CODE

```
import pandas as pd
import re
train=pd.read_csv("train.csv")
train.head()
train.drop("id",inplace=True ,axis=1)

import nltk
nltk.download()

from nltk.stem import PorterStemmer
stemmer = PorterStemmer()
def clean_sentences(text):
    text = text.lower()
    text = re.sub(r"[^a-z0-9^,!.\/']", " ", text)
    text = " ".join(text.split())
    text = " ".join(stemmer.stem(word) for word in text.split())
    return text
x = train['tweet']
y = train['label']
x = x.map(lambda a: clean_sentences(a))
x.head()

from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x,y,stratify=y,random_state=42)

x_train.head()

from sklearn.feature_extraction.text import TfidfVectorizer
vectorizer = TfidfVectorizer(stop_words='english')
x_train = vectorizer.fit_transform(x_train)
x_test = vectorizer.transform(x_test)
print(x_test)
from sklearn.svm import LinearSVC
model = LinearSVC(C=1.02, tol=0.3)
model.fit(x_train,y_train)

from sklearn.metrics import confusion_matrix, accuracy_score, precision_score,
f1_score,recall_score

print(confusion_matrix(y_test,model.predict(x_test)))
print(accuracy_score(y_test,model.predict(x_test)))
print(recall_score(y_test,model.predict(x_test)))
print(precision_score(y_test,model.predict(x_test)))
print(f1_score(y_test,model.predict(x_test)))
```

OUTPUT

Confusion Matrix: [[7369 61]
[228 333]]
Accuracy: 0.9638343136028031
Recall: 0.5935828877005348
Precision: 0.8451776649746193
F1 Score: 0.6973821989528797