Kaggle Competetion

Dear Professor : Mr.Manthouri

Produced By : Ghasemi,morteza

https://github.com/Morteza-Ghasemi

January 2021

Kaggle Competetion
Toxic Comment Classification Challenge

- https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge/overview

Kaggle Competetion Overview

- Discussing things you care about can be difficult. The threat of abuse and harassment online means that many people stop expressing themselves and give up on seeking different opinions. Platforms struggle to effectively facilitate conversations, leading many communities to limit or completely shut down user comments.

Kaggle Competetion Overview

- The Conversation AI team, a research initiative founded by Jigsaw and Google (both a part of Alphabet) are working on tools to help improve online conversation. One area of focus is the study of negative online behaviors, like toxic comments (i.e. comments that are rude, disrespectful or otherwise likely to make someone leave a discussion). So far they've built a range of publicly available models served through the Perspective API, including toxicity. But the current models still make errors, and they don't allow users to select which types of toxicity they're interested in finding (e.g. some platforms may be fine with profanity, but not with other types of toxic content).

Kaggle Competetion Overview

- In this competition, you're challenged to build a multi-headed model that's capable of detecting different types of of toxicity like threats, obscenity, insults, and identity-based hate better than Perspective's current models. You'll be using a dataset of comments from Wikipedia's talk page edits. Improvements to the current model will hopefully help online discussion become more productive and respectful.
- *Disclaimer: the dataset for this competition contains text that may be considered profane, vulgar, or offensive.*

Kaggle Competetion

Implementing Kaggle Competetion with Scikit-Learn

- **Importing libraries**

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

Kaggle Competetion

Implementing Kaggle Competetion with Scikit-Learn

- **Data Preprocessing**
  - ✓ Dividing the data into attributes and labels
  - ✓ dividing the data into training and testing sets
  - ✓ replace(r'^\s*$', np.NaN, regex=True)
  - ✓ dropna(axis = 0, how ='any')
  - ✓ reset_index(drop=True)

Kaggle Competetion

Implementing Kaggle Competetion with Scikit-Learn

- **Exploratory Data Analysis**
  - ✓ To get a feel of how our dataset actually looks, execute the following command:
    bankdata.head()
  - ✓ You can see that all of the attributes in the dataset are numeric. The label is also numeric i.e. 0 and 1.

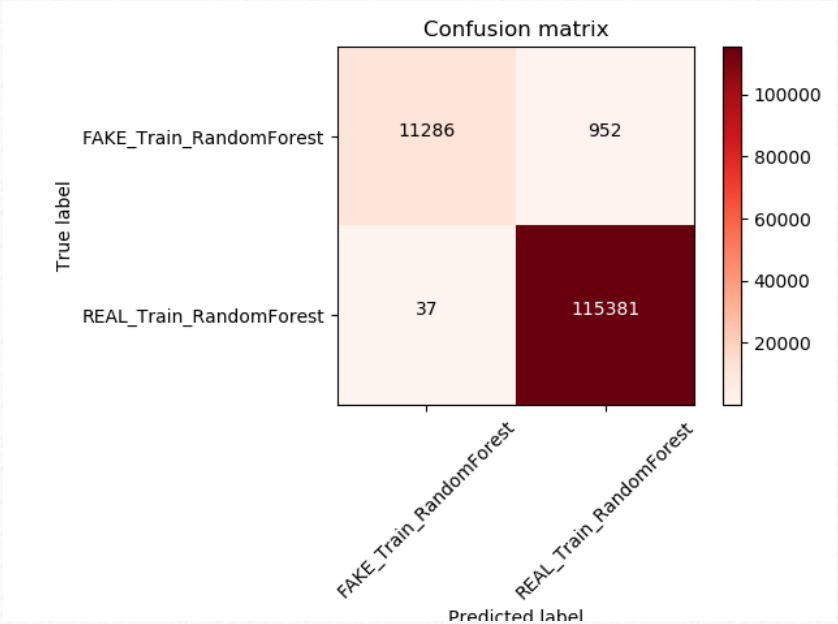
Kaggle Competetion

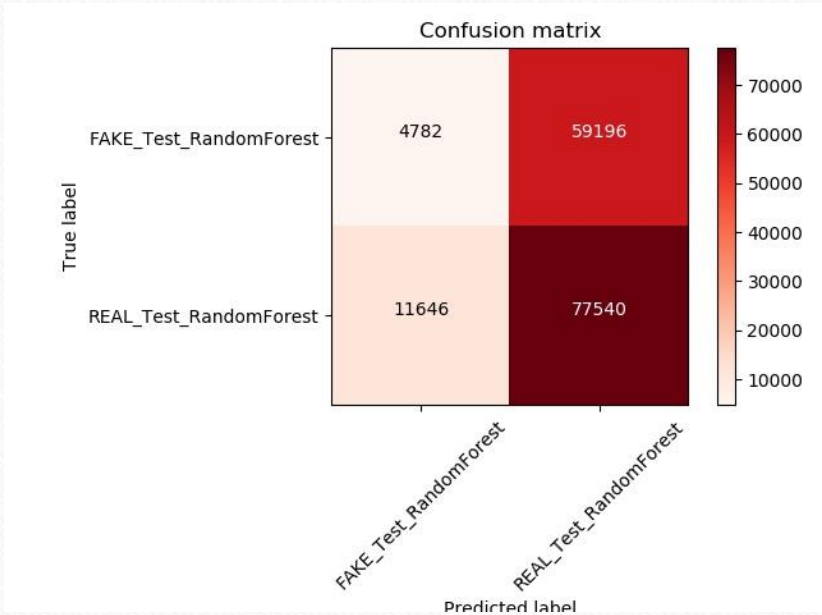
Implementing Kaggle Competetion with Scikit-Learn

