoulliNB	0.885	0.883	0.883	0.881
ComplementNB	0.559	0.678	0.525	0.663
GaussianNB	0.691	0.759	0.617	0.701
MultinomialNB	0.559	0.678	0.525	0.663
SVC	0.514	0.673	0.517	0.674
SVC	0..524	0.678	0525	0.674
SVC	0.924	0.921	0.892	0.893
DecisionTreeClassifier	0.960	0.960	0.867	0.857
AdaBoostClassifier	0.964	0.963	0.900	0.905

The ROC curves of the corresponding classifiers are as follows:

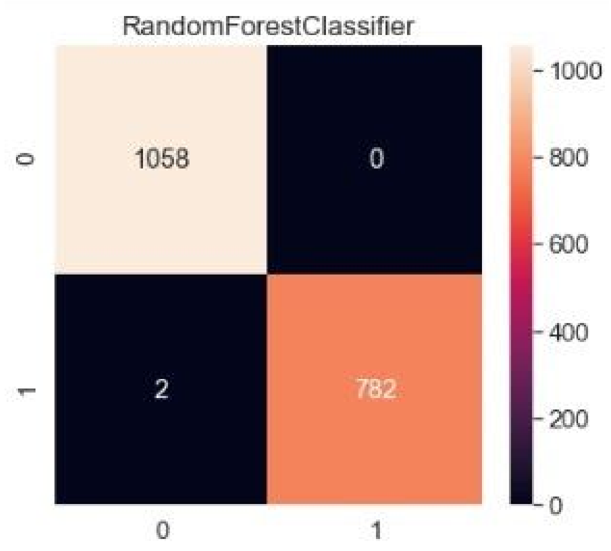

ROC Curves
tpr
fpr
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
KNeighborsClassifier
KNeighborsClassifier
RandomForestClassifier
RandomForestClassifier
MLPClassifier
LogisticRegression
LogisticRegression
BernoulliNB
ComplementNB
GaussianNB
MultinomialNB
SVC
SVC
SVC
DecisionTreeClassifier
AdaBoostClassifier

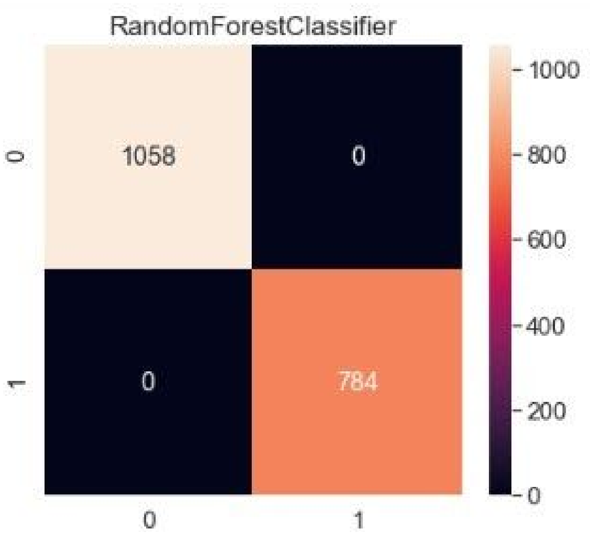
C. Database 3 (Manually collected)

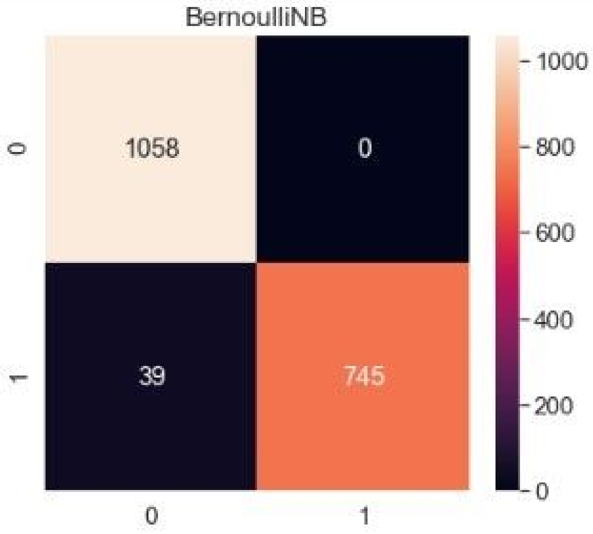
We manually collected 1842 posts consisting of 784 (42.5%) clickbait posts and 1058 (57.5%) non-clickbait posts. We then trained an ensemble of classifiers on this database and chose the best ones as follows: (Used the whole database as training data)

