

CaixaBank Tech Hackaton

1. Tweet dataset

	handle	text	year	month	day
0	abelac62	He hecho el repaso de todos los componentes de...	2022.0	4.0	9.0
1	LluisPerarnau	Els projectes que han presentat les empreses d...	2022.0	4.0	7.0
2	Pegaso121080	Por si no lo has visto, o no lo encuentras en ...	2022.0	4.0	4.0
3	zonavalue	📈 #BOLSA: El #Ibex35 abre en 🟢 \n\nes #Ibex35 ...	2022.0	4.0	5.0
4	EPeconomia	El #Ibex35 retrocede un 0,4% en marzo y un 3,0...	2022.0	3.0	31.0

“El análisis de sentimientos con la librería NLTK puede conseguir un puntaje para los tweets y usar ese score como una variable mas para el DataFrame original, (se combinan con las columnas year, month y day)”

	Open	High	Low	Close	Adj Close	Volume	Target	year	month	day	score
9829	9113.200195	9116.700195	9035.099609	9080.500000	9080.500000	148987100.0	0	2019	5	29	0.7003
9830	9120.799805	9175.200195	9114.099609	9157.799805	9157.799805	101389200.0	0	2019	5	30	0.0000
9831	9120.799805	9175.200195	9114.099609	9157.799805	9157.799805	101389200.0	0	2019	5	30	0.3612
9832	9120.799805	9175.200195	9114.099609	9157.799805	9157.799805	101389200.0	0	2019	5	30	0.5664
9833	9120.799805	9175.200195	9114.099609	9157.799805	9157.799805	101389200.0	0	2019	5	30	-0.2960

CaixaBank Tech Hackaton

2. Repetidos y NaN values

“Hay varios twits en un solo día, por lo que se calcula el promedio del score detectado por NLTK”
“Se eliminan las filas con valores NaN”

3. Entrenamiento

Se prueban 6 modelos para la clasificación

```
F1-score Logit      : 0.694
F1-score DecTree   : 0.957
F1-score RFC       : 0.960
F1-score AdaBoostC : 0.723
F1-score XGBoost    : 0.827
F1-score LightGBM  : 0.949
```

Se utilizan la combinación de DecisionTreeClassifier, RandomForest y LGBM classifier Para tener un F1-score final de 0.962

```
from sklearn.ensemble import VotingClassifier
from sklearn.metrics import f1_score

estimators = [
    ("DecTree", DecisionTreeClassifier()),
    ("RFC", RandomForestClassifier()),
    ("LightGBM", LGBMClassifier())
]

vc = VotingClassifier(estimators=estimators, voting="hard")
vc.fit(X_train, y_train)
y_pred = vc.predict(X_test)
f1 = f1_score(y_test, y_pred, average="macro")
print(f"F1-score: {f1:.3f}")
```

F1-score: 0.962