Leveraging Supervised Contrastive Learning with Attention Mechanisms and Emotion Synthesis for Detecting Implicit Hate Speech

Sri Akash Kadali & Harshith Chejerla

***Abstract*— Implicit hate speech detection presents significant challenges due to its subtle and indirect nature, often disguised as humor, sarcasm, or indirect references. Traditional hate speech detection models struggle to identify such implicit content, necessitating targeted approaches. To address this issue, we propose a novel framework that leverages supervised contrastive learning for implicit hate speech detection. Our methodology includes large-scale data augmentation, increasing the dataset size from 20,000 to 250,000 samples to enhance model robustness and generalization. The core of our approach combines supervised contrastive learning with binary cross-entropy loss, enabling the model to effectively distinguish between implicit hate speech and non-hateful content by learning discriminative representations. Furthermore, we incorporate emotion and sentiment features to capture underlying cues, improving the detection of subtle hate speech. We evaluate our framework on the ISHate and IHC datasets, demonstrating an average improvement of 12.4implicit hate speech across diverse contexts..**

***Index Terms*—Implicit Hate, Supervised Contrastive Learning, Emotion synthesis, Large-scale Augmentation**

.

I. Introduction

Social media channels are commonly utilized to spread harmful material, often targeting people based on factors like race, gender, religion, nationality, physical appearance, and occupation. This conduct frequently leads to online harassment and can have serious repercussions, including thoughts of self-harm. The effects of online harassment are sig nificant, causing increased stress and mental health problems for victims. The widespread nature of this behavior highlights the necessity for a more comprehensive understanding of its impacts to create effective strategies that minimize harmful online interactions. However, content on social media does not always explicitly appear as hate speech. Instead, it can take the form of implicit hate speech, which promotes hate without using direct hateful language, posing a challenge for detection systems to accurately identify and manage.

A. Implicit Hate Speech

Implicit hate speech refers to expressions that convey dis criminatory or prejudiced sentiments indirectly, often relying on coded language or implied meanings. Unlike explicit hate speech, which is overt and directly offensive, implicit hate speech may use more veiled language without violating plat form rules, making it challenging to identify and address. For example, implicit hate speech might use humor or irony to mask discriminatory undertones, making it challenging to de termine intent or impact based solely on keywords or phrases. Such content may exploit cultural or regional references that convey hateful connotations only to specific audiences who understand the subtext. Additionally, implicit hate speech often utilizes ambiguous language that requires an understanding of context, pushing the limits of conventional detection models.

B. Impact of Implicit Hate Speech

Addressing implicit hate speech is crucial due to its sig nificant psychological impact, particularly on individuals who are already emotionally vulnerable. Despite being expressed in a covert manner, implicit hate speech can induce substantial mental distress, exacerbating feelings of isolation, exclusion, and worthlessness. These emotional consequences can inten sify pre-existing mental health conditions and, in severe cases, lead to suicidal ideation. The insidious nature of implicit hate speech makes it a potent contributor to the deterioration of mental well-being, highlighting the necessity of its detection and mitigation. By effectively combating implicit hate speech, we contribute to fostering a more inclusive and supportive societal environment, thereby safeguarding mental health and mitigating the harmful effects of discriminatory language on susceptible individuals.

C. Exiting Approaches and Research Gaps

In the prior works on implicit hate speech detection, Hate BERT has been used predominantly for classification. Hate-BERT is a variant of BERT re-trained on over 1 million posts from banned Reddit communities. This model demonstrated strong performance in various bench marks. Prior works explored different techniques, such as data augmentation, knowledge graph infusion, and sentiment fusion to enhance implicit hate speech detection. A few works explored the use of classical approaches such as TF IDF and Glove methods to compare the performance with state-of-the-art transformer-based models such as BERT.However, we contend that ad-dressing implicit hate speech warrants a distinct approach from explicit forms due to its covert nature within linguistic expressions. Even in the absence of overtly offensive language, manifestations of hatred may still permeate discourse. While perusing numerous research papers, we observed a prevalent inclination towards this binary categorization, dichotomizing discourse as either ”Hate Speech” or ”Non-Hate Speech.” effects.

TABLE I
EXAMPLE OF IMPLICIT AND EXPLICIT HATE CONTENT

Comment	Type
I love white people! I am not interested in anyone else!	**Implicit**
Canada is an immigrant country do not change it to refugee country please.	**Implicit**
Death to collaborators of Zionists!	**Explicit**
Feminists are bad for the family.	**Explicit**
What an amazing picture.	**Non-Hate**
We will just have to wait and see.	**Non-Hate**

While prior works have made significant strides in implicit hate speech detection, there remain several research gaps that existing approaches have not fully addressed. One limitation is the insufficient use of large-scale data augmentation. This lack of extensive augmentation restricts the model's generalizability, especially when dealing with underrepresented or rare instances of implicit hate speech. Another gap lies in the integration of emotion synthesis in the detection pipeline. Despite some efforts to incorporate sentiment analysis, the use of emotion features as part of the learning process has not been thoroughly explored. Emotion synthesis could provide valuable insights into the emotional undertones of implicit hate speech, which are often subtle and context-dependent. The absence of this dimension limits the model's capability to capture complex expressions of hate that may not be overtly abusive but are embedded in emotional cues.

D. Our Solution

The proposed model follows a structured sequence of different modules to achieve effective classification. Initially, an extensive augmentation strategy is employed to diversify and expand the training dataset. This strategy includes eight techniques: Replace Named Entities (RNE), Replace Scalar Adverbs (RSA), Replace Adjectives (RA), Replace In-domain Expressions (RI), Add Adverbs to Verbs (AAV), Easy Data Augmentation (EDA), Back Translation (BT), and Generative AI (GAI). Following augmentation, feature extraction is performed using Deberta for contextual embeddings, NRC-Lex for emotion features, and CoreNLP for sentiment features. Subsequently, Bi-LSTM is employed to capture sequential dependencies in text, and a self-attention mechanism is employed to enhance feature representation. Subsequently, word-level attention is applied to the integrated features, combining text, emotion, and sentiment information. Contrastive learning is then utilized to refine the model's discriminative power during training. Finally, the model performs classification into three distinct categories using a final output layer.

This sequential approach, from data augmentation through feature extraction, attention mechanisms, and contrastive learning, is designed to ensure robust performance and adaptability in handling diverse textual data.

II. RELATED WORK

In this section, we provide an examination of prior research concerning the identification of abusive and hateful content on social media. Initially, we offer a summary of existing studies centered on the detection of general hate speech, specifically addressing abusive and hateful content. Subsequently, we delve into several works that concentrate on implicit hate speech detection, where instances of hate speech are concealed within sentences.

A. Existing Works on General Hate Speech Detection

In the dynamic landscape of hate speech detection, recent research has made significant strides, offering a narrative that unfolds across multiple dimensions. In 2021, MacAvaney et al. [10] tackled the intricacies of hate speech detection in their paper, "Hate speech detection: Challenges and solutions." This work not only identified but thoroughly examined the challenges confronted by online automatic approaches in discerning hate speech within text. Proposing a multi-view SVM approach, the authors achieved performance levels near the state-of-the-art, presenting a solution that is not only sophisticated but also simpler and more interpretable than prevailing neural methods.

