Design of a Real-Time, Multilingual, Emotion-Aware Cyberbullying Detection System using Multi-Teacher Knowledge Distillation and Explainable AI

Prof. Dhananjay R Raut, Harsh J Sakpal, Madhur V Shinde, Aman S Singh & Vishal R Yadav

ABSTRACT

Social media cyberbullying has propagated rapidly and is being experienced by individuals worldwide. It tends to be expressed using sarcasm, emotional language, and multiple languages, making it difficult to determine the identity of the perpetrator. Although automated detection systems are becoming increasingly prevalent, the majority of existing systems suffer from language issues, function only in offline batch mode, and are black-box models that cannot be interpreted. These constraints make it more difficult to intervene with speed and transparency.

This paper offers a real-time, multilingual system for detecting cyberbullying, using explainable AI, emotion and sarcasm detection, and Multi-Teacher Knowledge Distillation (MTKD) to address shortcomings.

The system leverages an ensemble of transformer-based teacher models, like mBERT, XLM-R, and IndicBERT, to capture language-specific features.

Keywords: cyberbullying, real-time NLP, multi-teacher knowledge distillation, explainable AI, XGBoost, SHAP, emotion detection, sarcasm detection, multilingual NLP.

Classification: DDC Code: 006.35

Language: English

Great Britain Journals Press

LJP Copyright ID: 975841
Print ISSN: 2514-863X
Online ISSN: 2514-8648

London Journal of Research in Computer Science & Technology

Volume 25 | Issue 4 | Compilation 1.0

Design of a Real-Time, Multilingual, Emotion-Aware Cyberbullying Detection System using Multi-Teacher Knowledge Distillation and Explainable AI

Prof. Dhananjay R Raut[α], Harsh J Sakpal[σ], Madhur V Shinde[ρ], Aman S Singh[Ↄ] & Vishal R Yadav[¥]

ABSTRACT

Social media cyberbullying has propagated rapidly and is being experienced by individuals worldwide. It tends to be expressed using sarcasm, emotional language, and multiple languages, making it difficult to determine the identity of the perpetrator. Although automated detection systems are becoming increasingly prevalent, the majority of existing systems suffer from language issues, function only in offline batch mode, and are black-box models that cannot be interpreted. These constraints make it more difficult to intervene with speed and transparency.

This paper offers a real-time, multilingual system for detecting cyberbullying, using explainable AI, emotion and sarcasm detection, and Multi-Teacher Knowledge Distillation (MTKD) to address shortcomings.

The system leverages an ensemble of transformer-based teacher models, like mBERT, XLM-R, and IndicBERT, to capture language-specific features. Then, the models collaborate to produce a lightweight XGBoost classifier. To assist with the interpretation of context, additional layers are incorporated to identify sarcasm and emotion. SHAP (SHapley Additive Explanations) is employed to provide each prediction token-level interpretability. Algorithmic and architectural design of a system that would form a transparent, efficient, and deployable solution to detect cyberbullying in different emotional and linguistic contexts is the focus of this study.

Keywords: cyberbullying, real-time NLP, multi-teacher knowledge distillation, explainable AI, XGBoost, SHAP, emotion detection, sarcasm detection, multilingual NLP.

Author α σ ρ Ↄ ¥: Computer Engineering Watumull Institute of Engneering and Technology Thane, India.

I. INTRODUCTION

Cyberbullying refers to the act of sending injurious, discriminatory, or insulting messages through digital media. This has increased in India with widespread smartphone penetration, low-cost internet, and regular use of apps such as WhatsApp, Instagram, Twitter (X), and Facebook. Teenagers, women, minority groups, and celebrities are most affected, and victims experience emotional trauma, reputational harm, and occasionally self-injury or suicide [1].

India's language heterogeneity presents a subtlety problem for detection systems. Users tend to post in several regional languages—Hindi, Bengali, Tamil, Telugu, Marathi, Kannada, etc.—and also often mix these with English, creating code-mixed utterances such as Hinglish or Tanglish. These are frequently posted in Roman alphabet rather than native scripts, employing casual grammar and phonetics (e.g., "tu bahut irritating hai yaar").

Conventional NLP systems and deep learning models usually fail with such code-mixed or transliterated text. Moreover, bullying content tends to include sarcasm, cultural hints, slang, emojis, and emotive tone. For instance, "wah kya smartness hai" may be used as praise or sarcastic

London Journal of Research in Computer Science & Technology

insult depending upon context—this kind of nuance that most models fail to capture [2].

Current cyberbullying detection systems have significant drawbacks:

They are created for a single language (such as English or Thai) and don't address multilingual and code-mixed content that is common in India.

They run offline in batch mode, incapable of detecting toxic posts as they go up.

They use black-box neural models (such as CNN or LSTM), providing no insight into why a post was detected.

They don't have emotion or sarcasm analysis, decreasing accuracy in real-world scenarios [3].

To cover these loopholes, we introduce a real-time multilingual detection framework, designed specifically for Indian languages. The approach employs Multi-Teacher Knowledge Distillation (MTKD) through the ensembling of transformer models (like mBERT, XLM-R, and IndicBERT) as teacher models. The collective outputs of these teacher models are distilled to a lightweight XGBoost student model to support efficient and high-speed inference.

Two auxiliary layers are also added to further improve contextual comprehension:

An emotion recognition module powered by GoEmotions-BERT, which is fine-tuned over Indian code-mixed social media.

An in-house sarcasm detection module trained on both Reddit sarcasm data and Indian code-mixed instances.

To promote interpretability, the system incorporates SHAP (Shapley Additive Explanations) to detect and flag distinctive words that have impacted the decision. This transparency increases trust for human moderators while aiding ethical deployment.

This work is concerned with the system's architectural and algorithmic design—ranging from data flow, model choice, knowledge distillation, emotion/sarcasm fusion, and explainability modules—giving an action plan for future execution in various Indian content environments.

II. LITERATURE REVIEW

Automatic cyberbullying identification has been a topical area of study in the past decade. Numerous approaches have been suggested based on machine learning, deep learning, and NLP. Most of the prior work, however, is constrained in language generality, interpretability of the model, and real-time processing capability. This section provides an overview of five important areas that pertain to our suggested design: knowledge distillation, multilingual NLP models, emotion recognition, sarcasm detection, and explainable AI.