- **Evaluating the Algorithm**
  - ✓ Confusion matrix, precision, recall, and F1 measures are the most commonly used metrics for classification tasks.
    - ✓ from sklearn.metrics import classification_report, confusion_matrix
    - ✓ print(confusion_matrix(y_test,y_pred))
    - ✓ print(classification_report(y_test,y_pred))

Kaggle Competetion
Random Forest

- The output of the **Random Forest** looks like this:





Kaggle Competetion
Random Forest

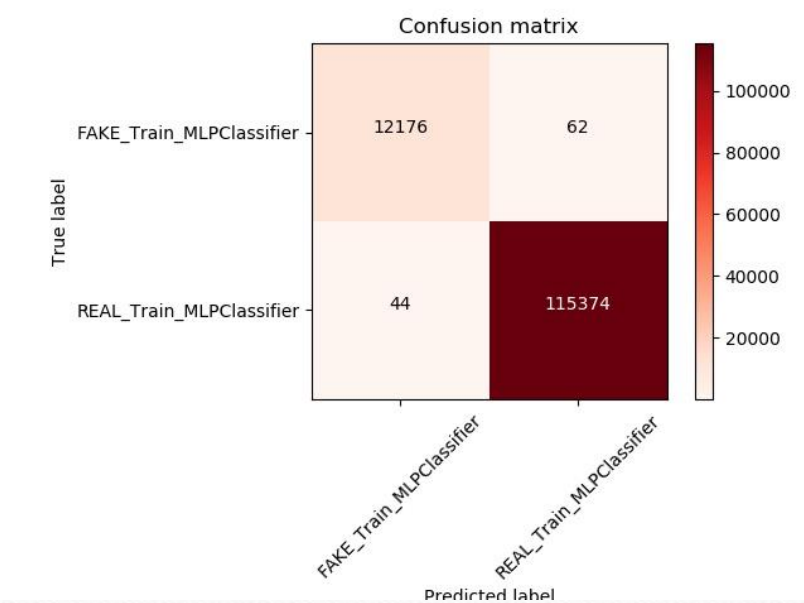
- The output of the **Random Forest** looks like this:

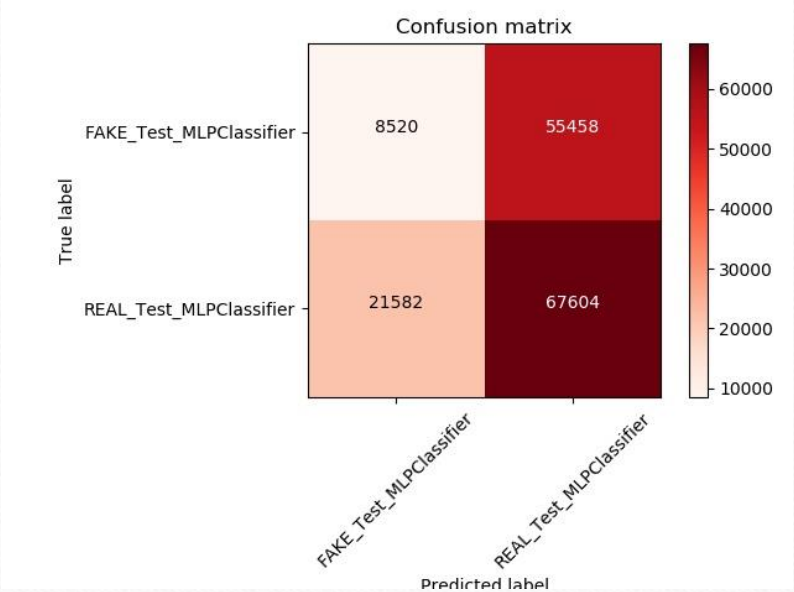
```
classification_report for RandomForestClassifier_train Model

             precision    recall  f1-score   support

         -1       0.99      1.00      1.00    115418
          1       1.00      0.92      0.96     12238

avg / total       0.99      0.99      0.99    127656


Accuracy_RandomForestClassifier_train:   0.992

Cohens kappa_train_RandomForestClassifier: 0.953763

[[115381     37]
[   952  11286]]
```

```
classification_report for RandomForestClassifier_test Model

             precision    recall  f1-score   support

         -1       0.57      0.87      0.69     89186
          1       0.29      0.07      0.12     63978

avg / total       0.45      0.54      0.45    153164


Accuracy_RandomForestClassifier_test:   0.537
```

