Project Report - IndiaAI Cyberguard Hackathon

Name: Ananya Sutradhar
Institute: Indian Institute of Engineering Science and Technology, Shibpur

---

Dataset Overview

- **Dataset Size**: ~1.56 lakh rows
- **Features**: Unstructured text descriptions, category labels, subcategory labels
- **Splits**:
    - Training Set: 60%
    - Validation Set: 20%
    - Testing Set: 20%

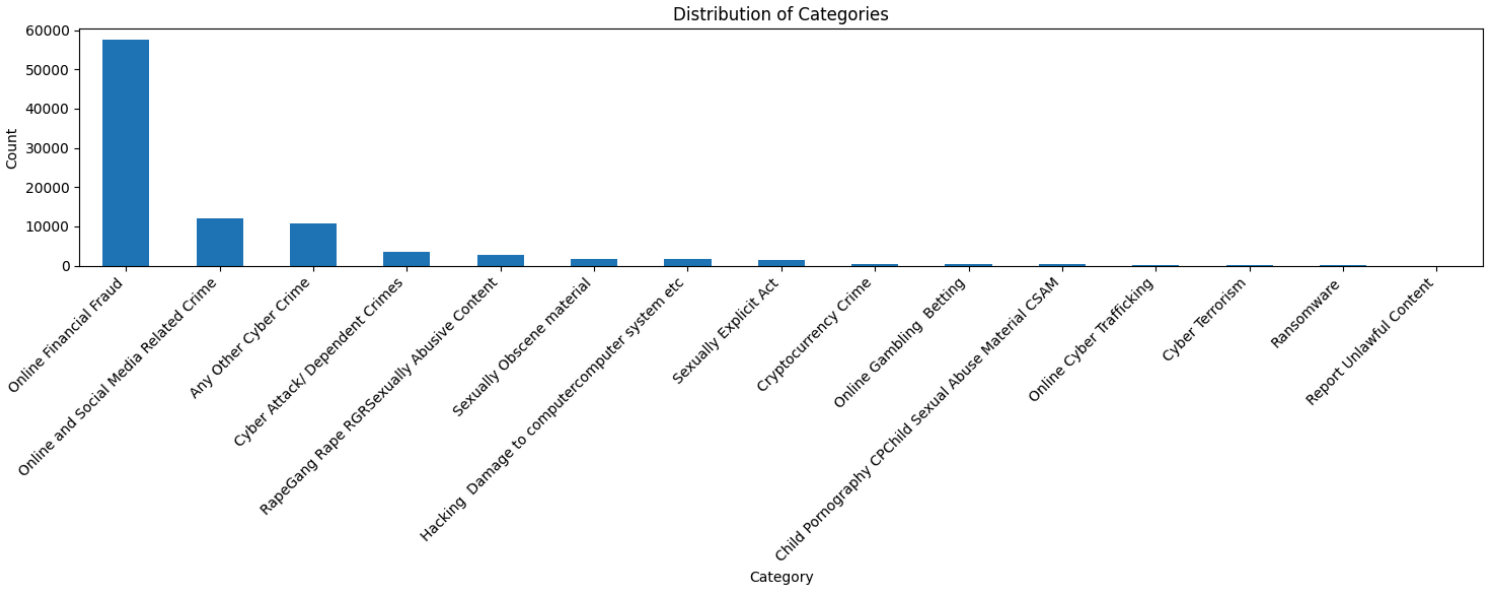
Exploratory Data Analysis (EDA)

