Summary of Approach, Findings, and Recommendations

---

1. Approach

The primary goal of this project was to develop a machine learning model capable of classifying cybercrime-related textual data into various categories based on the nature of the crime described in the text. The dataset provided contained labeled instances of crime reports, with each instance describing a specific cybercrime incident. The dataset consisted of three columns:

- **Category**: The main category of the crime (e.g., Online Financial Fraud, Cyber Terrorism, etc.).
- **Sub-Category**: A more granular classification within the main category (e.g., Debit/Credit Card Fraud, Fraud CallVishing).
- **Crime Description**: Textual description of the crime incident.

Steps Followed in the Approach:

1. **Data Preprocessing**:
    - The raw crime descriptions were preprocessed to remove noise. This included:
        - Lowercasing all text.
        - Removing punctuation and digits.
        - Tokenizing the text into individual words.
        - Removing common stopwords using the **NLTK stopwords corpus**.
        - Lemmatizing words to ensure that different forms of a word (e.g., "running" and "run") are treated as the same word.
2. **Feature Extraction**:
    - After preprocessing, the text was converted into numerical features using **TF-IDF Vectorizer**. TF-IDF helps capture the importance of words in the context of the corpus, balancing frequency with rarity across documents.
3. **Model Selection**:
    - Multiple classification models were evaluated to determine the best fit for the data:
        - **Logistic Regression**
        - **Random Forest Classifier**
        - **Support Vector Machine (SVM)**
    - The **Logistic Regression** model was chosen after experimentation due to its good balance of performance and simplicity, especially for this multi-class classification task.
4. **Model Training and Evaluation**:
    - The model was trained on the preprocessed training dataset and validated on a separate validation set.
    - **Cross-validation** was employed to ensure robust model performance and avoid overfitting.
    - **Evaluation metrics** such as **accuracy**, **precision**, **recall**, and **F1-score** were computed to assess model performance.
5. **Testing and Performance**:

    - Once the model was trained and evaluated on the validation set, it was tested on the unseen test data.
    - The model's performance on both the validation and test sets was documented.
6. **Saving Results**:
    - The evaluation metrics (accuracy, precision, recall, F1-score) for each class were saved to a **CSV file** as per the hackathon requirements.

---

2. Findings

The findings from the model evaluation indicate the following:

Performance on Validation Set:

- **Accuracy**: 74.42%
- **Precision**: 74.24%
- **Recall**: 74.42%
- **F1-Score**: 71.62%

The model performed reasonably well on the validation set, with a high **precision** and **recall** indicating that it was good at both identifying the correct categories (precision) and capturing most of the relevant crime reports (recall). The **F1-score** of 71.62% reflects a balanced performance, although there is still room for improvement.

Performance on Test Set:

- **Accuracy**: 73.57%
- **Precision**: 72.74%
- **Recall**: 73.57%
- **F1-Score**: 70.57%

On the test set, the model's performance remained consistent with its validation results. The slight drop in performance (compared to validation metrics) is expected in most real-world machine learning models, where a model typically performs slightly worse on previously unseen data.

Classification Report:

The classification report showed varying performance across different crime categories:

- Categories like **Cyber Attack/Dependent Crimes** and **Rape/Gang Rape** achieved near-perfect accuracy and F1-scores.
- However, certain classes like **Any Other Cyber Crime**, **Cyber Terrorism**, and **Ransomware** had very low scores, suggesting that the model struggled to correctly classify incidents in these categories.

Challenges Encountered:

- **Imbalanced Data**: Some crime categories had a significantly higher number of instances than others, which may have affected the model's ability to accurately classify rare categories.
- **Data Quality**: Some entries in the dataset had inconsistent or unclear descriptions, which may have led to confusion in classification.

---

3. Recommendations

Based on the findings, the following recommendations can help improve the model's performance:

1. Addressing Class Imbalance:

- The model's poor performance in some classes (e.g., **Cyber Terrorism** and **Ransomware**) may be due to the imbalance in the dataset, where certain classes are underrepresented. Techniques such as **oversampling** (e.g., SMOTE) or **undersampling** of the majority classes can help the model learn to better classify the minority classes.
- Alternatively, using algorithms that can handle class imbalance, such as **Balanced Random Forests** or **XGBoost**, may yield better results.

2. Data Augmentation:

- Augmenting the dataset by synthetically generating more samples for underrepresented categories can help improve performance. This can be done by paraphrasing existing data points or using text-generation models to create new, realistic crime descriptions.

3. Model Tuning:

- **Hyperparameter optimization** through grid search or random search for models such as **Random Forests** and **Support Vector Machines** could improve performance.
- Using more sophisticated models such as **BERT (Bidirectional Encoder Representations from Transformers)** for text classification could yield significant performance improvements due to its ability to understand context better than traditional models.

4. Text Preprocessing:

- Fine-tuning text preprocessing by:
  - Expanding contractions (e.g., "didn't" to "did not").
  - Removing any additional noise like excessive whitespace or irrelevant characters.
  - Including domain-specific stopwords or words that are crucial to the domain (e.g., terms like "fraud", "cyber", "attack").

5. Continuous Monitoring and Evaluation:

- Regular monitoring and retraining of the model on updated datasets will help ensure that it adapts to new types of cybercrimes. This can be done through **incremental learning** or periodic retraining with fresh data.

---

4. Conclusion

In conclusion, this project demonstrated a successful application of text classification techniques to cybercrime-related reports. The model achieved strong performance, especially for certain crime categories, while struggling with others due to class imbalance and data quality issues. By addressing these challenges and further refining the model, we can achieve even better results, providing a valuable tool for automatic categorization of cybercrime reports.