GenEx: A Commonsense-aware Unified Generative Framework for Explainable Cyberbullying Detection

Krishanu Maity[1*], **Raghav Jain**[1*], **Prince Jha**[1*], **Sriparna Saha**[1] and
Pushpak Bhattacharyya[2]
[1]Department of Computer Science and Engineering, Indian Institute of Technology Patna
[2] Department of Computer Science and Engineering, Indian Institute of Technology Bombay
krishanu_2021cs19@iitp.ac.in, raghavjain106@gmail.com *

Abstract

With the rise of social media and online communication, the issue of cyberbullying has gained significant prominence. While extensive research is being conducted to develop more effective models for detecting cyberbullying in monolingual languages, a significant gap exists in understanding code-mixed languages and the need for explainability in this context. To address this gap, we have introduced a novel benchmark dataset named *BullyExplain* for explainable cyberbullying detection in code-mixed language. In this dataset, each post is meticulously annotated with four labels: bully, sentiment, target, and rationales, indicating the specific phrases responsible for identifying the post as a bully. Our current research presents an innovative unified generative framework, *GenEx*, which reimagines the multitask problem as a text-to-text generation task. Our proposed approach demonstrates its superiority across various evaluation metrics when applied to the *BullyExplain* dataset, surpassing other baseline models and current state-of-the-art approaches.[1]

Disclaimer: The article contains profanity, an inevitable situation for the nature of the work involved. These in no way reflect the opinion of authors.

1 Introduction

Cyberaggression, which encompasses various types and forms of aggressive behavior conducted through information and communication technologies (ICT), including cyberbullying and hate speech. In this study, we are working on cyberbullying detection. Hate speech and cyberbullying are conceptualized differently. Hate speech (Hawdon et al., 2017) pertains to online assaults targeting collective identity, while cyberbullying (Kowalski et al., 2014) is characterized by a repetitive, harmful intent and a power imbalance. According to research from the Pew Research Center[2], approximately 40% of social media users have encountered cyberbullying, resulting in emotional distress, anxiety, diminished self-esteem, momentary fear, and even suicidal ideation (Sticca et al., 2013).

In the past decade, the majority of cyberbullying detection research has concentrated on monolingual social media data, using conventional machine learning (Dadvar et al., 2014; Dinakar et al., 2011) and deep learning (Agrawal and Awekar, 2018; Maity et al., 2023) models. However, these studies primarily sought to enhance detection performance without delving into the realm of explainability. As we enter the era of explainable artificial intelligence (Gunning et al., 2019), the need for providing interpretations behind machine learning decisions has become paramount. Furthermore, the prevalence of code-mixing (Myers-Scotton, 1997), where multiple languages are interchanged within speech, is rapidly increasing. A comprehensive analysis of over 50 million tweets revealed that approximately 3.5% of them incorporated code-mixing. Therefore, addressing the intricacies of code-mixed languages should be a central concern at this juncture.

Numerous studies on multitasking (Caruana, 1997) have demonstrated the effectiveness of incorporating closely related auxiliary tasks alongside the main task to improve the performance of primary tasks (such as cyberbullying detection (Maity and Saha, 2021b), complaint identification (Singh et al., 2021)). A common configuration for multitask models involves a shared encoder that consolidates data representations from diverse tasks, accompanied by multiple task-specific layers or heads linked to this central encoder. Nonetheless,

[*] The first three authors contributed equally to this work and are jointly the first authors.

[1]The code and dataset are available at https://github.com/MaityKrishanu/GenEx_Cybebullying.

[2]https://www.pewresearch.org/internet/2017/07/11/online-harassment-2017/

Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pages 16632–16645
December 6-10, 2023 ©2023 Association for Computational Linguistics

this approach is associated with certain limitations, including the risk of negative transfer (Crawshaw, 2020), wherein multiple tasks, instead of aiding the learning process, begin to hinder it. Additionally, concerns emerge regarding model capacity (Wu, 2019), as an overly expansive shared encoder can impede the effective transfer of information between various tasks.

In an attempt to overcome the above-mentioned challenges, in this paper, we have developed an explainable cyberbullying dataset (BullyExplain) and a unified generative approach to solving four tasks simultaneously. **Task 1 Cyberbullying Detection (CD):** Given a text, detect whether it is bully or non-bully. **Task 2 Sentiment Analysis (SA)** Given a text, detect whether it is positive, negative, or neutral. **Task 3 Target Identification (TI)** Detect the type of targets that each bully post points to. **Task 4 Rationales Detection (RD):** The task involves pinpointing text segments within the source text that substantiate a classification judgment. We introduce a commonsense-aware unified generative framework, $GenEx$, which excels in concurrently addressing all four tasks within a text-to-text generation environment, with a focus on prioritizing CD and RD as primary tasks, while treating TI and SA as secondary or auxiliary tasks. In summary, our contributions encompass two key aspects: (i) the achievement of explainable cyberbullying detection within code-mixed contexts and (ii) the innovative framing of the multi-task problem as a text-to-text generation endeavor.

2 Related Works

In this section, we have undertaken a review of cyberbullying detection research encompassing both monolingual and code-mixed data, as well as rationales.

Works on Monolingual Data: In the realm of monolingual data, Dinakar et al. (2011) conducted an investigation into cyberbullying detection, employing a dataset comprising 4500 YouTube comments and various binary and multiclass classifiers. Additionally, Balakrishnan et al. (2020) introduced a model that leverages diverse machine learning approaches and the psychological attributes of Twitter users for cyberbullying detection. Bu and Cho (2018) put forward an ensemble approach that amalgamates two deep learning models, employing character-level CNN and long-term recurrent convolutional networks (LRCN) for cyberbullying comment detection. CyberBERT, a BERT-based framework developed by Paul and Saha (2020), achieved state-of-the-art results across multiple datasets, including Formspring, Twitter, and Wikipedia.

Works on Code-mixed Data: In the context of code-mixed data, a cyberbullying dataset featuring Hindi-English code-mixed language was curated by Maity and Saha (2021a). They further proposed a deep learning framework that combines BERT and Capsule network methodologies, achieving an accuracy of 79.28%. Recent studies have delved into the impact of writing system changes (WSCs) on the Chinese language concerning affective and emotion analysis of social media text, as highlighted by Xiang et al. (2019). These studies have underscored the value of WSCs in enhancing various analytical tasks. Additionally, Kumar et al. (2018) contributed to this field by assembling an aggression-annotated corpus featuring 21k Facebook comments and 18k tweets in Hindi-English code-mixed language.

Works on Rationales: The realm of rationales was explored by Zaidan et al. (2007), who introduced the concept of rationales, wherein human annotators underlined text sections that substantiated their tagging decisions. Their findings demonstrated the utility of rationales in enhancing sentiment classification performance. Furthermore, Mathew et al. (2020) presented the HateXplain dataset, tailored for hate speech detection. Pavlopoulos et al. (2021) introduced the task of detecting toxic post spans in English texts responsible for the toxicity, and later released the "TOXICSPANS" dataset. Lastly, Ravikiran and Annamalai (2021) created the "DOSA" dataset, featuring English-Tamil code-mixed posts annotated with corresponding toxic spans.

3 Dataset Development

3.1 Data Collection

To commence, an extensive literature review was conducted to identify existing code-mixed cyberbullying datasets. Two such datasets, specifically in Hindi-English code-mixed tweets, were uncovered (Maity and Saha, 2021a),(Maity and Saha, 2021b). The dataset chosen for further annotation, referred to as BullySent (Maity and Saha, 2021b), had prior annotations for bully and sentiment labels. The labels for each task are detailed in Table 1.

Table 1: List the labels for each task.

Task	Class labels
Cyberbullying Detection (CD)	Bully, Non-Bully
Sentiment Analysis (SA)	Positive, Negative and Neutral
Target Identification (TI)	Religion, Gender, Organization, Community, Profession, Attacking-Relatives-and-Friends, and Miscellaneous
Rationales Detection (RD)	Each word of the input text is marked with 1 or 0, where 1 indicates the rationales.

3.2 Data Annotation

To better clarify the annotation process, we split the annotation section into two subsections: (i) Annotation Training and (ii) Main Annotation.

Annotation training Three Ph.D. scholars oversaw the annotation process, well-versed in cyberbullying and offensive content, and the actual annotations were conducted by three undergraduate students proficient in both Hindi and English. Initially, we hired a group of master's students in linguistics who volunteered via our department email list and compensated them with gift vouchers and an honorarium. Previously, the BullySent dataset (Maity and Saha, 2021b) included annotations for bully class (Bully / non-bully) and sentiment class (Positive / Neutral / Negative). For the specific annotations of rationales and target labels, we focused exclusively on the bully tweets. To train our annotators, we required gold standard samples with annotations for rationale and target labels. Our expert annotators randomly selected 300 samples (tweets), highlighted words as rationales for textual explanation, and assigned suitable target classes. For rationale annotation, we followed a similar strategy as outlined in (Mathew et al., 2020). Each word in a tweet was marked with either 0 or 1, where 1 indicated it was a rationale. We considered seven target classes (Religion, Sexual-Orientation, Attacking-Relatives-and-Friends, Organization, Community, Profession, and Miscellaneous) as defined in (Mathew et al., 2020) and (Pramanick et al., 2021). Expert annotators engaged in discussions to resolve any differences and created 300 gold standard samples with rationale and target annotations. These 300 annotated examples were divided into three sets, each containing 100 samples, to facilitate a three-phase training process. After each phase, expert annotators collaborated with novice annotators to correct any inaccuracies in the annotations, and the annotation guidelines were updated as needed. Following the conclusion of the third round of training, the top three annotators were selected to annotate the entire dataset.

Main annotation

We initiated our primary annotation process with a smaller batch of 100 samples, gradually increasing it to 500 as our annotators gained proficiency in their tasks. To maintain consistency and agreement among the annotators, we made corrections based on their previous errors in each batch. After completing each set of annotations, the final rationale labels were determined through a majority voting method. In cases where the selections of the three annotators diverged, we sought the assistance of an expert annotator to resolve any ties. We emphasized to the annotators that they should annotate the posts without any bias related to demographics, religion, or other factors. To assess the quality of the annotations, we utilized the Fleiss' Kappa score (Fleiss, 1971) to calculate the inter-annotator agreement (IAA), which yielded scores of 0.74 and 0.71 for the rationale detection (RD) and Target Identification (TI) tasks, respectively. These scores affirm the dataset's overall quality and reliability.

3.3 Dataset Statistics

Within the *BullyExplain* dataset, an average of 4.97 words are highlighted per post, with each tweet containing an average of 23.15 words. The dataset comprises a total of 6,084 samples, of which 3,034 are categorized as non-bully, and the remaining 3,050 samples are labeled as bully. Furthermore, the sentiment distribution in the dataset reveals 1,536 tweets with positive sentiments, 1,327 with neutral sentiments, and the remainder expressing negative sentiments. In Table 3, we present statistics regarding the Target class in the *BullyExplain* dataset, where it becomes apparent that roughly one-third of the total bully samples (3,050) fall into the category of "Attacking-Relatives-and-Friends" (ARF), accounting for 1,067 instances. This data sheds light on the prevalent nature of cyberbullying, where victims' relatives and friends often become the primary targets.

Sample entries from the *BullyExplain* dataset are showcased in Table 2. For a more comprehensive understanding of how inter-annotator agreement is computed on RD task, please refer to the appendix in Section G.

Table 2: Here are some examples from the annotated *BullyExplain* dataset, with rationale tokens highlighted in green.

Tweet	Bully Label	Sentiment Label	Target
T1: *Your sister is dumb, o sirf entertainment keliye Sahi hai* **Translation:** our sister is dumb; she is only suitable for entertainment.	Bully	Negative	ARF
T2: *Wo har pal khubsurat ho jata hai jisme tum shamil hote ho* **Translation:** All moments become beautiful in your presence..	Non-bully	Positive	NA

Table 3: Statistics of target classes.

Religion	Gender	ARF	Organization	Community	Profession	Miscellaneous
161	166	1067	525	173	364	594

4 Methodology

In this section, we introduce *GenEx* model for explainable cyberbullying detection, which is illustrated in Figure 1.

4.1 Redefining Cyberbullying Detection Task as Text to Text Generation Task

In this work, we put forth a text-to-text generation approach to tackle explainable cyberbullying detection and other related auxiliary tasks together in a unified framework. To formulate this as a text generation problem, we first create a natural language target sequence Y_i for each input sentence X_i during training. Y_i is constructed by concatenating the labels for all four tasks related to the input X_i. For the rationales detection task, we only take into account words $\{x\}$ in the input sentence that are associated with offensive labels in the set $R\_\{Labels\}$, represented as $R\_\{Off\}$. If the set of offensive labels $R\_\{Off\}$ is empty for a given input text, meaning there are no words tagged as offensive, we use the token NONE to indicate there are 0 offensive tokens in the text. This allows us to handle the case where no offensive content is present in a consistent manner. Finally, the target sequence Y_i is represented as :

$$Y_i = \{< R_{Off} > [b][t][s]\} \qquad (1)$$

where R_{Off}, b, t, and s represent the corresponding rationales, bully, target, and sentiment label of an input post X_i. We have shown two samples illustrating conversions of labels into natural language form in Table 11 (shown in Appendix E). To construct the final target sequence Y_i as in Equation 1, we add special delimiter characters after each task's prediction text. This allows us to extract the prediction for each individual task during testing and evaluation. With this formulation, the problem is reformulated as follows: Given an input sequence X_i, the goal is to generate an output sequence $Y_i^{'}$ that contains all the prediction texts as defined in Equation 1. This is achieved using a generative model G such that: $Y_i^{'} = G(X_i)$.

4.2 Commonsense Aware CD

We present *GenEx* (illustrated in Figure 1), a unified generative framework for explainable cyberbullying detection that incorporates commonsense knowledge. Our approach can be broken down into three main components: (1) ContextualCommonsense Extractor, (2) Commonsense-infused transformer model, and (3) Reward-Centric Learning Framework.

