Journal of Theoretical and Applied Information Technology
15st July 2026. Vol.104. No.13

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

MULTI-CLASS SEVERITY DETECTION OF CYBERBULLYING IN SAUDI-DIALECT TWEETS USING DEEP LEARNING

BADER AZI ALANAZI[1], CHIN-TENG LIN[2], YU-CHENG FRED CHANG[3]

[1]University of Technology Sydney (Australia).
[1]Jouf University (Kingdom of Saudi Arabia).
[2]University of Technology Sydney (Australia).
[3]University of Technology Sydney (Australia).
E-mail: [1]baenzi@ju.edu.sa,[2]Chin-Teng.Lin@uts.edu.au, [3]Fred.Chang@uts.edu.au.

ABSTRACT

Cyberbullying in Arabic social media poses a growing threat to users' psychological well-being, yet automated detection systems largely overlook both severity levels and dialectal variation, particularly the Saudi dialect. This gap limits the practical utility of existing systems for real-world content moderation. This study develops an automated deep-learning approach to classify the severity of cyberbullying in Saudi-dialect tweets into four levels: high, medium, low and non-cyberbullying. Twenty-eight experimental scenarios were constructed by systematically combining Arabic NLP toolkits, stop-word removal strategies and class-balancing techniques. Four deep learning models (CNN, LSTM, BiLSTM and GRU) were trained and evaluated. Data balancing and preprocessing improved classification performance; CAMeL with random oversampling produced the most consistent gains across models. Compared with previous machine learning baselines on the same dataset (maximum accuracy 92.23%), CNN achieved 95.92% and LSTM reached 95.59%. These findings have direct implications for automated content moderation platforms serving Arabic-speaking communities, enabling graduated responses calibrated to cyberbullying severity. To the best of our knowledge, this is the first study that uses deep learning to detect cyberbullying severity in Saudi-dialect tweets, providing a scalable, linguistically informed solution.
Keywords: *Text Classification, Deep Learning, Cyberbullying Detection, Arabic social media, Saudi dialect.*

1. INTRODUCTION

Social-media platforms have become powerful tools for communication and self-expression, enabling users to share ideas, exchange opinions and participate in public debate across diverse communities [1]. Twitter (now X) is especially popular in Saudi Arabia and a key digital interaction platform [2]. On the other hand, the proliferation of these sites has also given rise to cyberbullying that has serious psychological, emotional and social effects on the victim of cyberbullying. Cyberbullying is defined as the intentional and repeated sending of digital messages designed to upset, humiliate or threaten another individual. Persistent harassment online will often cause victims to experience anxiety, depression or even become socially isolated [3]. The growth of Arabic social-media content has made automatic detection an urgent research issue, complicated by the linguistic variations of Arabic and its dialects.

In the Arabic setting, recent research has applied deep learning to detect hate speech, offensive language and cyberbullying in social media. Studies have demonstrated that architectures such as CNN, LSTM, BiLSTM and GRU effectively capture linguistic patterns in Arabic text [4-11]. CNNs excel at extracting local n-gram features [4], while recurrent architectures like LSTM and BiLSTM model sequential dependencies and contextual information [5, 8] By combining convolutional and recurrent layers (e.g., CNN-LSTM, CNN-BiLSTM), hybrid approaches have achieved improved performance [7, 8, 11]. Multi-class detection efforts, such as DRNN-2 for hate-speech classification [6] show promising results for severity-aware systems.

Although studies demonstrate the effectiveness of deep learning to detect cyberbullying in Arabic text, still have limitations. Firstly, most focus on binary classification and overlook the different levels of severity that reflect real-world online aggression. Furthermore, there is a lack of research on dialectal Arabic, particularly the Saudi dialect,

despite its widespread use on Twitter and its distinct linguistic features that set it apart from Modern Standard Arabic (MSA) [9]. Due to limitations, there is a need for dialect-specific, severity-aware detection models that can capture the linguistic and contextual subtleties of abusive language.

In previous research [12], the authors introduced a machine learning (ML) framework designed to assess the severity of cyberbullying in tweets written in the Saudi dialect, categorising tweets into four levels: High, Medium, Low, and Non-Cyberbullying. The study utilised traditional algorithms, including Support Vector Machines (SVM) and Naïve Bayes (NB), across 28 experimental scenarios, incorporating data balancing and feature extraction techniques. The most effective model achieved an accuracy of 92.23%, demonstrating the potential of ML models for multi-class severity detection.

This study introduces a deep learning technique to assess the intensity of cyberbullying, specifically in tweets composed in the Saudi dialect. In contrast to conventional ML models, deep learning models can learn hierarchical representations and capture complex contextual patterns in text, making them well-suited to the linguistic variations and implicit severity cues in Saudi-dialect cyberbullying. Four deep learning models, CNN, LSTM, BiLSTM, and GRU, are systematically evaluated across 28 experimental scenarios involving a variety of preprocessing and data balancing techniques (random insertion, random oversampling, and synonym replacement). This study aims to identify the most effective combination of architecture, preprocessing, and balancing strategies for effective multi-class classification of cyberbullying severity.

The main contributions of this study are summarised as follows:

- Developed and tested a four-class severity detection system on tweets in the Saudi dialect, moving beyond the typical binary classification setup of existing DL studies on Arabic.
- Evaluated four DL models (CNN, LSTM, BiLSTM and GRU) in 28 different scenarios under different preprocessing and data-balancing configurations.
- Performed a comparison with results from existing ML studies on the same dataset, demonstrating the gains from DL approaches.
- Discussed the performance through confusion matrices, training-validation curves, and evaluation metrics, providing a clearer picture of generalisation.
- A contribution to the development of cyberbullying severity detection models for Saudi dialect, with the potential for practical applications in automated content moderation and digital safety tools.

Problem Statement: Despite increasing research on Arabic cyberbullying detection, three key gaps still exist in the literature. First, most current studies rely on binary classification, which fails to capture the severity of online cyberbullying [4-11]. Second, Saudi-dialect Arabic remains largely overlooked, despite its linguistic differences from Modern Standard Arabic and its widespread use on Twitter. As a result, there is no deep learning framework able to perform multi-class severity detection of cyberbullying, particularly in Saudi dialect tweets. This study tackles this gap directly by developing and systematically testing four deep learning models, CNN, LSTM, BiLSTM, and GRU, across 28 experimental scenarios that combine Arabic NLP preprocessing and data-balancing strategies.

Based on the identified gaps, this study is guided by the following research questions:

RQ1: Can deep learning models outperform traditional machine learning approaches in detecting four-class cyberbullying severity in Saudi-dialect tweets?

RQ2: Which deep learning architecture, CNN, LSTM, BiLSTM, or GRU, achieves the best performance for this task?

RQ3: How do Arabic NLP preprocessing toolkits and data-balancing strategies interact with model architecture to affect classification performance?

