In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
df=pd.read_csv(r"C3_bot_detection_data.csv")
df
```

Out[2]:

	User ID	Username	Tweet	Retweet Count	Mention Count	Follower Count	Verified	Bot Label	Loc
0	132131	flong	Station activity person against natural majori...	85	1	2353	False	1	Adki
1	289683	hinesstephanie	Authority research natural life material staff...	55	5	9617	True	0	Sande
2	779715	roberttran	Manage whose quickly especially foot none to g...	6	2	4363	True	0	Harris
3	696168	pmason	Just cover eight opportunity strong policy which.	54	5	2242	True	1	Martine
4	704441	noah87	Animal sign six data good or.	26	3	8438	False	1	Camach
...	...	...	...	...	...	...	...	...	
49995	491196	uberg	Want but put card direction know miss former h...	64	0	9911	True	1	Kimberly
49996	739297	jessicamunoz	Provide whole maybe agree church respond most ...	18	5	9900	False	1	Gree
49997	674475	lynncunningham	Bring different everyone international capital...	43	3	6313	True	1	Debor
49998	167081	richardthompson	Than about single generation itself seek sell ...	45	1	6343	False	0	Stephe
49999	311204	daniel29	Here morning class various room human true bec...	91	4	4006	False	0	Nova

50000 rows × 11 columns

In [3]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50000 entries, 0 to 49999
Data columns (total 11 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   User ID         50000 non-null  int64
 1   Username        50000 non-null  object
 2   Tweet           50000 non-null  object
 3   Retweet Count   50000 non-null  int64
 4   Mention Count   50000 non-null  int64
 5   Follower Count  50000 non-null  int64
 6   Verified        50000 non-null  bool
 7   Bot Label       50000 non-null  int64
 8   Location        50000 non-null  object
 9   Created At      50000 non-null  object
 10  Hashtags        41659 non-null  object
dtypes: bool(1), int64(5), object(5)
memory usage: 3.9+ MB
```

In [4]:
```
df=df.dropna()
```

In [5]:
```
df.isnull().sum()
```

Out[5]:
```
User ID           0
Username          0
Tweet             0
Retweet Count     0
Mention Count     0
Follower Count    0
Verified          0
Bot Label         0
Location          0
Created At        0
Hashtags          0
dtype: int64
```

In [6]:
```
df.describe()
```

Out[6]:

	User ID	Retweet Count	Mention Count	Follower Count	Bot Label
count	41659.000000	41659.000000	41659.000000	41659.000000	41659.000000
mean	548640.613097	49.950911	2.515207	4990.867928	0.500204
std	259990.806985	29.195286	1.709249	2880.947193	0.500006
min	100025.000000	0.000000	0.000000	0.000000	0.000000
25%	321829.500000	25.000000	1.000000	2493.500000	0.000000
50%	548396.000000	50.000000	3.000000	4997.000000	1.000000
75%	772751.500000	75.000000	4.000000	7475.500000	1.000000
max	999995.000000	100.000000	5.000000	10000.000000	1.000000

In [7]:
```
df["Bot Label"].value_counts()
```

Out[7]:
```
1    20838
0    20821
Name: Bot Label, dtype: int64
```

In [8]:
```
df1=df[['User ID','Retweet Count','Mention Count','Follower Count','Bot Label']
```

In [9]:
```
x=df1.drop('Bot Label',axis=1)
y=df1['Bot Label']
```

In [10]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,train_size=0.70)
```

In [11]:
```
from sklearn.ensemble import RandomForestClassifier
rfc=RandomForestClassifier()
rfc.fit(x_train,y_train)
```

Out[11]:
```
RandomForestClassifier()
```

In [12]:
```
parameters={'max_depth':[1,2,3,4,5],
            'min_samples_leaf':[5,10,15,20,25],
            'n_estimators':[10,20,30,40,50]}
```

In [13]:
```
from sklearn.model_selection import GridSearchCV
grid_search=GridSearchCV(estimator=rfc,param_grid=parameters,cv=2,scoring="accu
grid_search.fit(x_train,y_train)
```

Out[13]:
```
GridSearchCV(cv=2, estimator=RandomForestClassifier(),
             param_grid={'max_depth': [1, 2, 3, 4, 5],
                         'min_samples_leaf': [5, 10, 15, 20, 25],
                         'n_estimators': [10, 20, 30, 40, 50]},
             scoring='accuracy')
```

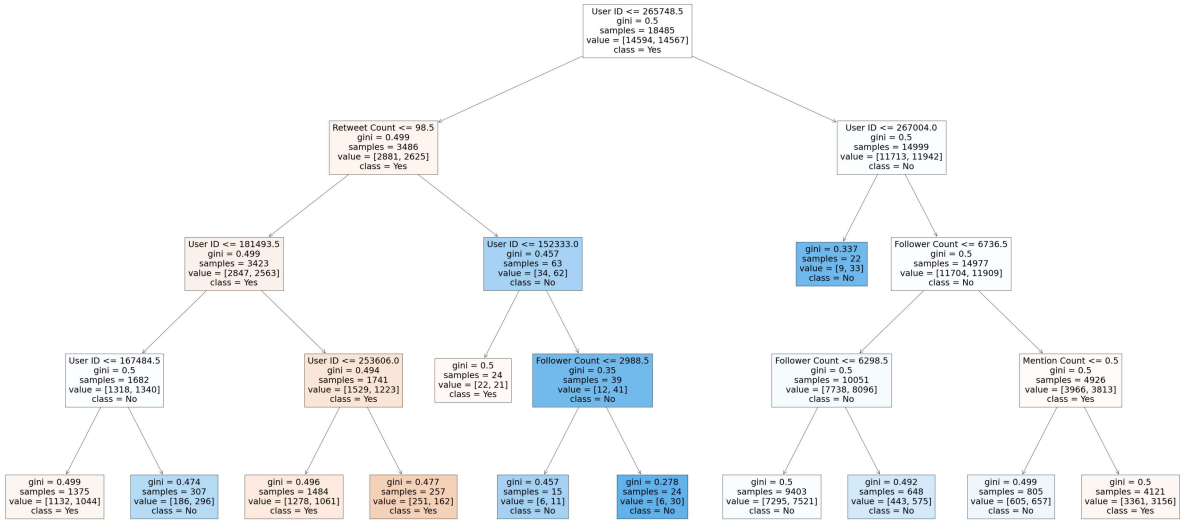
In [14]:
```
grid_search.best_score_
```

Out[14]:
```
0.5052638945452906
```

In [15]:
```
rfc_best=grid_search.best_estimator_
```

In [16]:
```
from sklearn.tree import plot_tree
plt.figure(figsize=(80,40))
plot_tree(rfc_best.estimators_[5],feature_names=x.columns,class_names=['Yes','N
```

Out[16]:
```
[Text(2399.4, 1956.96, 'User ID <= 265748.5\ngini = 0.5\nsamples = 18485\nval
ue = [14594, 14567]\nclass = Yes'),
 Text(1450.8, 1522.0800000000002, 'Retweet Count <= 98.5\ngini = 0.499\nsampl
es = 3486\nvalue = [2881, 2625]\nclass = Yes'),
 Text(892.8, 1087.2, 'User ID <= 181493.5\ngini = 0.499\nsamples = 3423\nvalu
e = [2847, 2563]\nclass = Yes'),
 Text(446.4, 652.3200000000002, 'User ID <= 167484.5\ngini = 0.5\nsamples = 1
682\nvalue = [1318, 1340]\nclass = No'),
 Text(223.2, 217.44000000000005, 'gini = 0.499\nsamples = 1375\nvalue = [113
2, 1044]\nclass = Yes'),
 Text(669.5999999999999, 217.44000000000005, 'gini = 0.474\nsamples = 307\nva
lue = [186, 296]\nclass = No'),
 Text(1339.1999999999998, 652.3200000000002, 'User ID <= 253606.0\ngini = 0.4
94\nsamples = 1741\nvalue = [1529, 1223]\nclass = Yes'),
 Text(1116.0, 217.44000000000005, 'gini = 0.496\nsamples = 1484\nvalue = [127
8, 1061]\nclass = Yes'),
 Text(1562.3999999999999, 217.44000000000005, 'gini = 0.477\nsamples = 257\nv
alue = [251, 162]\nclass = Yes'),
 Text(2008.8, 1087.2, 'User ID <= 152333.0\ngini = 0.457\nsamples = 63\nvalue
= [34, 62]\nclass = No'),
 Text(1785.6, 652.3200000000002, 'gini = 0.5\nsamples = 24\nvalue = [22, 21]
\nclass = Yes'),
 Text(2232.0, 652.3200000000002, 'Follower Count <= 2988.5\ngini = 0.35\nsamp
les = 39\nvalue = [12, 41]\nclass = No'),
 Text(2008.8, 217.44000000000005, 'gini = 0.457\nsamples = 15\nvalue = [6, 1