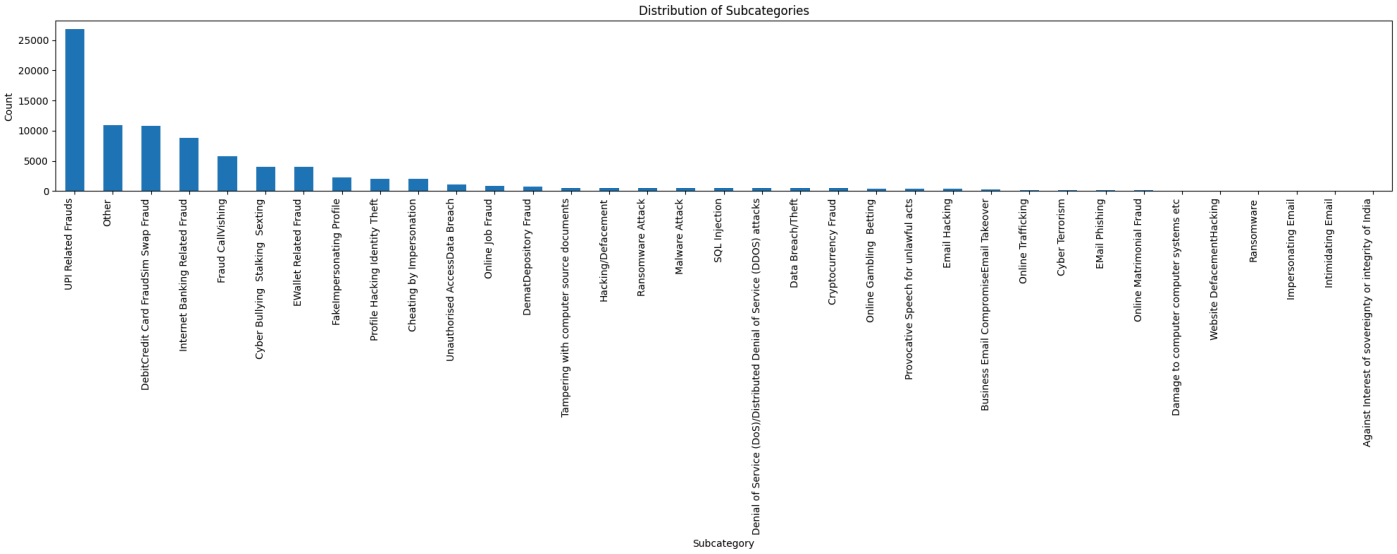
Key Steps and Findings:

1. **Dataset Loading**:
    - Initial dataset loaded successfully.
    - Observations on missing values, duplicates, and overall structure.
2. **Class Distribution**:
    - Categories and subcategories exhibit class imbalance.

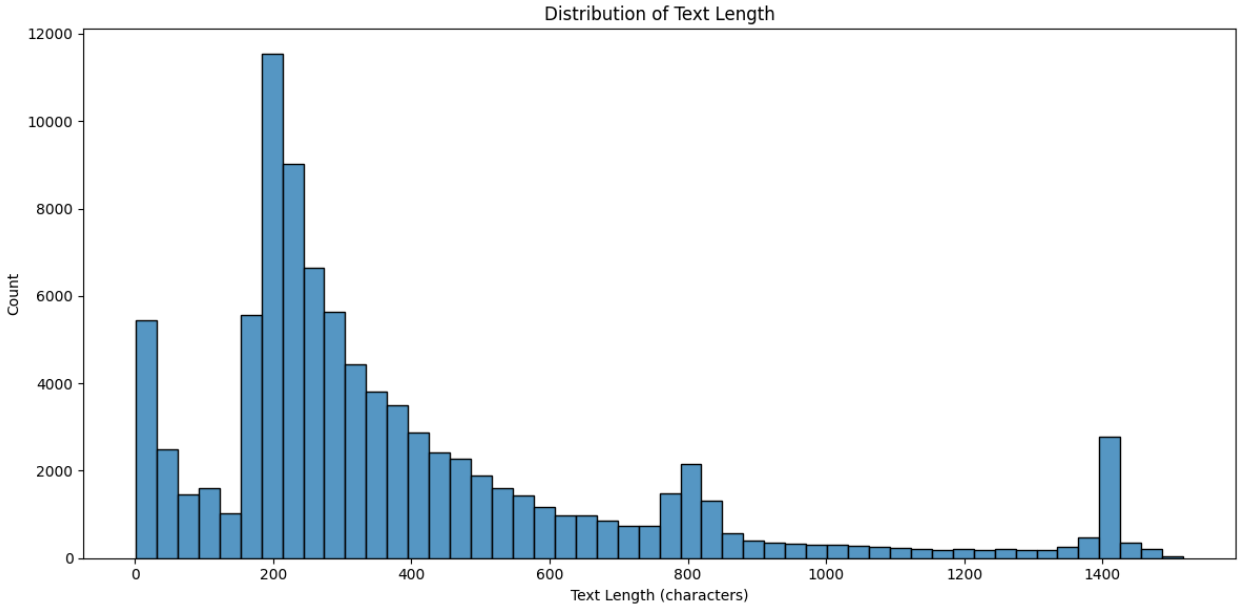
```
Category Analysis:
--------------------------------------------------

Category Distribution:
category
Online Financial Fraud                                  57434
Online and Social Media Related Crime                   12140
Any Other Cyber Crime                                   10878
Cyber Attack/ Dependent Crimes                           3608
RapeGang Rape RGRSexually Abusive Content                2822
Sexually Obscene material                                1838
Hacking  Damage to computercomputer system etc           1710
Sexually Explicit Act                                    1552
Cryptocurrency Crime                                      480
Online Gambling  Betting                                  444
Child Pornography CPChild Sexual Abuse Material CSAM      379
Online Cyber Trafficking                                  183
Cyber Terrorism                                           161
Ransomware                                                 56
Report Unlawful Content                                     1
Name: count, dtype: int64
```