RQ4: What is the most effective combination of architecture, preprocessing, and balancing strategy for severity-aware cyberbullying detection in Saudi-dialect text?

2. BACKGROUND

2.1 Cyberbullying

Cyberbullying refers to bullying conducted through social media, email, or instant messaging. It involves intentional aggression by an individual or group against a target with limited ability to respond. Researchers[3, 13, 14] similarly define cyberbullying as the deliberate use of platforms

such as Twitter to harm a person or community. Compared to traditional bullying, which usually stops when face-to-face contact ends, online bullying can continue nonstop 24/7. Cyberbullying adversely affects psychological health, self-esteem, and social relationships, often resulting in isolation and severe emotional damage [15]. Studies of victims show increased rates of anxiety and depression, and in severe cases, suicidal thoughts [16]. The impacts go beyond the individual to affect academic performance, peer relationships, family life, and the wider school and community. Survey evidence suggests that about 31% of school students have been bullied online at least once [17]. According to research conducted in Saudi Arabia, 18% of high school students were affected [18], whereas studies conducted in Jazan (42.8%) [19] and King Saud University (41.6%) [20] reported higher rates. The study results show the negative impact of cyberbullying on mental health and academic performance.

Twitter and other social media sites provide multiple options for bullying or harassing other users, including flaming, harassment, denigration, impersonation, and exclusion, each with its own social and psychological effects [21-23]. These methods are intended to harass, threaten, or target victims online.

2.2 The Arabic Language and Natural Language Processing

Natural Language Processing (NLP) allows computers to understand written and spoken language by modelling fundamental linguistic structures mainly syntax and semantics to derive meaning and facilitate further analysis [24-26]. It supports applications like translation, speech recognition, and sentiment analysis by converting unstructured text into structured formats used across industries [25, 27]. Arabic, an official language of the United Nations spoken by millions, presents unique challenges for NLP because of its rich morphology, extensive dialectal differences, and right-to-left 28-letter script; it includes Standard Arabic (Classical and Modern Standard) and various dialects used in daily communication [28, 29]. Despite these challenges, recent tools and resources have led to noticeable progress in Arabic NLP, especially in sentiment analysis and text categorisation, enhancing the effectiveness of Arabic text processing [30, 31].

2.3 Deep Learning in Cyberbullying Detection

Deep learning research has contributed to the advancement of automated detection of cyberbullying and abusive language in general. Many neural networks, such as CNN, LSTM, GRU, and hybrid models like CNN-GRU and CNN-BiLSTM, were used on Arabic content. As an example, Haidar et al.[9] applied FFNN model to detect Cyberbullying speech in Arabic text and achieved promising results. Tashtoush et al. [8] employed a CNN, LSTM, BiLSTM, and a CNN-LSTM hybrid on Arabic YouTube comments and demonstrated the effectiveness of deep learning in cyberbullying detection. Rachid et al.[10] used a CNN and RNN-based model with word-embedding representations to classify cyberbullying in YouTube comments. Meanwhile, Albayari et al.[11] compared various deep-learning models on Arabic cyberbullying datasets, including LSTM, GRU, LSTM with attention, and CNN-BiLSTM. However, most cyberbullying detection studies use binary classification, which limits usefulness when distinguishing severity. The linguistic complexity of Arabic, especially dialectal variation like Saudi dialect, further challenges current methods, and severity-level classification in Saudi dialectal remains unaddressed.

3. RELATED WORK

In recent years, researchers have focused on automating the detection of hate speech, offensive language, and cyberbullying on Arabic social media platforms such as Twitter and YouTube. Neural network architectures regularly outperform traditional feature-engineering approaches, including CNNs, recurrent models such as LSTM, GRU, BiLSTM, and hybrid models. However, three limitations remain: limited coverage of dialectal variation, a lack of multiclass severity annotations, and inconsistent evaluation practices under class imbalance.

3.1 Hate Speech and Offensive Language Detection

Alshalan et al.[4] presented early work on hate-speech detection in Saudi tweets using deep learning models, with a CNN delivering strong results. The system performed well within and out-of-domain evaluations based on the dataset of 9,316 labelled tweets, achieving an F1-score of 0.79 and an AUROC of 0.89 in both assessments. The study established CNNs as effective for Arabic hate-speech detection while underscoring the relative lack of Arabic-focused

research compared with English.

Building on offensive language detection, Aljuhani et al.[5] proposed a BiLSTM model enhanced with domain-specific word embeddings for Arabic Twitter content to identify offensive language. Their approach achieved 93% accuracy, demonstrating that task-specific and domain-specific representations significantly improve classification performance in culturally and contextually nuanced Arabic text. The integration of domain-specific embeddings addressed a key limitation of generic pre-trained models, which often fail to capture offensive expressions unique to Arabic dialects. However, the study focuses on binary classification and does not differentiate between the severity levels of offensive content.

Al Anezi [6] developed this research by applying a deep recurrent neural network (DRNN-2) model to detect multiple hate-speech categories across seven different types such as religious, racial and gender-based hate speech. The model achieved 99.73% accuracy for binary classification, 95.38% for three-class classification, and 84.14% for seven-class detection. These are promising results, but they also indicate significant challenges in multi-class classification, especially regarding how dataset size, label granularity, and data partitioning strategies affect generalizability. The 15.59-point difference from binary to seven classes is a clear indication of the challenge of accurate hate-speech categorisation.

Mohaouchane et al.[7] compared CNN, BiLSTM, BiLSTM with attention and a CNN-LSTM hybrid for detecting offensive Arabic content on YouTube. Using five-fold cross-validation and Bayesian hyperparameter optimisation, the hybrid CNN-LSTM achieved the highest recall of 83.46%. These results show that combining convolutional feature extraction with sequential modelling provides superior sensitivity by capturing local patterns and long-range dependencies in Arabic text. Nevertheless, the study is limited to binary classification and does not address dialectal variation in Arabic content.

A recent comprehensive survey by Itriq and Mohd Noor [32] reviewed deep learning studies on Arabic hate speech detection published between 2020 and 2024, covering CNN, RNN, and transformer-based approaches. The survey confirms that transformer models such as AraBERT and MarBERT achieve state-of-the-art F1 scores of 93-99%, yet highlights that Gulf Arabic dialects, including the Saudi dialect, remain systematically underrepresented and that severity-aware annotation is virtually absent from the literature. These findings directly corroborate the gap this study addresses.

3.2 Cyberbullying Detection

In the cyberbullying field, Tashtoush et al.[8] developed a system for detecting cyberbullying in Arabic YouTube comments. They evaluated CNN, LSTM, BiLSTM, and a CNN-LSTM hybrid in binary and multi-class settings. The CNN and hybrid models achieved the highest binary accuracy at 91.9%, while the LSTM and BiLSTM algorithms achieved the highest multi-class accuracy at 89.5%. The pattern indicates that recurrent encoders perform better in binary classification as label detail increases, although they show slightly lower performance in capturing fine-grained distinctions. Crucially, the dataset used is not dialect-specific, which limits its applicability to regional varieties such as the Saudi dialect.

A feed-forward neural network developed by Haidar et al.[9] was able to achieve 93% accuracy in addressing the limited prior research on Arabic cyberbullying. According to their study, standard deep learning methods, previously successful in other languages, can be adapted efficiently for classifying Arabic cyberbullying, despite the language's morphological complexity. The study does not address severity levels or dialect-specific language features.

Using CNN and RNN models with word-embedding representations, Rachid et al.[10] achieved an F1 of up to 0.84 on Arabic YouTube comments. They also highlighted persistent gaps, insufficiently balanced corpora, and underdeveloped Arabic-specific resources that constrain the domain's deep-learning performance.

Albayari et al.[11] conducted an extensive comparison of various deep-learning models on Arabic cyberbullying datasets, including LSTM, GRU, LSTM with attention, CNN-BiLSTM, CNN-LSTM, and LSTM-TCN. The hybrid model combining CNN-BiLSTM-GRU achieved the highest accuracy overall, demonstrating that well-crafted hybrid architectures can effectively leverage the strengths of individual components.

Al-Ajlan and Ykhlef introduced two frameworks that replace manual feature engineering with learned representations. The

OCDD [33] Twitter-based cyberbullying detection represents tweets as word vectors and uses metaheuristic parameter tuning to preserve semantic structure while reducing the dependency on noisy user metadata. In subsequent work, the authors proposed CNN-CB [34], a method of embedding word embeddings into a CNN to capture semantic similarity in cyberbullying content. CNN-CB achieved 95% accuracy on English Twitter and delivered efficiency gains over traditional feature-based approaches.

More recently, Shaziya et al. [35] evaluated DNN, CNN, LSTM, hybrid CNN+RNN, BERT, and AraBERT on the ArCyC Arabic cyberbullying corpus, with AraBERT achieving the highest accuracy of 95% on textual features. While confirming the effectiveness of deep learning for Arabic cyberbullying detection, the study relies exclusively on binary classification and uses a general Arabic corpus without dialect differentiation, precisely the limitations the present study is designed to address. Similarly, Husain [36] built a dialect-specific offensive language detection framework for Kuwaiti Arabic using fine-tuned AraBERT, demonstrating that Gulf-dialect NLP benefits from dialect-specific modelling; however, the system is confined to binary classification and does not distinguish between severity levels.

Table 1. Summary of Related Work.

Study	Task	Platform/Dialect	Labels/Data Size	Best Model
[4]	Hate Speech	Twitter/Saudi & Arabic mixed	Binary/9316	CNN vs GRU, CNN+GRU, BERT (F1=0.79; AUROC =0.89)
[5]	Offensive Language detection	Twitter/ Arabic mixed	Binary/ 500k	BiLSTM achieved accuracy 93%
[6]	Hate Speech	Social media Mixed / Not specific	Binary & 3-class & 7-class/4203	For Binary: DRNN-1 recognition rate of 99.73%, For multi-class: DRNN-2 95.38% for the three classes and 84.14% for the seven classes
[7]	Offensive Language detection	YouTube/ Arabic mixed	Binary/ 15,050	CNN, BiLSTM, Attn-BiLSTM, CNN-LSTM (best recall= 83.46%), other 80-82%
[8]	Cyberbullying	YouTube/ Arabic Mixed	Binary& multiclass/20,000	For binary the CNN, CMM-LSTM reaching 91.9%, for multiclass the LSTM , Bi-LSTM achieving 89.5%
[9]	Cyberbullying	Social media Mixed / Not specific	Binary/ two datasets	Feed-forward achieved accuracy 93%
[10]	Cyberbullying	News comments / Arabic Mixed	Binary /32000	CNN/LSTM/GRU achieving 84% F1-score
[11]	Cyberbullying	Social media Mixed/ Arabic Mixed	Binary & multiclass / four datasets	For multi-class: CNN-BLSTM achieved the highest accuracy, For Binary: CNN-BLSTM-GRU achieved the highest accuracy
[33]	Cyberbullying	Twitter/ English	Binary/20,000	CNN showed great results
[34]	Cyberbullying	Twitter/ English	Binary/39,000	CNN-CB achieved accuracy 95%
[35]	Cyberbullying	ArCyC Arabic	10000 of Arabic text	AraBERT 95%
[36]	offensive language	Kuwaiti dialect data	16.6k posts	AraBERT 90%

Even though deep-learning methods have proved successful in detecting cyberbullying in Arabic social media content, severe limitations exist (Table 1). Most research relies on binary labels that fail to adequately reflect the severity of cyberbullying. This is confirmed by the most recent literature: a 2025 survey of 42 Arabic hate speech detection studies [32] and a recent Arabic cyberbullying study achieving 95% accuracy [35] both operate within a binary classification framework, while the latest Gulf-dialect offensive language detection work [36] similarly does not address severity levels. Dialectal diversity, especially in the Saudi dialect, is also underrepresented despite its prevalence on social platforms.

The present study addresses these gaps and creates the following new knowledge: (1) it fills the absence of deep learning approaches for severity-aware cyberbullying detection in Saudi-dialect tweets, demonstrating that CNN, LSTM, BiLSTM, and GRU all surpass prior ML baselines on this task; (2) it provides systematic empirical evidence across 28 experimental scenarios on how preprocessing toolkits, stop-word strategies, and data-balancing techniques interact with CNN, LSTM, BiLSTM, and GRU architectures in a low-resource, dialect-specific setting; and (3) it contributes a 3,340-term Saudi-dialect cyberbullying lexicon as a reusable linguistic resource for future Arabic NLP research. Together, these contributions move the field beyond binary detection and general Arabic datasets toward severity-aware, dialect-specific modelling knowledge that is directly applicable to automated content moderation systems in Arabic-speaking online environments.

Journal of Theoretical and Applied Information Technology
15st July 2026. Vol.104. No.13

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4. METHODOLOGY

This research proposes a comprehensive deep-learning framework to automatically detect the severity of cyberbullying in Saudi dialect tweets (Figure 1). The methodology proceeds through six stages: (1) data collection and initial cleaning; (2) text preprocessing; (3) oversampling to address class imbalance; (4) feature extraction; (5) classification using multiple deep-learning architectures; and (6) evaluation across multiple scenarios. Figure 1 presents the system architecture implemented with CNN, LSTM, BiLSTM and GRU.

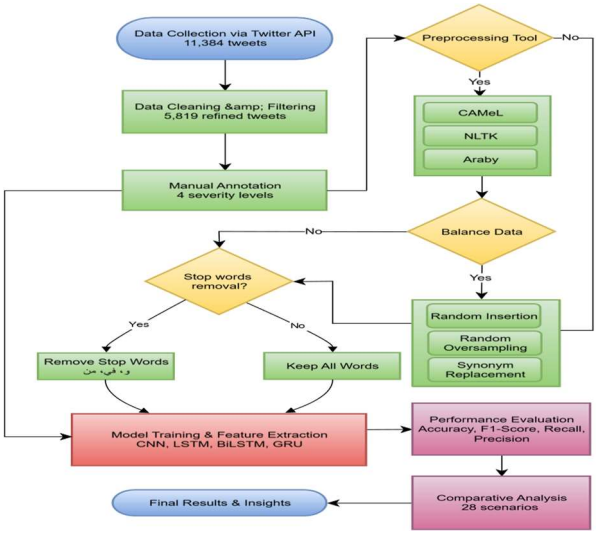


Figure 1. Proposed framework for detecting cyberbullying severity in Saudi-dialect tweets.

4.1 Dataset Construction

4.1.1 Data collection

The dataset was systematically gathered using the official X (Twitter) Developer API via a custom Python script. The data collection approach employed a comprehensive strategy based on keywords and filters, specifically targeting expressions related to cyberbullying and threats common in the Saudi dialect. The process focused on Saudi-specific hashtags and linguistic patterns typically associated with cyberbullying behaviour on Twitter.

A preliminary analysis of cyberbullying trends in Saudi dialects on Twitter discourse guided keyword selection. The collection script applied advanced filtering to preserve relevance and quality, including geographical targeting, language detection and content-type verification. Data were gathered over several months to capture temporal trends and seasonal variation in cyberbullying behaviour.

The automated pipeline retrieved 11,384 tweets. The corpus was then organised and exported into structured Excel workbooks, following rigorous de-duplication that removed redundant entries using text-similarity thresholds and metadata comparison. This structure facilitated subsequent manual review, annotation and quantitative analysis.

4.1.2 Data cleaning

The tweets were cleaned using a multi-stage pipeline to improve quality and reduce noise. A custom Python script initially removed digital noise and irrelevant content, focusing on: removing URLs to prevent bias toward external content; removing hashtags; removing @-mentions based on context; filtering numbers; and removing non-Arabic content to ensure linguistic consistency.

A subsequent manual review ensured consistency across Saudi dialects and removed commercial or promotional posts and non-text media descriptions. This process reduced the corpus from 11,384 to 5,819 high-quality tweets, yielding a carefully curated dataset for analysing the severity of Saudi dialect cyberbullying.

4.2 Data Annotation

4.2.1 Annotation framework

The annotation process employed a theoretically grounded methodology that followed established cyberbullying classification frameworks adapted to the Saudi dialect. Drawing on methods outlined in relevant literature [37], this research developed a comprehensive four-tier severity classification system to capture the nuanced nature of cyberbullying behaviour in Saudi digital discourse.

The annotation framework included cultural sensitivity considerations essential for accurate assessment in the Saudi dialect. This incorporated an understanding of local social norms, religious considerations, and cultural prohibitions that influence the perception and impact of cyberbullying behaviour.

4.2.2 Severity classification

The developed system categorises content into four severity levels with clear criteria: High (appearance-targeting or explicit sexual references, highest harm, violations of dignity values-e.g., body shaming, sexual harassment),

Medium (racial/religious attacks; systemic rather than personal-e.g., discrimination, sectarianism), Low (minor insults, general negativity, mild social aggression-e.g., criticism, mild name-calling, social exclusion) and Non-cyberbullying(neutral/positive/informational/constructive content). (Table 2) summarises the scheme.

Table 2. Cyberbullying Severity Levels by Category.

Category	Severity level
Sexual/ Appearance	High
Political / Racial	Medium
General	Low
Non	Non-cyberbullying

4.2.3 Annotation process and quality assurance

An extensive manual annotation protocol was implemented to ensure reliability and consistency. Each tweet was annotated, and the whole procedure was repeated three times to strengthen inter-annotator reliability, minimise subjective bias and secure accurate severity labels. Two native speakers of the Saudi dialect and cultural context conducted comprehensive reviews and verified the annotations. A lexicon of 3,340 cyberbullying terms specific to the Saudi dialect was compiled. (Table 3) presents representative examples from each severity category, illustrating the linguistic and contextual variation identified during the analysis.

Table 3. Examples of Tweets and Severity Classification.

Tweet	Translate	severity
أي شخص يحترمها أو يدافع عنها - سأبصق في وجهه.	Anyone who respects or defends her I'll spit in his face.	Low
مجموعة من الأوغاد يبالغون في تقدير بعضهم البعض - رغم أنوفكم.	A bunch of lowlifes hyping each other up despite your snouts.	Medium
إنه يطاردها بلا مبالاة - ماذا تتوقع منه؟ *اسم*، ابن المرأة المنبوذة.	He's blindly chasing after her what do you expect from him? *Name*, the outcast woman's son.	High
لأول مرة على تويتر، سورة البقرة كاملة في مقطع واحد؛ القارئ هو سعود الشريم. ضع التغريدة في المفضلة.	For the first time on Twitter, Surah Al-Baqarah is complete in one clip; the receiver is Saud Al-Shuraim. Put the tweet in your favourites.	Non-cyberbullying

4.2.4 Dataset Distribution

The final annotated dataset consists of 5,819 tweets across the four severity categories shown in (Table 4). . The distribution of the four classes reflects the occurrence of cyberbullying on Saudi social media, with non-cyberbullying tweets the most frequent (2,703), followed by high-severity (1,587), low-severity (1,299), and medium-severity (221) tweets. The class imbalance in this distribution required the application of oversampling techniques, which are explained in the next section.

Table 4. Distribution of Annotated Tweets by Severity Category.

Category	Annotated Tweets
High	1587
Medium	221
Low	1299
Non-cyberbullying	2703

4.3 Data Preprocessing

To address Arabic challenges, three established toolkits were evaluated: CAMeL [38], NLTK [39] and the Araby [40]. Because each provides distinct capabilities, a comparative assessment was necessary to identify the optimal configuration for this task.

- **CAMeL Tools -** comprehensive Arabic morphology, with advanced tokenisation, normalisation and morphological disambiguation.
- **NLTK -** general-purpose NLP with Arabic support (stemming, tokenisation and basic normalisation).
- **Araby -** lightweight preprocessing with emphasis on efficiency and language

handling.

The evaluation assessed effectiveness based on Saudi dialect content, processing accuracy, computational efficiency, and ease of integration into the larger pipeline.