1]\nclass = No'),
 Text(2455.2, 217.44000000000005, 'gini = 0.278\nsamples = 24\nvalue = [6, 3
0]\nclass = No'),
 Text(3348.0, 1522.0800000000002, 'User ID <= 267004.0\ngini = 0.5\nsamples =
14999\nvalue = [11713, 11942]\nclass = No'),
 Text(3124.7999999999997, 1087.2, 'gini = 0.337\nsamples = 22\nvalue = [9, 3
3]\nclass = No'),
 Text(3571.2, 1087.2, 'Follower Count <= 6736.5\ngini = 0.5\nsamples = 14977
\nvalue = [11704, 11909]\nclass = No'),
 Text(3124.7999999999997, 652.3200000000002, 'Follower Count <= 6298.5\ngini
= 0.5\nsamples = 10051\nvalue = [7738, 8096]\nclass = No'),
 Text(2901.6, 217.44000000000005, 'gini = 0.5\nsamples = 9403\nvalue = [7295,
7521]\nclass = No'),
 Text(3348.0, 217.44000000000005, 'gini = 0.492\nsamples = 648\nvalue = [443,
575]\nclass = No'),
 Text(4017.6, 652.3200000000002, 'Mention Count <= 0.5\ngini = 0.5\nsamples =
4926\nvalue = [3966, 3813]\nclass = Yes'),
 Text(3794.3999999999996, 217.44000000000005, 'gini = 0.499\nsamples = 805\nv
alue = [605, 657]\nclass = No'),
 Text(4240.8, 217.44000000000005, 'gini = 0.5\nsamples = 4121\nvalue = [3361,
3156]\nclass = Yes')]
```


User ID <= 265748.5
gini = 0.5
samples = 18485
value = [14594, 14567]
class = Yes
Retweet Count <= 98.5
gini = 0.499
samples = 3486
value = [2881, 2625]
class = Yes
User ID <= 267004.0
gini = 0.5
samples = 14999
value = [11713, 11942]
class = No
User ID <= 181493.5
gini = 0.499
samples = 3423
value = [2847, 2563]
class = Yes
User ID <= 152333.0
gini = 0.457
samples = 63
value = [34, 62]
class = No
gini = 0.337
samples = 22
value = [9, 33]
class = No
Follower Count <= 6736.5
gini = 0.5
samples = 14977
value = [11704, 11909]
class = No
User ID <= 167484.5
gini = 0.5
samples = 1682
value = [1318, 1340]
class = No
User ID <= 253606.0
gini = 0.494
samples = 1741
value = [1529, 1223]
class = Yes
gini = 0.5
samples = 24
value = [22, 21]
class = Yes
Follower Count <= 2988.5
gini = 0.35
samples = 39
value = [12, 41]
class = No
Follower Count <= 6298.5
gini = 0.5
samples = 10051
value = [7738, 8096]
class = No
Mention Count <= 0.5
gini = 0.5
samples = 4926
value = [3966, 3813]
class = Yes
gini = 0.499
samples = 1375
value = [1132, 1044]
class = Yes
gini = 0.474
samples = 307
value = [186, 296]
class = No
gini = 0.496
samples = 1484
value = [1278, 1061]
class = Yes
gini = 0.477
samples = 257
value = [251, 162]
class = Yes
gini = 0.457
samples = 15
value = [6, 11]
class = No
gini = 0.278
samples = 24
value = [6, 30]
class = No
gini = 0.5
samples = 9403
value = [7295, 7521]
class = No
gini = 0.492
samples = 648
value = [443, 575]
class = No
gini = 0.499
samples = 805
value = [605, 657]
class = No
gini = 0.5
samples = 4121
value = [3361, 3156]
class = Yes