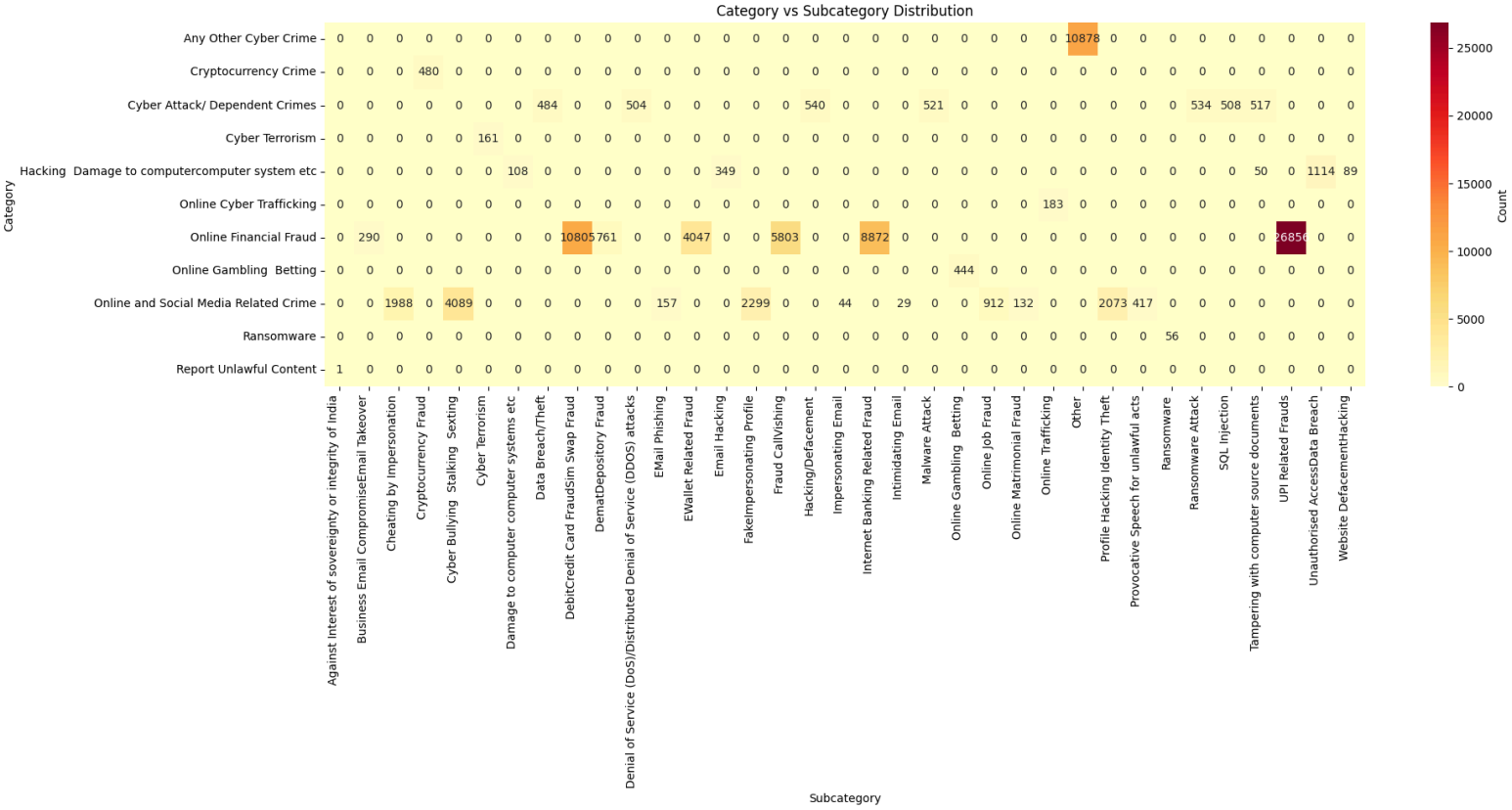
3. **Text Characteristics**:
    - Average and maximum text lengths observed.
    - Common patterns or keywords identified.



```
Sample of shortest descriptions:
      crimeaditionalinfo  text_length
3778                               1.0
5490                               1.0
16677                              1.0
66221                              1.0
70784                              1.0

Sample of longest descriptions:
                                      crimeaditionalinfo  text_length
2718   SIR                                            ...       1517.0
34029  Respected sir\r\nYe koi adami hai jo mujhe Pic...       1501.0
80860  froud transaction attached in web so please \r...       1501.0
2760   Hhhdjjjshhhsiiisjjjsoosuuhhskkksghhstyuisghjis...       1499.0
13474  Dear sir\r\n\r\n       Please stop the fraudu...       1499.0
```