4.3.1 Preprocessing pipeline implementation

The preprocessing pipeline applies a staged sequence of transformations that normalises Arabic text while retaining semantics relevant to cyberbullying detection:

- **Stage 1** Tokenisation. Text is split into tokens and routines specific to the Arabic script (directionality, right-to-left; connected letter shapes; word-boundary detection) are applied. For example, "اليوم السهرة ياسلام" is tokenised into "ياسلام", "السهرة", "اليوم".
- **Stage 2** Diacritics removal. Phonetic diacritics (tashkeel/ḥarakāt) are removed in order to eliminate visible variation and generate standard forms. For example, "السَّلامُ عَلَيْكُم ورَحْمَةُ اللهِ وَبَرَكاتُهُ" is reduced to "السلام عليكم ورحمة الله وبركاته".
- **Stage 3** Text normalisation. Orthographic variants are mapped to standard forms, with a focus on alif variants and other letters exhibiting spelling variability (e.g.,ٱ/إ/أ to ا), yielding consistent representations.
- **Stage 4** Letter repetition reduction. Emphatic elongations common in informal writing are collapsed to base forms while preserving meaning, e.g., "رااائع" to "رائع".

4.4 Oversampling Techniques for Dataset Balancing

The distribution of annotated data indicates the dataset is unbalanced. Class imbalance in machine learning models can lead to biasing the majority classes and poor performance on minority classes. Three oversampling methods were used: random insertion, oversampling, and synonym replacement. They were chosen based on prior success on text data.

- **Random insertion**: identify a content word, select an Arabic synonym, and insert it in a random position; continue with constraints until balance is reached [41].
- **Random oversampling**: minority-class tweets are duplicated until class counts are equal, with light variation to avoid exact duplicates that could promote overfitting [42].
- **Synonym replacement**: substitute chosen non-stop words with context-appropriate synonyms; apply context checks, frequency caps and automatic grammar validation [41].

Annotators manually reviewed a 100-random sample of each technique's outputs for semantic preservation, linguistic quality, and category consistency. Results indicated high accuracy across all criteria, supporting the integrity of the augmentation. As a result of oversampling, the dataset achieved perfect balance at 2,703 instances per class (Table 5).

Table 5. Distribution after Oversampling.

Category	Number
High	2703
Medium	2703
Low	2703
Non-cyberbullying	2703

4.5 Deep Learning Models

The study implements four deep learning models with complementary strengths for text classification and sequence modelling: CNN, LSTM, BiLSTM and GRU. CNNs are used for local pattern extraction, LSTM for long-range context, BiLSTM for bidirectional context, and GRU for a leaner recurrent option. This architectural diversity enables a comprehensive assessment of feature-extraction and sequence-modelling strategies, and provides insights into an optimal strategy for detecting cyberbullying severity in Saudi dialect text.

Hyperparameters were chosen through an empirical process. Experiments were conducted to determine the best embedding size using dimensions 128, 200, 300 and 400. The CNN model, which was well-suited to detect n-gram patterns, performed best with 128-sized embeddings. On the other hand, LSTM, BiLSTM, and GRU models were used with 200-sized embeddings to provide a more sufficient dimensionality for the sequential modelling of dependencies. The rest of the configuration, dropout scheduling, layer depth and pooling strategies were adapted to trade-off model capacity and regularisation, in order to achieve a reliable generalisation of

Saudi-dialect cyberbullying text.

4.5.1 Convolutional Neural Network (CNN)

The CNN leverages convolutional (Figure 2) operations to learn local, position-invariant features that are informative for cyberbullying severity [4, 43]. Tokens are first mapped to 128-dimensional pre-trained embeddings and regularised with spatial dropout (rate 0.2). Feature extraction is performed by three parallel convolutional branches with kernel sizes of 3, 5 and 7 (128 filters each), followed by batch normalisation and dim embeddings activation ($\alpha = 0.1$) to stabilise training and maintain gradient flow. Global max pooling then aggregates the most salient activations across the sequence. The concatenated representation is then passed through two dense layers of 256 and 128 units with dropout 0.4 followed by a softmax classification layer to predict the probabilities of each of the four classes. Using multiple kernel widths allows the model to capture signals at multiple granularities, from short phrases to longer patterns of contextual information, resulting in a final system that can provide accurate severity classification for Saudi dialect tweets.

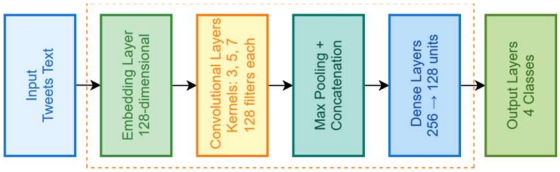


Figure 2. Architecture of the CNN model.

4.5.2 Long Short-Term Memory (LSTM)

The LSTM architecture (Figure 3) sequentially models text through gating mechanisms that selectively retain and update information across the sequence, a crucial capability for modelling contextual relations and temporal patterns in cyberbullying discourse [8, 11, 44]. Inputs are mapped to 200-dimensional embeddings with Gaussian noise (0.1) for robustness and are regularised with spatial dropout at 0.2. Three LSTM layers with 128, 128 and 64 units learn temporal structure under dropout and recurrent-dropout settings of 0.2, with layer normalisation applied after each layer to stabilise optimisation. The final representation combines global average and global max pooling and is subsequently processed by two dense layers (256 and 128 units) with batch normalisation and dropout, while L1-L2 regularisation helps prevent overfitting. The network employs focal loss to mitigate class imbalance inherent in severity detection, improving responsiveness to minority classes.

Figure 3.Architecture of the LSTM model.

4.5.3 Bidirectional Long Short-Term Memory (BiLSTM)

The BiLSTM architecture (Figure 4) extends standard LSTM by reading each sequence in both forward and backward directions, yielding a fuller view of context critical for accurate severity assessment in Arabic cyberbullying [8, 45]. The network comprises three bidirectional LSTM layers with a progressive reduction in 128, 128 and 64 units to capture rich temporal structure while gradually compressing the representation. The final outputs are pooled using global average and max operations to summarise sequence evidence, providing complementary statistics over the token stream. Regularisation is applied throughout via dropout, together with batch normalisation and layer normalisation, to stabilise optimisation. As class imbalance is intrinsic to the task, focal loss is used during training to emphasise harder, minority examples. By conditioning on both preceding and following words, the model resolves context that a unidirectional model may miss, improving severity classification.

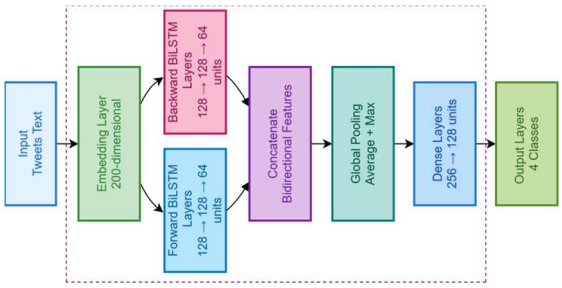


Figure 4. Architecture of the Bidirectional LSTM model.

4.5.4 Gated Recurrent Unit (GRU)

The GRU architecture (Figure 5) delivers computational efficiency while preserving effective sequence modelling, improving training speed and memory efficiency without forfeiting temporal context [11, 46]. Three GRU layers are

stacked with decreasing dimensionality to regulate capacity and improve stability. Feature information is obtained from pooled representations and then refined by dense layers. The outcome is a good model that adapts well to large datasets without significant loss in accuracy.

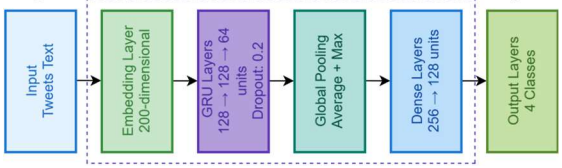


Figure 5. Architecture of the GRU model.

4.5.5 Training configuration and optimisation

To compare all models equally, the same base configuration was used for training. Optimisation was performed with Adam optimiser with gradient clipping (clipnorm=1.0) and a learning rate of 0.001, scheduled by ReduceLROnPlateau; and with a batch size of 32. Early stopping with a patience of 10 epochs and best weights restore was used, along with model checkpointing to automatically save the best-performing model. Regularisation with architecture-tuned dropout, strategic placement of batch normalisation to minimise internal covariate shift, L1-L2 penalties on dense layers and balanced class weights was applied to tackle the problem of class imbalance. Models were trained for up to 50 epochs with early stopping, with 15% of the training data held out for validation. Five-fold cross-validation was used to tune hyperparameters. Accuracy, precision, recall, F1-score and AUC were used as performance metrics.

4.5.6 Performance evaluation framework

Each scenario is evaluated with metrics suited to cyberbullying detection. Accuracy measures overall correctness across severity categories; precision reflects specificity by calculating the proportion of true positives among optimistic predictions; recall assesses sensitivity by the proportion of true positives among actual positives; and the F1-score provides a balanced summary of precision and recall [47]. The metrics are computed using standard definitions:

$$Accuracy = \frac{Number\ of\ Correct\ Prediction}{Total\ Number\ of\ Predictions} \quad (1)$$

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision \times Recall} \quad (2)$$

$$Recall = \frac{True\ Positives}{True\ Positives \times False\ Negatives} \quad (3)$$

$$Precision = \frac{True\ Positives}{True\ Positives \times False\ Positives} \quad (4)$$

5. EXPERIMENTS

5.1 Experimental design

The experimental design uses a full-factorial scheme to evaluate all possible combinations of factors and levels. The experimental factors are the original imbalanced dataset, the application of balancing methods (random insertion, random oversampling, and synonym replacement), the toolkit used to preprocess the dataset (CAMeL, NLTK, Araby and a non-preprocessed baseline), the use of stop-words in the corpus (removed or not) and the DL neural architecture used (CNN, LSTM, BiLSTM, and GRU). This approach yields a set of 28 experimental conditions, enabling a global evaluation of factor interactions to identify the best-performing combinations.

5.2 Result

This section reports the performance of four deep learning models, CNN, LSTM, BiLSTM, and GRU, applied to cyberbullying severity classification in Saudi dialect tweets. Each model was evaluated across 28 experimental scenarios, resulting from variations in data preprocessing and balancing techniques. The classification task involved four severity levels: high, medium, low, and non-cyberbullying. The models were evaluated using four standard metrics: accuracy, precision, recall, and F1 score [47]. All training and testing splits were stratified to preserve the distribution of severity classes.

The first data set was initially imbalanced. Then, three oversampling techniques were applied to create a balanced dataset: random insertion, random oversampling, and synonym replacement, yielding the other groups. Each imbalanced and balanced dataset was combined with different preprocessing pipelines using Arabic NLP tools: CAMeL, NLTK, and Araby, with and without stop-word removal. This systematic design produced seven scenarios per group, leading to 28 scenarios to evaluate each

deep learning model (Table 6).

Table 6. Summary of the 28 Data Scenarios Result.

Scenario	CNN				LSTM				BiLSTM				GRU			
	Accuracy	F1-Score	Recall	Precision	Accuracy	F1-Score	Recall	Precision	Accuracy	F1-Score	Recall	Precision	Accuracy	F1-Score	Recall	Precision
Original Data +No preprocessing	**92.38%**	0.871	0.859	0.884	84.87%	0.823	0.811	0.866	89.56%	0.852	0.835	0.879	84.91%	0.786	0.772	0.823
Original Data+ CAMeL	92.27%	0.895	0.878	0.917	88.08%	0.853	0.839	0.883	89.67	0.863	0.858	0.876	85.82%	0.828	0.821	0.853
Original Data+ CAMeL+No stop words	90.61%	0.882	0.885	0.879	**89.70%**	0.835	0.893	0.809	**90.85%**	0.894	0.886	0.908	89.23%	0.870	0.867	0.877
Original Data+ NLTK	92.20%	0.902	0.901	0.904	85.62%	0.771	0.851	0.761	90.82%	0.897	0.896	0.901	86.46%	0.853	0.857	0.860
Original Data+ NLTK+ No stop words	91.79%	0.885	0.895	0.877	86.63%	0.851	0.858	0.856	89.33%	0.877	0.881	0.881	86.90%	0.850	0.858	0.854
Original Data+ Araby	91.12%	0.890	0.878	0.903	86.43%	0.835	0.844	0.834	88.42%	0.867	0.863	0.881	83.53%	0.816	0.817	0.838
Original Data+ Araby+ No stop words	90.88%	0.870	0.861	0.880	83.09%	0.729	0.804	0.738	90.41%	0.874	0.853	0.906	**89.94%**	0.868	0.853	0.899

Table 6.(continue)

Scenario	CNN				LSTM				BiLSTM				GRU			
	Accuracy	F1-Score	Recall	Precision	Accuracy	F1-Score	Recall	Precision	Accuracy	F1-Score	Recall	Precision	Accuracy	F1-Score	Recall	Precision
Random Insertion balanced Data+No preprocessing	93.27%	0.932	0.932	0.932	93.33%	0.933	0.933	0.933	92.22%	0.922	0.922	0.922	90.90%	0.909	0.909	0.915
Random Insertion balanced Data+CAMeL	93.72%	0.937	0.937	0.937	93.19%	0.931	0.931	0.931	92.32%	0.924	0.923	0.928	92.41%	0.924	0.924	0.926
Random Insertion balanced Data+CAMeL+ No stop words	93.93%	0.939	0.939	0.939	93.27%	0.932	0.932	0.932	91.97%	0.920	0.919	0.925	71.30%	0.661	0.713	0.716
Random Insertion balanced Data+NLTK	93.48%	0.934	0.934	0.934	93.63%	0.936	0.936	0.936	92.29%	0.923	0.922	0.926	**91.58%**	0.916	0.915	0.919
Random Insertion balanced Data+NLTK+ No stop words	93.90%	0.938	0.939	0.939	93.48%	0.935	0.934	0.936	93.63%	0.936	0.936	0.937	91.28%	0.9133	0.912	0.917
Random Insertion balanced Data+Araby	**94.23%**	0.942	0.942	0.942	93.66%	0.936	0.936	0.936	93.10%	0.931	0.931	0.934	90.87%	0.909	0.908	0.915
Random Insertion balanced Data+Araby+ No stop words	94.08%	0.940	0.940	0.940	**93.75%**	0.937	0.937	0.937	**93.90%**	0.938	0.939	0.939	90.24%	0.902	0.902	0.905