Kaggle Competetion
MLP

- The output of the **MLP** looks like this:





Kaggle Competetion
MLP

- The output of the **MLP** looks like this:

classification_report for MLPClassifier_train Model

	precision	recall	f1-score	support
-1	1.00	1.00	1.00	115418
1	1.00	0.99	1.00	12238
avg / total	1.00	1.00	1.00	127656

Accuracy_MLPClassifier_train: 0.999

Cohens kappa_train_MLPClassifier: 0.995207

[[115374 44]
[62 12176]]

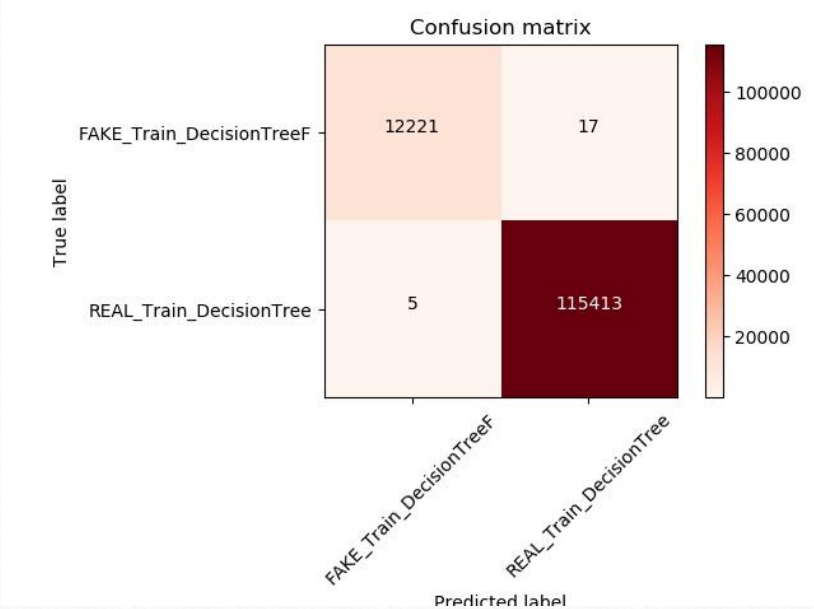
classification_report for MLPClassifier_test Model

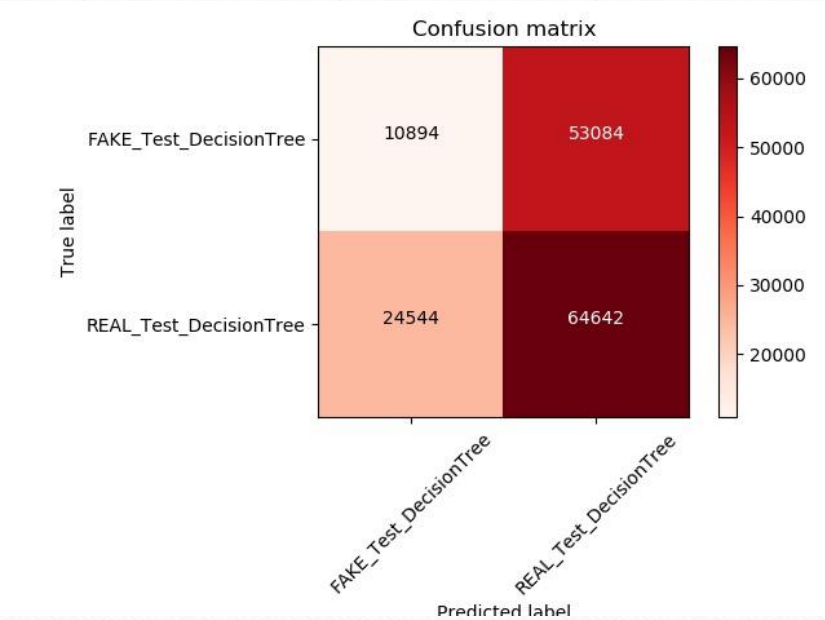
	precision	recall	f1-score	support
-1	0.55	0.76	0.64	89186
1	0.28	0.13	0.18	63978
avg / total	0.44	0.50	0.45	153164

Accuracy_MLPClassifier_test: 0.497

Kaggle Competetion
DecisionTree

- The output of the **DecisionTree** looks like this:





Kaggle Competetion
DecisionTree

- The output of the **DecisionTree** looks like this:

classification_report for DecisionTree_train Model

	precision	recall	f1-score	support
-1	1.00	1.00	1.00	115418
1	1.00	1.00	1.00	12238
avg / total	1.00	1.00	1.00	127656

Accuracy_DecisionTree_train: 1.000

classification_report for DecisionTree_test Model

	precision	recall	f1-score	support
-1	0.55	0.72	0.62	89186
1	0.31	0.17	0.22	63978
avg / total	0.45	0.49	0.46	153164

Accuracy_DecisionTree_test: 0.493

The End