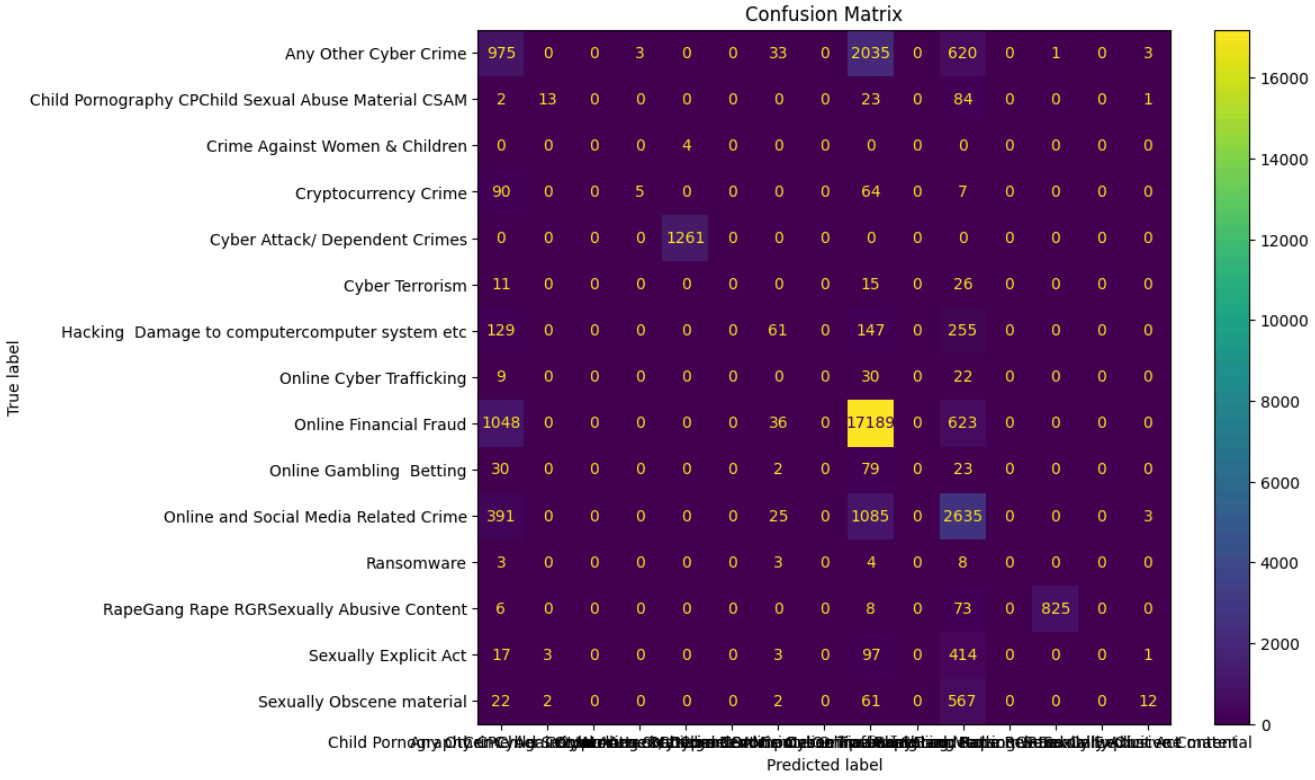
```
# Import necessary libraries
import pandas as pd
import numpy as np
import nltk
import string
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, classification_report,
import joblib
import matplotlib.pyplot as plt
import seaborn as sns
from wordcloud import WordCloud

# Download NLTK resources
nltk.download('stopwords')

# Load datasets
train_data = pd.read_csv('/content/train.csv')
test_data = pd.read_csv('/content/test.csv')

# Preview datasets
print("Training Data Preview:")
print(train_data.head(), "\n")
print("Test Data Preview:")
print(test_data.head(), "\n")

# Preprocessing function for cleaning and tokenizing text
def preprocess_text(text):
    if isinstance(text, str):  # Ensure input is string
        # Remove punctuation and digits
        text = ''.join([char for char in text if char not in string.punctuation and not char.isdigit()])

        # Convert to lowercase
        text = text.lower()

        # Remove stopwords
        stop_words = set(nltk.corpus.stopwords.words('english'))
        text = ' '.join([word for word in text.split() if word not in stop_words])

        # Apply stemming
        stemmer = nltk.stem.PorterStemmer()
        text = ' '.join([stemmer.stem(word) for word in text.split()])

        return text
    else:
        return ""

# Apply preprocessing
train_data['crimeaditionalinfo'] = train_data['crimeaditionalinfo'].apply(preprocess_text)
test_data['crimeaditionalinfo'] = test_data['crimeaditionalinfo'].apply(preprocess_text)

# Feature extraction using TF-IDF
tfidf_vectorizer = TfidfVectorizer(max_features=5000)
X_train = tfidf_vectorizer.fit_transform(train_data['crimeaditionalinfo'])
X_test = tfidf_vectorizer.transform(test_data['crimeaditionalinfo'])

# Labels
y_train = train_data['category']
y_test = test_data['category']

# Train-test split
X_train_split, X_val_split, y_train_split, y_val_split = train_test_split(X_train, y_train, test_size=0.2, r
```