As the journey through hate speech detection continued into the same year, a distinctive contribution emerged in the form of "Thirty years of research into hate speech: topics of interest and their evolution" [11]. This comprehensive analysis took a historical perspective, unraveling the evolution of hate speech topics over the past three decades. Providing a nuanced understanding of the development of hate speech and its various manifestations, the study contributed a unique layer to the evolving narrative.

The narrative leaped forward to 2022 with a focus on generalization capabilities in "Improving Generalization of Hate Speech Detection Systems to Novel Target Groups via Domain Adaptation" [12]. This research delved into the adaptability of deep learning models across different target groups of hate speech. Exploring the potential of domain adaptation techniques, the study aimed to enhance the performance of hate speech detection systems when faced with new and unseen target groups, introducing a dimension of adaptability to the overarching research.

Amidst the advancements, a survey paper emerged in the same year, titled "Hate Speech Detection: A Survey" [13]. Presented at the 4th International Conference on Advances in Computing, Communication Control, and Networking (ICAC3N), this comprehensive survey explored, summarized, and compared various hate speech detection methods. Serving as a compass in the expansive sea of techniques, the survey provided a panoramic view of the state-of-the-art approaches in hate speech detection, contributing a chapter to the ongoing narrative.

This extended to 2023 with a literature survey titled "A literature survey on multimodal and multilingual automatic hate speech identification" [14]. This work, weaving together a rich tapestry of hate speech definitions, motivational aspects for detection, and standard textual analysis methods, emphasized the importance of considering multiple modalities and languages in automatic hate speech identification. The narrative took a turn toward inclusivity, underlining the need for a holistic approach in identifying hate speech across diverse linguistic and modal contexts.

The methodologies discussed above primarily revolve around the classification of General Hate Speech, employing a binary classification system to determine whether a given piece of content falls into the category of hate speech or not. These approaches are specifically designed to address explicit instances of hate speech, where the content's nature is overt and easily identifiable.

However, it is essential to acknowledge the limitations of these methods, particularly in the context of Implicit Hate Speech Detection. Implicit hate speech refers to instances where the harmful intent is not explicitly expressed but may be subtly implied or concealed within the content. The existing methodologies, optimized for explicit cases, may not be as adept at discerning implicit forms of hate speech, thereby highlighting the need for specialized approaches to address the nuanced nature of implicit expressions. As the detection and classification of hate speech continue to evolve, researchers must explore strategies that encompass both explicit and implicit dimensions to enhance the comprehensiveness of hate speech detection systems.

B. Existing works on Implicit Hate Speech Detection :

In the realm of hate speech detection, the landscape has witnessed significant advancements over the years, with researchers pushing the boundaries of understanding and addressing implicit forms of hatred. ElSherief et al.'s seminal work in 2021, titled "Latent Hatred: A Benchmark for Understanding Implicit Hate Speech" [8], stands out as a pioneering contribution. This work introduces a meticulously crafted taxonomy of implicit hate speech, rooted in linguistic and pragmatic theories, providing a theoretical foundation to capture the nuanced expressions of concealed animosity. The authors further enhance the research landscape by creating a benchmark corpus with fine-grained labels, enabling a detailed analysis of both message types and implied meanings. This innovative approach facilitates the evaluation of hate speech detection models, marking a significant step forward in the field.

Building upon this foundation, Lin's 2022 paper, "Leveraging World Knowledge in Implicit Hate Speech Detection" [15], takes a bold step by incorporating Entity Linking (EL) techniques into the detection process. Breaking new ground, this work applies EL techniques to both explicit and implicit hate speech, demonstrating their effectiveness in identifying targets and contextual nuances. By linking textual mentions to real-world entities, Lin shows how EL enhances existing models' performance, offering a novel perspective in the ongoing evolution of hate speech detection strategies.

In 2023, Ocampo et al. delve deep into the heart of implicit hate speech with their work titled "An In-depth Analysis of Implicit and Subtle Hate Speech Messages" [3]. This comprehensive analysis systematically examines implicit hate speech within standard benchmarks for hate speech detection. Uncovering the challenges and limitations of existing datasets, the paper sheds light on issues such as unclear definitions, inconsistent annotations, and the diverse expressions of implicit hatred. Through rigorous scrutiny, Ocampo et al. contribute valuable insights that pave the way for refining future hate speech detection benchmarks.

Shifting focus to the tactical realm, the 2021 paper titled "Playing the Part of the Sharp Bully: Generating Adversarial Examples for Hate Speech Detection" [4] presents a unique approach. By systematically creating adversarial examples, modified versions of original messages designed to deceive NLP models, this work exposes the vulnerabilities of current systems. Employing linguistic strategies such as paraphrasing, negation, and irony, the authors offer targeted diagnostic insights, marking a notable advancement in understanding and fortifying hate speech detection models.

In 2022, the landscape evolves with the introduction of "ToxiGen: A Large-Scale Machine-Generated Dataset for Adversarial and Implicit Hate Speech Detection" [5]. This paper unveils ToxiGen, a groundbreaking machine-generated dataset produced using a generative adversarial network (GAN). Focused on toxic and benign statements about 13 minority groups, ToxiGen provides a realistic and diverse set of examples for training and testing hate speech detection models. This innovative approach contributes significantly to the development of more robust and inclusive detection systems.

Finally, the year 2023 witnesses the emergence of "Implicit Offensive Speech Detection Based on Multi-feature Fusion" [16]. Constructing the BMA (BERT-Mate-Ambiguity) model, this paper pioneers a BERT-based multi-task learning approach. By incorporating textual, emotional, and ambiguity features through multi-feature fusion, the authors achieve remarkable accuracy in detecting implicit offensive speech in real-world scenarios. This work underscores the importance of considering various linguistic dimensions for a more comprehensive understanding of subtle and complex characteristics in implicit offensive speech. In tandem, these research endeavors collectively propel the field of hate speech detection towards increased sophistication and effectiveness.

The majority of existing approaches predominantly emphasize Data Augmentation, with minimal attention given to Model Novelty. In contrast, our emphasis is dual, focusing on both augmenting the dataset and introducing innovations in the model architecture.

C. Research Objectives

Implicit hate speech detection presents unique challenges due to its subtle and indirect nature, often requiring an understanding of context beyond what traditional approaches can capture. Building upon these challenges, the primary objectives of this research are as follows:

- To develop a robust framework using supervised contrastive learning for the effective detection of implicit hate speech.
- To explore the impact of large-scale data augmentation in enhancing the model's generalization capability for implicit hate speech detection.
- To investigate the role of emotion and sentiment features in capturing affective cues for improved detection of implicit hate speech.
- To evaluate the proposed method's performance on benchmark datasets (ISHate and IHC) and compare it against state-of-the-art approaches.
- To conduct an ablation analysis to assess the contribution of key components, such as contrastive learning, data augmentation, and affective features, in improving model performance.

