Drishti - multi class classifier

Project Proposal

Light GBM based Multi class classifier of user compliants in categories and sub-categories.

Team Adios

Dec 2024

1. Introduction

In the context of cybercrime reporting and classification, accurately categorizing complaints is essential for enhancing case management efficiency and ensuring timely interventions. This project focuses on developing a robust multiclass classification model, *Drishti*, leveraging the LightGBM algorithm combined with TF-IDF vectorization to classify cybercrime complaints into three primary categories and 62 subcategories. The model emphasizes high accuracy and reliability, achieving **87% accuracy** and an **F1 score of 86.5%** for category classification, alongside **56% accuracy** for subcategory classification. This report outlines the methodologies employed, including data mapping, preprocessing, model training, evaluation, and the results achieved.

2. Dataset Description

The dataset utilized in this project comprises **1.34 lakh records** with three key columns:

1. **crimeaditionalinfo**: Contains textual descriptions of cybercrime complaints in Hinglish (a mix of Hindi and English) and English.
2. **category**: Represents one of the existing 15 high-level complaint categories.
3. **sub_category**: Represents one of the existing 35 detailed complaint subcategories.

The dataset is divided into two subsets:

- **Training Dataset**: 95k records used for training the model.
- **Testing Dataset**: 32k records used for evaluating the model's performance.

To align with the updated classification framework, the existing 15 categories and 35 subcategories are mapped to **3 new categories** and **62 new subcategories** using heuristic rules and semantic relationships.

The textual nature of the dataset presents challenges such as multilingual processing, imbalanced subcategory distribution, and variations in expression, making robust preprocessing and classification techniques crucial for achieving high accuracy.

3. Methodology

This section outlines the step-by-step approach adopted for developing **Drishti**, a multiclass classification model that predicts cybercrime categories and subcategories using **LightGBM** combined with **TF-IDF vectorization**.

3.1 Data Loading and Preprocessing

3.1.1 Data Loading

The dataset was provided as pre-split into two subsets:

- **Training Dataset**: Contains 93,686 records (80%) for model training.
- **Testing Dataset**: Contains 31,229 records (20%) for model evaluation.

Both subsets were loaded into pandas DataFrames for subsequent processing.

3.1.2 Mapping Existing Categories and Subcategories

To align with the updated classification framework, the **15 existing categories** and **35 existing subcategories** were mapped to **3 new categories** and **62 new subcategories** using heuristic rules and semantic relationships. This mapping ensures consistency and contextual alignment across diverse cybercrime cases.

3.1.3 Preprocessing

The **crimeaditionalinfo** column, containing textual descriptions of cybercrime complaints, was preprocessed to prepare it for feature extraction and model training. The preprocessing pipeline included:

1. **Hinglish to English Conversion**: Transliteration of Hinglish terms to English using a predefined mapping for improved semantic understanding.
2. **Text Cleaning**: Removal of special characters, numbers, and irrelevant symbols to standardize the text.
3. **Stopword Removal**: Elimination of common stopwords to retain meaningful terms.
4. **Lemmatization**: Conversion of words to their base forms to reduce dimensionality.

5. **Tokenization**: Splitting text into individual tokens for feature extraction.

The resulting processed text column was used for feature extraction.

3.2 Feature Extraction

To represent the processed text numerically, **TF-IDF vectorization** was employed.

- **TF-IDF (Term Frequency-Inverse Document Frequency)**: Captures the importance of words in a document relative to the corpus.
- Parameters:
  - **max_features**: Limited to 5,000 for computational efficiency.
  - **ngram_range**: (1, 2) to capture both unigrams and bigrams for richer contextual representation.

The vectorized output served as input features for model training.

3.3 Model Development

3.3.1 LightGBM Classifier

The **LightGBM** algorithm was selected due to its high efficiency and ability to handle large-scale datasets. Two separate models were developed:

1. **Category Classification Model**: Trained to classify the data into 3 main categories.
2. **Subcategory Classification Model**: Trained to classify the data into 62 subcategories.