2.1 MTKD with XGBoost for Cyberbullying

Our project base work was presented by Sathit Prasomphan [4], in which a combination of Multi-Teacher Knowledge Distillation (MTKD) and XGBoost was suggested to detect cyberbullying in Thai social media. Several transformer models were used as teacher models to give soft probability outputs. These were distilled into a student XGBoost model to enhance efficiency.

Although the method had good precision, it was limited to Thai material and did not support features such as multilinguality, emotion detection, or explainability. Furthermore, it was developed for static data instead of real-time.

2.2 GoEmotions – Emotion Detection using BERT

GoEmotions is a large dataset developed by Google with more than 58,000 Reddit comments annotated with 27 emotion categories and a neutral category. High accuracy in emotion classification has been demonstrated by a fine-tuned BERT model on this dataset [5]. It comes handy in interpreting the tone of text — if it conveys anger, happiness, sadness, etc.

But GoEmotions was built primarily for English text and is not designed for online abuse or cyberbullying detection. Furthermore, the model has not been implemented in multilingual or

code-mixed environments such as are typical in India.

2.3 SHAP -Explainable AI for NLP

SHAP (SHapley Additive Explanations) is a widely used framework to explain the predictions of machine learning models. It provides a contribution score to each word or token towards the final prediction [6]. SHAP has been applied to various fields like healthcare and finance to increase model transparency.

SHAP is primarily used in NLP with classifiers such as XGBoost or BERT. But there has not been a lot of work on SHAP for cyberbullying detection, particularly in multilingual or affect-based scenarios. Furthermore, SHAP explanations are computationally costly, which makes them less practical to use in real-time

2.4 Sarcasm Detection using Deep Learning

It is challenging to identify sarcasm since it tends to rely on context and implied meaning. Researchers such as Mishra et al. [7] have employed BiGRUs, LSTMs, and CNNs that were trained on Reddit or Twitter to recognize sarcastic posts. The models are moderately successful but tend to be trained on English data alone.

There is no current model that entirely enables real-time sarcasm detection in Indian code-mixed languages. In addition, sarcasm detectors are typically standalone and not incorporated into cyberbullying detection systems.

2.5 RoERTa for Toxiuc Language Classification

RoBERTa is a state-of-the-art variant of BERT and has been extensively applied in toxicity detection tasks, such as datasets like Jigsaw Toxic Comments. It provides excellent accuracy in classifying hateful or offensive speech [8].

Nonetheless, RoBERTa is heavy on resources, opaque, and not optimized for real-time inference. It also does not support multiple Indian languages or emotional and sarcastic content.

2.6 Comparitive Summary of Prior Work

Sr. No.	Paper/Approach	Method	Limitations
1	MTKD with XGBoost [4]	Knowledge Distillation + XGBoost	Thai-only, offline, no emotion or XAI support
2	GoEmotions [5]	Emotion-labeled BERT model	English-only, not focused on bullying
3	SHAP NLP [6]	Word-level explanation for XGBoost	Not integrated with cyberbullying pipeline, high computational cost
4	Sarcasm Detection [7]	BiGRU and CNN on Reddit/Twitter	English-only, not real-time or multilingual
5	RoBERTa Toxicity [8]	Toxic comment classification	Black-box model, lacks emotion/sarcasm modules and multilingual support

III. CONCLUSION OF LITRATURE REVIEW

The review vehemently points out that none of the available systems provide real-time, multilingual, emotion-sensing, and explainable cyberbullying detection in an integrated manner. Particularly in a culturally and linguistically diverse nation like India, where language, emotion, and sarcasm combine in intricate manners, available solutions are inadequate.

The system proposed tries to bridge this gap by:

- Utilizing MTKD for ensemble of multilingual teacher models,
- Including SHAP for explainability purposes,
- Combining emotion and sarcasm detectors for improved context,
- Utilizing a light XGBoost model for efficient inference.

This combined design offers a scalable and deployable backbone for abusive content detection over India's diverse linguistic and social terrain.

IV. PROBLEM STATEMENT

Cyberbullying has emerged as a developing issue in India with the upsurge of online activity on social media like Twitter, Instagram, Facebook, and WhatsApp. In contrast to physical bullying, cyberbullying can take place at any time and from anywhere and even anonymously—resulting in long-term psychological trauma, particularly among young people, women, and marginalized groups [9]. In spite of great improvement in natural language processing (NLP) and machine learning, the existing cyberbullying detection mechanisms are unable to meet the real-world requirements of India's multilingual and culturally rich internet population.

The most significant challenge among them is diversity of languages and code-mixing. Indian users typically write in Hindi, Tamil, Bengali, Telugu, or Marathi—or code-mix them with English (e.g., Hinglish or Tanglish). Such posts are usually composed in Romanized script with heavy usage of non-standard grammar, abbreviations, emojis, and web slang (e.g., "Tu kya bakwaas kar raha hai bro ????♂"). The standard monolingual models learned on formal English data are not able to handle such noisy and informal content efficiently [10].

Secondly, the majority of current systems operate offline in batch mode, analyzing pre-gathered datasets [1]. This introduces a lag between when something is posted and when it's analyzed—making the system useless for sites that require real-time moderation. Without instantaneous discovery, bullying or toxic comments can spread virally before any moderation is done, amplifying its damage.

One of the most important gaps is explainability. Recent transformer-based models such as BERT and RoBERTa provide outstanding NLP accuracy but are black-boxes, providing no or minimal information on why a choice was made. This transparency issue is concerning in legal, educational, or institutional environments where decisions must be justified, confirmed, or audited [6][8].

In addition, the majority of systems do not consider emotional tone and sarcasm, which are particularly prevalent in Indian online discourse. Such a statement as "Waah kya sanskaar hai!" might be an honest compliment or caustic sarcasm depending on the situation. With no consideration for sarcasm or measurement of emotional intensity, systems can either fail to notice problematic content or produce false positives, eroding trust in automated moderation tools [5][7].

Considering these constraints, the need for a real-time, emotion-sensitive, explainable, and multilingual cyberbullying detector, specifically for Indian users, is highly urgent. The system should:

- Be able to handle multilingual and code-mixed text input, particularly from Indian languages
- Function in real time so continuous monitoring and instant alerts are possible
- Detect emotion and sarcasm to enhance contextual categorization
- Be explainable so that human moderators can comprehend and rely on model decisions
- Be light-weight and efficient, allowing deployment in real-world applications

To fill this gap, we present a design that employs Multi-Teacher Knowledge Distillation (MTKD) to merge the strengths of diverse transformer models like IndicBERT, mBERT, and XLM-R, all trained on varying language domains. The teachers impart their soft-label knowledge into an efficient and light-weight XGBoost student model, which is deployable for real-time prediction. We also add emotion and sarcasm detection layers to understand user tone and intent, and lastly apply SHAP (SHapley Additive Explanations) to make each decision interpretable on the token level.

This combined system—tuned to India's digital linguistics—seeks to greatly enhance the detection of toxic behavior across social networks and make detection such that it becomes actionable, ethical, and transparent.

V. OBJECTIVES

The main aim of this project is to create a real-time, multilingual, emotionally intelligent, and explainable cyberbullying detection system custom-made for the nuances of Indian social media. The following is the system's particular objectives in a precise way, ranging from the entire architecture to linguistic diversity, affective comprehension, and moderator interaction design.

5.1 Real-Time Cyberbullying Detection

The system to be proposed is such that it runs in real-time, processing posts upon posting. Contrary to batch-processing models that work on static datasets after they are gathered, this architecture makes use of the Tweepy (Twitter) and PRAW (Reddit) APIs to scan social feeds in real-time.

For instance, when someone tweets "You're such a burden, nobody wants you here," the system needs to process immediately, classify, and alert the moderators in a matter of seconds. Early warning is essential to avoid escalation, especially in high-risk scenarios with youths or vulnerable communities [1], [9].

5.2 Processing Indian Multilingual and Code-Mixed Content.

Considering India's linguistic diversity, the users tend to switch between languages frequently (e.g., "Tum kya bakwas kar rahe ho?" in Hinglish). To deal with this, the system is equipped with:

- Multiple Indian languages: Hindi, Tamil, Bengali, Telugu, Marathi.
- Code-mixed and Romanized scripts.

This is accomplished through IndicBERT, mBERT, and XLM-R, each trained on multilingual corpora with the ability to comprehend local phonetics, grammar variation, and transliteration. These models assist in identifying offensive content for different linguistic inputs [2], [10].

5.3 Multi-Teacher Knowledge Distillation (MTKD)

The architecture employs a Multi-Teacher Knowledge Distillation method in which multiple high-performance transformer models serve as teachers. Each teacher is fine-tuned on a target language or code-mixed data and produces soft probabilities as output.

For example, IndicBERT is good for Indian regional scenarios and XLM-R is good with low-resource multilingual data. Such outputs are consolidated (frequently through weighted average taking reliability of models into consideration) and distilled into a student model, enhancing multilingual generalization without sacrificing speed [1].

5.3 Lightweight XGBoost Student Model

The end student model is an XGBoost classifier, selected because it has:

- Low latency
- High interpretability
- Simple integration with SHAP for explainability

This model receives the distilled soft targets and is tuned using grid search (e.g., max_depth, learning_rate, lambda) to maximize F1-score. It can detect severe phrases like “Go away forever. You’re useless” with minimal computational overhead—ideal for real-time deployment [1], [6].

5.4 Emotion Detection Layer

Emotions are an essential aspect of cyber-abuse, particularly of indirect bullying. This framework incorporates an emotion classifier trained on GoEmotions-BERT, which projects each post to one out of 27 emotion categories: anger, sadness, and fear.

A subtle post like “I’m tired of pretending to be happy ” may not be toxic, but signals distress. This classifier, adapted for code-mixed Indian language content, flags emotionally vulnerable posts to ensure protective action [5].

5.5 Sarcasm Detection Layer

Indian social media often contains sarcasm that masks hostility. Consider a sarcastic comment like

"Wah kya smartness hai!", which can be both humorous and derogatory depending on context.

A BiGRU or attention-based LSTMs sarcasm classifier is employed, which is trained on labeled sarcasm data from Twitter and Reddit. The module detects sarcastic posts based on language indicators, emojis, and context shifting, greatly improving the accuracy of classification [7].

5.6 Explainable AI (SHAP-based)

Model explainability is facilitated through SHAP (SHapley Additive Explanations). SHAP calculates token-level attribution, allowing moderators to know why a post was detected.

For instance, in the statement "Nobody wants you around anymore", SHAP will underline "nobody" and "anymore" as primary triggers. These visual justifications increase transparency and establish trust, particularly in moderation environments that mandate human justification or auditing [6].

5.7 Severity Classification of Cyberbullying

Posts are categorized into three levels of severity based on their content:

- "You are annoying" → Mild
- "You should disappear" → Moderate
- "You deserve to die" → Severe

This is attained through the use of a mix of toxicity scores, emotion classes, and sarcasm indicators. By introducing severity levels, the system allows moderators to prioritize high-risk content and act accordingly [3], [9].

5.8 Moderator Dashboard for Monitoring and Action

To aid human moderators, a ReactJS dashboard is created. It shows flagged posts with:

- Detected language
- Emotion
- Sarcasm
- SHAP explanation
- Severity level

They may take steps like "Ignore," "Report," "Delete," or "Export logs." This interface offers complete transparency, accountability, and usability to platform teams, educators, or legal authorities that need to track abuse patterns [11].

VI. PROPOSED DESIGN

6.1 Data Collection and Sources from Social Media

To create a good system for finding cyberbullying, we need lots of different kinds of data we can trust. Social media sites like X, Facebook, Instagram, and YouTube have tons of content made by users, and that's where bullying happens. We start by grabbing posts and comments using official tools or data sources we pay for. For example, we can use X's API to grab tweets that have certain words or hashtags linked to bullying, like loser, hate, or idiot, or mentions of user accounts. We can also get data from Reddit and YouTube comments using tools that are available to everyone. We make sure to keep user data private by removing personal info and keeping everything safe.

In addition to open-source datasets, curated code-mixed language corpora—especially from Indian social media—are included. Users often mix English, Hindi, Bengali, Tamil, and other languages within the same post, which complicates detection. Many Hindi words also appear in Roman script (for example, "ch-**iya") or as slang variants, raising the difficulty. To address this, the dataset includes both monolingual and code-mixed samples across multiple languages to reflect diverse community contexts.

