Problem Statement

Classification of Articles into One of 5 categories
(Sports,Business,Technology,Politics,Entertainment)

Importing Libraries

```
import pandas as pd
import matplotlib.pyplot as plt
import re
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.stem import WordNetLemmatizer
from sklearn.preprocessing import OrdinalEncoder
from sklearn.feature_extraction.text import CountVectorizer,
TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import classification_report, confusion_matrix,
roc_auc_score, precision_score, recall_score, f1_score
import seaborn as sns
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.multiclass import OneVsRestClassifier
from sklearn.preprocessing import label_binarize
from sklearn.metrics import roc_curve, auc

# To ignore all warnings
import warnings

# For reading & manipulating the data
import pandas as pd
import numpy as np

# For visualizing the data

import matplotlib.pyplot as plt
import seaborn as sns

# To use Regular Expressions
import re

# To use Natural Language Processing
import nltk

# For tokenization
from nltk.tokenize import word_tokenize
nltk.download('punkt')
```

```
# To remove stopwords
from nltk.corpus import stopwords
nltk.download('stopwords')

# For lemmetization
from nltk import WordNetLemmatizer
nltk.download('wordnet')

# For BoW & TF-IDF
from sklearn.feature_extraction.text import CountVectorizer,
TfidfVectorizer

# For encoding the categorical variable

import category_encoders as ce

# To try out different ML models
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import MultinomialNB
from sklearn.ensemble import RandomForestClassifier

# To perform train-test split
from sklearn.model_selection import train_test_split

# Performace Metrics for evaluating the model
from sklearn.metrics import accuracy_score, roc_auc_score, f1_score,
precision_score, recall_score
from sklearn.metrics import confusion_matrix, classification_report

warnings.simplefilter('ignore')
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\Sharat\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\Sharat\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\Sharat\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

```
---------------------------------------------------------------------
-----
ModuleNotFoundError                       Traceback (most recent call
last)
Cell In[33], line 36
     32 from sklearn.feature_extraction.text import CountVectorizer,
TfidfVectorizer
     34 # For encoding the categorical variable
```

```
---> 36 import category_encoders as ce
     38 # To try out different ML models
     39 from sklearn.tree import DecisionTreeClassifier

ModuleNotFoundError: No module named 'category_encoders'
```

Loading Dataset

```
# Load the dataset

df = pd.read_csv("flipitnews-data.csv")

# Display the first few rows of the dataset
df.head()
```

```
         Category                                            Article
0      Technology  tv future in the hands of viewers with home th...
1        Business  worldcom boss  left books alone  former worldc...
2          Sports  tigers wary of farrell  gamble  leicester say ...
3          Sports  yeading face newcastle in fa cup premiership s...
4   Entertainment  ocean s twelve raids box office ocean s twelve...
```

Exploring the Dataset

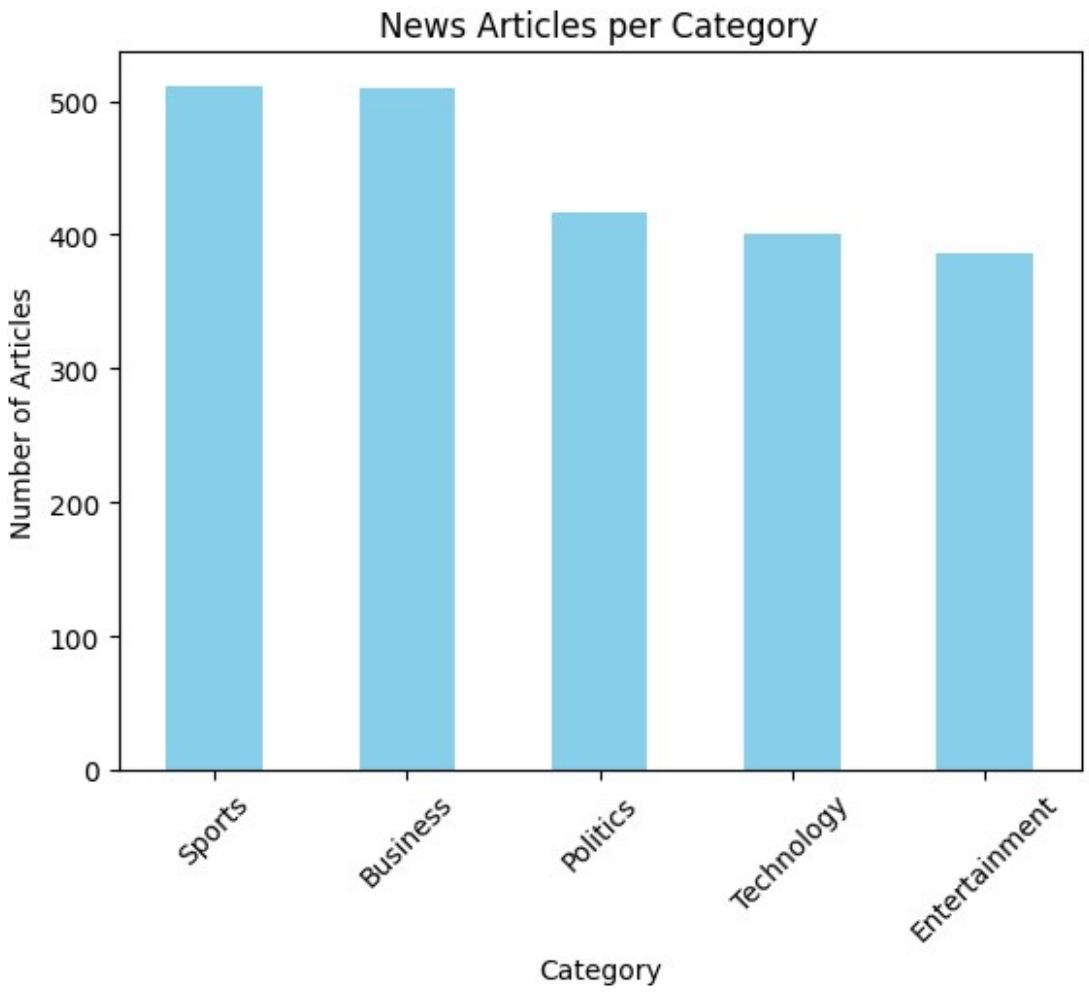
```
df.shape
```

```
(2225, 2)
```

```
# News articles per category
category_distribution = df['Category'].value_counts()
print("News articles per category:")
print(category_distribution)
```

```
News articles per category:
Category
Sports           511
Business         510
Politics         417
Technology       401
Entertainment    386
Name: count, dtype: int64
```

```
# Plot category distribution
category_distribution.plot(kind='bar', color='skyblue')
plt.title('News Articles per Category')
plt.xlabel('Category')
plt.ylabel('Number of Articles')
plt.xticks(rotation=45)
plt.show()
```




```
df["Article"][2]
```

```
'tigers wary of farrell  gamble  leicester say they will not be rushed
into making a bid for andy farrell should the great britain rugby
league captain decide to switch codes.   we and anybody else involved
in the process are still some way away from going to the next stage
tigers boss john wells told bbc radio leicester.  at the moment  there
are still a lot of unknowns about andy farrell  not least his medical
situation.  whoever does take him on is going to take a big  big
gamble.  farrell  who has had persistent knee problems  had an
operation on his knee five weeks ago and is expected to be out for
another three months. leicester and saracens are believed to head the
list of rugby union clubs interested in signing farrell if he decides
to move to the 15-man game.  if he does move across to union  wells
believes he would better off playing in the backs  at least initially.
i m sure he could make the step between league and union by being
involved in the centre   said wells.  i think england would prefer him
to progress to a position in the back row where they can make use of
some of his rugby league skills within the forwards.  the jury is out
```

```
on whether he can cross that divide.  at this club  the balance will
have to be struck between the cost of that gamble and the option of
bringing in a ready-made replacement.'
```

Text Processing

```
import re
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.stem import WordNetLemmatizer

# Download necessary NLTK data
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('wordnet')

# Initialize Lemmatizer
lemmatizer = WordNetLemmatizer()

# User defined function to process text
def process_text(text):
    # Remove non-letters
    text = re.sub("[^a-zA-Z]", " ", text)

    # Tokenize text
    words = word_tokenize(text.lower())

    # Remove stopwords
    words = [word for word in words if word not in
stopwords.words('english')]

    # Lemmatize words
    words = [lemmatizer.lemmatize(word) for word in words]

    return " ".join(words)

# Apply the function to the 'Article' column
df['Processed_Article'] = df['Article'].apply(process_text)

# Display a single news article before and after processing
print("Original Article:", df['Article'][0])
print("\n\n\nProcessed Article:", df['Processed_Article'][0])
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\Sharat\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\Sharat\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

```
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\Sharat\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

Original Article: tv future in the hands of viewers with home theatre systems plasma high-definition tvs and digital video recorders moving into the living room the way people watch tv will be radically different in five years time. that is according to an expert panel which gathered at the annual consumer electronics show in las vegas to discuss how these new technologies will impact one of our favourite pastimes. with the us leading the trend programmes and other content will be delivered to viewers via home networks through cable satellite telecoms companies and broadband service providers to front rooms and portable devices. one of the most talked-about technologies of ces has been digital and personal video recorders (dvr and pvr). these set-top boxes like the us s tivo and the uk s sky+ system allow people to record store play pause and forward wind tv programmes when they want. essentially the technology allows for much more personalised tv. they are also being built-in to high-definition tv sets which are big business in japan and the us but slower to take off in europe because of the lack of high-definition programming. not only can people forward wind through adverts they can also forget about abiding by network and channel schedules putting together their own a-la-carte entertainment. but some us networks and cable and satellite companies are worried about what it means for them in terms of advertising revenues as well as brand identity and viewer loyalty to channels. although the us leads in this technology at the moment it is also a concern that is being raised in europe particularly with the growing uptake of services like sky+. what happens here today we will see in nine months to a years time in the uk   adam hume the bbc broadcast s futurologist told the bbc news website. for the likes of the bbc there are no issues of lost advertising revenue yet. it is a more pressing issue at the moment for commercial uk broadcasters but brand loyalty is important for everyone. we will be talking more about content brands rather than network brands   said tim hanlon from brand communications firm starcom mediavest. the reality is that with broadband connections anybody can be the producer of content. he added: the challenge now is that it is hard to promote a programme with so much choice.   what this means said stacey jolna senior vice president of tv guide tv group is that the way people find the content they want to watch has to be simplified for tv viewers. it means that networks in us terms or channels could take a leaf out of google s book and be the search engine of the future instead of the scheduler to help people find what they want to watch. this kind of channel model might work for the younger ipod generation which is used to taking control of their gadgets and what they play on them. but it might not suit everyone the panel recognised. older generations are more comfortable with familiar schedules and channel brands because they know what they are getting. they perhaps do not want so much of

the choice put into their hands mr hanlon suggested. on the other end you have the kids just out of diapers who are pushing buttons already - everything is possible and available to them   said mr hanlon. ultimately the consumer will tell the market they want.   of the 50 000 new gadgets and technologies being showcased at ces many of them are about enhancing the tv-watching experience. high-definition tv sets are everywhere and many new models of lcd (liquid crystal display) tvs have been launched with dvr capability built into them instead of being external boxes. one such example launched at the show is humax s 26-inch lcd tv with an 80-hour tivo dvr and dvd recorder. one of the us s biggest satellite tv companies directtv has even launched its own branded dvr at the show with 100-hours of recording capability instant replay and a search function. the set can pause and rewind tv for up to 90 hours. and microsoft chief bill gates announced in his pre-show keynote speech a partnership with tivo called tivotogo which means people can play recorded programmes on windows pcs and mobile devices. all these reflect the increasing trend of freeing up multimedia so that people can watch what they want when they want.

Processed Article: tv future hand viewer home theatre system plasma high definition tv digital video recorder moving living room way people watch tv radically different five year time according expert panel gathered annual consumer electronics show la vega discus new technology impact one favourite pastime u leading trend programme content delivered viewer via home network cable satellite telecom company broadband service provider front room portable device one talked technology ce digital personal video recorder dvr pvr set top box like u tivo uk sky system allow people record store play pause forward wind tv programme want essentially technology allows much personalised tv also built high definition tv set big business japan u slower take europe lack high definition programming people forward wind advert also forget abiding network channel schedule putting together la carte entertainment u network cable satellite company worried mean term advertising revenue well brand identity viewer loyalty channel although u lead technology moment also concern raised europe particularly growing uptake service like sky happens today see nine month year time uk adam hume bbc broadcast futurologist told bbc news website like bbc issue lost advertising revenue yet pressing issue moment commercial uk broadcaster brand loyalty important everyone talking content brand rather network brand said tim hanlon brand communication firm starcom mediavest reality broadband connection anybody producer content added challenge hard promote programme much choice mean said stacey jolna senior vice president tv guide tv group way people find content want watch simplified tv viewer mean network u term channel could take leaf google book search engine future instead scheduler help people find want watch kind channel model might work younger ipod generation used taking control gadget

```
play might suit everyone panel recognised older generation comfortable
familiar schedule channel brand know getting perhaps want much choice
put hand mr hanlon suggested end kid diaper pushing button already
everything possible available said mr hanlon ultimately consumer tell
market want new gadget technology showcased ce many enhancing tv
watching experience high definition tv set everywhere many new model
lcd liquid crystal display tv launched dvr capability built instead
external box one example launched show humax inch lcd tv hour tivo dvr
dvd recorder one u biggest satellite tv company directtv even launched
branded dvr show hour recording capability instant replay search
function set pause rewind tv hour microsoft chief bill gate announced
pre show keynote speech partnership tivo called tivotogo mean people
play recorded programme window pc mobile device reflect increasing
trend freeing multimedia people watch want want
```

Encoding And Transforming The Data

```
from sklearn.preprocessing import OrdinalEncoder

# Encode the target variable
ordinal_encoder = OrdinalEncoder()
df['Category'] = ordinal_encoder.fit_transform(df[['Category']])



df
```

```
      Category                                            Article  \
0          4.0  tv future in the hands of viewers with home th...
1          0.0  worldcom boss  left books alone  former worldc...
2          3.0  tigers wary of farrell  gamble  leicester say ...
3          3.0  yeading face newcastle in fa cup premiership s...
4          1.0  ocean s twelve raids box office ocean s twelve...
...        ...                                                ...
2220       0.0  cars pull down us retail figures us retail sal...
2221       2.0  kilroy unveils immigration policy ex-chatshow ...
2222       1.0  rem announce new glasgow concert us band rem h...
2223       2.0  how political squabbles snowball it s become c...
2224       3.0  souness delight at euro progress boss graeme s...

                                      Processed_Article
0     tv future hand viewer home theatre system plas...
1     worldcom bos left book alone former worldcom b...
2     tiger wary farrell gamble leicester say rushed...
3     yeading face newcastle fa cup premiership side...
4     ocean twelve raid box office ocean twelve crim...
...                                                 ...
2220  car pull u retail figure u retail sale fell ja...
2221  kilroy unveils immigration policy ex chatshow ...
2222  rem announce new glasgow concert u band rem an...
```

```
2223  political squabble snowball become commonplace...
2224  souness delight euro progress bos graeme soune...

[2225 rows x 3 columns]
```

Choice between TF-IDF and BOW

```
# Function to vectorize data
def vectorize_data(method='tfidf'):
    if method == 'bow':
        vectorizer = CountVectorizer()
    elif method == 'tfidf':
        vectorizer = TfidfVectorizer()
    else:
        raise ValueError("Method should be 'bow' or 'tfidf'")

    X = vectorizer.fit_transform(df['Processed_Article'])
    return X

# Vectorize using TF-IDF (example)
X = vectorize_data(method='tfidf')
y = df['Category']
```

Function to use different models and checking their Classification Metrics

```
# Binarize the labels for ROC AUC
y_bin = label_binarize(y, classes=[0, 1, 2, 3, 4])
n_classes = y_bin.shape[1]

# Split the data
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)
y_train_bin = label_binarize(y_train, classes=[0, 1, 2, 3, 4])
y_test_bin = label_binarize(y_test, classes=[0, 1, 2, 3, 4])

# Function to train and evaluate model
def train_and_evaluate(model, X_train, y_train, X_test, y_test,
y_test_bin):
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    y_score = model.predict_proba(X_test)

    # Evaluate the model
    print(f"{model.__class__.__name__} Classification Report:")
    print(classification_report(y_test, y_pred))
    print(f"{model.__class__.__name__} Confusion Matrix:")
    print(confusion_matrix(y_test, y_pred))
    print(f"{model.__class__.__name__} ROC AUC Score:
{roc_auc_score(y_test_bin, y_score, multi_class='ovr')}")
```