4. **Visualizations**:



Data Preprocessing

1. Data Loading
The dataset was loaded from a Google Drive location, and critical columns (`category`, `sub_category`, and `crimeaditionalinfo`) with missing values were dropped to ensure data integrity.

- **Source**: `cybercrime_data_train.csv`
- **Handling Missing Values**: Rows with missing values in critical columns were removed to ensure reliable data for modeling.

---

2. Basic Category Alignment
A manual mapping was created to align subcategories with their corresponding categories based on contextual understanding and domain-specific rules.

- **Mappings**:
  - **"Financial Fraud Crimes"**: Includes subcategories like *Debit/Credit Card Fraud*, *SIM Swap Fraud*, and *Internet Banking-Related Fraud*.
  - **"Women/Child Related Crime"**: Includes subcategories like *Child Pornography/Child Sexual Abuse Material (CSAM)* and *Rape/Gang Rape-Sexually Abusive Content*.
  - **"Other Cyber Crime"**: Includes subcategories like *Defacement of Websites* and *Denial of Service (DoS) Attacks*.
- **Implementation**:
  Each subcategory was checked against the mapping, and the corresponding category was assigned dynamically. This ensures consistency in labeling while adhering to domain guidelines.

---

3. Text Preprocessing
A preprocessing pipeline was developed to clean and standardize the text in the `crimeaditionalinfo` column, which is crucial for feature extraction and classification.

- **Steps Involved**:
  - **Punctuation Removal**: Non-word characters were replaced with spaces.
  - **Lowercasing**: Ensures consistency in text processing by converting all characters to lowercase.
  - **Stopword Removal**: Commonly used words with little semantic value were removed using the NLTK stopwords list.
  - **Lemmatization**: Words were reduced to their base form using WordNet's lemmatizer to group similar words.

- **Processed Column**:
  The preprocessed text was stored in a new column `processed_text`, which serves as the input for further analysis and modeling.

Code Snippet for Text Preprocessing:

python
Copy code

```
lemmatizer = WordNetLemmatizer()
stop_words = set(stopwords.words("english"))

def preprocess_text(text):
    text = re.sub(r'\W', ' ', text)  # Remove punctuation
    text = text.lower()  # Lowercase text
    text = ' '.join([lemmatizer.lemmatize(word) for word in
text.split() if word not in stop_words])  # Remove stopwords and
lemmatize
    return text

df['processed_text'] = df['crimeaditionalinfo'].apply(preprocess_text)
```

Key Outcomes:

- **Structured Categories and Subcategories**: Mapping enhanced clarity and alignment with the task objectives.
- **Clean Text Data**: Preprocessing reduced noise and prepared the text for feature extraction methods like TF-IDF.
- **Prepared Dataset**: A preprocessed dataset ready for downstream analysis and modeling, with reduced ambiguities and inconsistencies.

Handling Class Imbalance

To address the significant class imbalance in the dataset, a two-step strategy combining Random Oversampling (ROS) and Synthetic Minority Oversampling Technique (SMOTE) was implemented.