4.2.1 ContextualCommonsense Extractor

To enhance the context and depth of typically short and brief tweets, we deploy a module specialized in extracting commonsense reasoning. Drawing from the extensive ATOMIC dataset (Sap et al., 2019), our module enriches tweet content for better interpretability. Comprising a rich tapestry of 880,000 triplets, the ATOMIC database forms the backbone of this module. Each triplet is constructed as (e, r, cs) where e denotes an event, r denotes a commonsense relation, and cs denotes the inferred commonsense reasoning. The ATOMIC commonsense knowledge base contains events e, each with associated commonsense reasoning cs across six relation types r that describe inferences about the event's entity. For instance, xEffect captures the effect on the entity, while xNeed describes the entity's need from the event. In our cyberbullying detection problem, the event is the user's tweet. To understand the tweet's impact and motivation, we utilize two relevant ATOMIC relations – xEffect and xIntent. By focusing only on these relations rather than reproducing all details of the knowledge base, we extract pertinent commonsense information while avoiding excessive similarity to the original source. To produce commonsense insights from tweets, we utilize a BART-based language model (Lewis et al., 2020) called COMET (Hwang

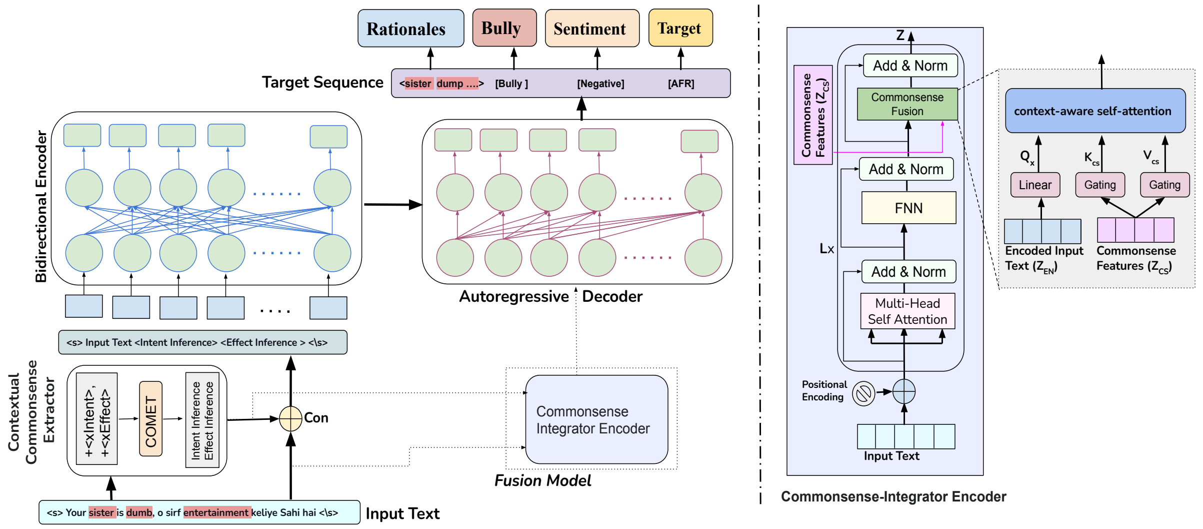
Rationales
Bully
Sentiment
Target
Target Sequence
<sister dump ....> [Bully] [Negative] [AFR]
Bidirectional Encoder
Autoregressive Decoder
<s> Input Text <Intent Inference> <Effect Inference > <\s>
Contextual Commonsense Extractor
+<xIntent>, +<xEffect>
COMET
Intent Inference Effect Inference
Con
Commonsense Integrator Encoder
Fusion Model
<s> Your sister is dumb, o sirf entertainment keliye Sahi hai <\s>
Input Text
Z
Add & Norm
Commonsense Fusion
Commonsense Features (Z_CS)
Add & Norm
FNN
Lx
Add & Norm
Multi-Head Self Attention
Positional Encoding
Input Text
Commonsense-Integrator Encoder
context-aware self-attention
Q_x
K_cs
V_cs
Linear
Gating
Gating
Encoded Input Text (Z_EN)
Commonsense Features (Z_CS)

Figure 1: The overall architecture of the proposed mode, *GenEx*. The two variations of our proposed model; 1) *GenEx-Con*: where input text and commonsense reasoning are concatenated and then fed into an encoder-decoder module, and 2) *GenEx-Fuse*: where input text and commonsense reasoning are fed into a commonsense aware encoder (see right side) followed by decoder

et al., 2020), which has been previously trained and later fine-tuned on the aforementioned ATOMIC dataset. This training makes the model particularly adept at generating commonsense reasoning for novel events (Sahand Sabour, 2021). Due to the absence of a commonsense reasoning dataset in Hindi-English code-mixed language and given that COMET is primarily designed for English, we first convert our code-mixed tweets into English. We adopt the translation strategy outlined in prior research for this purpose (Gautam et al., 2021). Our ContextualCommonsense Extractor operates as follows:

First, for each input tweet X_i, we append the xIntent and xEffect relation tokens to the English translation. We then input each relation-appended tweet to the pre-trained COMET model, which generates commonsense reasoning texts $cs^{r_{Intent}}$ and $cs^{r_{Effect}}$ corresponding to the xIntent and xEffect relations. To get the final commonsense reasoning CS for tweet X_i, we concatenate the $cs^{r_{Intent}}$ and $cs^{r_{Effect}}$ texts as follows: $CS = cs^{r_{Intent}} \oplus cs^{r_{Effect}}$.

4.2.2 Commonsense-infused transformer model

To utilize the commonsense reasoning gleaned from the previous module, we introduce two versions of an encoder-decoder architecture equipped with commonsense awareness ($GenEx-Con$ and $GenEx-Fuse$). These architectures are designed to integrate CS into their sequence-to-sequence learning mechanisms, and are described in detail below:

***GenEx-Con* (Concatenation based *GenEx*):** Given an input tweet X_i and corresponding commonsense reasoning CS, the task to generate the target sequence $Y_i^{'}$ can be modeled as the following conditional text generation model: $P_\theta(Y_i^{'}|X_i, CS)$, where θ is a set of model parameters. *GenEx-Con* models this conditional probability as follows:

First, we construct the input T_i by concatenating the input tweet X_i and the corresponding commonsense reasoning text CS. We feed this T_i input to the encoder module, which outputs a hidden representation Z_{EN}. Next, we provide Z_{EN} along with the previous decoded tokens $Y_{<t}$ up to time step $t-1$ to the decoder module. This produces the decoder hidden state Z_{DE}^t at time step t. To predict the output token at time step t given T_i and the previous $t-1$ tokens, we apply the softmax function to Z_{DE}^t:

$$P_\theta(Y_t^{'}|T, Y_{<t}) = F_{softmax}(\theta^T Z_{DE}^t) \qquad (2)$$

where $F_{softmax}$ represents softmax computation and θ denotes weights/parameters of our model.

***GenEx-Fuse* (Fusion based *GenEx*):** We also introduce an approach called *GenEx-Fuse* that integrates commonsense knowledge into the model using a CommonsenseIntegrator Encoder.

Our proposed CommonsenseIntegrator Encoder extends the standard transformer encoder architecture (Vaswani et al., 2017) to combine insights from both the input text and commonsense knowledge. First, the input text X_i is tokenized and converted into embedded sequences. Positional encodings are added to retain order information. This input is fed into our proposed CommonsenseIntegrator Encoder. Beyond the standard Multi-head Self-Attention (MSA) and Feedforward Network (FFN) sub-layers, we introduce a new Commonsense Fusion (CSF) sub-layer to integrate commonsense knowledge.

The encoded input representation Z_{EN} from the MSA and FFN sub-layers is fed into the CSF sub-layer, along with the commonsense feature vector Z_{CS}. Unlike the standard transformer encoder, we implement context-aware self-attention in CSF to enable information exchange between Z_{EN} and Z_{CS}, inspired by Yang et al. (2019). We create query, key, and value matrix triplets for both Z_{EN} (as Q_x,K_x,V_x) and Z_{CS} (as Q_{cs},K_{cs},V_{cs}). The Z_{EN} triplets are obtained by linearly projecting Z_{EN}. The Z_{CS} triplets are generated through a gating mechanism, as described in Yang et al. (2019), which operates as follows: To balance integrating information from the commonsense representation Z_{CS} and retaining original knowledge from the text representation Z_{EN}, we learn matrices λ_K and λ_V. These are used to generate context-aware versions of K_{cs} and V_{cs}, as shown in Equation 3.

$$\begin{bmatrix} K_{cs} \\ V_{cs} \end{bmatrix} = (1 - \begin{bmatrix} \lambda_K \\ \lambda_V \end{bmatrix}) \begin{bmatrix} K_x \\ V_x \end{bmatrix} + \begin{bmatrix} \lambda_K \\ \lambda_V \end{bmatrix} (G_{CS} \begin{bmatrix} U_K \\ U_V \end{bmatrix}) \tag{3}$$

where U_K and U_V are learnable parameters and matrices λ_K and λ_V are computed as follows:

$$\begin{bmatrix} \lambda_K \\ \lambda_V \end{bmatrix} = \sigma(\begin{bmatrix} K_x \\ V_x \end{bmatrix} \begin{bmatrix} W_K^X \\ W_V^X \end{bmatrix} + G_{CS} \begin{bmatrix} U_K \\ U_V \end{bmatrix} \begin{bmatrix} W_K^{CS} \\ W_V^{CS} \end{bmatrix}) \tag{4}$$

where W_K^X, W_V^X, W_K^{CS} and W_V^{CS} all are learnable parameters. σ represents the sigmoid function computation.

After obtaining K_{cs} and V_{cs}, we apply the dot product attention based fusion method over Q_x, K_{cs} and V_{cs} to obtain the final commonsense aware input representation Z computed as follows: $Z = softmax(\frac{Q_x K_{cs}^T}{\sqrt{d_k}})V_{cs}$. Finally, we input the commonsense-enriched input representation Z to an autoregressive decoder.

4.2.3 Reward-Centric Learning Framework

We first initialize our model weights θ using a pre-trained sequence-to-sequence generative model. Then we fine-tune the model with two training objectives: 1) A maximum likelihood estimation (MLE) supervised objective that optimizes the weights θ, as shown in Equation 5.

$$\max_{\theta} \prod_{t=0}^{T} P_\theta(Y_t^{'}|X_i, Y_{<t}) \tag{5}$$

2)In addition to the MLE objective, we also employ a reinforcement learning (RL) reward-based training objective, inspired by Sancheti et al. (2020). Specifically, we use a BLEU-based reward function that measures the overlap between the target sequence Y_i and predicted sequence $Y_i^{'}$. BLEU is used rather than other similarity measures because optimizing based on this reward will encourage the model to generate sequences with a higher overlap with the target. The BLEU-based reward R_{BLEU} is defined as shown at a high level in Equation 6:

$$R_{BLEU} = (BLEU(Y_i^{'}, Y_i) - BLEU(Y_i^g, Y_i)), \tag{6}$$

In the above, $Y_i^{'}$ represents an output sequence sampled from the conditional probability distribution at each decoding timestep (Equation 2), while Y_i^g is the output sequence obtained by greedily maximizing the probability distribution at each step. To maximize the expected BLEU reward R_{BLEU} of $Y_i^{'}$, we employ a policy gradient technique, summarized in Equation 7.

$$\nabla_\theta J(\theta) = R_{BLEU} \cdot \nabla_\theta log P(Y_i^{'}|X_i, CS; \theta) \tag{7}$$

5 Experiments and Results

5.1 Baselines Setup

Standard Baselines: We have experimented with different standard baseline techniques like CNN-GRU, BiRNN, BiRNN-Attention, BERT-finetuned, BART and T5 (Detailed explanation given in Appendix C)

5.2 Findings from Experiments

Table 4 shows and compares the results of CD and RD (main tasks) of our proposed model, *GenEx* with different baseline models. Single task results are also shown in Table 5. From all these reported results, we can conclude the following: **(1)** It can be observed from table 4 that BERT performs best

Table 4: The outcomes of various baseline models and the two novel frameworks, *GenEx-Con* and *GenEx-Fuse* are presented within a multi-task configuration. Regarding the bully tasks, the results are measured in terms of macro-F1 score (F1) and Accuracy (Acc) values. JS: Jaccard Similarity, HD: Hamming distance, and ROS: Ratcliff-Obershelp Similarity. Bold-faced values represent the maximum scores attained; Gray Highlight: statistically significant; ±: standard deviation scores, CD: Cyberbullying Detection, SA: Sentiment Analysis, TI: Target Identification and RD: Rationales Detection.

Model	**CD+RD**					CD+RD+TI					CD+RD+SA					CD+RD+TI+SA				
	Bully		Rationales			Bully		Rationales			Bully		Rationales			Bully		Rationales		
	Acc	F1	JS	HD	ROS	Acc	F1	JS	HD	ROS	Acc	F1	JS	HD	ROS	Acc	F1	JS	HD	ROS
Standard Baselines																				
BiRNN	79.42	79.51	49.51	45.13	50.13	79.44	79.49	49.91	46.78	51.11	79.67	79.66	49.89	46.52	50.18	79.99	79.79	49.96	47.01	51.13
BiRNN+Att	79.93	80.02	50.05	46.14	51.06	79.89	80.03	50.69	46.99	51.75	80.02	80.01	50.48	46.81	51.88	80.15	80.21	51.11	47.23	51.91
CNN GRU	79.22	79.12	50.07	45.15	51.21	79.56	79.51	50.22	46.07	51.25	79.51	79.63	50.21	45.89	51.07	79.86	80.01	51.03	46.62	51.29
BERT	80.41	80.34	53.76	49.52	55.87	80.81	80.72	54.06	49.98	56.13	80.41	80.32	53.61	49.65	55.89	81.06	80.71	54.51	50.03	55.99
	± 0.10	± 0.11	± 0.23	± 0.14	± 0.13	± 0.09	± 0.12	± 0.19	± 0.16	± 0.17	± 0.07	± 0.07	± 0.18	± 0.16	± 0.17	± 0.12	± 0.11	± 0.22	± 0.18	± 0.16
Generative Baselines																				
T5	81.03	81.01	62.26	54.61	69.11	81.14	81.11	62.32	54.31	69.16	80.82	80.71	62.25	54.6	69.2	81.20	81.17	62.32	54.53	69.35
BART	81.73	81.54	62.12	53.83	69.12	81.73	81.74	62.17	54.53	69.22	81.74	81.53	62.17	53.92	69.22	81.92	82.04	62.31	54.62	69.56
mBART	82.13	82.09	62.40	54.53	69.5	82.21	82.13	62.54	54.53	69.51	81.91	81.92	62.34	54.63	69.61	**82.31**	82.30	62.45	54.53	**69.64**
	± 0.05	± 0.06	± 0.12	± 0.09	± 0.09	± 0.04	± 0.05	± 0.15	± 0.10	± 0.11	± 0.04	± 0.04	± 0.12	± 0.08	± 0.89	± 0.06	± 0.07	± 0.13	± 0.11	± 0.10
Proposed Model																				
GenEx-Con	83.23	83.19	64.52	56.61	71.20	83.42	83.40	64.53	56.51	71.32	83.21	83.05	64.42	56.52	71.11	**83.59**	83.57	64.89	56.92	**71.43**
	± 0.03	± 0.03	± 0.07	± 0.06	± 0.05	± 0.03	± 0.04	± 0.07	± 0.08	± 0.08	± 0.03	± 0.03	± 0.06	(± 0.07)	(± 0.07)	(± 0.03)	(± 0.03)	(± 0.08)	(± 0.06)	(± 0.06)
GenEx-Fuse	83.20	83.17	64.34	56.61	71.05	83.31	83.29	64.43	56.14	71.22	83.14	82.91	64.13	56.34	71.08	83.48	83.36	64.59	56.58	71.27
	± 0.02	± 0.02	± 0.04	± 0.03	± 0.04	± 0.03	± 0.03	± 0.05	± 0.06	± 0.06	± 0.02	± 0.02	± 0.04	± 0.05	± 0.05	± 0.03	± 0.03	± 0.04	± 0.05	(± 0.06)

Table 5: Results of different baselines and the two proposed frameworks, *GenEx-Con* and *GenEx-Fuse* in a single task setting (Target sequence has only one particular task's output token).

Model	**CD**		**TI**		**SA**		**RD**		
	Acc	**F1-Score**	**Acc**	**F1-Score**	**Acc**	**F1-Score**	**JS**	**HD**	**ROS**
Standard Baselines									
BiRNN	78.63	78.52	51.14	48.04	65.29	65.21	49.52	45.09	50.12
BiRNN+Atn	79.57	79.43	51.98	48.81	65.12	65.26	50.13	46.07	50.69
CNN GRU	78.72	78.31	51.93	48.87	65.66	65.61	50.11	45.12	51.13
BERT	80.10	**80.09**	54.76	**49.12**	68.71	**67.41**	52.81	49.16	**55.01**
	± 0.11	± 0.12	± 0.23	± 0.24	± 0.13	± 0.13	± 0.15	± 0.12	± 0.12
Generative Baselines									
T5	80.6	80.52	57.08	**55.18**	69.71	69.74	62.35	54.72	69.11
BART	81.1	80.91	55.11	54.14	71.14	**71.15**	62.28	54.23	69.14
mBART	81.58	**81.55**	57.43	54.23	71.22	70.53	62.42	54.57	**69.25**
	± 0.08	± 0.09	± 0.17	± 0.19	± 0.09	± 0.10	± 0.12	± 0.10	± 0.11
Proposed Models									
GenEx-Con	82.52	**82.47**	60.72	**56.44**	73.53	**73.16**	64.14	56.38	**71.17**
	± 0.03	± 0.03	± 0.07	± 0.08	± 0.06	± 0.07	± 0.09	± 0.06	± 0.09
GenEx-Fuse	82.35	82.34	60.35	56.19	73.28	72.89	63.93	56.87	71.12
	± 0.05	± 0.05	± 0.12	± 0.13	± 0.09	± 0.09	± 0.11	± 0.07	± 0.09

in CD and RD tasks over all the multitask variants as compared to other standard baselines. However, all the generative baselines and our proposed models (*GenEx-Con* and *GenEx-Fuse*) can outperform the BERT model by a huge margin showing the superiority of pre-trained sequence to sequence language models. **(2)** In CD+RD+TI+SA multitask setting, our proposed generative model *GenEx-Con* outperformed the best standard baseline BERT by 2.53% (Acc) and 15.44% (ROS) for CD and RD tasks, respectively. As our problem statement has both classification tasks (CD, SA, TI) and sequence labeling tasks (RD), we need a robust model that can handle both types of tasks effectively. Though standard classification models achieve a comparable result for classification tasks, there is a massive fall in performance for the rationale detection task (-15.44%) compared to the proposed generative model. This finding validates our idea of *re-framing the multitask problem as a text-to-text generation task* when solving different types of tasks using a single unified model. **(3)** It is also evident from table 4 that mBART consistently outperforms both T5 and BART baseline models over all the multitask variants as mBART is pre-trained on 50 languages, making it more suitable for a code-mixed dataset. That is why we select mBART as the base model for our proposed models (*GenEx-Con* and *GenEx-Fuse*). **(4)** Both *GenEx-Con* and *GenEx-Fuse* outperform the vanilla mBART model by a margin of: 1) 1.28% and 1.17% in accuracy for CD task, respectively, and 2) 2.27% and 2.00% on an average over JS, HD and ROS metrics for RD task, respectively. **(5)** Surprisingly, the performance of *GenEx-Fuse* does not exhibit notable improvement when compared to *GenEx-Con* as their scores remain closely aligned across all multi-task variations for both tasks. This trend could be attributed to the fact that fusion techniques tend to excel when combining various modalities like vision or acoustics with text data, while certain studies (Sridhar and Yang, 2022) have also indicated that direct concatenation methods yield comparable results. Additionally, the relatively modest dataset size of approximately 6,000 samples may have limited the effectiveness of training a fusion-based model. **(6)** Table 5 reports the result for individual tasks where we train the model only for one task at a time. In a single task setting also, our proposed models can consistently outperform all the standard and generative baselines across all the tasks. **(7)** When we compare table 4 with table 5 (which contains results for single tasks), we can observe that when we add RD task, there is an improvement in performance for CD task for both *GenEx-Con* (Accuracy: 82.52 to 83.23 and F1-score: 82.47 to

83.19) and *GenEx-Fuse* (Accuracy: 82.35 to 83.20 and F1-score: 82.34 to 83.17) models. This illustrates that adding the RD task as an auxiliary task helps the model in making better predictions showing that proposed models can learn the mapping between these two tasks efficiently during the decoding step. **(8)** In the Target Identification (TI) task, the *GenEx-Con* model achieves the highest F1 scores of 56.44 and 58.92 in single-task and multi-task settings, respectively. The relatively lower accuracy in the TI task may be attributed to the imbalance distribution of the Target class. **(9)** Furthermore, we conducted experiments on the English HateExplain dataset to assess our proposed model's robustness. As depicted in Table 8, our proposed model demonstrates remarkable performance gains compared to the state-of-the-art (SOTA) in terms of the explainability tasks (Rationales) while achieving comparable performance in the hate speech detection task.

Please refer to Appendix D in where we showed the results for sentiment analysis and target identification in multitask settings. We can observe from Table 10 that both *GenEx-Con* and *GenEx-Fuse* also outperformed all baselines for SA and TI tasks. We performed a statistical t-test on the outcomes from five distinct runs of our proposed model and the other baseline models, revealing a p-value below 0.05.

5.2.1 Ablation Study

We performed an ablation study of our proposed model to show the effect of reinforcement learning and Commonsense knowledge (Table 6). Removing commonsense knowledge from *GenEx-Con* results in a drop of 1.16% and 1.20% in accuracy and F1-score of CD task, respectively. This performance drop shows that the commonsense extractor module provides extra context to the model, leading to increased CD task performance. When we remove the RL component from *GenEx-Con* and *GenEx-Fuse*, we can see that there is not much drop in the performance of the CD task as compared to the drop when we remove the CS component. This shows that commonsense has more effect on CD tasks than RL-based training. It can also be observed from table 6 that when we remove commonsense from *GenEx-Con*, there is an average drop of 1.55% over all the metrics for the RD task. However, when we remove the RL component from the model, there is an average drop of 1.81% in performance for the RD task. Removing RL from *GenEx-Fuse* also results in an average drop of 1.56% for the RD task. This shows that RL training plays a vital role in improving the performance for RD tasks as the BLEU-based reward function (Equation 6) encourages the model to generate a target sequence close to the golden target sequence. Based on this, we can conclude that commonsense and RL training both helped the model to increase the performance of CD and RD tasks, respectively.

Please see our proposed work's limitation and error analysis in Appendix 6 and A, respectively.