III. Methodology

This section presents a comprehensive overview of our proposed approach for the implicit hate speech classification task. This approach incorporates components, namely a super-contrastive learning, a fusion of word-level attention features, Bi-LSTM features, sentiment and emotion features. The classifier comprises a network of seven fully connected layers strategically orchestrated to optimize performance in a classification task with three final classes.

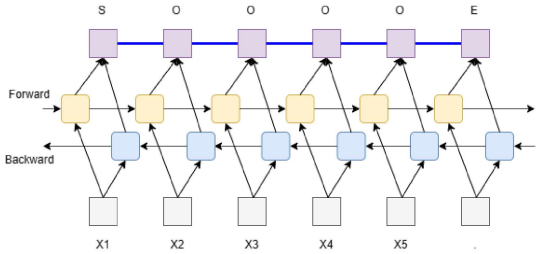

Fig. 2. Bi-LSTM

A. Data Preprocessing

Effective data preprocessing plays a crucial role in ensuring the quality of the subsequent analyses. The first step in our preprocessing pipeline involved text normalization and cleaning. We employ techniques such as lowercasing, removal of special characters, and handling of contractions to create a standardized textual format. This normalization process is vital to mitigate inconsistencies in language usage and ensure a uniform representation of text across the dataset. Moreover, careful consideration was given to the removal of noisy or irrelevant information, such as URLs, user mentions, and emojis, to streamline the input data and focus the model on the linguistic aspects relevant to hate speech identification.

Following text normalization, we addressed the challenge of handling imbalanced class distribution, inherent in hate speech datasets. This approach helped prevent the model from being biased towards the majority class and enhanced its ability to discern subtle patterns associated with implicit hate speech.

B. Data Augmentation

In addressing the challenge of an imbalanced dataset, we employ the paraphrasing-based data augmentation (DA) technique using a generative language model. To enhance the dataset further, we employ the following methods:

1) Replace Named Entities (RNE): To diversify the dataset, we introduce the Replace Named Entities method. This involves substituting named entities in sentences with alternatives selected from a pre-collected list, determined by their similarity using pre-trained Fast Text embeddings.

2) Replace Scalar Adverbs (RSA): The Replace Scalar Adverbs method substitutes emphasizing adverbs with alternatives that either intensify or diminish emphasis, exemplified by altering the emphasis on an adjective/verb in a sentence.

3) Add Adverbs to Verbs (AAV): To accentuate verbs, the Add Adverbs to Verbs method introduces speculative adverbs, such as "certainly" and "likely," into sentences.

4) Replace Adjectives (RA): The Replace Adjectives technique substitutes adjectives with their synonyms to diversify the language in sentences.

5) Replace In-Domain Expressions (RI): The Replace In-Domain Expressions method replaces manually-crafted expressions often used in hate speech messages with semantically similar alternatives, aiming to further diversify the dataset.

6) Easy Data Augmentation (EDA): EDA involves four random operations applied to sentences, including synonym replacement, insertion, word swapping, and word removal, contributing to data diversity.

7) Back Translation (BT): Back Translation involves translating messages into a different language and then back into the original language, contributing to variation in expressions.

8) Generative Models (GM): GM fine-tunes generative language models with instances from minority classes, utilizing a human-in-the-loop approach to review and re-annotate generated examples.

These augmentation methods contribute to dataset diversity, enhancing the model's ability to handle imbalanced classes. Parameters such as the percentage of words to change in a sentence are configurable for certain methods, allowing for flexibility in implementation.

C. Feature Extraction

1) Text Feature Extraction: We employ Deberta for textual feature extraction. The `bert--base--uncased` is used for tokenization, breaking down raw text into tokens. For each sample in the training data, the tokenized input is passed through Deberta. The model is configured with a maximum sequence length of 128 tokens, enabling truncation and padding for uniform sizes. The Deberta model processes the set of tokens $\mathbf{w}_t$, where $t \in \{1, 2, \ldots, 128\}$, producing the feature matrix Q_{txt}.

$$Q_{\text{txt}}(i) = \overrightarrow{\text{Deberta}}(w_i) \tag{1}$$

Here, w_{t_i} represents the i-th token in the sequence.

$$\mathbf{Q}_{\text{txt}} \in \mathbb{R}^{128 \times 768} \tag{2}$$

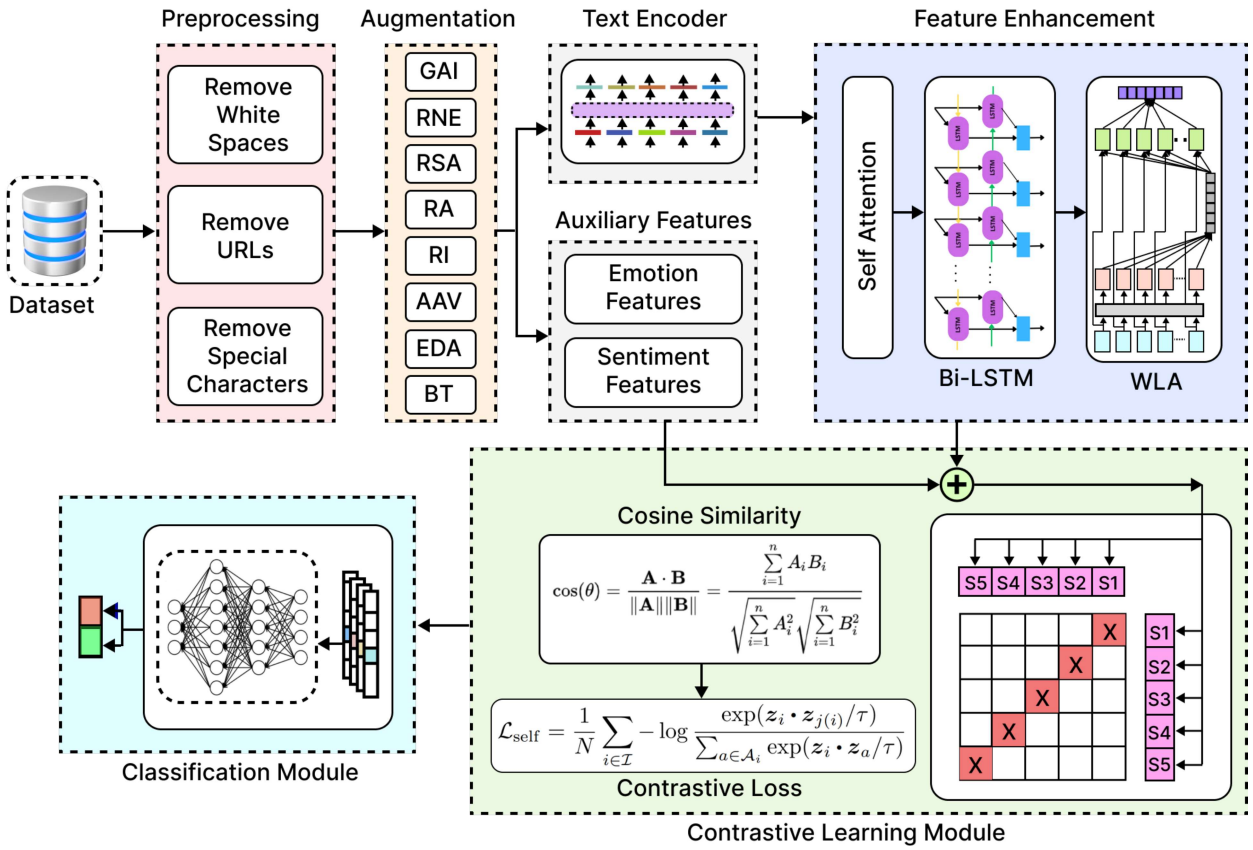

Fig. 1. System Architecture

The integration of the robust Deberta , known for its state-of-the-art performance in natural language processing tasks, enriches the model with advanced contextualized embeddings. This integration significantly enhances the model's capacity to discern intricate patterns within the input data. In the course of our research, we employed Deberta, a state-of-the-art transformer-based language model, for text feature extraction. The extracted text features served as a foundation for our subsequent analyses.

2) Emotion and Sentiment Feature Extraction: We utilized the NRCLex, an approach rooted in lexicon-based methods that link words to specific emotional categories. By employing the NRCLex, we encoded textual data with emotional cues. To extract emotion features, each word in the text is mapped to its underlying emotion category {fear, anger, anticipation, trust, surprise, positive, negative, sadness, disgust, joy}. This approach allowed us to measure and examine the emotional subtleties present in our dataset, illuminating the affective dimensions of the text. The application of the NRC Lexicon facilitated a detailed exploration of emotional content, imparting valuable insights into the emotional landscape of the text corpus under scrutiny. Furthermore, we retrieve sentiment values for each text sample. The sentiment categories span from 0 to 4, encompassing Very Negative, Negative, Neutral, Positive, and Very Positive sentiment categories

D. Attention-based Context Learning

1) Context Learning : The incorporation of Context Learning played a pivotal role in enhancing the depth of our text analysis. Leveraging Bidirectional Long Short-Term Memory (Bi-LSTM) networks, we aimed to capture the intricate contextual relationships within the textual data. The bidirectional nature of the LSTM allowed us to efficiently process information in both forward and backward directions, providing a comprehensive understanding of how words relate to each other in a sequential manner.

Mathematically, the Bi-LSTM operation can be represented as follows:

$$\overrightarrow{h_t} = \sigma(W_{ih}x_t + b_{ih} + W_{oh}\overrightarrow{h_{t-1}} + b_{oh}) \tag{3}$$

$$\overleftarrow{h_t} = \sigma(W_{ih}x_t + b_{ih} + W_{oh}\overleftarrow{h_{t+1}} + b_{oh}) \tag{4}$$

Here, x_t is the input at time t, $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ are the hidden states in the forward and backward directions, and W_{ih}, b_{ih}, W_{oh}, and b_{oh} are weight and bias parameters.

This contextual learning mechanism proved crucial in extracting nuanced features from the text, enabling us to discern the subtleties that might be overlooked in a unidirectional analysis. The Bi-LSTM served as a foundational component, laying the groundwork for a more sophisticated examination of the sequential dynamics inherent in our dataset.

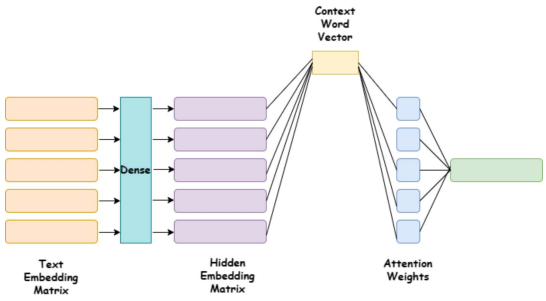

Fig. 3. Word Level Attention

2) Word Level Attention: To further refine our analysis, we integrated a Word Level Attention mechanism into our methodology. This attention mechanism was designed to dynamically weigh the importance of individual words in the text, allowing us to focus on those words that contribute significantly to the overall context.

Given u_t, the concatenated hidden state representation for each time step, it is subjected to a non-linear transformation to obtain a new hidden representation, as shown in Eq (5), where W and b are learned weight and bias parameters, respectively.

Mathematically, the Word Level Attention mechanism can be represented as:

$$P_{it} = \tanh(W_1 u_{it} + b_1) \tag{5}$$

$$a_{it} = \frac{\exp(p_{it} u_1)}{\sum_k \exp(p_{it} u_1)} \tag{6}$$

$$v_{it} = \sum_{k=1}^{K} a_{it} u_{it} \tag{7}$$

Here, v_{it} represents the importance attention score of each word.

By assigning varying levels of attention to different words, we were able to highlight key terms that played a pivotal role in shaping the meaning of the text. This granular attention to word-level details facilitated a more targeted analysis, emphasizing the specific linguistic elements that carried substantial semantic weight. The incorporation of Word Level Attention enriched our understanding of the text, enabling us to extract and emphasize the most salient components for subsequent stages of our research.

3) Feature Fusion : At this stage, feature fusion and classification involve the combination of the outputs from HAN and the Capsule Network after CLA (S_i), along with the auxiliary features (C_i), through the concatenation operation, resulting in a unified feature vector f_c:

$$f_i = [r_i, s_i, c_i] \tag{8}$$

Following the fusion, the combined feature vector f_c undergoes processing through two dense layers with Rectified Linear Unit (ReLU) activation functions. After the first dense layer transformation, f_d1 is obtained with an output dimension of 256. Subsequently, the second dense layer transforms f_d1 into f_d2 with an output dimension of 128. The transformations applied in each dense layer are represented as:

$$f_d1 = \text{ReLU}(W_{d1} f_c + b_{d1}) \tag{9}$$

$$f_d2 = \text{ReLU}(W_{d2} f_d1 + b_{d2}) \tag{10}$$

Here, W_{d1} and b_{d1} are the weight matrix and bias vector associated with the first dense layer, and W_{d2} and b_{d2} are the corresponding parameters for the second dense layer.

The processed feature vector f_d2 is then forwarded through the classification layer with a sigmoid activation function, generating the predicted output $\hat{Y}$. Mathematically, this can be expressed as:

$$\hat{Y} = \sigma(W_c f_d2 + b_c) \tag{11}$$

where W_c represents the weight matrix associated with the classification layer, and b_c is the bias vector.

For the purpose of training, the model employs the binary cross-entropy loss ($\mathcal{L}$) defined as:

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^{N} \Big(Y_i \log(\hat{Y}_i) + (1 - Y_i) \log(1 - \hat{Y}_i) \Big) \tag{12}$$

where N denotes the number of samples, Y_i is the ground truth label for the ith sample, and $\hat{Y}_i$ is the corresponding predicted probability.

In an effort to create a comprehensive feature set, we adopted Feature Fusion by concatenating the outputs of two distinct components: the NRC Lexicon and the Word Level Attention mechanism.

Here, Concatenate denotes the concatenation operation.