```
    print(f"{model.__class__.__name__} Precision Score:
{precision_score(y_test, y_pred, average='weighted')}")
    print(f"{model.__class__.__name__} Recall Score:
{recall_score(y_test, y_pred, average='weighted')}")
    print(f"{model.__class__.__name__} F1 Score: {f1_score(y_test,
y_pred, average='weighted')}")

    # Confusion Matrix
    conf_matrix = confusion_matrix(y_test, y_pred)
    sns.heatmap(conf_matrix, annot=True, fmt='d')
    plt.title(f'{model.__class__.__name__} Confusion Matrix')
    plt.show()

    # ROC AUC
    plot_roc_auc(y_test_bin, y_score, f'{model.__class__.__name__} ROC
AUC')


X_test.shape[0]
```

```
445
```

```
X_train.shape[0]
```

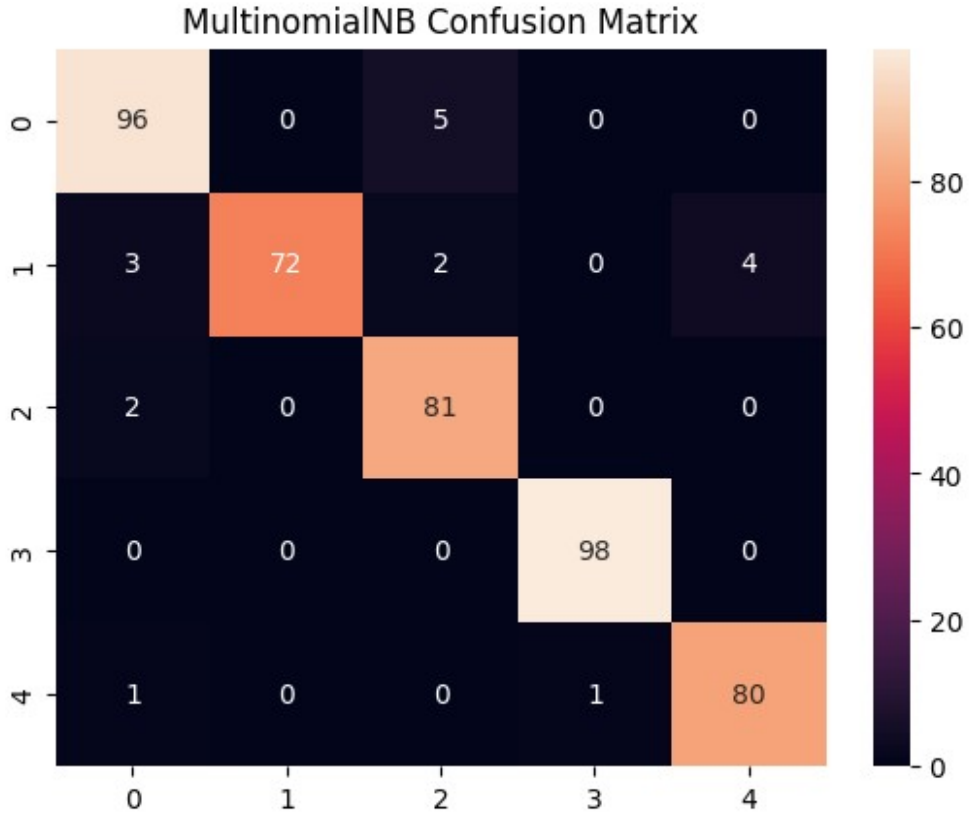
```
1780
```

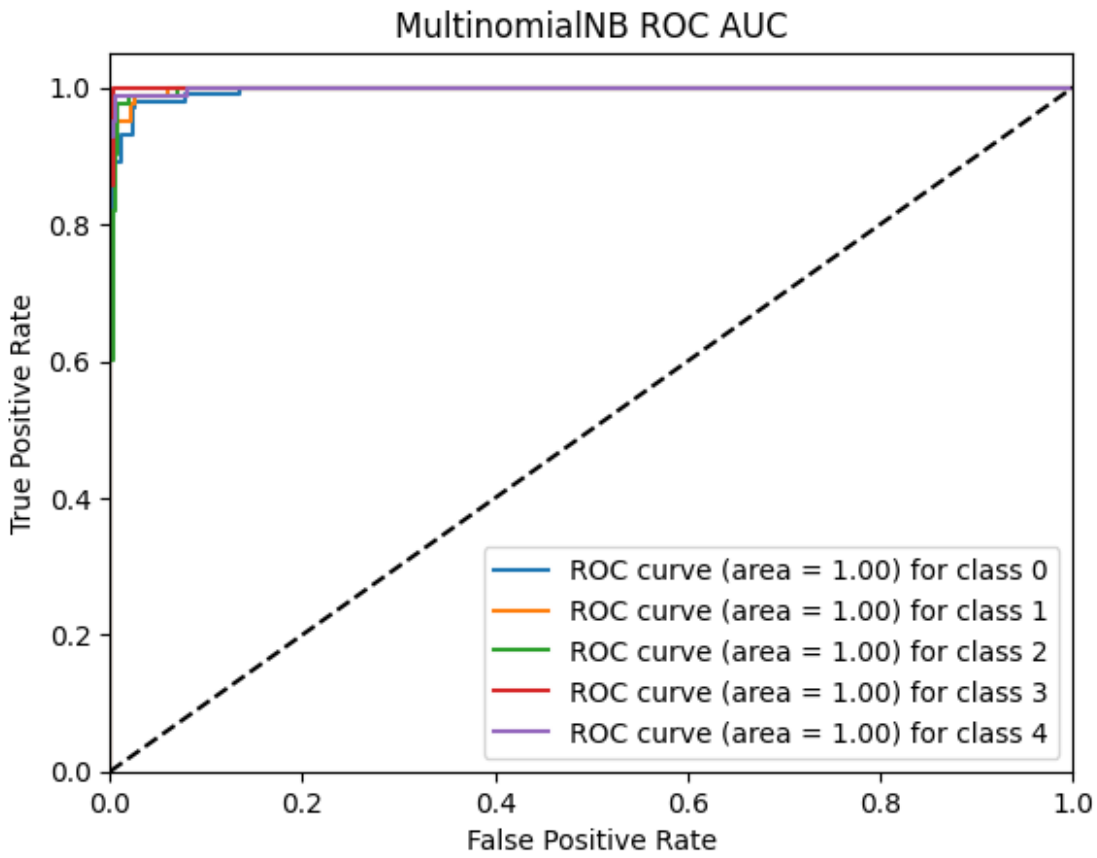
Naive-Bayes Model

```
from sklearn.naive_bayes import MultinomialNB

# Train and evaluate models

# Naive Bayes
nb_model = MultinomialNB()
train_and_evaluate(nb_model, X_train, y_train, X_test, y_test,
y_test_bin)
```

```
MultinomialNB Classification Report:
              precision    recall  f1-score   support

         0.0       0.94      0.95      0.95       101
         1.0       1.00      0.89      0.94        81
         2.0       0.92      0.98      0.95        83
         3.0       0.99      1.00      0.99        98
         4.0       0.95      0.98      0.96        82

    accuracy                           0.96       445
   macro avg       0.96      0.96      0.96       445
weighted avg       0.96      0.96      0.96       445
```

```
MultinomialNB Confusion Matrix:
[[96  0  5  0  0]
 [ 3 72  2  0  4]
 [ 2  0 81  0  0]
 [ 0  0  0 98  0]
 [ 1  0  0  1 80]]
MultinomialNB ROC AUC Score: 0.997989677299946
MultinomialNB Precision Score: 0.9608132357471417
MultinomialNB Recall Score: 0.9595505617977528
MultinomialNB F1 Score: 0.9593992119336041
```



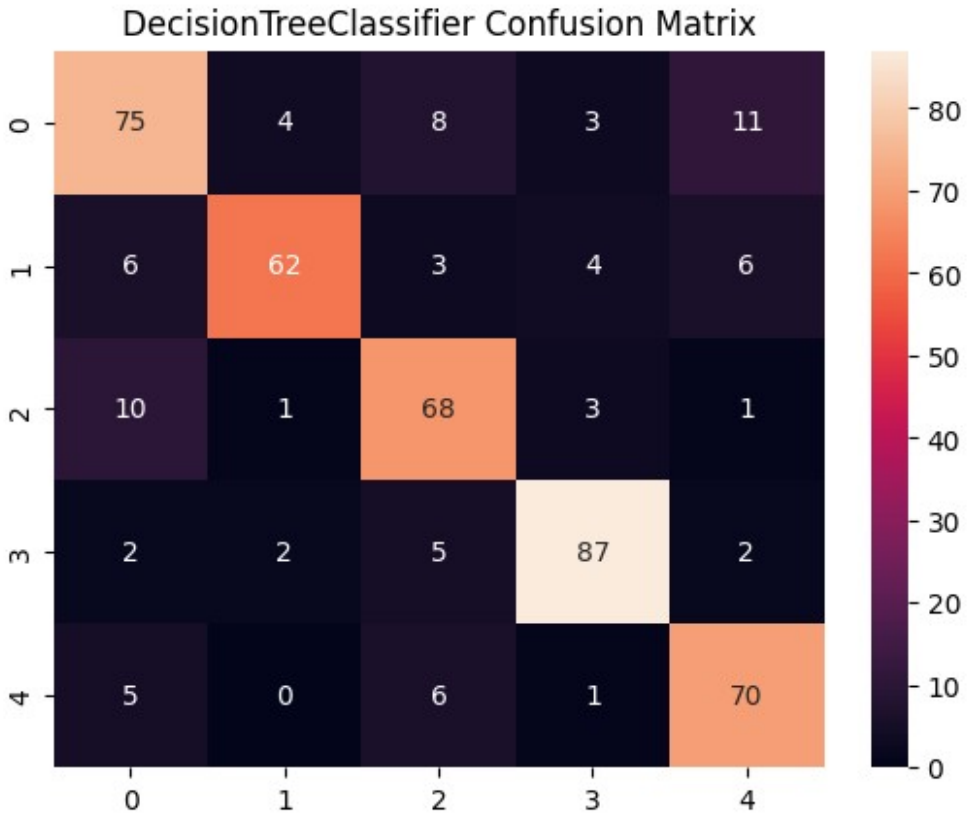


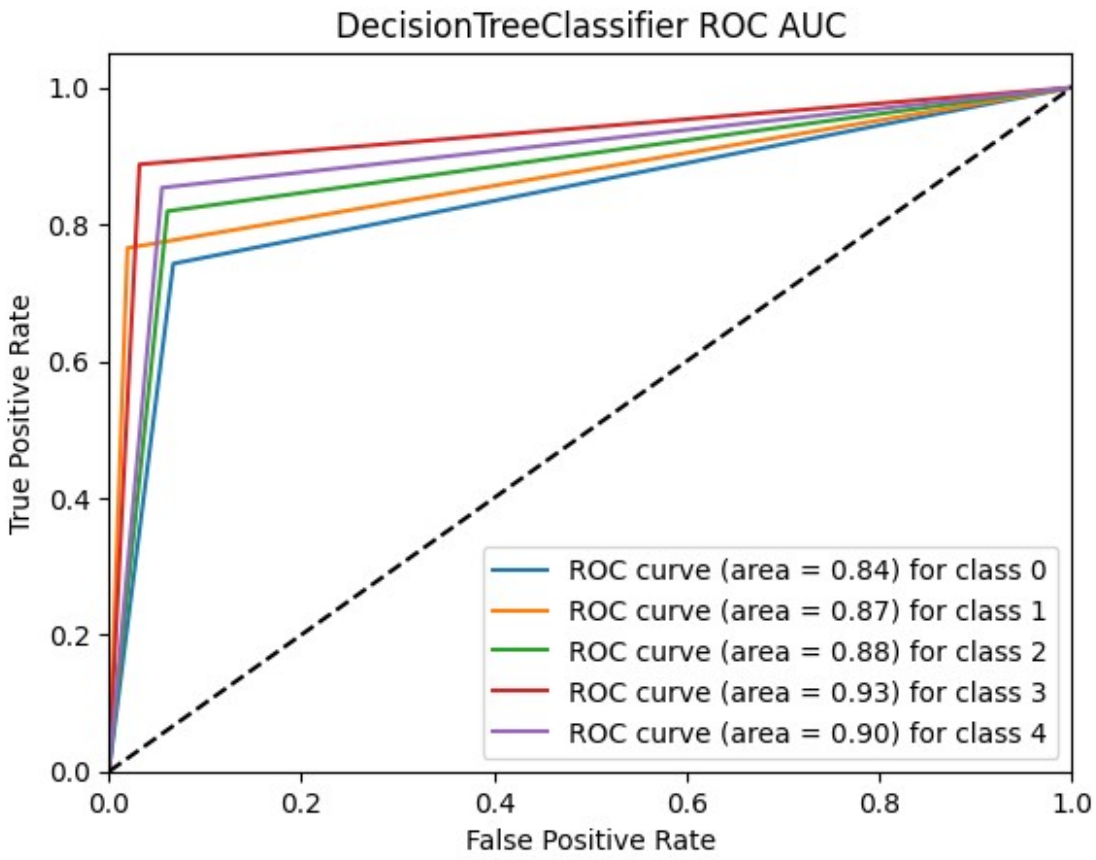
Decision Tree Model