Table 6.(continue

Scenario	CNN				LSTM				BiLSTM				GRU			
	Accuracy	F1-Score	Recall	Precision	Accuracy	F1-Score	Recall	Precision	Accuracy	F1-Score	Recall	Precision	Accuracy	F1-Score	Recall	Precision
Random Oversampling balanced Data+No preprocessing	95.56%	0.955	0.955	0.956	94.85%	0.948	0.948	0.948	94.43%	0.944	0.944	0.946	94.52%	0.945	0.945	0.946
Random Oversampling balanced Data+CAMeL	95.86%	0.958	0.958	0.959	95.00%	0.950	0.950	0.950	**94.61%**	0.946	0.946	0.948	**94.53%**	0.945	0.945	0.947
Random Oversampling balanced Data+CAMeL+ No stop words	**95.92%**	0.959	0.959	0.959	94.20%	0.942	0.942	0.944	94.55%	0.945	0.945	0.948	94.82%	0.948	0.948	0.949
Random Oversampling balanced Data+NLTK	95.65%	0.956	0.956	0.956	**95.59%**	0.955	0.955	0.956	94.17%	0.942	0.941	0.944	94.23%	0.942	0.942	0.944
Random Oversampling balanced Data+NLTK+No stop words	95.62%	0.956	0.956	0.956	95.12%	0.951	0.951	0.952	94.43%	0.944	0.944	0.947	94.40%	0.944	0.944	0.945
Random Oversampling balanced Data+Araby	95.71%	0.957	0.957	0.957	94.88%	0.949	0.948	0.950	94.40%	0.944	0.944	0.946	94.38%	0.944	0.943	0.945
Random Oversampling balanced Data+Araby+No stop words	94.93%	0.957	0.957	0.957	94.72%	0.947	0.947	0.948	94.58%	0.946	0.945	0.948	94.11%	0.941	0.941	0.943

Table 6.(continue)

Scenario	CNN				LSTM				BiLSTM				GRU			
	Accuracy	F1-Score	Recall	Precision	Accuracy	F1-Score	Recall	Precision	Accuracy	F1-Score	Recall	Precision	Accuracy	F1-Score	Recall	Precision
Synonym Replacement balanced Data+No preprocessing	93.90%	0.939	0.939	0.939	93.16%	0.931	0.931	0.933	92.74%	0.928	0.927	0.933	91.88%	0.919	0.918	0.920
Synonym Replacement balanced Data+CAMeL	94.38%	0.943	0.943	0.943	93.84%	0.938	0.938	0.938	93.39%	0.9343	0.933	0.935	92.26%	0.923	0.922	0.925
Synonym Replacement balanced Data+CAMeL+No stop words	**94.67%**	0.946	0.946	0.946	**94.70%**	0.947	0.947	0.947	93.66%	0.936	0.936	0.938	92.28%	0.928	0.928	0.930
Synonym Replacement balanced Data+NLTK	93.57%	0.935	0.935	0.936	93.84%	0.938	0.938	0.938	93.78%	0.938	0.937	0.939	92.44%	0.924	0.924	0.927
Synonym Replacement balanced Data+NLTK+No stop words	94.08%	0.940	0.940	0.940	93.81%	0.938	0.938	0.938	93.69%	0.937	0.936	0.937	**92.56%**	0.925	0.925	0.927
Synonym Replacement balanced Data+Araby	93.54%	0.935	0.935	0.935	93.81%	0.938	0.938	0.938	**94.02%**	0.940	0.940	0.940	90.75%	0.907	0.907	0.909
Synonym Replacement balanced Data+Araby+No stop words	93.87%	0.938	0.938	0.938	94.20%	0.942	0.942	0.942	92.56%	0.926	0.925	0.931	89.71%	0.897	0.897	0.900

5.3 Model Performance Across Scenarios

The CNN model consistently outperforms all other models in all experimental scenarios. First, the accuracy of the original imbalanced dataset ranged from 90.61% to 92.38%. However, significant improvements were observed when data balancing was applied. In the random insertion group, CNN achieved up to 94.23% accuracy when combined with Araby preprocessing. Through a random oversampling group, the CNN achieved its highest accuracy of 95.92% using CAMeL with stop-word removal. Additionally, the synonym replacement technique achieved strong results, with an accuracy of 94.67% using CAMeL and stop-word removal. According to these results, CNN is particularly well-suited to this task, especially when augmented with aggressive data augmentation and effective preprocessing.

The LSTM model also demonstrated significant improvements on balanced datasets. The performance of LSTM was lower than that of CNN on the original imbalanced data, with accuracies ranging from 83.09% to 89.70% across preprocessing methods. The performance of the LSTM improved after data balancing was implemented. A random insertion group procedure yielded an accuracy of 93.75% for LSTM with Araby and stopword removal. The random oversampling group achieved a maximum of 95.59 % with NLTK preprocessing. The model achieved 94.70% accuracy using CAMeL with stopword removal in the synonym replacement group. These findings confirm the impact of oversampling on LSTM's sequential learning capabilities, which appear to benefit from more balanced exposure to each severity class.

The BiLSTM model showed consistent results across all experiments. In the first group, BiLSTM outperformed LSTM and GRU in some scenarios, achieving an accuracy of 90.85% when combined with CAMeL and stop word removal. As with the other models, performance improved with balancing techniques, especially in oversampling and synonym replacement scenarios. The higher accuracy in this model was 94.61% in the random oversampling scenario with CAMeL preprocessing. Moreover, the BiLSTM consistently scored highly on precision and recall. BiLSTM achieved 94.02% accuracy in the synonym replacement group using Araby, demonstrating its adaptability to various preprocessing tools.

The GRU model performed best under the same conditions as the others, specifically in random oversampling and CAMeL preprocessing scenarios. In this scenario, GRU achieved an accuracy of 94.53%, nearly matching the top performance of all other models. The model performed relatively poorly on the original dataset, with a maximum accuracy of 89.94% on the Araby dataset with stop word removal. In general, the performance of this model was improved especially in balancing data scenarios. Though GRU occasionally matched the accuracy of BiLSTM and CNN, its precision and recall metrics remained consistently high. This suggests that GRU can effectively model long-term dependencies in text data when supported by appropriate preprocessing and balanced input.