This fusion aimed to synergize the strengths of emotional encodings and contextually significant words, providing a more holistic representation of the textual data. By combining the emotional nuances captured by the NRC Lexicon with the contextually relevant information highlighted through Word Level Attention, we created a feature set that encapsulated both semantic and emotional dimensions. This integrated approach not only enriched the information available for analysis but also facilitated a more nuanced interpretation of our dataset. The Feature Fusion step was instrumental in ensuring that our subsequent analyses benefited from a comprehensive representation of both emotional and contextual aspects, contributing to the overall depth and accuracy of our findings.

E. Contrastive Learning

We incorporated the Super Contrastive Loss (SupConLoss) as a critical component for enhancing feature representations in self-supervised learning tasks. The loss function, combines a traditional cross-entropy loss (ce-loss) with a negative transfer cross-entropy loss (cl-loss). The nt-xent-loss method within the class computes the latter by employing a carefully designed mask to exclude positive pairs, ensuring a meaningful contrastive learning process. This method leverages anchor and target embeddings to calculate logits, subsequently computing log probabilities and deriving a negative transfer loss.

Mathematically, the Super Contrastive Loss (SupConLoss) can be represented as follows:

$$\text{SupConLoss}(\text{ce-loss}, \text{cl-loss}) = -\frac{1}{N}\sum_{i=1}^{N}(\text{ce-loss}_i + \text{cl-loss}_i) \tag{13}$$

Here, N represents the number of samples, and ce-loss_i and cl-loss_i are the individual cross-entropy and contrastive losses for each sample.

Cross-entropy loss, often referred to as log loss, is a common loss function used in classification problems. It measures the difference between two probability distributions: the true labels and the predicted probabilities. The cross-entropy loss for a single sample can be defined as:

$$\mathcal{L} = -\sum_{c=1}^{C} y_c \log(\hat{y}_c) \tag{14}$$

Here:

- C is the number of classes.
- y_c is a binary indicator (0 or 1) if class label c is the correct classification for the given observation.
- $\hat{y}_c$ is the predicted probability that the observation belongs to class c.

In the context of a binary classification problem, where there are only two classes (e.g., y and $1-y$), the cross-entropy loss simplifies to:

$$\mathcal{L} = -\left[y \log(\hat{y}) + (1-y)\log(1-\hat{y})\right] \tag{15}$$

This loss function penalizes predictions that are further from the true label, with larger penalties for predictions that are more confident but incorrect. The cross-entropy loss is widely used because it is convex and has a well-defined gradient, which is crucial for optimization using gradient descent algorithms.

The core of the cl-loss function is nt-xent-loss, which creates a binary mask to identify positive sample pairs (i.e., pairs with the same label) and excludes self-comparisons. The dot products between the anchor and target vectors are computed and scaled by the temperature. To enhance numerical stability, the logits are normalized by subtracting the maximum value in each row, and the exponentiated logits are calculated. The function then applies the mask to isolate the positive pairs, computes the log-probabilities, and ensures that no division by zero occurs by adjusting the mask sums. The final loss is derived by averaging the negative log-likelihoods of the positive pairs across all samples in the batch. This loss function effectively encourages the model to bring together representations of similar samples (those with the same label) while pushing apart representations of dissimilar samples, making it a key component in contrastive learning frameworks.

Mathematically, the nt-xent-loss for a single sample i can be expressed as [illegible]:

$$\mathcal{L}_i = -\log \frac{\exp\left(\frac{\text{sim}(\mathbf{z}_i, \mathbf{z}_j)}{\tau}\right)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp\left(\frac{\text{sim}(\mathbf{z}_i, \mathbf{z}_k)}{\tau}\right)} \tag{16}$$

The cosine similarity function is defined as:

$$\text{sim}(\mathbf{z}_i, \mathbf{z}_j) = \frac{\mathbf{z}_i \cdot \mathbf{z}_j}{\|\mathbf{z}_i\| \|\mathbf{z}_j\|} \tag{17}$$

The overall loss across the batch is:

$$\mathcal{L} = \frac{1}{2N}\sum_{i=1}^{2N} \mathcal{L}_i \tag{18}$$

The incorporation of SupConLoss into our training pipeline contributed to the development of more discriminative and contextually rich feature representations.

Furthermore, our experiments revealed that the Super Contrastive Loss, with its adaptive combination of supervised and contrastive components, significantly improved the performance of our models in self-supervised learning scenarios. The temperature parameter (temp) and the balancing factor (alpha) in the loss function provided a flexible framework for fine-tuning the emphasis between the two loss components.

Mathematically, the temperature-adjusted SupConLoss can be expressed as:

$$\text{SCL} = -\frac{1}{N}\sum_{i=1}^{N} \text{temp} \cdot \log\left(\frac{\exp(\text{ce}_i)}{\exp(\text{ce}_i) + \text{alpha} \cdot \exp(\text{cl}_i)}\right) \tag{19}$$

The flexibility allowed us to achieve a harmonious blend of supervised learning through cross-entropy and self-supervised learning through negative transfer contrastive loss. The inclusion of SupConLoss not only enhanced the discriminative power of the learned features but also facilitated the extraction of more nuanced contextual information from the data, thereby contributing to the overall success of our research endeavors.

The introduction of Super Contrastive Loss augments the model's learning process by prioritizing the discrimination of positive and negative pairs, thereby refining feature representations. This augmentation significantly contributes to the overall robustness and discriminative capabilities of the model.

Employing a distinctive concatenation approach, our model amalgamates word-level attention embeddings, bi-LSTM embeddings, and NRC lex embeddings. This unique fusion ensures a holistic representation of input data, capturing both sequential and contextual information to augment classification performance.

With a depth of seven layers, the model attains a heightened level of abstraction and feature extraction. This multi-layered architecture facilitates the learning of hierarchical representations, thereby enabling the extraction of intricate features and patterns present in the input data.

Tailored explicitly for a classification task with three final classes, our model aligns precisely with the specific requirements of the intended application.

Its robust architecture, coupled with the amalgamation of Deberta, Super Contrastive Loss, and diverse embeddings, demonstrates a formidable solution for handling classification tasks involving high-dimensional data.