Throughout, ethical standards are applied by anonymizing personal information such as names, email addresses, and phone numbers. Because bullying instances are relatively rare, data augmentation techniques like synonym replacement, back-translation, and paraphrasing are used to expand the dataset. This reduces bias toward non-bullying examples and improves the model's ability to generalize across content types.

6.2 Preprocessing and Cleaning

Online posts can be a mess. People misspell words, use emojis to show feelings, and often use

abbreviations or loose grammar. If you don't clean things up first, it's easy for a model to miss teasing or bullying. This part gets the text ready but keeps the original meaning.

- Noise and Slang Removal – We fix things like umm, lol, and btw. We also squeeze repeated letters in words like looooser and make them loser. We keep a list of slang terms (including bad ones from different areas) and replace them with standard markers, so the models understand them better.
- Emoji Expansion – Emojis carry clear signals: "😂" can hint at sarcasm, while "😡" suggests anger. Following Felbo et al. [12], the pipeline maps emojis to short text descriptors (e.g., 😂 → "laughing-crying") so text-only models still capture affective cues.
- Language Identification & Romanized Text Handling – Many Indian users write Hindi or Tamil in Roman script, so a language ID step labels tokens as English or a regional language. Then, we turn words back to native scripts with tools like IndicTrans [13], ensuring slang like "madarchod" is recognized despite spelling variants.
- Normalization & Tokenization – We swap URLs, hashtags, and user names for tags like <URL> and <USER> to reduce clutter but keep post structure. We split the text into smaller bits (tokens) with tools made for multiple languages—this lets models learn from cleaner data.

Together, these steps turn noisy, informal text into clean, structured, and semantically rich sequences that are ready for deep representation learning.

6.3 Multi-Teacher Ensemble (mBERT + XLM-R + MuRIL)

Understanding the many ways people express themselves in different languages takes more than a single model. Instead of relying on just one, we bring together several powerful language models, each trained for multilingual tasks. This "multi-teacher ensemble" blends their strengths to make our system both broader and more dependable.All the teacher models are trained on the same dataset, and instead of just using hard labels, we merge their soft probability outputs to gain richer insights. This method preserves uncertainty and allows us to better understand cross-lingual details.

- mBERT (Multilingual BERT) [14] – Supports over 100 languages with strong cross-lingual capabilities for various NLP tasks. Its subword tokenization helps navigate complex morphology in Indian languages and reduces out-of-vocabulary issues.
- XLM-RoBERTa (XLM-R) [15] – Trained on an extensive multilingual corpus, it excels particularly at zero-shot transfer across languages. It's especially effective for context-rich cues, like cross-lingual sarcasm, offering strong, language-agnostic contextual embeddings.
- MuRIL (Multilingual Representations for Indian Languages) [16] – Built specifically with Indian languages and code-mixing in mind, including Romanized text. This makes it ideal for identifying harassment or bullying in mixed scripts such as Hinglish and Tanglish.

The ensemble combines the soft predictions from each teacher to use their complementary strengths and minimize blind spots. For example, if mBERT outputs a high likelihood for "hate" while MuRIL detects a Roman-script Hindi slur, the combined result offers a more reliable understanding than either model alone.

6.4 Knowledge Distillation

While teacher models tend to be highly accurate, they often require major computational resources, making them less suitable for real-time applications. To address this challenge, we use knowledge distillation (KD) [17]. This technique involves using the softened probability outputs of a large teacher ensemble to guide the training of a smaller, more efficient student model.

The core idea is that the ensemble of teachers doesn't just say 'yes' or 'no' to bullying—instead, it gives nuanced scores (for example, "70% bullying, 20% sarcasm, 10% neutral"). The student model then learns from these slightly softened, richer targets, instead of hard yes/no labels. This lets it

understand more about the grey areas between classes.

During training, the student tries to match the teachers by minimizing the difference (using KL divergence) between their predictions. We also use "temperature scaling" to smooth the probabilities, helping the student grasp a wider range of outcomes. In the end, the student model learns not only to make correct classifications but to mirror the deeper understanding of its teachers—while running much faster on ordinary computers.

6.5 Student Model (XGBoost Classifier)

After distillation, we pass the learned information to a lightweight XGBoost classifier. XGBoost is fast, easy to understand, and runs well even on basic hardware (like school desktops). It takes a mix of raw transformer outputs—summarized into embeddings—and a handful of hand-crafted features to spot bullying in social media posts. These features include:

- Term frequency–inverse document frequency (TF-IDF): How rare or common each word is in the dataset.
- Sentiment polarity scores: Does the post "sound" positive, negative, or neutral?
- Stylometric features: Punctuation counts, upper/lowercase ratios, number of exclamation marks, etc.
- Code-mixing index: For example, what percent of a post is in Hindi versus English?

By combining deep text understanding with these simple, readable features, XGBoost gives us predictions that are reliable and still easy to explain. Because it runs efficiently on CPUs, it's practical to use for real-time checks on everyday school computers.

6.6 Emotion & Sarcasm Detection

Bullies don't always come out and say what they mean—instead, they often hide behind sarcasm or emotional language. That's why our system includes two special detectors:

- Emotion Detector — Built on GoEmotions-BERT [], this model classifies messages into emotions like anger, sadness, joy, disgust, or fear. Bullying is often linked to strong negative emotions (especially anger and disgust), so detecting these feelings helps us judge how serious a message is.
- Sarcasm Detector — Sometimes, a message looks positive but actually carries a cutting or mean tone—for instance, "Wow, you're such a genius 🙄." Our sarcasm detector tackles this using a BiGRU with attention network [19], which has been trained on real sarcastic posts (and uses emoji cues, too). The model "pays extra attention" to words and symbols that hint at sarcasm, making it easier for us to spot when someone is being sneaky.

The outputs from both detectors feed into XGBoost's final decision, making it much less likely that hidden or subtle bullying slips by unnoticed.

