Hotel review sentiment analysis

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
import pandas as pd
df = pd.read_csv('/content/drive/MyDrive/Python_dataset/train.csv')
```

```
df.shape
```

```
(38932, 5)
```

```
df.head()
```

	User_ID	Description	Browser_Used	Device_Used	Is_Response
0	id10326	The room was kind of clean but had a VERY stro...	Edge	Mobile	not happy
1	id10327	I stayed at the Crown Plaza April -- - April -...	Internet Explorer	Mobile	not happy
2	id10328	I booked this hotel through Hotwire at the low...	Mozilla	Tablet	not happy
3	id10329	Stayed here with husband and sons on the way t...	InternetExplorer	Desktop	happy
4	id10330	My girlfriends and I stayed here to celebrate ...	Edge	Tablet	not happy

```
#Removing columns
df.drop(columns = ['User_ID', 'Browser_Used', 'Device_Used'], inplace = True)
```

```
# clean dataset
```

```
import re
import string

#This function converts to lower-case, removes numbers and punctuation,remove square brackets
def text_clean_1(text):
    text = text.lower()
    text = re.sub('\[.*?\]', '', text)
    text = re.sub('[%s]' % re.escape(string.punctuation), '', text)
    text = re.sub('\w*\d\w*', '', text)
    return text

cleaned1 = lambda x: text_clean_1(x)


# Let's take a look at the updated text
df['cleaned_description'] = pd.DataFrame(df.Description.apply(cleaned1))
df.head(10)
```

	Description	Is_Response	cleaned_description
0	The room was kind of clean but had a VERY stro...	not happy	the room was kind of clean but had a very stro...
1	I stayed at the Crown Plaza April -- - April -...	not happy	i stayed at the crown plaza april april th...
2	I booked this hotel through Hotwire at the low...	not happy	i booked this hotel through hotwire at the low...
3	Stayed here with husband and sons on the way t...	happy	stayed here with husband and sons on the way t...
4	My girlfriends and I stayed here to celebrate ...	not happy	my girlfriends and i stayed here to celebrate ...
5	We had - rooms. One was very nice and clearly ...	happy	we had rooms one was very nice and clearly ha...
6	My husband and I have stayed in this hotel a f...	not happy	my husband and i have stayed in this hotel a f...
7	My wife & I stayed in this glorious city a whi...	happy	my wife i stayed in this glorious city a whil...
8	My boyfriend and I stayed at the Fairmont on a...	happy	my boyfriend and i stayed at the fairmont on a...
9	Wonderful staff, great location, but it was de...	not happy	wonderful staff great location but it was defi...

```
# cleaning quotes and new line
def text_clean_2(text):
```

```
    text = re.sub('[‘’“”…]', '', text)
    text = re.sub('\n', '', text)
    return text

cleaned2 = lambda x: text_clean_2(x)


df['cleaned_description_new'] = pd.DataFrame(df['cleaned_description'].apply(cleaned2))
df.head(10)
```

	Description	Is_Response	cleaned_description	cleaned_description_new
0	The room was kind of clean but had a VERY stro...	not happy	the room was kind of clean but had a very stro...	the room was kind of clean but had a very stro...
1	I stayed at the Crown Plaza April -- - April -...	not happy	i stayed at the crown plaza april april th...	i stayed at the crown plaza april april th...
2	I booked this hotel through Hotwire at the low...	not happy	i booked this hotel through hotwire at the low...	i booked this hotel through hotwire at the low...
3	Stayed here with husband and sons on the way t...	happy	stayed here with husband and sons on the way t...	stayed here with husband and sons on the way t...
4	My girlfriends and I stayed here to celebrate ...	not happy	my girlfriends and i stayed here to celebrate ...	my girlfriends and i stayed here to celebrate ...
5	We had - rooms. One was very nice and clearly ...	happy	we had rooms one was very nice and clearly ha...	we had rooms one was very nice and clearly ha...
	My husband and I have stayed in this		my husband and i have stayed in this	my husband and i have stayed in this

```
from sklearn.model_selection import train_test_split

Independent_var = df.cleaned_description_new
Dependent_var = df.Is_Response
X_train, X_test, Y_train, Y_test = train_test_split(Independent_var, Dependent_var, test_size = 0.1, random_state = 225)



from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
```

```
tvec = TfidfVectorizer()
clf2 = LogisticRegression()



from sklearn.pipeline import Pipeline
# In pipeline first perform vectorization and next perform classification(similar to dense layer)
model = Pipeline([('vectorizer',tvec),('classifier',clf2)])
model.fit(X_train, Y_train)

from sklearn.metrics import confusion_matrix



predictions = model.predict(X_test)

confusion_matrix(predictions, Y_test)

     array([[2418,  305],
            [ 153, 1018]])


# Model Prediction

from sklearn.metrics import accuracy_score, precision_score, recall_score

print("Accuracy : ", accuracy_score(predictions, Y_test))
print("Precision : ", precision_score(predictions, Y_test, average = 'weighted'))
print("Recall : ", recall_score(predictions, Y_test, average = 'weighted'))

     Accuracy :  0.8823831535695943
     Precision :  0.8890590818181386
     Recall :  0.8823831535695943


# Trying on new review
example = ["I'm  sad"]
result = model.predict(example)

print(result)
```

```
['not happy']
```

```
# Trying on new review
example = ["I'm delight"]
result = model.predict(example)

print(result)
```

```
['happy']
```