TABLE II
PERFORMANCE METRICS FOR VARIOUS MODELS ON ISHATE DATASET

Method	NON-HS			Explicit			Implicit			Overall
	Precision	Recall	F1-score	Precision	Recall	F1-Score	Precision	Recall	F1-Score	Accuracy
BERT	0.888	0.902	0.895	0.813	0.811	0.812	0.469	0.37	0.414	0.848
HateBERT	0.892	0.894	0.892	0.811	0.815	0.813	0.382	0.349	0.365	0.843
Roberta	0.902	0.902	0.902	0.823	0.843	0.833	0.427	0.344	0.381	0.857
USE+SVM	0.891	0.862	0.872	0.761	0.809	0.783	0.402	0.361	0.382	0.82
DeBERTa	0.901	0.9	0.9	0.822	0.835	0.828	0.416	0.371	0.392	0.855
HateBERT+ALL	0.903	0.896	0.899	0.827	0.827	0.827	0.502	0.559	0.529	0.84
USE+TFIDF	0.82	0.918	0.866	0.785	0.709	0.745	0.667	0.043	0.081	0.816
USE+Count Vectorizer	0.834	0.872	0.853	0.741	0.718	0.73	0.321	0.194	0.242	0.79
LSTM based Comparative	0.85	0.815	0.832	0.773	0.636	0.698	0.103	0.312	0.155	0.732
Bi-CHAT Model	0.853	0.826	0.84	0.725	0.733	0.733	0.24	0.253	0.246	0.776
Paper 2 Published	0.9	0.9	0.9	0.81	0.85	0.83	0.45	0.82	0.58	-
Proposed Method	0.912	0.941	0.926	0.825	0.891	0.856	0.629	0.596	0.612	0.889

IV. EXPERIMENTAL EVALUATIONS

A. Evaluation Metrics

We aim to demonstrate that our method achieves consistent performance across various evaluation metrics. Therefore, we utilize four metrics for comparison: Accuracy (Acc), Precision (P), Recall (R), and F1-score (F1). The F1-score is particularly useful for evaluating unbalanced classes.

Let A_{TP}, A_{FP}, A_{TN}, and A_{FN} represent the sets of labels correctly predicted as abusive, incorrectly predicted as abusive, correctly predicted as non-abusive, and incorrectly predicted as non-abusive, respectively. Precision is calculated by dividing the number of true positive abusive labels A_{TP} by the total number of predicted abusive labels. Eq. (20) presents the formula for precision.

$$\text{Precision} = \frac{|A_{TP}|}{|A_{TP}| + |A_{FP}|} \tag{20}$$

Recall is calculated by dividing the number of true positive abusive labels A_{TP} by the sum of true positive abusive labels and false negative abusive labels A_{FN}. The recall formula is shown in Eq. (21).

$$\text{Recall} = \frac{|A_{TP}|}{|A_{TP}| + |A_{FN}|} \tag{21}$$

The F1-score metric takes both precision and recall into account, and is calculated as their harmonic mean. The formula for the F1-score is given in Eq. (22).

$$\text{F1-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{22}$$

Accuracy is determined by dividing the total number of correctly predicted comments, whether abusive or non-abusive, by the total number of comments in the dataset. Since our dataset is relatively balanced, we include accuracy in our evaluation. The formula for accuracy is provided in Eq. (23).

$$\text{Accuracy} = \frac{|A_{TP}| + |A_{TN}|}{|A_{TP}| + |A_{FP}| + |A_{TN}| + |A_{FN}|} \tag{23}$$

We proceed by discussing the experimental results on both datasets using these evaluation metrics.

B. Comparison on ISHate Dataset

To demonstrate the robustness of our proposed method, we compare its performance with multiple baselines across three categories on the ISHate dataset: Non-HS, Explicit, and Implicit hate speech. The comparative results are shown in Table II. Our method achieves an accuracy of 88.9% with high precision, recall, and F1-scores for each category, surpassing the performance of classical and transformer-based models by notable margins. For the Non-HS category, our method attains precision, recall, and F1-scores of 91.2%, 94.1%, and 92.6%, respectively. This demonstrates a significant improvement over transformer-based baselines like BERT and HateBERT, which achieve F1-scores of 89.5% and 89.2%, respectively. These gains highlight our model's ability to distinguish Non-HS content more effectively than existing methods. In the Explicit hate category, our method achieves an F1-score of 85.6%, outperforming other baselines such as Roberta and DeBERTa, which report F1-scores of 83.3% and 82.8%, respectively. The contextual information embedded in our model enables it to capture explicit hate expressions with greater accuracy and precision.

For the Implicit hate category, our method attains precision, recall, and F1-scores of 62.9%, 59.6%, and 61.2%, respectively. This demonstrates our model's superior ability to identify implicit hate, surpassing baseline models such as HateBERT and Roberta, which achieve lower F1-scores of 36.5% and 38.1%, respectively. The improvement emphasizes our model's capability to capture implicit hate speech, which traditional methods struggle to classify accurately. Furthermore, our method shows an overall accuracy improvement of 4.1% over BERT and 3.2% over Roberta. This enhancement is largely due to our model's use of contextual embeddings, which enables a more comprehensive understanding of hate speech across varying intensities and subtleties. The substantial performance gains in all categories and across all metrics validate the effectiveness of our approach on the ISHate dataset.

C. Comparison on IHC dataset

In addition to evaluating our method on the ISHate dataset, we also perform comparisons on the IHC dataset to further

TABLE III
PERFORMANCE METRICS FOR VARIOUS MODELS ON IHC DATASET

Method	NON-HS			Explicit			Implicit			Overall
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score	Accuracy
BERT	0.827	0.829	0.828	0.635	0.656	0.645	0.512	0.386	0.441	0.751
HateBERT	0.817	0.836	0.827	0.624	0.632	0.638	0.538	0.343	0.419	0.745
Roberta	0.816	0.844	0.83	0.641	0.631	0.636	0.509	0.344	0.41	0.75
USE+SVM	0.829	0.739	0.777	0.549	0.681	0.612	0.333	0.312	0.322	0.7
DeBERTa	0.818	0.826	0.822	0.621	0.634	0.627	0.474	0.344	0.398	0.741
HateBERT+ALL	0.812	0.802	0.807	0.709	0.765	0.745	0.503	0.723	0.593	-
USE+TFIDF	0.746	0.875	0.805	0.614	0.489	0.545	0.826	0.117	0.204	0.713
USE+Count Vectorizer	0.745	0.818	0.78	0.553	0.493	0.521	0.372	0.178	0.241	0.681
LSTM based Comparative	0.756	0.708	0.731	0.475	0.54	0.505	0.253	0.233	0.243	0.63
Bi-CHAT Model	0.76	0.68	0.717	0.463	0.589	0.518	0.284	0.166	0.209	0.624
Paper	-	-	-	-	-	-	58.6	59.1	58.6	-
Proposed method	0.8487	0.8453	0.847	0.6742	0.6551	0.663	0.5867	0.6802	0.627	0.792

validate its effectiveness. The performance metrics across different models on the IHC dataset are presented in Table III. Similar to the results on ISHate, our proposed method demonstrates superior performance across Non-HS, Explicit, and Implicit hate categories. For the Non-HS category, our method achieves precision, recall, and F1-scores of 84.87%, 84.53%, and 84.7%, respectively, outperforming baseline models such as BERT and DeBERTa, which have F1-scores of 82.8% and 82.2%, respectively. This shows our model's enhanced capability in correctly identifying Non-HS instances. In the Explicit hate category, our method attains an F1-score of 66.3%, providing a performance advantage over models like Roberta and HateBERT, which achieve F1-scores of 63.6% and 63.8%, respectively. The contextual features in our model allow it to detect explicit hate content more accurately.

For the Implicit hate category, our method achieves precision, recall, and F1-scores of 58.67%, 68.02%, and 62.7%, respectively. This represents a notable improvement over traditional models, which typically struggle with the nuanced nature of implicit hate detection. For instance, HateBERT and Roberta show lower F1-scores of 41.9% and 41.0%, respectively, on this category. Overall, our method achieves an accuracy of 79.2% on the IHC dataset, outperforming BERT and Roberta by 4.1% and 4.2%, respectively. The consistent improvements across all categories and evaluation metrics further establish the robustness and adaptability of our proposed approach in handling diverse hate speech datasets.

D. Ablation Analysis

In this section, we present a detailed ablation analysis to gain a deeper understanding of the factors contributing to the performance of our proposed method. The analysis in this section focuses exclusively on the ISHate dataset to ensure a consistent and comprehensive examination of the model's performance. By isolating and modifying specific aspects of the model, we aim to shed light on the critical elements that enable the models to excel in various tasks.

1) Influence of Attention: To validate the effectiveness of our proposed method, we conducted an ablation analysis to assess the impact of word-level attention (WLA) and self-attention mechanisms on model performance. The results are summarized in Table IV. When word-level attention is removed (WLA), the model achieves precision, recall, and F1-scores of 0.8955, 0.9343, and 0.9145, respectively, for the NON-HS class. In the explicit category, precision, recall, and F1-scores are 0.8036, 0.8873, and 0.8434, respectively. For the implicit class, the model achieves a precision of 0.6144, recall of 0.5689, and F1-score of 0.5908. The overall accuracy without WLA is 0.8689.

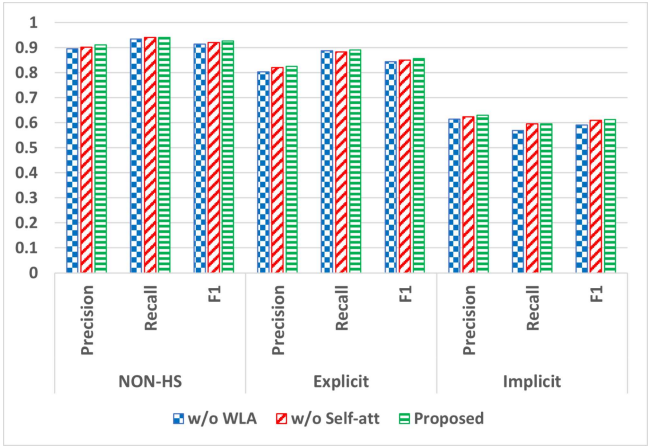

Fig. 4. Influence of Different Attention Mechanisms on Binary Classification Performance

When self-attention is removed (elf-att), the model shows a slight improvement over the model without WLA. It achieves precision, recall, and F1-scores of 0.9012, 0.9409, and 0.9206, respectively, for the NON-HS class. For the explicit class, these metrics are 0.8198, 0.8838, and 0.8506. In the implicit category, the model attains a precision of 0.6236, recall of 0.5960, and F1-score of 0.6095. The overall accuracy without self-attention is 0.8774. Our proposed method, which incorporates both word-level attention and self-attention mechanisms, significantly outperforms the ablated models. For the NON-HS class, it achieves precision, recall, and F1-scores of 0.9116, 0.9408, and 0.9260, respectively. In the explicit class, it attains a precision of 0.8248, recall of 0.8906, and F1-score of 0.8564. For the implicit class, precision, recall, and F1-scores are 0.6297, 0.5960, and 0.6124, respectively. The overall accuracy

TABLE IV
ABLATION ANALYSIS ON ATTENTION MECHANISMS

Features	NON-HS			Explicit			Implicit			Overall
	Precision	Recall	F1score	Precision	Recall	F1Score	Precision	Recall	F1Score	Accuracy
w/o WLA	0.8955	0.9343	0.9145	0.8036	0.8873	0.8434	0.6144	0.5689	0.5908	0.8689
w/o Self-att	0.9012	0.9409	0.9206	0.8198	0.8838	0.8506	0.6236	0.5960	0.6095	0.8774
Proposed	0.9116	0.9408	0.9260	0.8248	0.8906	0.8564	0.6297	0.5960	0.6124	0.8895

TABLE V
ABLATION ANALYSIS ON CL LOSS AND CE LOSS

Features	NON-HS			Explicit			Implicit			Accuracy
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score	Accuracy
w/o CL Loss	0.9070	0.8863	0.8965	0.8089	0.8629	0.8354	0.6280	0.5286	0.5737	0.8542
w/o CE Loss	0.9285	0.9003	0.9142	0.8133	0.8771	0.8445	0.6344	0.5494	0.5895	0.8656
Proposed	0.9116	0.9408	0.9260	0.8248	0.8906	0.8564	0.6297	0.5960	0.6124	0.8895

of the proposed method is 0.8895. This ablation analysis demonstrates the importance of both word-level attention and self-attention in enhancing the model's performance. The superior results of the proposed method across all categories highlight the effectiveness of incorporating these attention mechanisms for abusive content detection.

2) Influence of Different Losses: We performed an ablation analysis focusing on supervised contrastive loss (CL Loss) and cross-entropy loss (CE Loss). The performance metrics for this analysis are summarized in Table V. In our experiments, we observed that removing CL Loss from the model led to a decrease in performance. Specifically, the precision, recall, and F1-score for the NON-HS category dropped to 0.9070, 0.8863, and 0.8965, respectively. For explicit detection, the F1-score fell to 0.8354, and the implicit detection metrics also showed a decline, with precision, recall, and F1-scores dropping to 0.6280, 0.5286, and 0.5737. These results demonstrate the importance of CL Loss in maintaining robust performance across all categories.

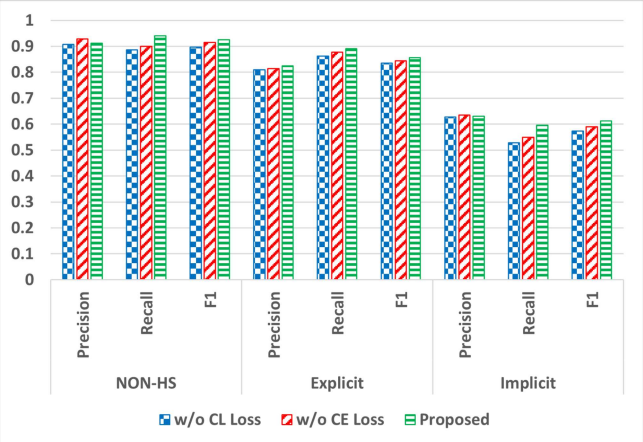

Fig. 5. Influence of Different Losses on Binary Classification Performance

Similarly, omitting CE Loss also negatively impacted the model's effectiveness. The NON-HS F1-score dropped to 0.9142, while explicit detection metrics decreased, with an F1-score of 0.8445, and implicit detection metrics showed declines, with an F1-score of 0.5895. These findings highlight the critical role of CE Loss in sustaining the model's overall performance. Our proposed model, which incorporates both CL Loss and CE Loss, outperformed the ablated versions. For the NON-HS category, the model achieved an F1-score of 0.9260, and for the explicit and implicit classes, it attained F1-scores of 0.8564 and 0.6124, respectively. The overall accuracy of the proposed model reached 0.8895, further confirming its superior performance. In summary, both supervised contrastive loss and cross-entropy loss contribute significantly to the efficacy of our proposed method. The ablation study underscores their importance, demonstrating that the removal of either component results in notable declines in performance across multiple metrics.

