Machine learning

Q-8. Quora question pair similarity, you need to find the Similarity between two questions by mapping the words in the questions using TF-IDF, and using a supervised Algorithm you need to find the similarity between the questions.

Dataset This is the Dataset You can use this dataset for this question.

In [1]:

```
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
import warnings
warnings.filterwarnings('ignore')
```

Loading the dataset

In [6]:

```
data = pd.read_csv('Downloads/train.csv/train.csv')  # Update the path to your dataset file
```

In [7]:

```
data.head()
```

Out[7]:

	id	qid1	qid2	question1	question2	is_duplicate
0	0	1	2	What is the step by step guide to invest in sh...	What is the step by step guide to invest in sh...	0
1	1	3	4	What is the story of Kohinoor (Koh-i-Noor) Dia...	What would happen if the Indian government sto...	0
2	2	5	6	How can I increase the speed of my internet co...	How can Internet speed be increased by hacking...	0
3	3	7	8	Why am I mentally very lonely? How can I solve...	Find the remainder when [math]23^{24} [/math] i...	0
4	4	9	10	Which one dissolve in water quikly sugar, salt...	Which fish would survive in salt water?	0

In [8]:

```
data.shape
```

Out[8]:

```
(404290, 6)
```

In [9]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 404290 entries, 0 to 404289
Data columns (total 6 columns):
 #   Column        Non-Null Count   Dtype 
---  ------        --------------   ----- 
 0   id            404290 non-null  int64 
 1   qid1          404290 non-null  int64 
 2   qid2          404290 non-null  int64 
 3   question1     404289 non-null  object
 4   question2     404288 non-null  object
 5   is_duplicate  404290 non-null  int64 
dtypes: int64(4), object(2)
memory usage: 18.5+ MB
```

Preprocess the data

In [10]:

```
data.isnull().sum()
```

Out[10]:

```
id              0
qid1            0
qid2            0
question1       1
question2       2
is_duplicate    0
dtype: int64
```

In [11]:

```
# Drop rows with missing values
data.dropna(inplace=True)

# Split the data into question pairs and labels
questions = data[['question1', 'question2']]
labels = data['is_duplicate']

```

Split the data into training and testing sets

In [12]:

```
questions_train, questions_test, labels_train, labels_test = train_test_split(questions, labels, te

```

Apply TF-IDF transformation on the training data

In [13]:

```
tfidf = TfidfVectorizer()
tfidf_train = tfidf.fit_transform(questions_train['question1'] + ' ' + questions_train['question2']
```

Train a supervised algorithm (Logistic Regression)

In [16]:

```
model = LogisticRegression()
model.fit(tfidf_train, labels_train)
```

Out[16]:

```
LogisticRegression()
```

```
 In a Jupyter environment, please rerun this cell to show the HTML representation or trust the
notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with
nbviewer.org.
```

Apply TF-IDF transformation on the testing data and predict similarity

In [17]:

```
tfidf_test = tfidf.transform(questions_test['question1'] + ' ' + questions_test['question2'])
predictions = model.predict(tfidf_test)
```

Evaluate the model

In [18]:

```
accuracy = accuracy_score(labels_test, predictions)
print("Accuracy:", accuracy)
```

```
Accuracy: 0.7549654950654233
```

In [ ]:

```

```