6.7 Final Decision Layer

The final decision brings together the student model, emotion detector, and sarcasm detector all send their findings into a final decision layer, which produces a single, easy-to-use result. The layer returns three outputs:

- Binary Decision – A simple Yes/No on whether the post is considered bullying.
- Severity Score – A number between 0 and 1, showing how intense the bullying is (so you can decide, for example, if a warning is enough or if a block is needed).
- SHAP Explanation [6] – For full transparency, the system uses SHAP (SHapley Additive exPlanations) to break down which words or features contributed most to the final call—for example, the score might show that a certain abusive word or sarcastic emoji tipped the scales.

This kind of explainability helps moderators trust the system, makes decisions fairer, and fits best practices for ethical AI.

6.8 Moderator Dashboard

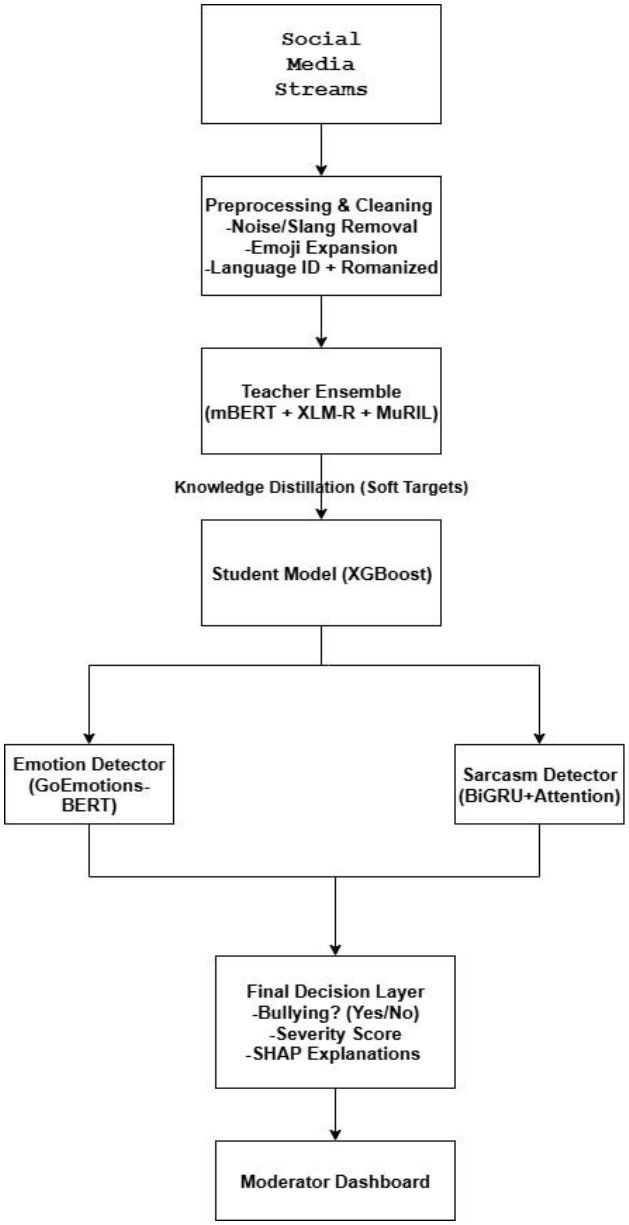
After processing, a Moderator Dashboard turns complex outputs into simple, actionable views. It

clearly indicates what requires immediate attention and offers sufficient context to enable quick, confident decisions.

- Flagged Content Viewer – Shows the original post and clearly marks which parts are abusive, making it easy to spot and review problem language fast.
- Severity Analytics – The dashboard brings trends to life: track which terms, languages, or types of abuse are flaring up over time, and catch new slang or patterns before they spread.
- Real-Time Alerts – If something urgent—like a threat or hate speech—shows up, the dashboard pings the moderator right away. That way, high-severity issues never slip through the cracks.
- Explainability Reports – When a post gets flagged, moderators aren't left guessing. The dashboard lays out a straightforward summary (thanks to SHAP), showing exactly why the system flagged that message.

The goal here is to keep the human in charge. By keeping controls simple and explanations easy to follow, the dashboard supports ethical, transparent content moderation—helping moderators protect their communities and make decisions with confidence [9].

A. System Workflow Diagram.



VII. TOOLS AND TECHNOLOGY

7.1 Social Media API Integration (Free Alternatives)

To keep costs down and still collect enough data, the system leans on free API options, with pacing and batching to avoid rate caps.

- Twitter (Tweepy with free API v2 access)
  - Free developer tiers allow limited monthly volume (policy-dependent, often a few hundred thousand tweets).
  - Real-time filters by keywords or hashtags make targeted streaming easy.
  - JSON output drops straight into preprocessing and NLP steps.
- Reddit (PRAW)
  - Works free with personal/app credentials within rate limits.
  - Streams or polls posts and comments from chosen subreddits tied to bullying detection.
- Other free platforms
  - Pushshift API: Handy for historical Reddit backfill and exploration, availability varies.
  - Mastodon API: Federated, open-source streaming from selected instances.
  - Facebook/Instagram alternatives: Crowd Tangle offers limited free research access for public data.

Run short, staggered jobs (every 10–30 seconds) to stay within free limits, and write line-delimited JSON from small Python scripts for durable, auditable NLP pipelines [20].

7.2 Language Detection and Handling.

Free tools for multilingual detection and processing:

- langdetect (Python): Open-source, detects 55+ languages, and works with simple rules to handle code-mixed text.
- langid.py: A lightweight, fully offline detector that's easy to run in scripts or on devices.
- Indic NLP Library: Tokenizes and provides utilities for Indian languages to produce clean inputs.
- IndicTrans: Free transliteration from Romanized text to native scripts (e.g., Hinglish → Hindi).

Example:

```
from langdetect import detect
text = "Tum kya kar rahe ho bro?"
lang = detect(text)  # returns 'hi' for Hindi
```

Free Handling for Indian Languages:

- Hindi, Tamil, Bengali → By using IndicTrans + IndicNLP.
- Code-mixed → Combine outputs from mBERT + XLM-R (free HuggingFace models).

7.3 NLP Processing (Free Libraries)

It uses open-source libraries for all NLP steps.

- NLTK: It handles tokenization, stopword removal, and sentence splitting.
- SpaCy: Fast tokenization and dependency parsing for production use.
- PyThaiNLP: This is optional and helps with Thai benchmarks or multilingual tests.
- Emoji (Python package): Maps emojis to text so sentiment can be captured.
- Slang normalization dictionaries: It uses free, custom mappings to normalize social media slang.