3) Influence of Different Features: The results from Table VI illustrate the impact of NRCLex and CoreNLP features on the model's ability to detect abusive content across various categories. In the NON-HS category, excluding NRCLex reduces the F1-score to 0.9145, while removing CoreNLP results in an F1-score of 0.8971. The proposed method, using both NRCLex and CoreNLP, achieves a superior F1-score of 0.9260, underscoring the combined effectiveness of these features.

For explicit content detection, the F1-score drops to 0.8475 and 0.8411 when NRCLex and CoreNLP are excluded, respectively. In contrast, the proposed method achieves an F1-score of 0.8564, demonstrating its improved capability in identifying explicit abusive language by leveraging both emotion and sentiment scores. In the implicit category, the model with both features reaches an F1-score of 0.6124, outperforming the ablated models, which achieve 0.6002 without NRCLex and 0.5964 without CoreNLP. This performance boost highlights the importance of both feature sets in capturing subtle, context-dependent expressions of implicit abuse. Overall, the proposed method achieves an accuracy of 88.95%, higher than the ablations without NRCLex (86.56%) and CoreNLP (86.12%). These results underscore the importance of integrating NRCLex and CoreNLP features, as they significantly enhance the model's precision, recall, and F1-score across all categories,

TABLE VI
ABLATION ANALYSIS ON AUGMENTATION, EMOTION AND SENTIMENT

Features	NON-HS			Explicit			Implicit			Accuracy
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score	Accuracy
w/o NRCLex	0.9047	0.9267	0.9145	0.8327	0.8617	0.8475	0.6290	0.5731	0.6002	0.8656
w/o CoreNLP	0.9066	0.8899	0.8971	0.8296	0.8530	0.8411	0.5975	0.5954	0.5964	0.8612
Proposed	0.9116	0.9408	0.9260	0.8248	0.8906	0.8564	0.6297	0.5960	0.6124	0.8895

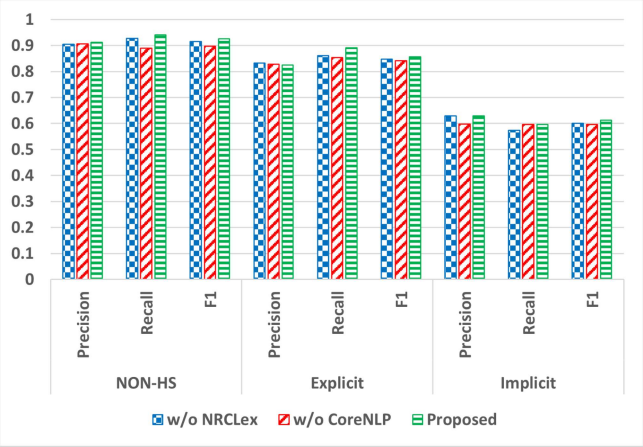

Fig. 6. Influence of NRCLex and CoreNLP Features on Binary Classification Performance

leading to more accurate abusive content detection.

V. CONCLUSION

In conclusion, our findings indicate that leveraging supervised contrastive learning with large-scale data augmentation significantly enhances the detection of implicit hate speech, effectively addressing the challenges posed by its subtle and indirect nature. By incorporating emotion and sentiment features, our approach captures the underlying affective cues, further improving the model's performance. We have demonstrated the effectiveness of our method on two benchmark datasets, ISHate and IHC, where it consistently outperforms state-of-the-art methods in terms of macro-F1 score. Additionally, our ablation analysis highlights the impact of key components, such as data augmentation and the inclusion of affective features, underscoring the robustness and adaptability of our proposed framework for implicit hate speech detection.

REFERENCES

[1] A. Sharma, A. Kabra, and M. Jain, "Ceasing hate with moh: Hate speech detection in hindi–english code-switched language," *Information Processing & Management*, vol. 59, no. 1, p. 102760, 2022.

[2] T. Caselli, V. Basile, J. Mitrović, and M. Granitzer, "Hatebert: Retraining bert for abusive language detection in english," in *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, 2021, pp. 17–25.

[3] N. B. Ocampo, E. Sviridova, E. Cabrio, and S. Villata, "An in-depth analysis of implicit and subtle hate speech messages," in *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 2023, pp. 1997–2013.

[4] N. B. Ocampo, E. Cabrio, and S. Villata, "Playing the part of the sharp bully: Generating adversarial examples for implicit hate speech detection," in *Findings of the Association for Computational Linguistics: ACL 2023*, 2023, pp. 2758–2772.

[5] T. Hartvigsen, S. Gabriel, H. Palangi, M. Sap, D. Ray, and E. Kamar, "Toxigen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2022, pp. 3309–3326.

[6] Y. Kim, S. Park, and Y.-S. Han, "Generalizable implicit hate speech detection using contrastive learning," in *Proceedings of the 29th International Conference on Computational Linguistics*, 2022, pp. 6667–6679.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 4171–4186.

[8] M. ElSherief, C. Ziems, D. Muchlinski, V. Anupindi, J. Seybolt, M. De Choudhury, and D. Yang, "Latent hatred: A benchmark for understanding implicit hate speech," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 345–363.

[9] A. R. Jafari, G. Li, P. Rajapaksha, R. Farahbakhsh, and N. Crespi, "Fine-grained emotions influence on implicit hate speech detection," *IEEE Access*, 2023.

[10] S. MacAvaney, H.-R. Yao, E. Yang, K. Russell, N. Goharian, and O. Frieder, "Hate speech detection: Challenges and solutions," *PloS one*, vol. 14, no. 8, p. e0221152, 2019.

[11] A. Tontodimamma, E. Nissi, A. Sarra, and L. Fontanella, "Thirty years of research into hate speech: topics of interest and their evolution," *Scientometrics*, vol. 126, pp. 157–179, 2021.

[12] F. Ludwig, K. Dolos, T. Zesch, and E. Hobley, "Improving generalization of hate speech detection systems to novel target groups via domain adaptation," in *Proceedings of the Sixth Workshop on Online Abuse and Harms (WOAH)*, 2022, pp. 29–39.

[13] S. Kumar, A. Nagar, A. Kumar, and A. Singh, "Hate speech detection: A survey," in *2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*. IEEE, 2022, pp. 171–176.

[14] A. Chhabra and D. K. Vishwakarma, "A literature survey on multimodal and multilingual automatic hate speech identification," *Multimedia Systems*, vol. 29, no. 3, pp. 1203–1230, 2023.

[15] J. Lin, "Leveraging world knowledge in implicit hate speech detection," *arXiv preprint arXiv:2212.14100*, 2022.

[16] T. Guo, L. Lin, H. Liu, C. Zheng, Z. Tu, and H. Wang, "Implicit offensive speech detection based on multi-feature fusion," in *International Conference on Knowledge Science, Engineering and Management*. Springer, 2023, pp. 27–38.