PROBLEM 4 : L1 feature selection on text

In [1]:
```
from sklearn.datasets import fetch_20newsgroups
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import Lasso
from sklearn.naive_bayes import MultinomialNB
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
import numpy as np

# Load 20NG dataset
newsgroups = fetch_20newsgroups(subset='all')
X, y = newsgroups.data, newsgroups.target

# Split data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, ra

# Preprocess using TF-IDF vectorization
vectorizer = TfidfVectorizer(stop_words='english', max_df=0.5)
X_train_vec = vectorizer.fit_transform(X_train)
X_test_vec = vectorizer.transform(X_test)

# L1-regularized regression (Lasso)
alpha_value = 0.01  # Adjust this value as needed to obtain the desired spa
lasso = Lasso(alpha=alpha_value)
lasso.fit(X_train_vec, y_train)

# Select top 200 features based on regression coefficients
coef_abs = np.abs(lasso.coef_)
top_indices = coef_abs.argsort()[-200:][::-1]
X_train_reduced = X_train_vec[:, top_indices]
X_test_reduced = X_test_vec[:, top_indices]

# Run a classification task
clf = MultinomialNB()
clf.fit(X_train_reduced, y_train)
predictions = clf.predict(X_test_reduced)
accuracy = accuracy_score(y_test, predictions)
print(f"Accuracy using top 200 features from L1-regularized regression: {ac
```

```
Accuracy using top 200 features from L1-regularized regression: 0.2724
```