7.4 Transformer Teacher Models (Free and Open-Source).

The MTKD framework uses free pretrained transformer models available on HuggingFace.

- mBERT: bert-base-multilingual-cased.
- XLM-RoBERTa: xlm-roberta-base.
- MuRIL: Free Indian multilingual model for Hindi, Bengali, Tamil, and Telugu.
- ThaiBERT: This is optional and used for benchmarking.

All models are publicly available, with no paid subscription required [23][24].

7.5 Knowledge Distillation Framework

It relies on free and open-source tools for training and inference.

- XGBoost (Python): Free, open-source tree boosting library.
- scikit-learn: It handles preprocessing, train–test splits, and metrics.
- NumPy / pandas: These are used for data handling.

Training process:

- It distills soft labels from the teacher models.
- It takes a weighted average by F1-score per language.
- It trains XGBoost on the distilled labels for fast inference.

7.6 Emotion and Sarcasm Detection (Free)

- GoEmotions-BERT: It uses a HuggingFace model trained on 28 emotion classes; this is free to download and fine-tune.
- BiGRU Sarcasm Detection: It uses a Keras/TensorFlow implementation; both frameworks are free.
- Datasets: It uses free Hindi-English tweet datasets from Kaggle or research repositories for training.

7.7 Explainability with SHAP

- SHAP library: This is a free, Python-based interpretability tool.
- Generates token-level explanations for XGBoost predictions.
- It is critical for moderator transparency and ethical AI.

7.8 Backend Infrastructure (Free Options)

- FastAPI: Free, async Python framework for REST APIs.
- Flask: It is a lightweight alternative for quick prototypes.
- Server Hosting:
  - Railway Free Tier: It offers about 500–1000 hours per month.
  - Render Free Tier: It offers about 750 free hours per month.
  - Replit: Supports free prototyping and hosting.

7.8 Frontend and User Interface (Free)

- ReactJS: It is free and open-source library for building user interfaces.
- Charts/Heatmaps: react-chartjs-2 or Plotly.js, both free.
- Deployment: Firebase Hosting (free tier) for static React apps.

7.9 Database and Hosting (Free Alternatives)

MongoDB Atlas (Free Tier): 512 MB storage; a good fit for JSON documents.

- Firebase Firestore (Free Tier): Up to 50,000 reads/writes per day.
- Hosting: Free tiers on Railway, Render, or Firebase Hosting for small deployments.

VIII. EXPECTED OUTCOMES

The proposed approach to multilingual cyberbullying detection in code-mixed Indian languages is expected to produce outcomes across multiple technical, practical, and social levels. The outcomes will be divided into four main areas: (1) system design and overall project outcomes, (2) technology and methods, (3) deployment and project management, and (4) societal and user impact.

8.1 Systems Design and Final Project Outcomes

This project will be based on modular and scalable architecture that integrates the data source and acquisition, pre-processing, model training, classification, and visualization layers.

- User Interface (UI): The principal interface will be a web dashboard built with React (frontend) and FastAPI (backend) that will enable monitoring of real-time social media streams from Twitter, Reddit, YouTube, and Facebook. Importantly, it will have multilingual input support for English, Hindi, Hinglish, Bengali, Tamil, and Indic Languages, providing support for communities across various languages.
- Data Pipeline: Data will be obtained using free APIs for Twitter (via Tweepy [20]), Reddit (PRAW [29]), Pushshift [30], and YouTube Data API v3 [31]. Furthermore, public reports

(such as from the Facebook Transparency Center [32]) will also be appended to use as published benchmark references for harmful content detection all at once. The preprocessing pipeline will consist of tokenization, transliteration, and emoji treatment, primarily using IndicNLP [21] and emoji libraries [25] library, so that the system can detect sarcasm, codeswitching, and the emotion of a situation in context.

- Visualization and Alerts: The dashboard will be designed as a professional moderation console. The left panel will show feeds of live data streams (tweets, posts, comments); the middle panel will display classification results; and the right panel will describe the Bystander Bullying scale, severity, toxicity categorizations, and emotional intensity output. Alerts will be sent to the moderators when the system detects high-risk bullying content.

8.2 Technical and Methodological Outcomes

The technical outcomes will focus on enhancing detection and monitoring accuracy; improving efficiency of model performance; and explainability in multilingual and code-mixed contexts.

- Enhanced Multilingual Detection: The improved multilanguage detection process will deploy pre-trained transformers, such as MuRIL [16] (a multilingual variant of BERT), IndicTrans [13] (a multi-script transformer appropriate for Indic languages), and RoBERTA [8], which will have all subsequently outperform traditional monolingual baselines. The inclusion of emoji in detection [12] will also improve sentiment detection performance, especially within sarcasm or humour-orientated bullying context.
- Evaluation Metrics: In addition to accuracy, recall, and F1-score [18], the system will utilize ROC curve analysis [33] to present the trade-offs of values between sensitivity and specificity for more reliable, actionable outcomes in high-stakes decision making.
- Comparative Model Benchmarks: The results of the proposed pipeline will be compared to the most recent abusive language detection models such as HateBERT [34] to show how pre-trained domain-specific models improve realizable outcomes in abusive and cyber bullying context.
- Explainability: To enhance trust and incentivize adoption, explainable AI [6] processes will be incorporated in which words, phrases, or emojis are highlighted as influences that provoke bullying classification. For example, in a Hinglish tweet that states “Tu loser hai 😂”, the system will identify “loser” as a related toxic word and that the laughing emoji as reinforcement sarcastically.
- Efficiency through Knowledge Distillation: Lightweight transformer-based models [17] will be trained to mimic the performance of large models while requiring less resources, creating a knowledge transfer model that can be utilized in data-poor settings.

8.3 Project Execution and Development

The project will encompass a complete step-by-step procedure, from data collection through to live deployment to be a scalable end to end solution.