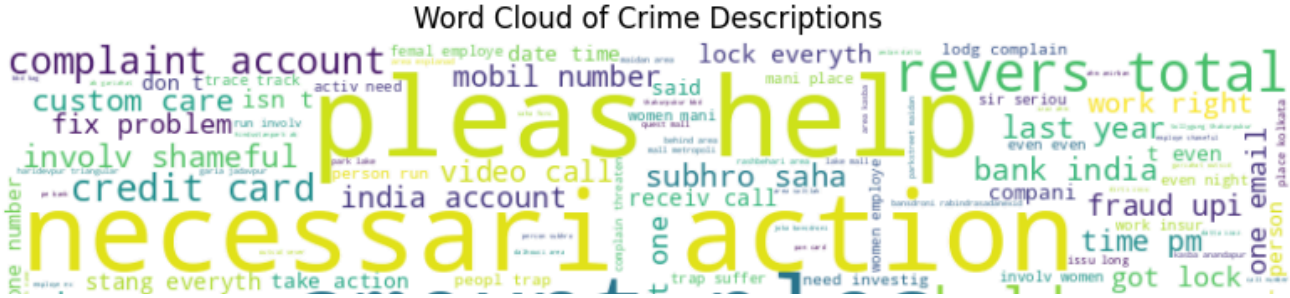
```
# Train model
model = MultinomialNB()
model.fit(X_train_split, y_train_split)

# Evaluate on validation set
y_val_pred = model.predict(X_val_split)
print("\nValidation Set Performance:")
print(classification_report(y_val_split, y_val_pred))

# Evaluate on test set
y_test_pred = model.predict(X_test)
test_metrics = classification_report(y_test, y_test_pred, output_dict=True)
test_accuracy = test_metrics["accuracy"]
print("\nTest Set Performance:")
print(f"Accuracy: {test_accuracy:.4f}")
print(classification_report(y_test, y_test_pred))

# Save the model and vectorizer
joblib.dump(model, 'text_classification_model.pkl')
joblib.dump(tfidf_vectorizer, 'tfidf_vectorizer.pkl')

# Confusion Matrix
class_labels = sorted(test_data['category'].unique())
conf_matrix = confusion_matrix(y_test, y_test_pred, labels=class_labels)
plt.figure(figsize=(10, 8))
ConfusionMatrixDisplay(confusion_matrix=conf_matrix, display_labels=class_labels).plot(cmap='viridis', ax=pl
plt.title('Confusion Matrix')
plt.show()

# Word Cloud
all_text = ' '.join(train_data['crimeaditionalinfo'].astype(str))
wordcloud = WordCloud(width=800, height=400, background_color='white').generate(all_text)
plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.title('Word Cloud of Crime Descriptions')
plt.show()

# Bar Plot for Category Distribution
category_counts = train_data['category'].value_counts()
plt.figure(figsize=(10, 6))
category_counts.plot(kind='bar', color='skyblue')
plt.title('Distribution of Crime Categories')
plt.xlabel('Category')
plt.ylabel('Count')
plt.xticks(rotation=45, ha='right')
plt.show()

# Save metrics
metrics_df = pd.DataFrame([{
    'accuracy': test_metrics['accuracy'],
    'precision': test_metrics['weighted avg']['precision'],
    'recall': test_metrics['weighted avg']['recall'],
    'f1_score': test_metrics['weighted avg']['f1-score']
}])
metrics_df.to_csv('evaluation_metrics.csv', index=False)
print("\nMetrics saved to evaluation_metrics.csv.")
