Toxic Comments Filter Project – Guidelines

Project Overview

In recent years, online content moderation has become a critical challenge for social platforms facing an increasing volume of potentially harmful comments. These comments may include insults, threats, obscene content, or hate speech. Manual moderation is ineffective at scale, and traditional algorithms often fail to capture the complexity and variety of offensive language.

DeepCortex AI Solutions is developing an advanced deep learning system to automate and improve moderation. The core of the project is a recurrent neural network (RNN) model designed to classify comments into multiple toxicity categories.

Problem Statement

TechTalk, a technology-focused forum, has noticed that a significant number of user comments contain hate speech or insults, degrading discussion quality. With growing platform popularity, traditional moderation cannot keep up. TechTalk has requested a deep learning-based automated moderation solution to filter toxic comments in real-time.

Use Case Example:
Mario Rossi, TechTalk's community manager, manually moderates content daily. As traffic increases, manual moderation is no longer feasible. The system should automatically filter offensive, threatening, or obscene comments without disrupting the user experience.

Project Objectives

The goal is to build a multi-label deep learning model capable of detecting toxicity across six categories:

1. Toxic
2. Severely Toxic
3. Obscene
4. Threat
5. Insult
6. Identity Hate

Value for TechTalk:

- **Automation:** Reduce manual moderation workload and process more comments in real-time.
- **Efficiency:** Recurrent layers capture context and nuances, improving classification accuracy.
- **Scalability:** System can handle growing volumes of comments and users.
- **Integration:** Direct integration into TechTalk's commenting system ensures immediate automatic filtering without harming UX.

Dataset

The dataset contains ~160,000 user comments, each labeled with one or more toxicity categories.

- Comments can have zero or more active labels.
- Download: Filter_Toxic_Comments_dataset.csv

Technical Requirements

- **Task:** Multi-label classification (6 categories).
- **Model Architecture:** Recurrent layers (LSTM or GRU) to handle sequential text data.
- **Output:** For each comment, produce a 6-element binary vector (1 = presence of label, 0 = absence).

Project Phases

1. Data Preprocessing

- Tokenize text into numerical sequences.
- Normalize and balance the dataset to ensure all toxicity categories are represented.

2. Model Development

- Build a deep learning model with recurrent layers (LSTM/GRU).
- Design for multi-label classification.

3. Model Training

- Split dataset into training, validation, and test sets.
- Apply optimization techniques to improve convergence and performance.

4. Inference & Prediction

- For each new comment, generate a 6-element binary vector indicating toxicity labels.

5. Validation

- Evaluate using metrics such as:
  - Accuracy per label
  - F1-score per label
  - Overall precision for multi-label predictions

Key Deliverables

- Automated, scalable toxic comment filtering system.
- Real-time integration into TechTalk platform.
- Metrics-driven evaluation demonstrating model performance and robustness.