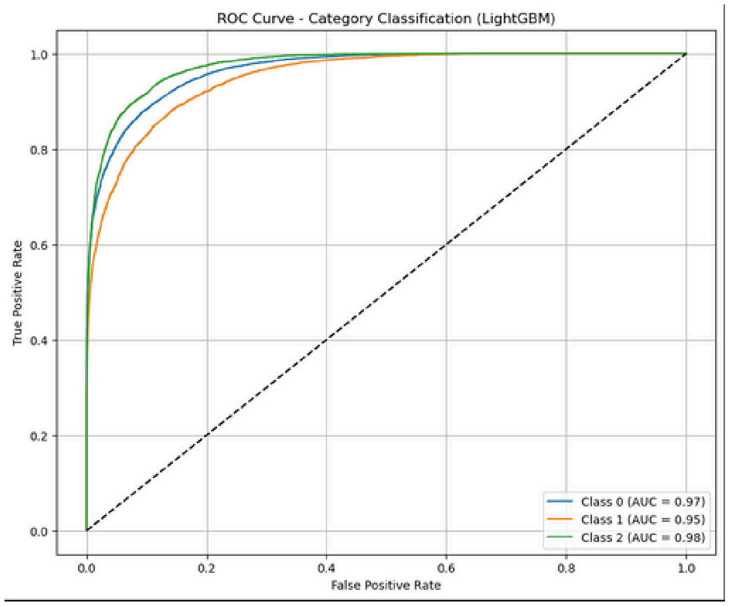
LightGBM Configuration:

- **Boosting Type**: Gradient Boosting Decision Tree (GBDT).
- **Objective**:
  - Multiclass classification for category classification.
  - Multilabel classification using One-vs-Rest for subcategory classification.
- **Evaluation Metric**: Multi-log loss to measure classification performance.
- **Device**: GPU acceleration for faster training using NVIDIA RTX 3060.

3.4 Visualizations

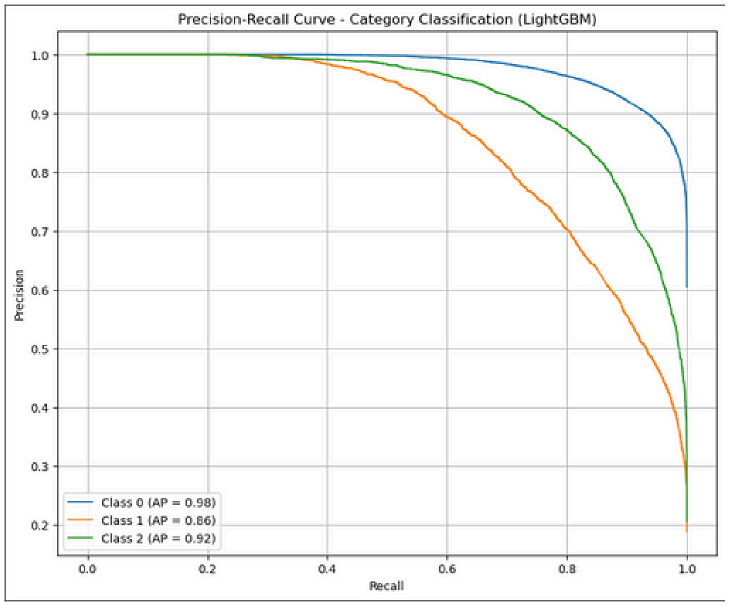
3.4.1 ROC Curve

The ROC (Receiver Operating Characteristic) curve illustrates the model's ability to distinguish between compliant and non-compliant entities. The area under the ROC curve (AUC) quantifies this ability, with a value of 1.0 representing perfect discrimination.



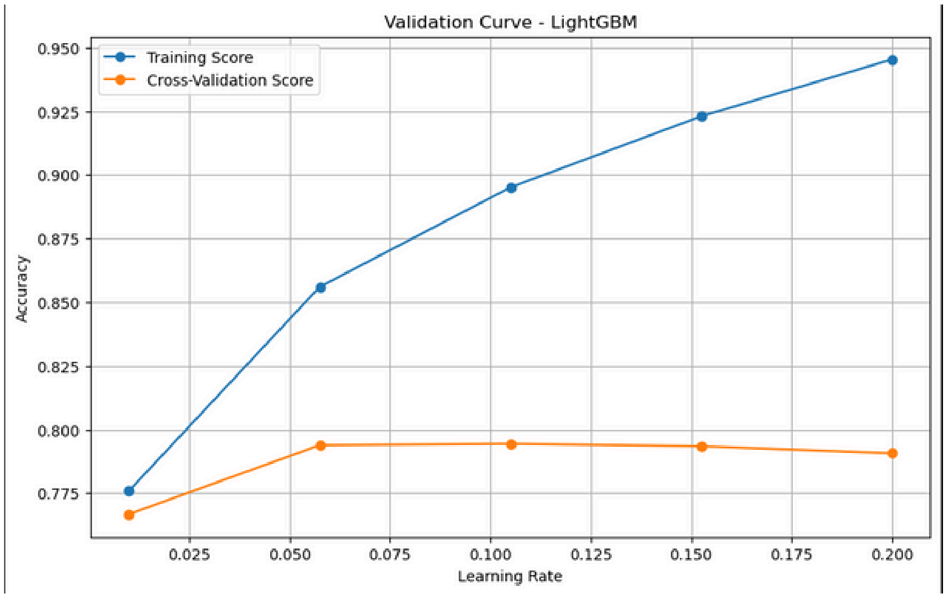
3.4.2 Precision Recall curve

This curve shows the relationship between precision and recall, providing insight into the model's ability to predict positive cases. A higher area under the curve indicates better model performance.



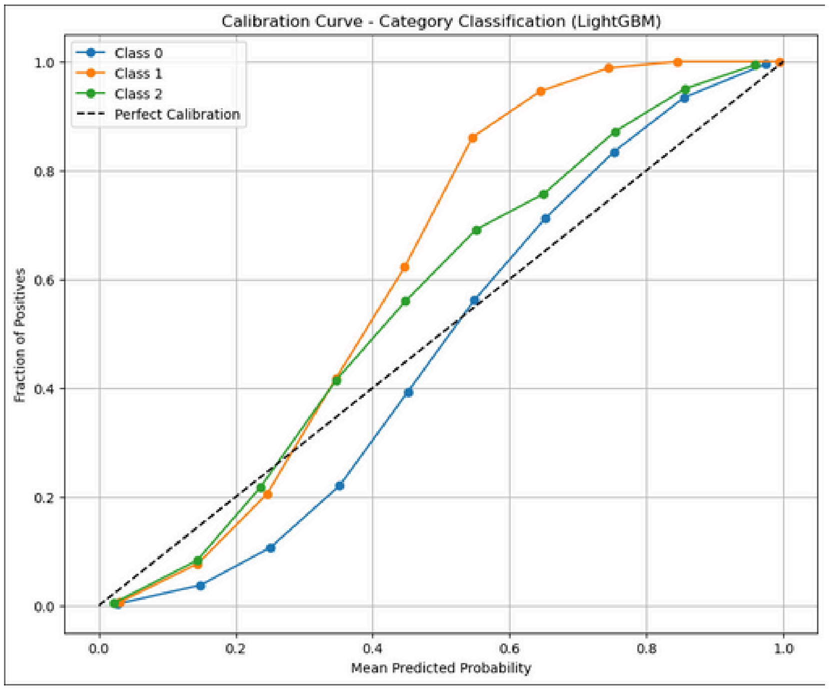
3.4.3 Validation Curve

A plot showing F1 score against the max_depth parameter helps visualize how model complexity influences performance. This assists in selecting an optimal learning rate for the trees.