```

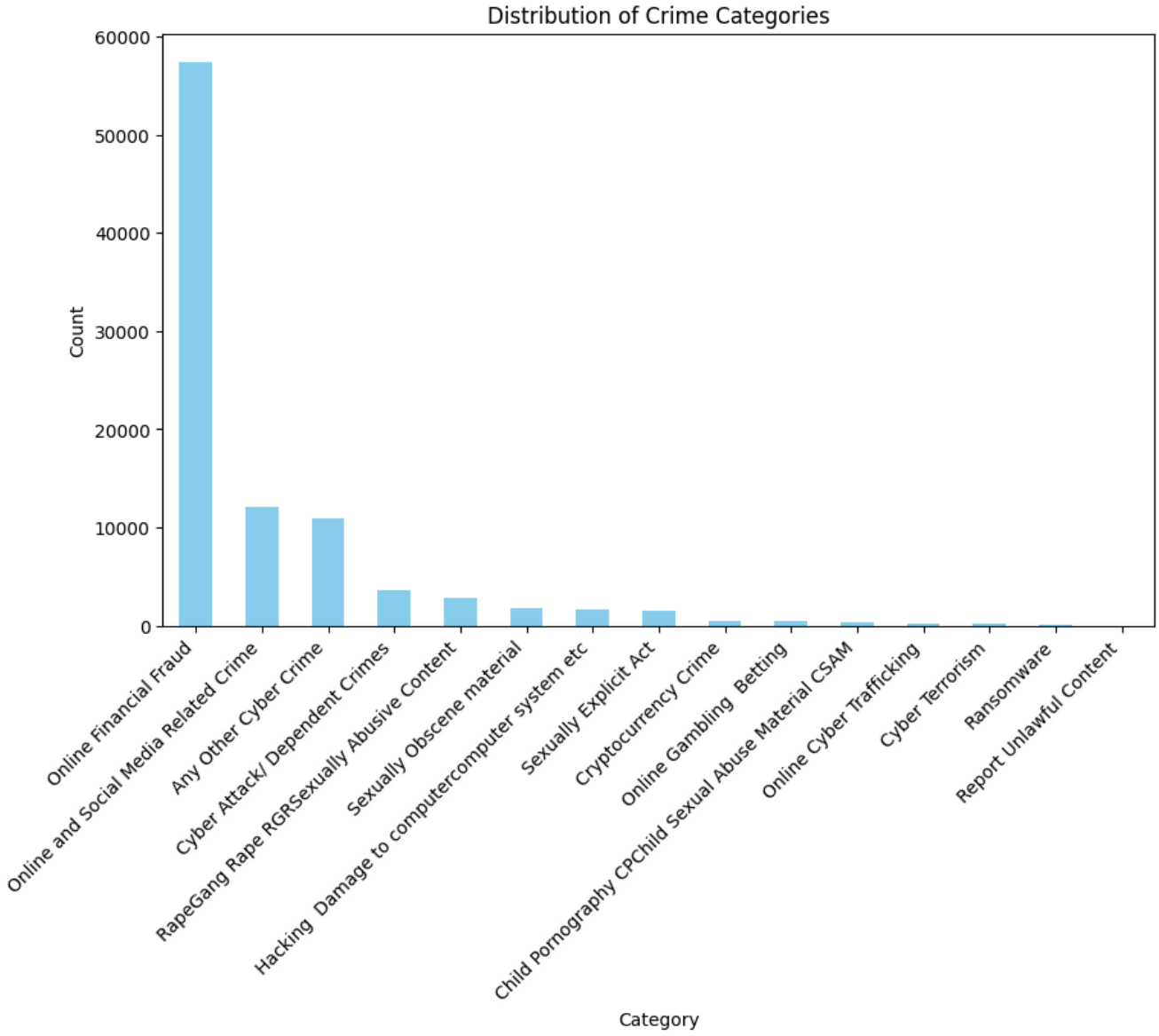
```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
Training Data Preview:
                              category                      sub_category  \
0  Online and Social Media Related Crime  Cyber Bullying  Stalking  Sexting
1                Online Financial Fraud                  Fraud CallVishing
2              Online Gambling  Betting          Online Gambling  Betting
3  Online and Social Media Related Crime                   Online Job Fraud
4                Online Financial Fraud                  Fraud CallVishing

                                crimeaditionalinfo
0  I had continue received random calls and abusi...
1  The above fraudster is continuously messaging ...
2  He is acting like a police and demanding for m...
3  In apna Job I have applied for job interview f...
4  I received a call from lady stating that she w...

Test Data Preview:
                                 category  \
0  RapeGang Rape RGRSexually Abusive Content
1                    Online Financial Fraud
2            Cyber Attack/ Dependent Crimes
3                    Online Financial Fraud
4                     Any Other Cyber Crime

                         sub_category  \
0                                 NaN
1  DebitCredit Card FraudSim Swap Fraud
2                        SQL Injection
3                    Fraud CallVishing
4                               Other

                                crimeaditionalinfo
0  Sir namaskar  mein Ranjit Kumar PatraPaise neh...
1         KOTAK MAHINDRA BANK FRAUD\r\nFRAUD AMOUNT
2  The issue actually started when I got this ema...
3  I am amit kumar from karwi chitrakoot I am tot...
4  I have ordered  saree and  blouse from rinki s...


Validation Set Performance:
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1531: UndefinedMetricWarning
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1531: UndefinedMetricWarning
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1531: UndefinedMetricWarning
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
                                                    precision    recall  f1-score   support

                                 Any Other Cyber Crime       0.36      0.28      0.32      2091
Child Pornography CPChild Sexual Abuse Material CSAM       0.93      0.19      0.31        69
                                Cryptocurrency Crime       1.00      0.03      0.06        96
                        Cyber Attack/ Dependent Crimes       1.00      1.00      1.00       765
                                     Cyber Terrorism       0.00      0.00      0.00        31
      Hacking  Damage to computercomputer system etc       0.37      0.11      0.17       341
                               Online Cyber Trafficking       0.00      0.00      0.00        34
                                 Online Financial Fraud       0.83      0.91      0.87     11471
                               Online Gambling  Betting       0.00      0.00      0.00        97
                  Online and Social Media Related Crime       0.50      0.64      0.56      2453
                                            Ransomware       0.00      0.00      0.00        11
            RapeGang Rape RGRSexually Abusive Content       1.00      0.90      0.95       565
                               Report Unlawful Content       0.00      0.00      0.00         1
                                 Sexually Explicit Act       0.00      0.00      0.00       322
                              Sexually Obscene material       0.67      0.01      0.02       391

                                               accuracy                           0.74     18738
                                              macro avg       0.44      0.27      0.28     18738
                                           weighted avg       0.72      0.74      0.72     18738

/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1531: UndefinedMetricWarning
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1531: UndefinedMetricWarning
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1531: UndefinedMetricWarning
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1531: UndefinedMetricWarning
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))

Test Set Performance:
Accuracy: 0.7357
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1531: UndefinedMetricWarning
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1531: UndefinedMetricWarning
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1531: UndefinedMetricWarning
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
```

	precision	recall	f1-score	support
Any Other Cyber Crime	0.36	0.27	0.30	3670
Child Pornography CPChild Sexual Abuse Material CSAM	0.72	0.11	0.18	123
Crime Against Women & Children	0.00	0.00	0.00	4
Cryptocurrency Crime	0.62	0.03	0.06	166
Cyber Attack/ Dependent Crimes	1.00	1.00	1.00	1261
Cyber Terrorism	0.00	0.00	0.00	52
Hacking Damage to computercomputer system etc	0.37	0.10	0.16	592
Online Cyber Trafficking	0.00	0.00	0.00	61
Online Financial Fraud	0.82	0.91	0.87	18896
Online Gambling Betting	0.00	0.00	0.00	134
Online and Social Media Related Crime	0.49	0.64	0.55	4139
Ransomware	0.00	0.00	0.00	18
RapeGang Rape RGRSexually Abusive Content	1.00	0.90	0.95	912
Sexually Explicit Act	0.00	0.00	0.00	535
Sexually Obscene material	0.60	0.02	0.03	666
accuracy			0.74	31229
macro avg	0.40	0.26	0.27	31229
weighted avg	0.70	0.74	0.71	31229








```
Metrics saved to evaluation_metrics.csv.
```