- Data Collection and Storage: Where data is collected from Twitter [20], Reddit [29] and YouTube [32] APIs, data will be maintained in either MongoDB Atlas [27] or Firebase Firestore [28] in support of scaling. Data normalization and cleaning will be conducted in preprocessing queues, prior to feeding the model.
- Training and Evaluation: Training will utilize either Google Colab [26] or Kaggel [21] cloud services with the support of any available GPU hardware. Evaluation will involve multi-class classification metrics supplemented with ROC analysis [33].
- Deployment Infrastructure: The backend model will be deployed via FastAPI [26], whereas the web dashboard will connect with REST APIs. Demonstrations for the public or academic instances, would utilize Heroku’s [35] free cloud-hosting service and therefore eliminate the need for expensive infrastructure.

- Monitoring and Human-in-the-Loop Feedback Loop: A feedback loop will log moderation actions taken by the moderators in order to facilitate re-training suggesting models learn from adapting slang, changes in cultural variances, and detect emerging bullying patterns.

8.4 Social and User-level Outcomes.

Any effects at the societal and user level are equally as important as the technical deliverables of the system.

- Intervening Early to Online Harassment: The system will help mods, NGOs, schools, and online platforms catch early stages of harassment and act on it early on, as prolonged exposure to online harassment has substantial psychological and emotional costs [9].
- Safer and More Inclusive Digital Environment: Organizations including schools and workplaces will be able to implement the system in order to protect vulnerable groups, notably children/teens. This is also consistent with the UNESCO "Feasibility Study for Safe and Inclusive Digital Learning Spaces for Children" [36].
- Supporting Multilingual Communities: The system will cater to the many Indic languages and code-mixing prevalent in South Asian communities, creating informal collective leverage. This also closes the gap of moderating in pre-dominantly monolingual systems [2].
- Different Applicability Use-Case Scenarios: The solution will be applicable to more than just social media moderation, as it will also have applications in education, workplace harassment, and law enforcement to name a few.
- Contributing to Research and Open Resources: The projects data sets and trained models will be published as multilingual open-access research resources for the academic and industry to innovate on [15].

IX. FUTURE SCOPE

Detecting cyberbullying in multilingual and code-mixed contexts is a burgeoning area of research and the current work provides a number of pathways for future work.

9.1 Extending to More Languages and Dialects

The current system works with English, Hindi and a subset of Indic languages. Future expanding this framework to other regional dialects and minority languages, such as Assamese, Odia, and Konkani, has benefits, particularly because many of these languages have few or no labelled datasets [2]. With the increasing availability of cross-lingual transfer learning techniques [13], the thinking would be that those scarce languages could be modelled much easier, without large annotated corpora, but with some distance from the original dataset.

9.2 Syncing with Audio and Video Streams

This study primarily concerns itself with the text data related to cyberbullying. Future work may explore what it takes to bulk out the system, in a multimodal way, with YouTube videos, live audio chats, and memes. Work integrating APIs, such as the YouTube Data API v3 [31], and even speech-to-text technologies in real time should be explored, to collect comments, transcripts, and captions as they occur. The multimodal approach will strengthen detection on platforms that provide bullying in multiple formats.

9.3 Real-Time Deployment at Scale

This implementation illustrates a scalable cloud deployment of FastAPI [26] on Heroku [35]. Additional work could focus on edge deployments for mobile devices and interconnection of moderation pipelines of social platforms. This enables real-time detection of abusive content and flagging it and removing it before the use becomes widespread.

9.4 Contextual and Psychological Analysis

Cyberbullying detection cannot be limited to keyword or sentiment analysis. The future scope should provide, in addition, some form of

psychological perspectives and behavioral modelling to distinguish between banter, sarcasm, or real bullying [12]. This can be accomplish when offer the expertise of psychologists, educators and NGOs [36] or a more human-centered moderation.

9.5 Adaptive Learning with Continuous Feedback

The models may become obsolete as internet slang, memes, and emoji use develop and change. Therefore, masters of all Interactive Learning can be embedded in the future with a feedback link so moderators' decision can automatically retrain models and provide ongoing adaptability [6].

9.6 Other Application Domains

There are additional domains, other than social media, this framework could be applied to:

- Education: Monitoring forum or chats in classrooms are the right environment and opportunity to inform students they are not subjected to online harassment.
- Work settings: Employees should be assured their corporate communication tools are free from bullying and harassment.
- Law enforcement: To monitor that organized cyber harassment and campaign hate or negative organizations are not beginning to take shape [34].

9.7 Engagement with Global Digital Safety Initiatives

Future systems can collaborate with programs led by UNESCO [36] and other digital safety initiatives to build global databases of annotated bullying data. This would support greater model quality and cross-jurisdictional collaboration to combat cyber harassment.

X. CONCLUSION

This project proposed a multilingual cyberbullying detection system for the heterogeneity both linguistically and culturally of India, with features such as code-mixed language, the use of emojis, and sarcasm detection. Implementation drew on plug-in free APIs for real time data acquisition (Twitter [20], Reddit [29] and YouTube [31]), state-of-the-art natural language processing (NLP) models (MuRIL [16] and HateBERT [34]), and cloud deployment options (Heroku [35]). The result is a well-defined system that is built to be robust and scalable.

The contributions of this work are summarized as follows:

1. A composite pipeline of cyberbullying detection in multilingual and code-mixed space.
2. Apart from sample rates presented using the baseline models, the translator-based models that lead to improvements can see significant increases in detection rates.
3. A dashboard that provides the moderator with real-time data with visual representation for alerts.
4. The work meets international standards for online safety (UNESCO [36]), which gives the work immediate topicality.

On the societal side, the project highlighted an AI-powered systems can be a source of promoting a safer and more inclusive digital scenario by its use with vulnerable populations, in particular children, adolescents and minority groups [9], [36].

To conclude this study, the focus has been to address the technical side of the multilingual cyberbullying and detection problem, with more importance placed on the deployment and social impact. As the project grows to involve more multimodal, multimodal learning and global engagement, the project has the ability to be a worldwide model for online safety.

