## 1. Objective

The project’s goal was to build a reliable cyberbullying detection system using natural language processing (NLP) and machine learning (ML) techniques.

## 2. Dataset Preparation

We started by importing a labeled dataset. Data cleaning involved removing null values, duplicates, and irrelevant content such as ads or non-textual entries. Class imbalance was found and addressed using stratified data splitting to ensure fair representation across all classes. Sure! Here's a paraphrased version of the selected text:

**Objective**: The aim of this project was to create a dependable system for detecting cyberbullying using natural language processing (NLP) and machine learning (ML) techniques.

**Dataset Preparation**: We began by importing a labeled dataset. The data cleaning process involved removing null values, duplicates, and irrelevant content such as advertisements or non-textual entries. To address class imbalance, we used stratified data splitting to ensure fair representation across all classes.

## 3. Text Preprocessing

Text data was preprocessed by:  
- Lowercasing  
- Removing punctuation, stopwords, and special characters  
- Tokenizing the text  
- Applying lemmatization to reduce words to their root forms

## 4. Feature Representation

Two main types of feature representations were used:  
- TF-IDF (Term Frequency-Inverse Document Frequency) for traditional ML models  
- Word2Vec and Transformer Embeddings for deep learning models

## 5. Model Implementation

We implemented both traditional and deep learning models:  
- Traditional ML: Logistic Regression, Naive Bayes, SVM, Random Forest  
- Deep Learning: LSTM and BERT

## 6. Evaluation Metrics

Models were evaluated using:  
- Precision, Recall, F1-Score, AUC-ROC  
- Confusion Matrix to visualize performance

## 7. Hyperparameter Tuning

We used GridSearchCV to fine-tune parameters like regularization strength and kernel type for optimal model performance.

## 8. Model Training & Testing

Data was split into:  
- 70% for training  
- 15% for validation  
- 15% for testing  
All splits were stratified to maintain class balance.

## 9. Results & Model Selection

Logistic Regression and BERT gave the best results. BERT was chosen for its high F1-score and ability to minimize false negatives by better understanding semantic context.

## 10. Challenges & Solutions

We tackled class imbalance, noisy data, and training deep models efficiently using stratification, text normalization, and careful model selection.

## 11. Conclusion

The final pipeline combines traditional ML and deep learning to provide an accurate, interpretable, and scalable cyberbullying detection system.

Lab Screenshots: