Q4 - Naive Bayes Classifier (EMAIL Dataset) - Detailed Answers

Problem statement

Use Naive Bayes classification algorithms to build a classifier for EMAIL dataset. Pre-process the dataset as required. Compute Accuracy, Precision, Recall and F1 measure.

Why Naive Bayes

MultinomialNB works well for discrete counts like TF or TF-IDF transformed text. It is fast and often a strong baseline for spam detection.

Sample Python code

```
# Naive Bayes (Multinomial) on EMAIL dataset
from sklearn.naive_bayes import MultinomialNB
from sklearn.pipeline import make_pipeline
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics import classification_report
import pandas as pd

df = pd.read_csv('emails.csv')  # columns: 'text','label'
X_train, X_test, y_train, y_test = train_test_split(df['text'], df['label'], test_size=0.2, stra
pipeline = make_pipeline(TfidfVectorizer(max_df=0.95, min_df=2), MultinomialNB())
pipeline.fit(X_train, y_train)
y_pred = pipeline.predict(X_test)
print(classification_report(y_test, y_pred, digits=4))
```

Expected outputs & interpretation

Report accuracy, precision, recall, and F1. Compare with Decision Tree and k-NN. Typically MultinomialNB yields good recall for spam with proper preprocessing. Discuss confusion matrix and common errors.