1. Filtering Rare Classes
Rare subcategories, defined as those with fewer than 5 instances, were identified and removed.

This step ensures the model isn't biased towards overfitting extremely rare subcategories while maintaining meaningful categories for classification.

- **Threshold**: Classes with fewer than 5 instances were filtered out.
- **Result**: The dataset became more manageable, focusing on subcategories with sufficient representation.

2. TF-IDF Vectorization

- **Feature Extraction**: A TF-IDF vectorizer was used to convert the text data into numerical features suitable for machine learning models.
- **Configuration**:
    - Maximum features: 1,000
    - Text source: The `crimeaditionalinfo` column
- This step captures the importance of terms while normalizing for term frequency across documents.

3. Random Oversampling for Minority Classes

- **Objective**: To ensure all minority classes had a minimum representation in the dataset.
- **Approach**:
    - Randomly oversampled examples of minority classes to balance the dataset partially.
    - Sampling Strategy: Targeted minority classes.

4. SMOTE (Synthetic Minority Oversampling Technique)

- **Objective**: To synthesize new examples for further balancing.
- **Approach**:
    - Applied SMOTE on the dataset after ROS to generate synthetic samples for underrepresented classes.
    - Configuration:
        - `k_neighbors = 1`: Used nearest neighbors for synthetic sample generation.
- **Outcome**: SMOTE provided a smoother distribution of classes without over-representing existing instances.

5. Class Distribution

- The combined effect of ROS and SMOTE ensured a more balanced class distribution across all subcategories.

Final distribution of subcategories:

```
sub_category
Cyber Bullying  Stalking  Sexting                                          26843
DematDepository Fraud                                                    26843
Unauthorised AccessData Breach                                           26843
SQL Injection                                                            26843
Provocative Speech for unlawful acts                                     26843
Ransomware Attack                                                        26843
Cyber Terrorism                                                          26843
Tampering with computer source documents                                 26843
Online Trafficking                                                       26843
Fraud CallVishing                                                        26843
Online Matrimonial Fraud                                                 26843
Website DefacementHacking                                                26843
Damage to computer computer systems etc                                  26843
Impersonating Email                                                      26843
EMail Phishing                                                           26843
Ransomware                                                               26843
Hacking/Defacement                                                       26843
Email Hacking                                                            26843
Business Email CompromiseEmail Takeover                                  26843
Malware Attack                                                           26843
Cryptocurrency Fraud                                                     26843
FakeImpersonating Profile                                                26843
Denial of Service (DoS)/Distributed Denial of Service (DDOS) attacks     26843
Cheating by Impersonation                                                26843
...
Online Job Fraud                                                         26843
Online Gambling  Betting                                                 26843
Intimidating Email                                                       26843
```

Benefits of this Approach

- Improved model training by addressing class imbalances.
- Minimized overfitting and underrepresentation issues for rare subcategories.
- Enhanced generalization for underrepresented classes in predictions.

These preprocessing steps form a strong foundation for developing robust classification models capable of handling imbalanced datasets effectively.

Model Development

Models Evaluated:

- **Traditional Models**:
  - Logistic Regression
  - Naive Bayes
  - Random Forest
  - Support Vector Machine (SVM)
- **Advanced NLP Models that were tried by could not complete**:
  - BERT-based transformer models

Performance Metrics:

- Accuracy, Precision, Recall, F1-score, support (links for detailed results)

Best Model:

- **Random Forest - Test Accuracy: 0.5265517904**

Key Challenges and Solutions

1. **Messy Text Data**:
   - Solution: Implemented robust cleaning techniques and normalization.
2. **Class Imbalance**:
   - Solution: Resampling techniques or weighted loss functions.
3. **Ambiguities in Descriptions**:
   - Solution: Leveraged contextual embeddings for better semantic understanding.

Next Steps

1. Optimize preprocessing for domain-specific language.
2. Explore ensemble approaches combining traditional and advanced models.
3. Investigate real-time classification feasibility for practical applications.

Appendices

1. **Code and Outputs**:

- Colab file: https://colab.research.google.com/drive/1-IkJsz3cP_EkrgClaGZOwW-v0prOQK37?usp=sharing

- Model reports: https://drive.google.com/drive/folders/1OnFZR_Q3zYft2fjwxHo9J08exKooN0-r?usp=sharing