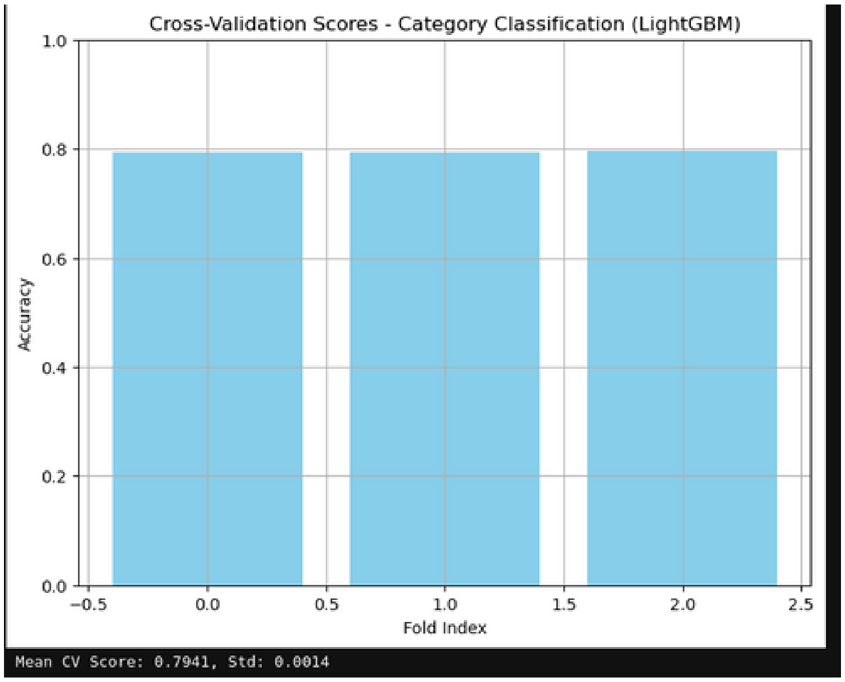
3.4.4 Calibration Curve

This curve assesses the alignment between predicted probabilities and actual outcomes, indicating how well the model's probability estimates reflect the true likelihood of compliance.



3.4.5 Cross-validation Curve

The cross-validation curve provides insights into the model's stability and performance consistency across different training sets.



```
Mean CV Score: 0.7941, Std: 0.0014
```

3.5 Evaluation metrics

To evaluate the model's performance, several metrics were calculated:

- Accuracy: Overall proportion of correct predictions.
- Precision: The ratio of true positives to the total predicted positives, reflecting the accuracy of positive predictions.
- Recall: The ratio of true positives to actual positives, measuring the model's ability to identify non-compliant cases.
- F1 Score: The harmonic mean of precision and recall, providing a balance between the two.
- AUC-ROC: Area under the ROC curve, indicating the model's ability to discriminate between classes across various thresholds.
- Log Loss: Measures the performance of the classification model where the prediction is a probability value between 0 and 1.
- Confusion Matrix: A detailed breakdown of true positives, true negatives, false positives, and false negatives.
- Classification Report: A comprehensive report providing precision, recall, F1 score, and support for each class.

4.Overview of LightGBM

LightGBM (Light Gradient Boosting Machine) is a gradient-boosting framework designed to be efficient, fast, and scalable, particularly for large datasets. It is based on the principle of boosting, where weak learners (decision trees) are combined to form a strong learner.

4.1 Mathematical Model

Objective Function

The objective function for LightGBM consists of two components: the loss function and the regularization term. The overall objective is:

$$L = \Sigma[i=1 \text{ to } n]\ L(y\_i, y\_hat\_i) + \Sigma[k=1 \text{ to } K]\ \Omega(f\_k)$$

Where:

- **L(y_i, y_hat_i)** is the loss function that measures the difference between the actual value **y_i** and the predicted value **y_hat_i**.
- **Ω(f_k)** is the regularization term to control the complexity of the model, defined as:

$$\Omega(f\_k) = (1/2) * \lambda * ||w\_k||\text{^}2$$

Here:

- **w_k** represents the weights of the leaf nodes.
- λ is the regularization parameter penalizing the leaf weights.

Additive Model

XGBoost builds an additive model by sequentially adding new trees f_k to correct the errors made by the previous ensemble. The model can be expressed as:

$$\hat{y} = \sum(k=1 \text{ to } K)\ f\_k(x)$$

Where:

- $\hat{y}$ is the final prediction for an instance.
- K is the total number of trees in the ensemble.
- Each tree f_k is learned to fit the residual errors of the previous trees.

Gradient Descent Optimization

To optimize the objective function, LightGBM uses gradient descent. The loss function is approximated using a second-order Taylor expansion:

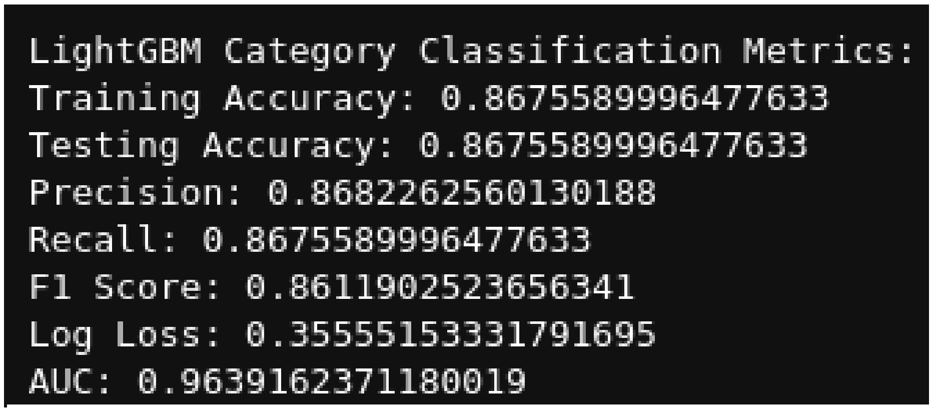
$$L(y\_i, y\_hat\_i + \delta) \approx L(y\_i, y\_hat\_i) + g\_i * \delta + (1/2) * h\_i * \delta^2$$

Where:

- **$g\_i = \partial L / \partial y\_hat\_i$** is the first-order gradient of the loss function.
- **$h\_i = \partial^2 L / \partial y\_hat\_i^2$** is the second-order gradient (Hessian) of the loss function.
- δ is the update to the model.

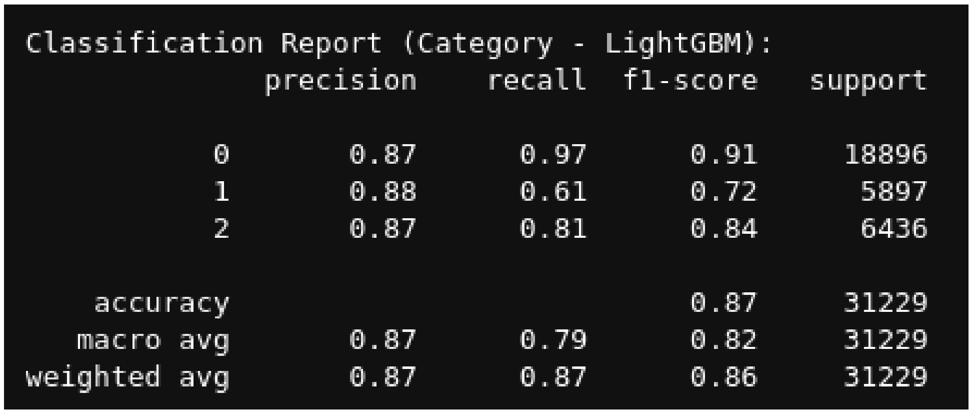
The gradients and Hessians are used to determine the optimal splits and compute the leaf weights for each tree.