REFERENCES

1. S. Prasomphan, "Enhance Social Network Bullying Detection Using Multi-Teacher Knowledge Distillation With XGBoost Classifier," *IEEE Access*, 2025. Available: https://www.researchgate.net/publication/392212453_Enhance_Social_Network_Bullying_Detection_using_Multi-Teacher_Knowledge_Distillation_with_XGBoost_Classifier
2. A. Patra, A. Das, B. Das, and S. Das, "Sentiment Analysis of Code-Mixed Indian

Languages: SAIL Code-Mixed Shared Task," arXiv preprint arXiv:1803.06745, 2018. Available: https://arxiv.org/abs/1803.06745
3. S. Mehendale, D. Dodia, and H. Palshetkar, "A Review on Cyberbullying Detection Using Machine Learning in English and Hinglish," *SSRN*, 2022. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4116153
4. S. Prasomphan, "Enhance Social Network Bullying Detection Using Multi-Teacher Knowledge Distillation With XGBoost Classifier," *IEEE Access*, 2025. [Online]. Available: https://www.researchgate.net/publication/392212453
5. D. Demszky, D. Movshovitz-Attias, J. Ko, A. Cowen, G. Nemade, and S. Ravi, "GoEmotions: A Dataset of Fine-Grained Emotions," *arXiv preprint arXiv:2005.00547*, 2020. [Online]. Available: https://arxiv.org/abs/2005.00547
6. S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in *Advances in Neural Information Processing Systems*, vol. 30, 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf
7. A. Mishra, A. Jain, and P. Bhattacharyya, "A Deep Learning Approach to Sarcasm Detection in Social Media," *arXiv preprint arXiv:1605.01159*, 2016. [Online]. Available: https://arxiv.org/abs/1605.01159
8. Y. Liu, M. Ott, N. Goyal, et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," *arXiv preprint arXiv:1907.11692*, 2019. [Online]. Available: https://arxiv.org/abs/1907.11692
9. UNICEF India, "UNICEF calls for concerted action to prevent online bullying," *Press Release*, 2021. Available: https://www.unicef.org/india/press-releases/safer-internet-day-unicef-calls-concerted-action-prevent-bullying-and-harassment
10. V. Srivastava and M. Singh, "Challenges and Considerations with Code-Mixed NLP for Multilingual Societies," *arXiv preprint arXiv:2106.07823*, 2021. Available: https://arxiv.org/abs/2106.07823
11. K. Maity, R. Jain, P. Jha, and S. Saha, "Explainable Cyberbullying Detection in Hinglish: A Generative Approach," *IEEE Transactions on Computational Social Systems*, vol. 11, no. 3, pp. 3338–3347, 2024. [Online]. Available: https://doi.org/10.1109/TCSS.2023.3333675
12. C. Felbo, A. Mislove, A. Søgaard, I. Rahwan, and S. Lehmann, "Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm," *Proc. Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1615–1625, 2017. [Online]. Available: https://aclanthology.org/D17-1169
13. A. Bapna et al., "IndicTrans: An effective transformer-based model for English–Indic translation," *Proc. 2022 Conf. North American Chapter of the ACL: Human Language Technologies*, 2022. [Online]. Available: https://aclanthology.org/2022.naacl-main.58
14. J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *Proc. NAACL-HLT*, pp. 4171–4186, 2019. [Online]. Available: https://aclanthology.org/N19-1423
15. A. Conneau et al., "Unsupervised cross-lingual representation learning at scale," *Proc. 58th Annual Meeting of the ACL*, pp. 8440–8451, 2020. [Online]. Available: https://aclanthology.org/2020.acl-main.747 K. Khanuja et al., "MuRIL: Multilingual representations for Indian languages," arXiv preprint arXiv:2103.10730, 2021. [Online]. Available: https://arxiv.org/abs/2103.10730
16. K. Khanuja et al., "MuRIL: Multilingual representations for Indian languages," arXiv preprint arXiv:2103.10730, 2021. [Online]. Available: https://arxiv.org/abs/2103.10730
17. G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," arXiv preprint arXiv:1503.02531, 2015. [Online]. Available: https://arxiv.org/abs/1503.02531
18. T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, pp. 785–794, 2016. [Online].

Available: https://doi.org/10.1145/2939672.2939785
19. A. Mishra, A. Jain, and P. Bhattacharyya, "A deep learning approach to sarcasm detection in social media," arXiv preprint arXiv:1605.01159, 2016. [Online]. Available: https://arxiv.org/abs/1605.01159
20. Twitter Developers, "Tweepy Python Client for Twitter API v2," [Online]. Available: https://docs.tweepy.org/en/stable/
21. GitHub, "IndicNLP Library for Indian Languages NLP," [Online]. Available: https://github.com/anoopkunchukuttan/indic_nlp_library
22. GitHub, "IndicTrans: Transformer-Based English to Indic Transliteration," [Online]. Available: https://github.com/AI4Bharat/indicTrans
23. Hugging Face, "Transformers Library," [Online]. Available: https://huggingface.co/docs/transformers/index
24. Google AI, "MuRIL: Multilingual Representations for Indian Languages," [Online]. Available: https://ai.googleblog.com/2021/03/muril-multilingual-representations-for.html
25. Python emoji Package Documentation, [Online]. Available: https://pypi.org/project/emoji/
26. FastAPI Documentation, [Online]. Available: https://fastapi.tiangolo.com/
27. MongoDB Atlas, [Online]. Available: https://www.mongodb.com/cloud/atlas
28. Firebase Firestore, [Online]. Available: https://firebase.google.com/docs/firestore
29. PRAW (Python Reddit API Wrapper), [Online]. Available: https://praw.readthedocs.io/
30. Pushshift Reddit API, [Online]. Available: https://pushshift.io/
31. Google Developers, "YouTube Data API v3," *Google Developers*, 2025. [Online]. Available: https://developers.google.com/youtube/v3
32. Meta, "Community Standards Enforcement Report," *Facebook Transparency Center*, 2023. [Online]. Available: https://transparency.fb.com
33. T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006. [Online]. Available: https://doi.org/10.1016/j.patrec.2005.10.010
34. M. Caselli, V. Basile, E. Mitrović, and B. Nissim, "HateBERT: Retraining BERT for abusive language detection in English," *arXiv preprint*, arXiv:2010.12472, 2020. [Online]. Available: https://arxiv.org/abs/2010.12472
35. Salesforce, "Heroku: Free cloud application hosting," *Heroku*, 2025. [Online]. Available: https://www.heroku.com/free UNESCO, "Ending cyberbullying: Promoting safe and inclusive digital spaces for children," *UNESCO Report*, 2022. [Online]. Available: https://unesdoc.unesco.org/ark:/48223/pf0000381649