Table 6: Ablation Study (Only CD+RD+TI+SA setting is shown). Here, RL: Reinforcement Learning and CS: Commonsense

Model	**CD+RD+TI+SA**				
	Bully		**Rationales**		
	Acc	**F1**	**JS**	**HD**	**ROS**
GenEx-Con	**83.59**	83.57	64.89	56.92	**71.43**
-RL	83.11	83.06	62.43	55.44	69.95
-CS	82.43	82.37	62.65	55.82	70.13
GenEx-Fuse	83.48	83.36	64.59	56.58	71.27
-RL	82.82	82.91	62.35	55.27	70.14
-(RL+CS)	82.31	82.30	62.45	54.53	69.64

5.3 Comparison with SOTA

Only one prior work exists on the topic of cyberbullying detection in code-mixed Indian languages (Maity and Saha, 2021b), which introduced a multi-task model combining CD as the primary task and SA as a secondary task. To compare our novel "GenEx" model with the current state-of-the-art (SOTA) approach for Hindi-English code-mixed CD, we present the results in Table 7. Our study includes a multi-task variant focusing on two tasks (CD+SA) similar to the SOTA model, where our proposed model outperforms existing techniques with a 1.24% accuracy and a 0.83% F1-score advantage. Furthermore, we evaluated the SOTA results against our "GenEx-Con" model

Table 7: Results of state-of-the-art models and the proposed model on *BullyExplain* dataset; ST: Single Task, MT: Multi-task

Model	**Bully**	
	Acc	**F1**
SOTA		
ST- BERT+VecMap(Maity and Saha, 2021b)	79.97	80.13
MT-BERT+VecMap(Maity and Saha, 2021b)	**81.12**	**81.50**
Ours		
GenEx-Con (CD+SA)	82.36	82.33
GenEx-Con (CD+RD+TI+SA)	**83.59**	**83.57**
Improvements (CD+SA)	1.24	0.83
Improvements (CD+RD+TI+SA)	2.47	2.07

(CD+RD+TI+SA), which surpasses the current state-of-the-art methods by 2.47% in accuracy and 2.07% in F1-score. This comparison underscores the significance of integrating rationale detection and target identification tasks in enhancing CD identification processes.

5.4 Evaluating *GenEx-Con* model on HateExplain dataset

Table 8: Performance of our proposed model on the HateExplain (HE) dataset; We use the same metrics (IOUF1, TokenF1, AUPRC) for RD task, as mentioned in (Mathew et al., 2020) for a fair comparison.

Model	**Hate**		**Rationales**		
	Acc	**F1**	**IOUF1**	**TokenF1**	**AUPRC**
BERT-HE [Attn]	**0.698**	**0.687**	0.120	0.411	0.626
BiRNN-HE [Attn]	0.629	0.629	**0.222**	**0.506**	**0.841**
GenEx-Con (Ours)	0.682	0.678	**0.243**	**0.553**	**0.851**

Further, to check the robustness of our proposed model, we have experimented with the existing English *HateExplain* dataset (Mathew et al., 2020). The choice of baseline models can be adapted based on the dataset used. For the HateExplain dataset, models like BART or T5 can be selected, while for the BullyExplain dataset, we need to use the multilingual variations of these models (mBART or mT5) due to the presence of Hindi-English code-mixed language.

From Table 8, we can observe that our proposed model significantly outperforms the SOTA for the HateExplain dataset in the case of the explainability tasks (Rationales) and slightly underperforms for the HSD detection task. BERT-HateExplain-Attn model achieved an F1 score of 0.687 and IOUF1 of 0.120 in HSD and RD tasks, respectively. In contrast, GenEx achieved 0.678 and 0.243 for HSD and RD tasks, respectively. Another model, BiRNN-HateExplain-Attn, attained 0.629 F1 score and 0.222 IOUF1 score for HSD and RD tasks, respectively. The difference between these models is quite noticeable. When one SOTA model excels in the HSD task, it performs poorly in the RD task, and vice versa is observed for the other model (BiRNN-HateExplain-Attn). In this context, our proposed GenEx model has the novelty of being able to attend SOTA results for both tasks. The reason behind achieving good results with the GenEx model is the idea of using a generative model for different categories of tasks, including classification tasks (CD, SA, TI) and sequence labeling tasks (RD). As stated by researchers (Mathew et al., 2020), models that excel in classification may not always be able to provide reasonable and accurate rationales for their decisions. Therefore, our proposed model attempts to bridge this research gap.

6 Conclusion and Future Works

This research tackles the challenge of cyberbullying detection in a code-mixed linguistic context while emphasizing the aspect of explainability. The contributions of this study can be summarized into two key components: (a) the creation of the explainable cyberbully detection (CD) dataset in a code-mixed language, featuring annotations for rationale/phrases used in decision-making alongside bully labels, sentiment labels, and target labels; (b) the introduction of a unified generative framework, (*GenEx*) founded on common-sense knowledge and reinforcement learning, which adeptly addresses four distinct tasks. By formulating the multitask problem as a text-to-text generation task and harnessing the capabilities of large pre-trained sequence-to-sequence models, our proposed model surpasses the state-of-the-art with an enhanced accuracy score of 2.47% for the CD task.

Future attempts will be made to extend explainable cyberbullying detection in a multimodal setting considering image and text modalities.

Limitations

Our endeavor aimed to construct a multitask framework and introduce a benchmark dataset, *BullyExplain* tailored for explainable cyberbullying detection, encompassing target and sentiment identification within code-mixed language. Nonetheless, it is important to acknowledge certain limitations inherent in our proposed approach and dataset, including: (1) The current explainability feature operates at the word token level, offering explanations solely on the lexical scale. (2) Implicit or indirect hate expressions were not included in this study, with the focus primarily on explicit markers. Future work is planned to address the development of datasets and models capable of detecting implicit/indirect toxic posts. (3) Users often incorporate images alongside text in their social media posts. As it stands, our system does not accommodate multi-modal inputs, limiting its capacity to detect cyberbullying in such diverse content forms.

Acknowledgement: The Authors would like to acknowledge the support of Ministry of Home Affairs (MHA), India for conducting this research.

References

Sweta Agrawal and Amit Awekar. 2018. Deep learning for detecting cyberbullying across multiple social media platforms. In *European conference on information retrieval*, pages 141–153. Springer.

Vimala Balakrishnan, Shahzaib Khan, and Hamid R Arabnia. 2020. Improving cyberbullying detection using twitter users' psychological features and machine learning. *Computers & Security*, 90:101710.

Seok-Jun Bu and Sung-Bae Cho. 2018. A hybrid deep learning system of cnn and lrcn to detect cyberbullying from sns comments. In *Hybrid Artificial Intelligent Systems: 13th International Conference, HAIS 2018, Oviedo, Spain, June 20-22, 2018, Proceedings 13*, pages 561–572. Springer.

Rich Caruana. 1997. Multitask learning. *Machine Learning*, 28.

Michael Crawshaw. 2020. Multi-task learning with deep neural networks: A survey.

Maral Dadvar, Dolf Trieschnigg, and Franciska de Jong. 2014. Experts and machines against bullies: A hybrid approach to detect cyberbullies. In *Canadian conference on artificial intelligence*, pages 275–281. Springer.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Karthik Dinakar, Roi Reichart, and Henry Lieberman. 2011. Modeling the detection of textual cyberbullying. In *Proceedings of the International Conference on Weblog and Social Media 2011*. Citeseer.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Devansh Gautam, Prashant Kodali, Kshitij Gupta, Anmol Goel, Manish Shrivastava, and Ponnurangam Kumaraguru. 2021. Comet: Towards code-mixed translation using parallel monolingual sentences. In *CALCS*.

Soumitra Ghosh, Swarup Roy, Asif Ekbal, and Pushpak Bhattacharyya. 2022. Cares: Cause recognition for emotion in suicide notes. In *European Conference on Information Retrieval*, pages 128–136. Springer.

David Gunning, Mark Stefik, Jaesik Choi, Timothy Miller, Simone Stumpf, and Guang-Zhong Yang. 2019. Xai—explainable artificial intelligence. *Science Robotics*, 4(37):eaay7120.

James Hawdon, Atte Oksanen, and Pekka Räsänen. 2017. Exposure to online hate in four nations: A cross-national consideration. *Deviant behavior*, 38(3):254–266.

Jena D. Hwang, Chandra Bhagavatula, Ronan Le Bras, Jeff Da, Keisuke Sakaguchi, Antoine Bosselut, and Yejin Choi. 2020. Comet-atomic 2020: On symbolic and neural commonsense knowledge graphs.

Robin M Kowalski, Gary W Giumetti, Amber N Schroeder, and Micah R Lattanner. 2014. Bullying in the digital age: a critical review and meta-analysis of cyberbullying research among youth. *Psychological bulletin*, 140(4):1073.

Ritesh Kumar, Aishwarya N Reganti, Akshit Bhatia, and Tushar Maheshwari. 2018. Aggression-annotated corpus of hindi-english code-mixed data. *arXiv preprint arXiv:1803.09402*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. pages 7871–7880.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation.

Krishanu Maity and Sriparna Saha. 2021a. Bert-capsule model for cyberbullying detection in code-mixed indian languages. In *International Conference on Applications of Natural Language to Information Systems*, pages 147–155. Springer.

Krishanu Maity and Sriparna Saha. 2021b. A multi-task model for sentiment aided cyberbullying detection in code-mixed indian languages. In *International Conference on Neural Information Processing*, pages 440–451. Springer.

Krishanu Maity, Sriparna Saha, and Pushpak Bhattacharyya. 2023. Emoji, sentiment and emotion aided cyberbullying detection in hinglish. *IEEE Trans. Comput. Soc. Syst.*, 10(5):2411–2420.

Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. 2020. Hatexplain: A benchmark dataset for explainable hate speech detection. *arXiv preprint arXiv:2012.10289*.

Carol Myers-Scotton. 1997. *Duelling languages: Grammatical structure in codeswitching*. Oxford University Press.

Sayanta Paul and Sriparna Saha. 2020. Cyberbert: Bert for cyberbullying identification. *Multimedia Systems*, pages 1–8.

John Pavlopoulos, Jeffrey Sorensen, Léo Laugier, and Ion Androutsopoulos. 2021. SemEval-2021 task 5: Toxic spans detection. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 59–69, Online. Association for Computational Linguistics.

Shraman Pramanick, Dimitar Dimitrov, Rituparna Mukherjee, Shivam Sharma, Md Akhtar, Preslav Nakov, Tanmoy Chakraborty, et al. 2021. Detecting harmful memes and their targets. *arXiv preprint arXiv:2110.00413*.

Manikandan Ravikiran and Subbiah Annamalai. 2021. DOSA: Dravidian code-mixed offensive span identification dataset. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 10–17, Kyiv. Association for Computational Linguistics.

Minlie Huang Sahand Sabour, Chujie Zheng. 2021. Cem: Commonsense-aware empathetic response generation. *arXiv preprint arXiv:2109.05739*.

Abhilasha Sancheti, Kundan Krishna, Balaji Srinivasan, and Anandhavelu Natarajan. 2020. Reinforced rewards framework for text style transfer.

Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A Smith, and Yejin Choi. 2019. Atomic: An atlas of machine commonsense for if-then reasoning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 3027–3035.

Apoorva Singh, Sriparna Saha, Md Hasanuzzaman, and Kuntal Dey. 2021. Multitask learning for complaint identification and sentiment analysis. *Cognitive Computation*, pages 1–16.

Rohit Sridhar and Diyi Yang. 2022. Explaining toxic text via knowledge enhanced text generation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 811–826, Seattle, United States. Association for Computational Linguistics.

Fabio Sticca, Sabrina Ruggieri, Françoise Alsaker, and Sonja Perren. 2013. Longitudinal risk factors for cyberbullying in adolescence. *Journal of community & applied social psychology*, 23(1):52–67.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Sen Wu. 2019. Emmental: A framework for building multimodal multi-task learning systems.

Rong Xiang, Qin Lu, Ying Jiao, Yufei Zheng, Wenhao Ying, and Yunfei Long. 2019. Leveraging writing systems changes for deep learning based chinese affective analysis. *International Journal of Machine Learning and Cybernetics*, 10:3313–3325.

Baosong Yang, Jian Li, Derek Wong, Lidia Chao, Xing Wang, and Zhaopeng Tu. 2019. Context-aware self-attention networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:387–394.

Omar Zaidan, Jason Eisner, and Christine Piatko. 2007. Using "annotator rationales" to improve machine learning for text categorization. In *Human language technologies 2007: The conference of the North American chapter of the association for computational linguistics; proceedings of the main conference*, pages 260–267.

A Error Analysis

We have manually checked some samples from the test set to examine how machine-generated rationales and bully labels differ from the human annotator's decision. Table 9 shows the predicted rationales and bully labels of a few test samples obtained by different baselines and our proposed models (*GenEx-Con* and *GenEx-Fuse*). **(I)** It can be observed that the human annotator labeled the T1 tweet as Non-Bully. In contrast, all the models (both baselines and *GenEx* models) predicted the label as Bully highlighting the offensive word *gandu* (Asshole), supporting their predictions. This shows that the model cannot comprehend the context of this offensive word as it is not directed at anyone and has been used more in a sarcastic manner, highlighting the model's limitation in understanding indirect and sarcastic statements. **(II)** All the models (both baselines and our proposed models) predicted the correct label for tweet T2. But if we see the rationales predicted (highlighted part), none of the baseline models performed well compared to human decisions. Both GenEx-Con and *GenEx-Fuse* can predict all the words present in the ground truth rationale, but it also predicts other phrases as the rationale. In this case, we can also notice that *GenEx-Fuse* predicts some tokens not present in the original sentence (highlighted in yellow) as generative models like BART are designed to generate output based on the pre-trained vocabulary. So there can be some instances where the model generates rationales that contain some information that is not present in the input text.

B Experimental Settings

In this section, we detail various hyperparameters and experimental settings used in our work. We have performed all the experiments on Tyrone machine with Intel's Xeon W-2155 Processor having 196 Gb DDR4 RAM and 11 Gb Nvidia 1080Ti GPU. We have randomly chosen 80% of the data for training, 5% for validation, and the remaining 15% for testing. We have executed all of the models five times, and the average results have

Table 9: In comparison to human annotators, rationales identified by several models are shown. Green highlights indicate agreements between the human annotator and the model. Orange highlighted tokens are predicted by models, not by human annotators. Yellow highlighted tokens are predicted by models but are not present in the original text.

Model	**Text**	**Bully Label**
Human annotator (T1)	Semi final tak usi bnde ne pahochaya hai jisko tu gandu bol raha .	Non-Bully
Translation	**The person you are calling ass*ole is the one that helped us to reach the semi finals.**	
BERT	Semi final tak usi bnde ne pahochaya hai jisko tu gandu bol raha .	Bully
mBART	Semi final tak usi bnde ne pahochaya hai jisko tu gandu bol raha .	Bully
GenEx-Con	Semi final tak usi bnde ne pahochaya hai jisko tu gandu bol raha .	Bully
GenEx-Fuse	Semi final tak usi bnde ne pahochaya hai jisko tu gandu bol raha .	Bully
Human annotator (T2)	Abey mc gb road r ki pehle customer ki najayaz auladte	Bully
Translation	**You are the illegitimate children of first customer of GB road.**	
BERT	Abey mc gb road r ki pehle customer ki najayaz auladte	Bully
mBART	Abey mc gb road r ki pehle customer ki najayaz auladte	Bully
GenEx-Con	Abey mc gb road r ki pehle customer ki najayaz auladte	Bully
GenEx-Fuse	Abey mc gb road r ki pehle customer ki najayaz auladeeeeee.......	Bully

been reported. We have used mBART (Liu et al., 2020) as the base model for both *GenEx-Con* and *GenEx-Fuse*. Both these models are trained for a maximum of 60 epochs and batch size of 16. Adam optimizer is used to train the model with an epsilon value of 0.00000001. All the models are implemented using Scikit-Learn[3] and pytorch[4] as a backend. For the CD, TI and SA tasks, accuracy and macro-F1 metrics are used to evaluate predictive performance. For the quantitative assessment of the RD task, we used the Jaccard Similarity (JS), Hamming Distance (HD), and Ratcliff-Obershelp Similarity (ROS) metrics as mentioned in (Ghosh et al., 2022).

There are four different multitask variants based on the number of tasks we aim to address simultaneously. As we have four tasks and our main objective is explainable cyberbullying detection, CD and RD are kept common for any multitask variant. So we have four multitask variants, i.e., CD+RD, CD+RD+SA, CD+Rd+TI and CD+RD+SA+TI.

C Standard Baselines

We have developed the following standard baselines as done in (Mathew et al., 2020). Those baselines can be used for both single-task and multi-task settings. For single-task settings, the final features are passed through one task-specific layer. A task-specific layer is made by a fully connected layer followed by an output layer. In multi-task scenarios, there are task-specific layers designed to address multiple tasks concurrently.

1. **CNN-GRU**: The input is sent through a 1D CNN with window sizes of 2, 3, and 4, each with 100 filters. We employ the GRU layer for the RNN portion and then max-pool the output representation from the GRU architecture's hidden layers. This hidden layer is processed via a fully connected layer to output the prediction logits.

2. **BiRNN**: We pass input text through BiRNN followed by a dense layer to obtain a shared representation of the textual feature. This shared representation is then passed through task-specific layers.

3. **BiRNN-Attention**: We pass input text to BiRNN followed by the attention layer. Attended features of the text are passed through a dense layer to obtain shared representation which will be further passed through different multitask channels.

4. **BERT-finetune** BERT's (Devlin et al., 2018) pooled output with dimension 768 was fed to a softmax output layer.

D Sentiment Analysis and Target Identification Results

Table 10 shows and compares the performance of our proposed model with different baseline models for sentiment analysis and target identification across different multitask variants.

[3] https://scikit-learn.org/stable/
[4] https://pytorch.org/

Table 10: Results of different baselines and the two proposed frameworks, *GenEx-Con* and *GenEx-Fuse* in a multi task setting for Sentiment analysis and Target identification tasks.

Model	**CD+RD+TI**		**CD+RD+SA**		**CD+RD+SA+TI**			
	Target		**Senitment**		**Target**		**Sentiment**	
	ACC	**F1-Score**	**ACC**	**F1-score**	**ACC**	**F1-Score**	**ACC**	**F1-Score**
Standard Baselines								
BiRNN	51.76	48.53	65.72	65.61	51.91	49.09	65.92	65.91
BiRNN+Attention	51.92	48.83	65.88	65.89	52.08	49.47	66.72	66.63
CNN GRU	52.33	49.16	66.71	66.61	52.93	49.91	66.26	66.21
BERT	54.81	**49.98**	68.23	**68.03**	55.34	**50.44**	69.91	**69.88**
Generative Baselines								
T5	59.34	56.11	72.22	72.40	60.22	56.51	73.42	73.18
BART	59.02	55.77	72.10	72.08	59.43	56.19	73.19	73.22
mBART	60.72	**56.70**	73.15	**73.88**	60.77	**56.47**	74.11	**74.12**
Proposed Approach								
GenEx-Con	61.18	**58.23**	74.66	**74.56**	62.23	**58.92**	74.68	**74.63**
GenEx-Fuse	61.27	58.15	74.43	74.32	62.17	58.47	74.23	74.18

Table 11: Sample transformation of labels into single target sequences. See translation of tweets T1 and T2 in Table 2

Input Sentence	**Bully Label**	**Rationales**	**Target**	**Sentiment**	**Target Sequence**
T1: Larkyaaan toh jaisyyy bht hi phalwaan hoti. Ak chipkali ko dekh kr tm logon ka sans rukk jataa	Bully	[**1**,0,0,0,0,**1**,0,0,**1**,0,0,0,0,0,0,**1**,**1**,**1**]	Gender	Negative	<Larkyaaan phalwaan chipkali sans rukk jataa> [Bully] [Gender] [Negative]
T2: Laal phool gulaab phool shahrukh bhaiya beautifu	Non Bully	[0,0,0,0,0,0,0]	Not Applicable	Positive	<NONE> [Non Bully] [Not Applicable] [Positive]

E Transformation of labels into single target sequences

Table 11 shows few example instances of target sequences constructed from labels for training the proposed models.

F Discussion on errors during codemixed-English translation

When we translated the code-mixed data to English using automated translation tools, we manually checked examples to find patterns and observed where the model failed to translate. We have shown some examples (highlighted in red) in Table 12 where the model fails to translate some offensive words into corresponding English words. We made a dictionary of such words and used that dictionary to replace those codemixed offensive words with their correct corresponding English offensive words. Further, we have engaged three senior annotators (Master's students in Linguistics) to verify the translation quality in terms of fluency (F) and adequacy (A). Fluency evaluates whether the translation is syntactically correct or not, whereas Adequacy checks the semantic quality. Each annotator marked randomly selected 500 translated sentences with an ordinal value from a scale of 1-5[5] for both F and A. We attain high average F and A scores of 4.23 and 4.58, respectively, showing that the translations are of good quality.

G How was inter-annotator agreement computed on RD?

The RD task is the sequence labeling task where we have highlighted words or phrases responsible for annotating the post as a bully. During RD task annotation, the annotator has to mark each word as either 0 or 1, where 1 means rationale. Each bully sentence is encoded with a boolean vector with length equal to the number of tokens in the sentence. Table 13 shows the RD annotation of the input sentence "saleko kon mic de diya, voice dekoh hizra jaisa hai" (Translation: He sounds like

[5] **Fluency** - 5: Flawless, 4: Good, 3: Non-native, 2: Disfluent, 1: Incomprehensible; **Adequacy** - 5: All, 4: Most, 3: Much, 2: Little, 1: None

Table 12: Errors in translation from code-mixed to English

Code mixed Tweet	**English translated (Using Google Translator)**	**Corrected English translated sentence**
Twitter kholo to har koi alag hi randi rona daal k baitha hota h	If you open Twitter, everyone would be sitting crying differently	If you open Twitter, everyone would be sitting crying like a wh*re.
Hum apni mehnat se paisa kamana jante Sun re musselman katue roz logo k	We know how to earn money with our hard work, listen to the Muslims who are bitter everyday	We know how to earn money with our hard work, listen to the bastard Muslims who are bitter everyday
What about you? Kitne logon ko chuthiya banaya bhai?	What about you? Kitne logon ko chuthiya banaya bhai?	What about you? Brother, how many people have you made a bi*ch?

Table 13: An example of annotation procedure for RD task

Input Sentence	saleko	kon	mic	de	diya	voice	dekho	hizra	jaisa	hai
Annotator 1	1	0	1	0	0	1	1	1	1	0
Annotator 2	1	0	0	0	0	1	0	1	1	0
Annotator 3	1	0	0	0	0	1	1	1	1	0
Final Label	1	0	0	0	0	1	1	1	1	0
	saleko kon mic de diya, voice dekoh hizra jaisa hai									

faggot, who gave the mic to this idiot ?). The inter-annotator agreement (IAA) score based on Fleiss' Kappa measure of this sentence is 0.73. In this way, we calculated each sentence's IAA score and reported the mean value of 0.74 for the RD task.