5. Results



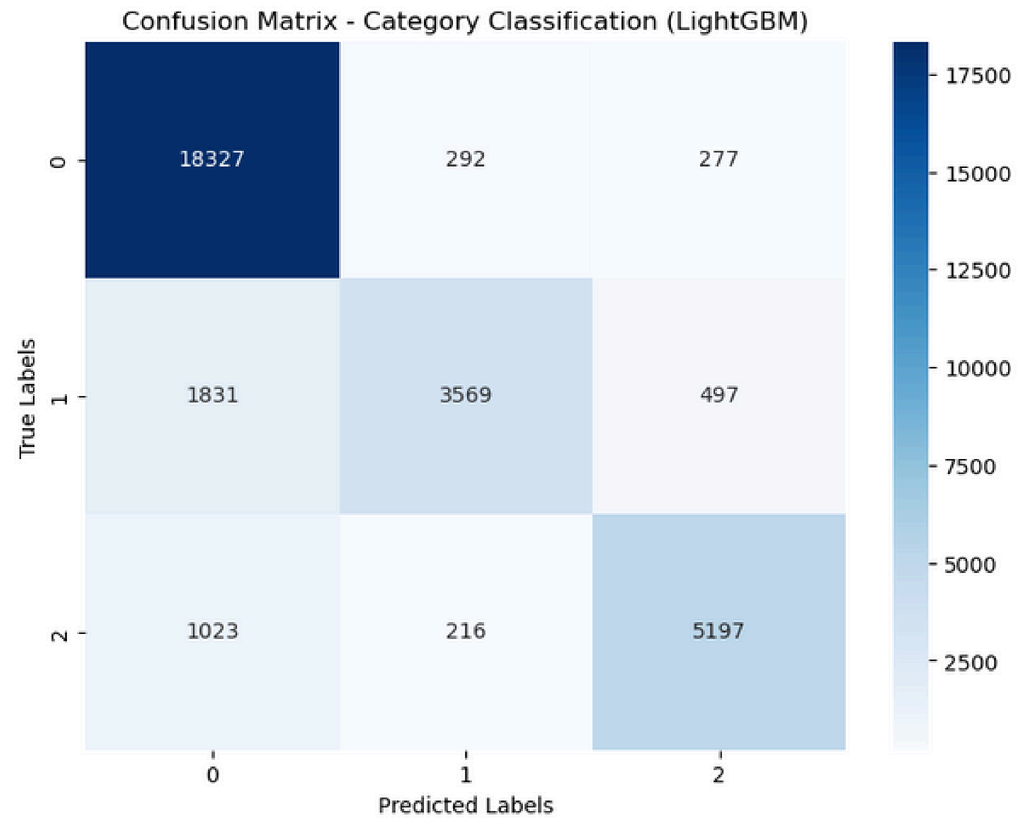
```
LightGBM Category Classification Metrics:
Training Accuracy: 0.8675589996477633
Testing Accuracy: 0.8675589996477633
Precision: 0.8682262560130188
Recall: 0.8675589996477633
F1 Score: 0.8611902523656341
Log Loss: 0.35555153331791695
AUC: 0.9639162371180019
```

Evaluation metrics for category classification

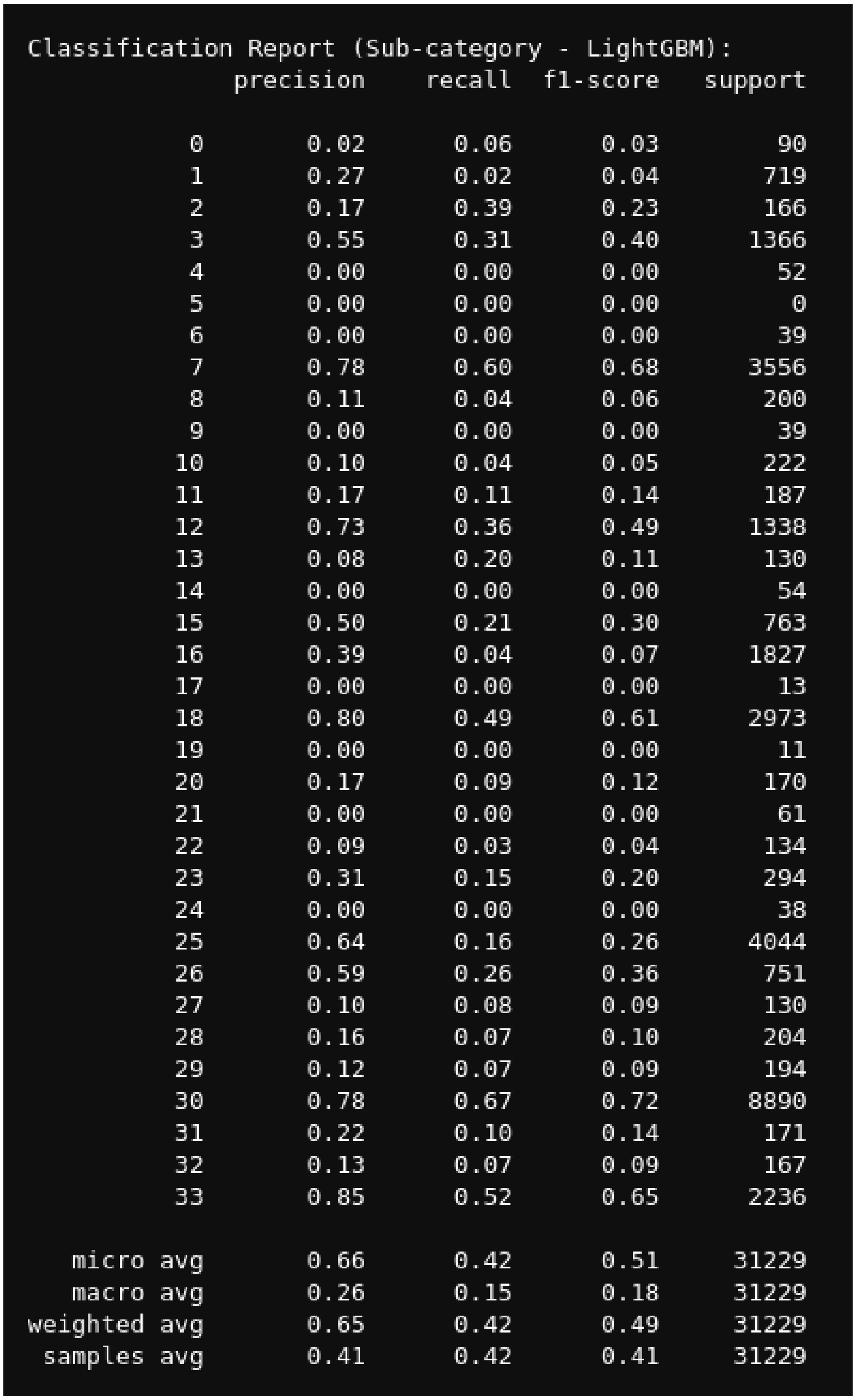


```
Classification Report (Category - LightGBM):
              precision    recall  f1-score   support

           0       0.87      0.97      0.91     18896
           1       0.88      0.61      0.72      5897
           2       0.87      0.81      0.84      6436

    accuracy                           0.87     31229
   macro avg       0.87      0.79      0.82     31229
weighted avg       0.87      0.87      0.86     31229
```