```
# Decision Tree
dt_model = DecisionTreeClassifier()
train_and_evaluate(dt_model, X_train, y_train, X_test, y_test,
y_test_bin)
```

```
DecisionTreeClassifier Classification Report:
              precision    recall  f1-score   support

         0.0       0.77      0.74      0.75       101
         1.0       0.90      0.77      0.83        81
         2.0       0.76      0.82      0.79        83
         3.0       0.89      0.89      0.89        98
         4.0       0.78      0.85      0.81        82

    accuracy                           0.81       445
   macro avg       0.82      0.81      0.81       445
weighted avg       0.82      0.81      0.81       445

DecisionTreeClassifier Confusion Matrix:
[[75  4  8  3 11]
 [ 6 62  3  4  6]
 [10  1 68  3  1]
```

```
 [ 2  2  5 87  2]
 [ 5  0  6  1 70]]
DecisionTreeClassifier ROC AUC Score: 0.883503568132384
DecisionTreeClassifier Precision Score: 0.8170054291053663
DecisionTreeClassifier Recall Score: 0.8134831460674158
DecisionTreeClassifier F1 Score: 0.8136705488227823
```

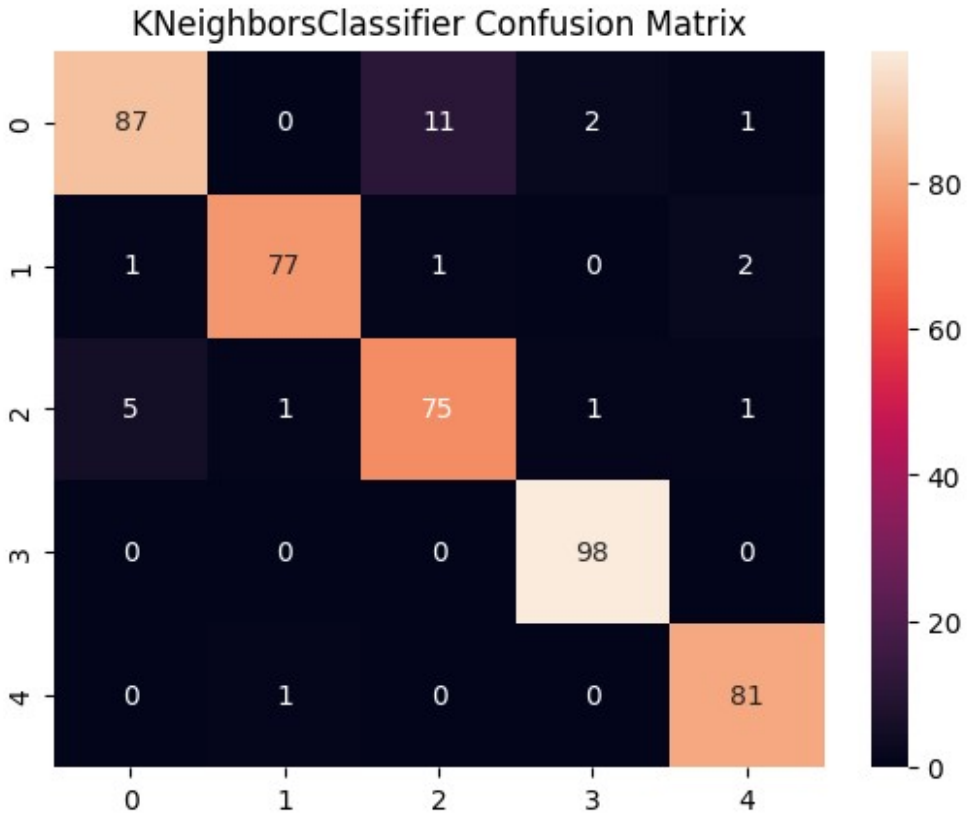


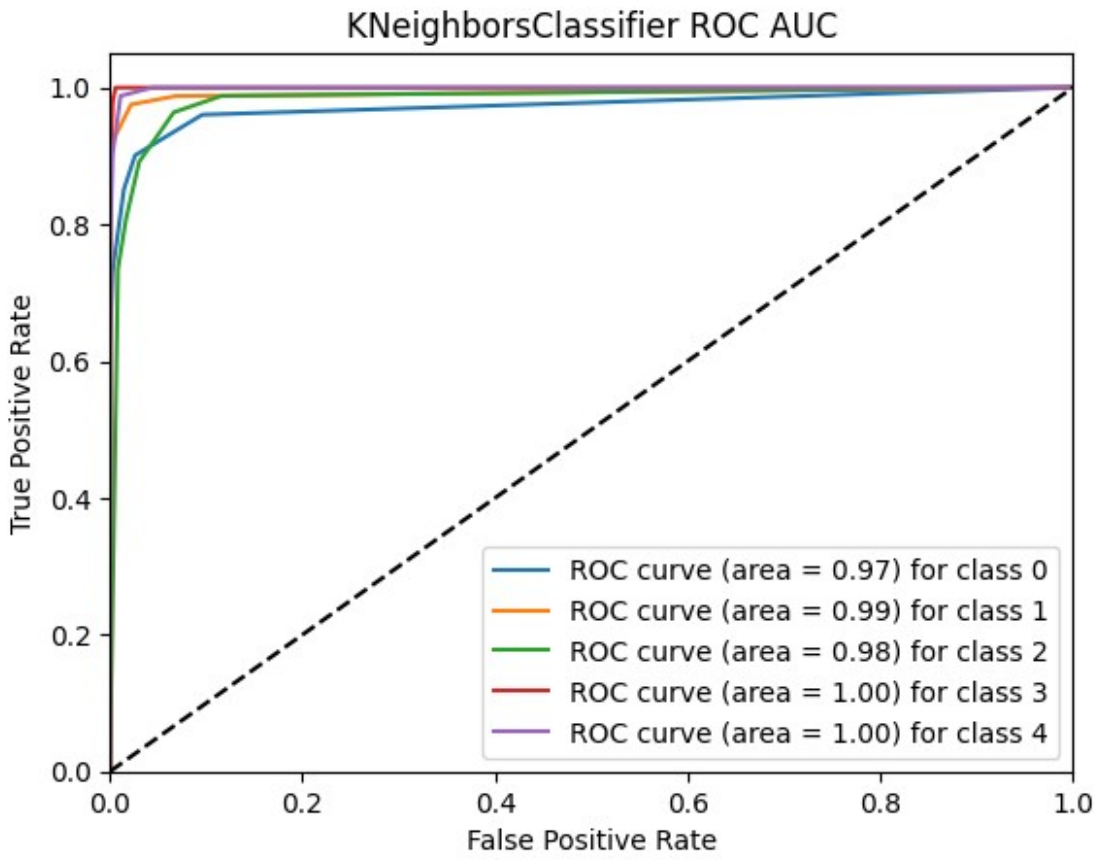


Nearest Neighbours Model

```
# Nearest Neighbors
knn_model = KNeighborsClassifier()
train_and_evaluate(knn_model, X_train, y_train, X_test, y_test,
y_test_bin)

KNeighborsClassifier Classification Report:
              precision    recall  f1-score   support

         0.0       0.94      0.86      0.90       101
         1.0       0.97      0.95      0.96        81
         2.0       0.86      0.90      0.88        83
         3.0       0.97      1.00      0.98        98
         4.0       0.95      0.99      0.97        82

    accuracy                           0.94       445
   macro avg       0.94      0.94      0.94       445
weighted avg       0.94      0.94      0.94       445

KNeighborsClassifier Confusion Matrix:
[[87  0 11  2  1]
 [ 1 77  1  0  2]
 [ 5  1 75  1  1]
```

```
 [ 0  0  0 98  0]
 [ 0  1  0  0 81]]
KNeighborsClassifier ROC AUC Score: 0.9889872127275445
KNeighborsClassifier Precision Score: 0.9398095451140671
KNeighborsClassifier Recall Score: 0.9393258426966292
KNeighborsClassifier F1 Score: 0.938995384783357
```

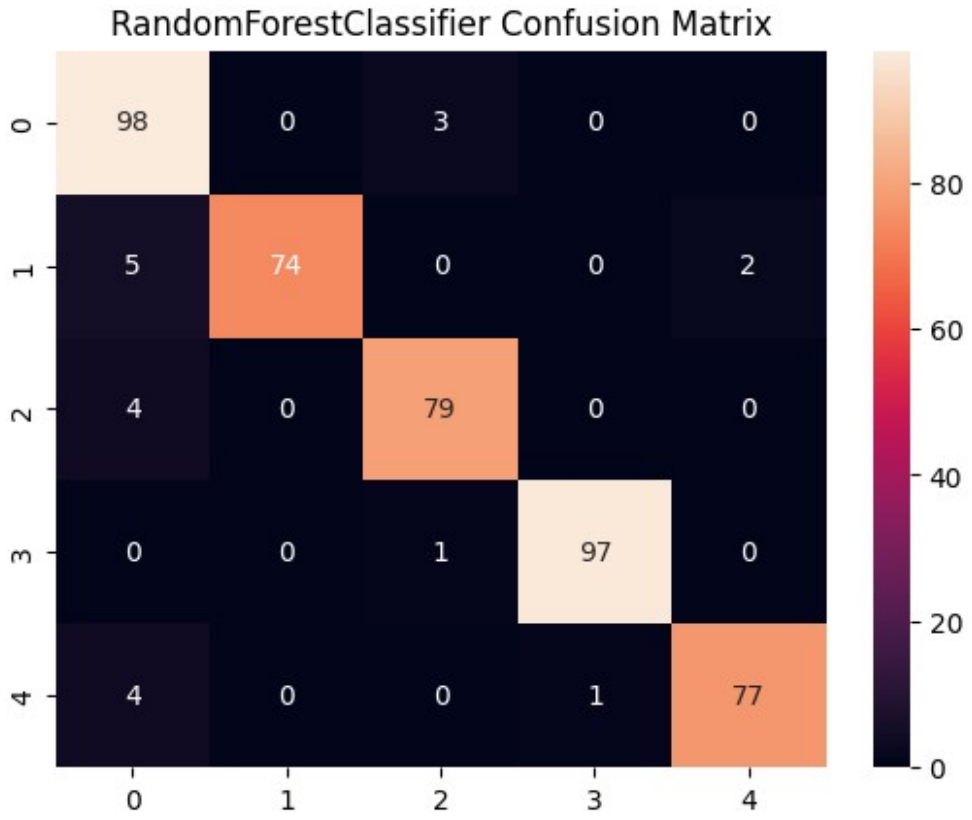


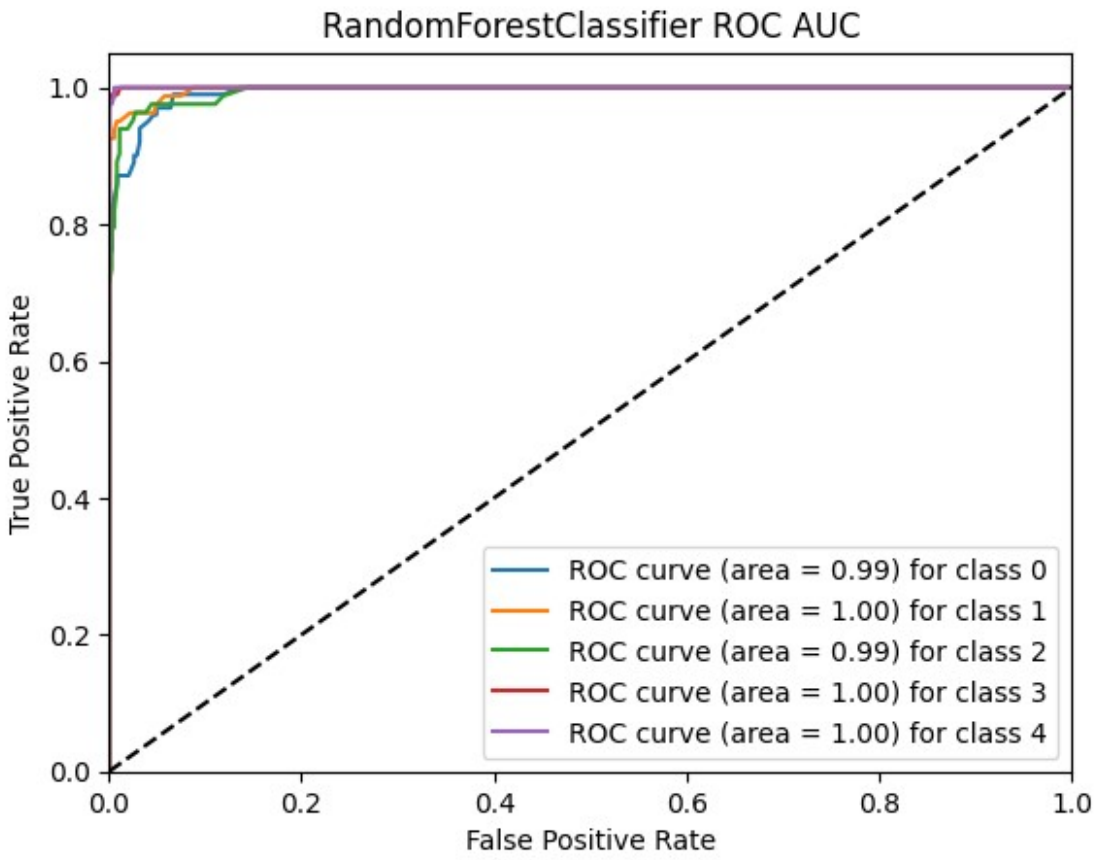


Random Forest Model

```
# Random Forest
rf_model = RandomForestClassifier()
train_and_evaluate(rf_model, X_train, y_train, X_test, y_test,
y_test_bin)

RandomForestClassifier Classification Report:
              precision    recall  f1-score   support

         0.0       0.88      0.97      0.92       101
         1.0       1.00      0.91      0.95        81
         2.0       0.95      0.95      0.95        83
         3.0       0.99      0.99      0.99        98
         4.0       0.97      0.94      0.96        82

    accuracy                           0.96       445
   macro avg       0.96      0.95      0.96       445
weighted avg       0.96      0.96      0.96       445

RandomForestClassifier Confusion Matrix:
[[98  0  3  0  0]
 [ 5 74  0  0  2]
 [ 4  0 79  0  0]
```

```
 [ 0  0  1 97  0]
 [ 4  0  0  1 77]]
RandomForestClassifier ROC AUC Score: 0.9970816602404744
RandomForestClassifier Precision Score: 0.9575173523687249
RandomForestClassifier Recall Score: 0.9550561797752809
RandomForestClassifier F1 Score: 0.9554024192879386
```




RandomForestClassifier ROC AUC
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (area = 0.99) for class 0
ROC curve (area = 1.00) for class 1
ROC curve (area = 0.99) for class 2
ROC curve (area = 1.00) for class 3
ROC curve (area = 1.00) for class 4