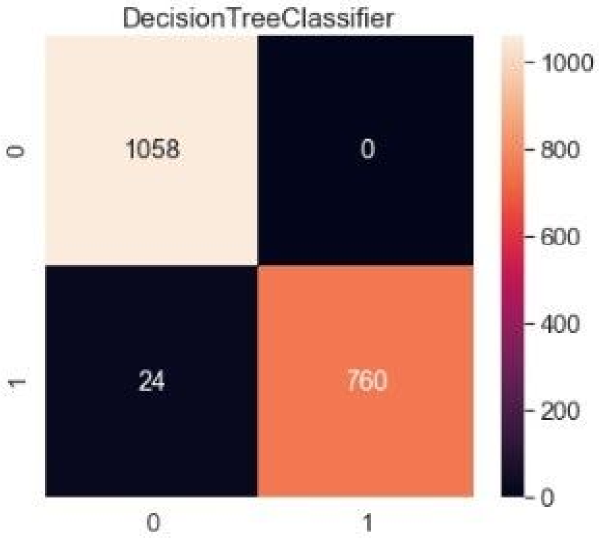
Classifier Type	Training Accuracy	Training F1 Score
RandomForestClassifier	0.999	0.999
RandomForestClassifier	1.000	1.000
BernoulliNB	0.979	0.974
DecisionTreeClassifier	0.987	0.984
AdaBoostClassifier	1.000	1.000

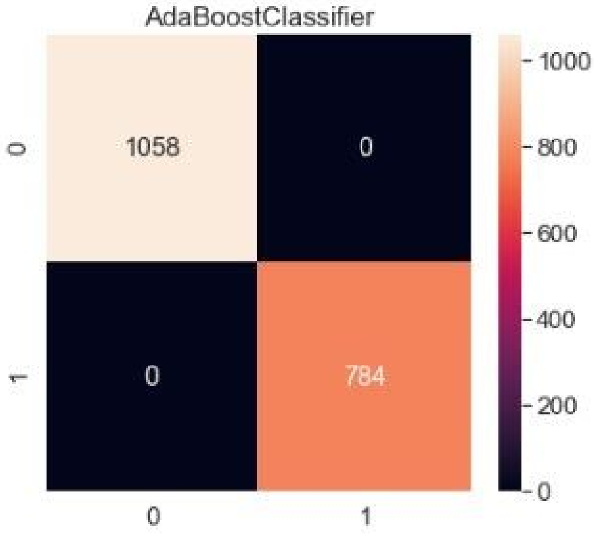
The confusion matrices of the corresponding classifiers and their ROC curves are as follows:

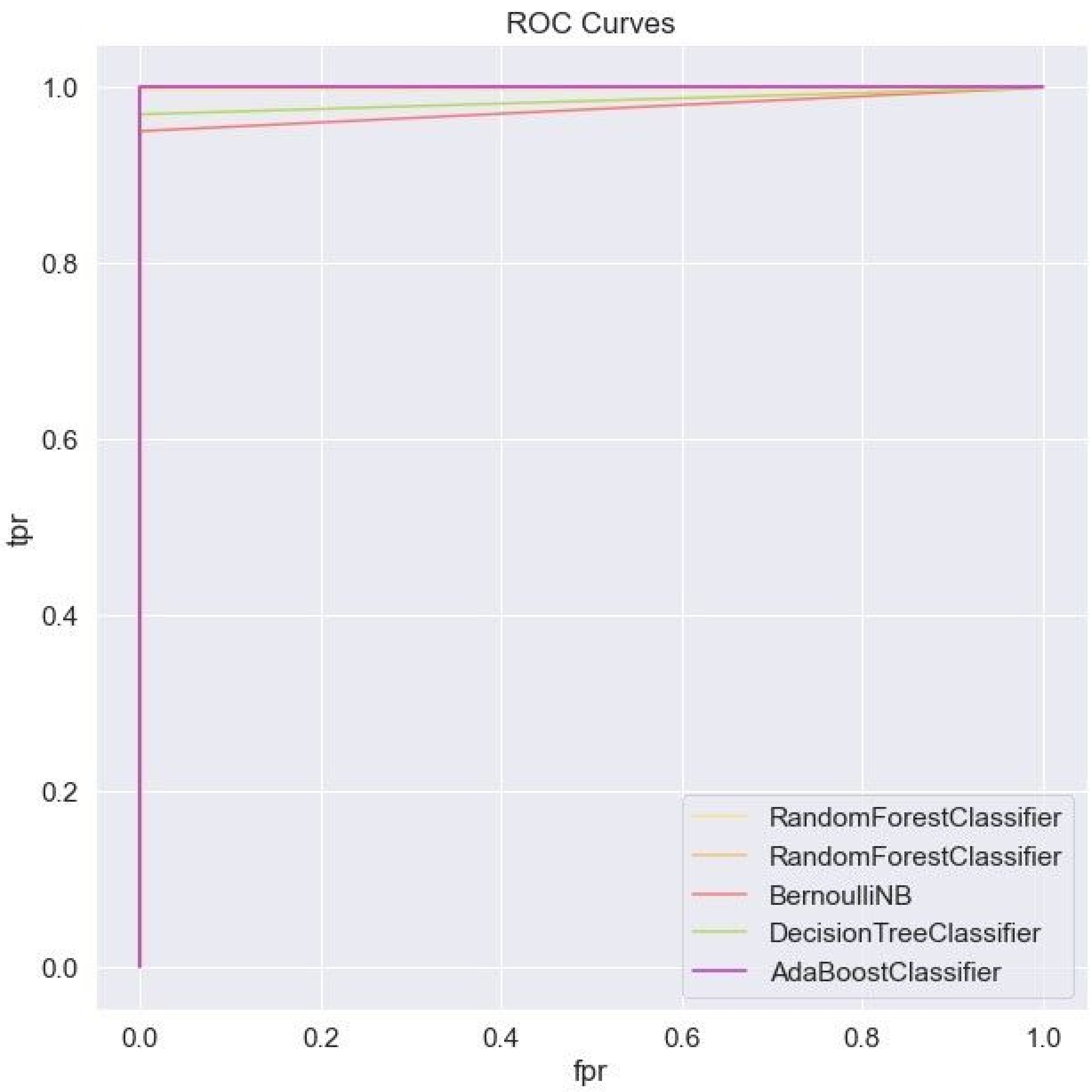





BernoulliNB
1058
0
39
745
0
1
0
1
1000
800
600
400
200
0


DecisionTreeClassifier
1058
0
24
760
0
1
0
1
1000
800
600
400
200
0

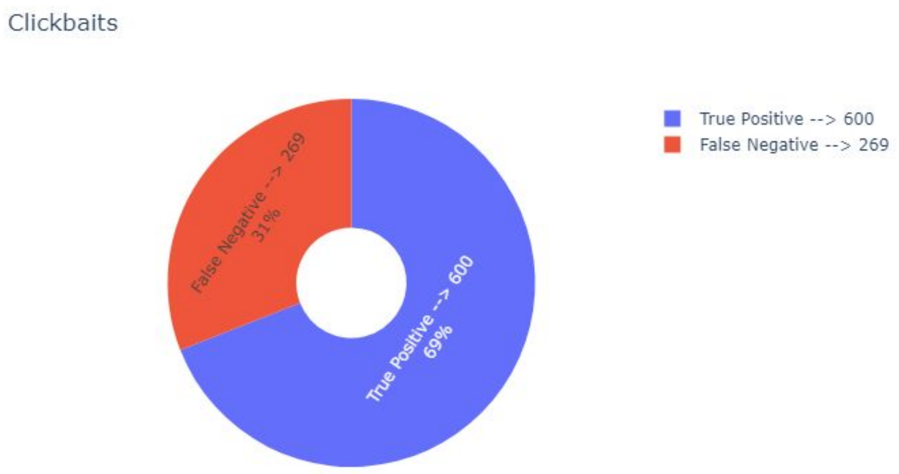

AdaBoostClassifier
1058
0
0
784
0
1
0
1
1000
800
600
400
200
0



D. Actual Performance Analysis

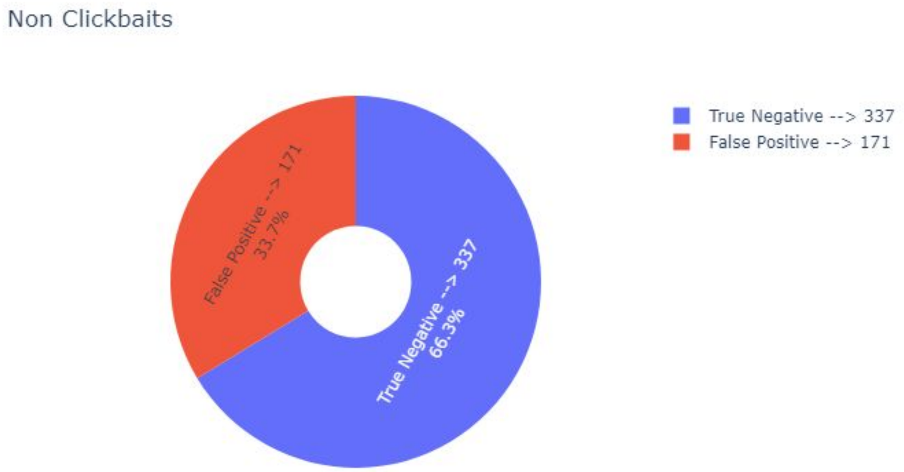
We performed an analysis of how the app actually performs using links of posts and users. We collected clickbait and non-clickbait posts on Instagram using an online tool called Phantom Buster [Link]. This tool also outputs the usernames, so we used those to analyse InstaAsli's performance on classifying users who tend to post more clickbait. Using this tool we collected and ran the app on:

1. **869 clickbait posts on Instagram and got an accuracy of 69%.**



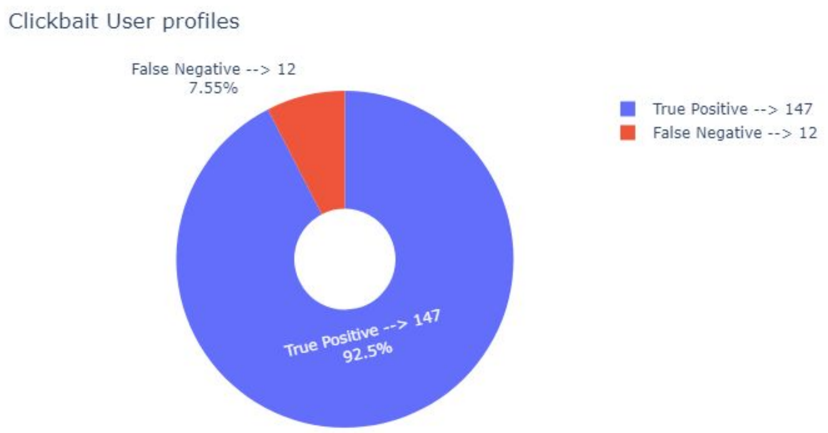
Clickbaits

2. **508 non-clickbait posts and got an accuracy of 66.3%.**

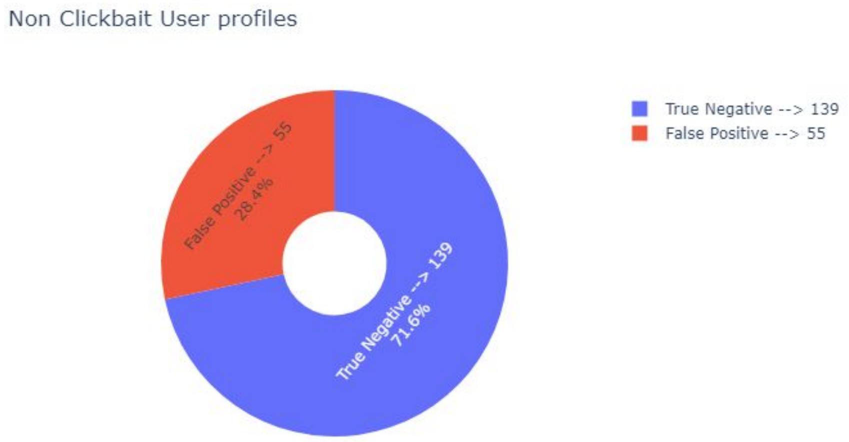


Non Clickbaits

3. **159 clickbait user profiles (extracted from the clickbait posts) and got an accuracy of 92.5%.**



4. **194 non-clickbait user profiles (extracted from the non-clickbait posts) and got an accuracy of 71.6%.**



Future Plans for Improvement

We plan to increase the accuracy of the classifiers even more and collect even more posts to make our self-collected database bigger and train our classifiers on them to make them even more accurate. This will lead the classifiers to work more accurately on a given post.

We also plan to improve the speed of the app, so that the classification is done faster and the result is displayed sooner. Other than that, we also plan to improve the GUI of the app to make it more user-friendly.

We will add more complex Machine Learning and Natural Language Processing methods while keeping the execution time low, to make the classification even better and more accurate and precise.

We plan to add a link collector that collects the links put in by the users. With these links, we will also collect users' opinions on whether the posts/users are clickbait/spam or not. These links will then be combined with the database of our classifiers and the classifiers will be retrained (at the backend, the user won’t have to wait for the run to finish). This will make the app self-improving.

References

1. https://medium.com/@mike_liu/predicting-instagram-clickbait-posts-f1dba34c86cb - Clickbait Database
2. https://www.kaggle.com/free4ever1/instagram-fake-spammer-genuine-accounts - Spam User Database
3. https://buildmedia.readthedocs.org/media/pdf/instalooter/latest/instalooter.pdf
4. https://towardsdatascience.com/read-text-from-image-with-one-line-of-python-code-c22ede074cac
5. https://medium.com/@adamaulia/crawling-instagram-using-instalooter-2791edb453ff
6. https://www.youtube.com/watch?v=n7Yw7VUrMxs
7. https://almeta.io/en/blog/how-to-detect-clickbait-headlines-using-nlp/
8. https://www.researchgate.net/publication/320241720_Machine_Learning_Based_Detection_of_Clickbait_Posts_in_Social_Media
9. http://www.sscnet.ucla.edu/comm/jjoo/web/icwsm18-clickbait-instagram.pdf
10. https://arxiv.org/pdf/1806.07713v1.pdf
11. https://venngage.com/blog/7-reasons-why-clicking-this-title-will-prove-why-you-clicked-this-title/
12. https://phantombuster.com/
13. https://docs.djangoproject.com/en/3.0/
14. https://pypi.org/project/InstagramAPI/
15. https://developer.android.com/docs

Team Members

1. Abhishrut Khanna (2017006)
2. Pankaj Yadav (2017074)
3. Shashwat Jain (2017103)
4. Tanish Jain (2017115)
5. Porvil (2017304)
6. Sanchit Mittal (2017312)

Video

https://youtu.be/DjltwBIM9zw

Blog