Classification report

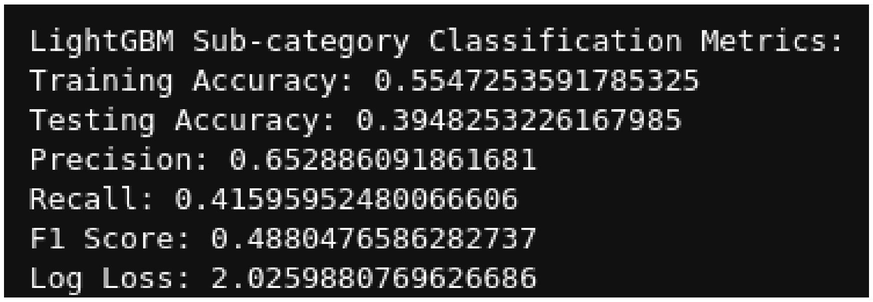


Confusion matrix for category classification



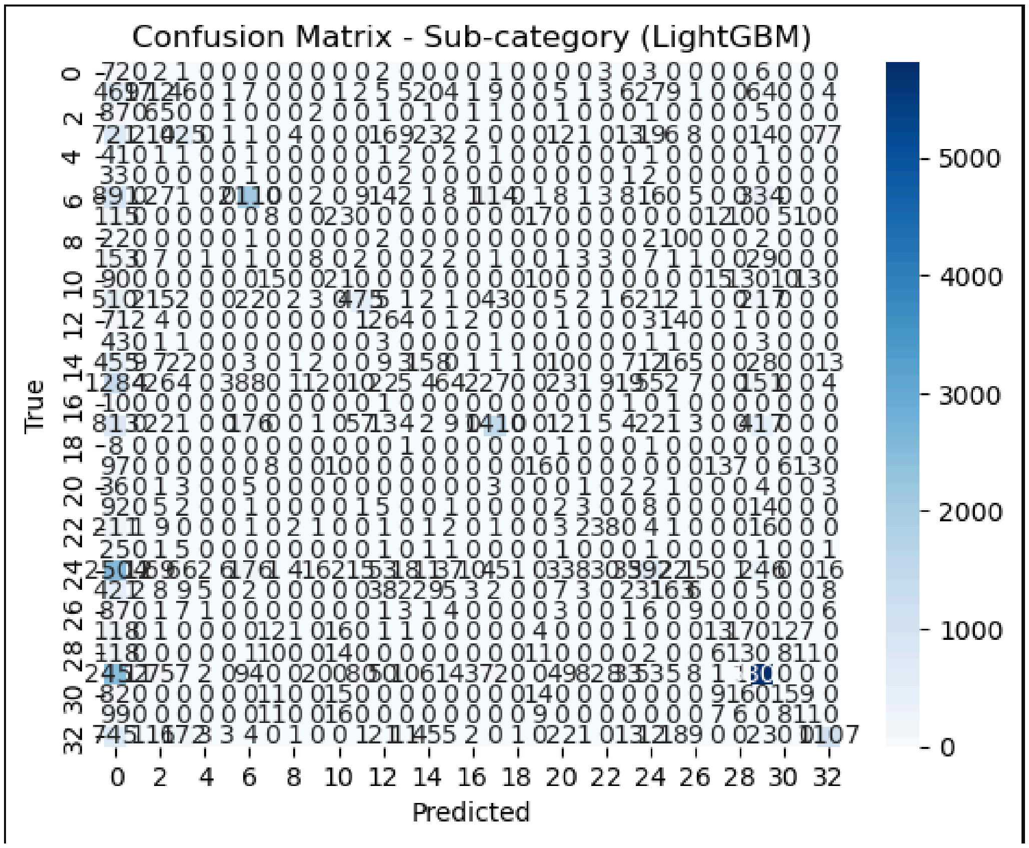
```
Classification Report (Sub-category - LightGBM):
              precision    recall  f1-score   support

           0       0.02      0.06      0.03        90
           1       0.27      0.02      0.04       719
           2       0.17      0.39      0.23       166
           3       0.55      0.31      0.40      1366
           4       0.00      0.00      0.00        52
           5       0.00      0.00      0.00         0
           6       0.00      0.00      0.00        39
           7       0.78      0.60      0.68      3556
           8       0.11      0.04      0.06       200
           9       0.00      0.00      0.00        39
          10       0.10      0.04      0.05       222
          11       0.17      0.11      0.14       187
          12       0.73      0.36      0.49      1338
          13       0.08      0.20      0.11       130
          14       0.00      0.00      0.00        54
          15       0.50      0.21      0.30       763
          16       0.39      0.04      0.07      1827
          17       0.00      0.00      0.00        13
          18       0.80      0.49      0.61      2973
          19       0.00      0.00      0.00        11
          20       0.17      0.09      0.12       170
          21       0.00      0.00      0.00        61
          22       0.09      0.03      0.04       134
          23       0.31      0.15      0.20       294
          24       0.00      0.00      0.00        38
          25       0.64      0.16      0.26      4044
          26       0.59      0.26      0.36       751
          27       0.10      0.08      0.09       130
          28       0.16      0.07      0.10       204
          29       0.12      0.07      0.09       194
          30       0.78      0.67      0.72      8890
          31       0.22      0.10      0.14       171
          32       0.13      0.07      0.09       167
          33       0.85      0.52      0.65      2236

   micro avg       0.66      0.42      0.51     31229
   macro avg       0.26      0.15      0.18     31229
weighted avg       0.65      0.42      0.49     31229
 samples avg       0.41      0.42      0.41     31229
```

Classification report for sub_category metrices



```
LightGBM Sub-category Classification Metrics:
Training Accuracy: 0.5547253591785325
Testing Accuracy: 0.3948253226167985
Precision: 0.652886091861681
Recall: 0.41595952480066606
F1 Score: 0.4880476586282737
Log Loss: 2.0259880769626686
```

Classification report for sub_category metrices



Confusion matrix for sub-category classification