A comparative analysis shows that all four models benefited from balanced, preprocessed datasets, particularly those produced by random oversampling. Across preprocessing strategies, CAMeL frequently featured in the strongest combinations, often with stop-word removal. By accuracy alone, CNN ranked first at 95.92%, followed by LSTM at 95.59%, BiLSTM at 94.61%, and GRU at 94.53% (Table 7). Although there are slight differences, they highlight each architecture's strengths when tuned to favourable data settings. Consistent improvements in F1 scores, recall and precision were also observed across all models, indicating balanced classification across all severity levels.

Table 7. Best Result for each Model.

Model	Data scenario	Best Accuracy
CNN	Random Oversampling balanced Data+CAMeL+ No stop words	95.92%
LSTM	Random Oversampling balanced Data+NLTK	95.59%
BiLSTM	Random Oversampling balanced Data+CAMeL	94.61%
GRU	Random Oversampling balanced Data+CAMeL	94.53%

Figure 6 presents the best-performing configurations in the first scenario (imbalanced data). All four models achieved accuracy above 89%, with CNN demonstrating the strongest performance at 92.38% using original data and no preprocessing, followed closely by BiLSTM ,LSTM, and GRU with CAMeL/Araby preprocessing and no stop-word removal.

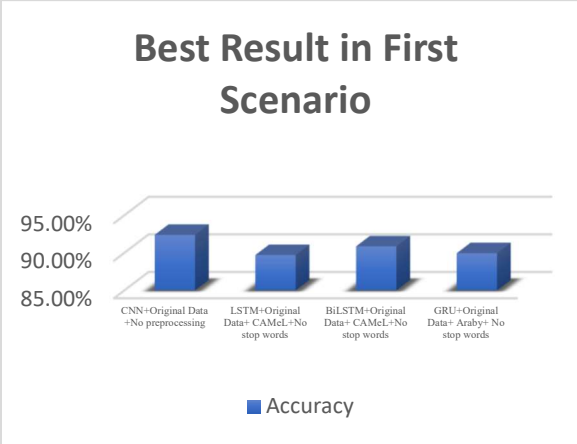


Figure 6. Best Result in First Scenario Imbalanced Data.

Figure 7 shows the best results for the second scenario using random insertion for data balancing. With Araby preprocessing, the CNN achieved an accuracy of 94.61%, higher than that of the imbalanced baseline. LSTM, BiLSTM and GRU also showed consistent performance gains with Araby/NLTK preprocessing, indicating that random insertion effectively addresses class imbalance in Saudi-dialect cyberbullying detection.

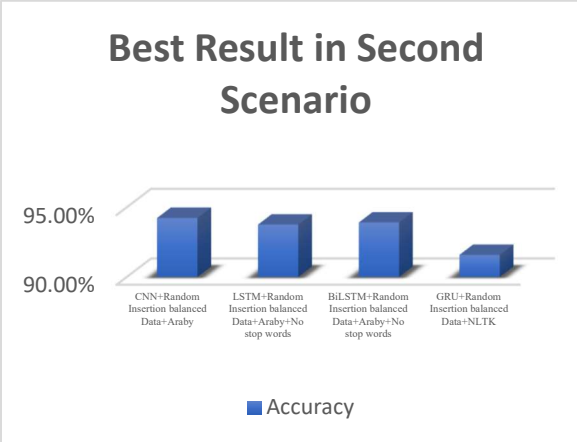


Figure 7. Best Result in Second Scenario Random Insertion Balanced Data.

Figure 8 shows the best performance in the third scenario, which uses random oversampling to balance the data. The CNN achieved an accuracy of 95.92% with CAMeL preprocessing and stop word removal, while LSTM achieved 95.59% with NLTK preprocessing. Across all experiments, this scenario yielded the highest result, demonstrating that combining random oversampling with CAMeL preprocessing is the most effective approach for evaluating the severity of cyberbullying in tweets in the Saudi dialect.

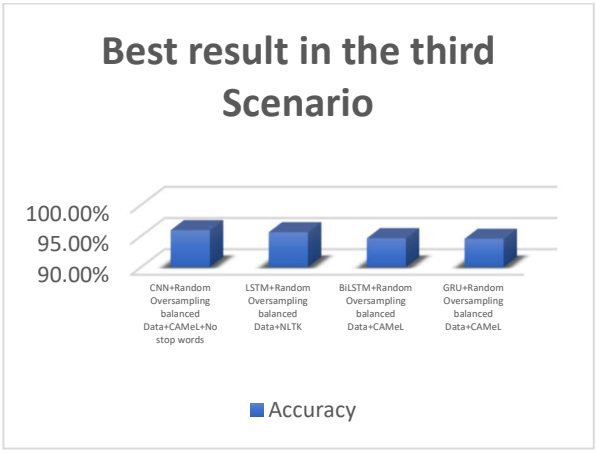


Figure 8. Best result in the third Scenario Random Oversampling Balanced Data.

Figure 9 displays the best results for the fourth scenario using synonym replacement for data balancing. With CAMeL preprocessing, the LSTM achieved the highest accuracy of 94.70%, while the CNN attained a close 94.67% under the same conditions. Synonym replacement shows good results across all architectures.

Journal of Theoretical and Applied Information Technology
15st July 2026. Vol.104. No.13

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

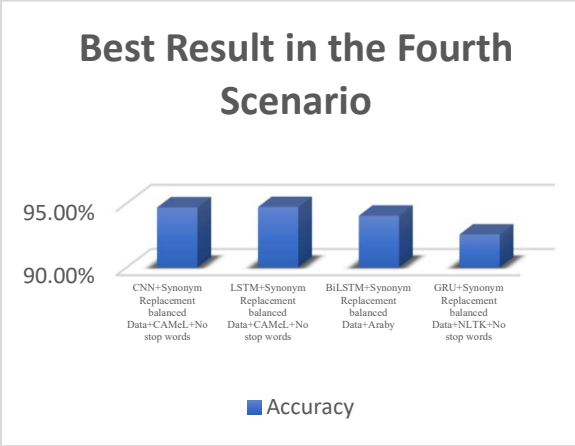


Figure 9. Best Result in the Fourth Scenario Synonym Replacement Balanced Data.

5.4 Confusion Matrix and Training Curves

Class-wise behaviour was analysed using confusion matrices for the two best configurations. For CNN (Figure 10), which achieved 95.92% with CAMeL preprocessing, stop-word removal, and random oversampling, precision was high across all classes; the "Medium" class was correctly classified, and precision for "High" exceeded 95%. The remaining errors were few and concentrated at the "Non-Cyberbullying" and "Low" boundary, a common ambiguity in social media text. LSTM (Figure 10) reached 95.59% with a similar setup. Although slightly lower in overall accuracy than CNN, it provided balanced precision and recall, also achieving perfect classification for "Medium" and firm results for "High" and "Low".

Learning curves over 50 epochs reflect efficient, stable training. For CNN (Figure 12), the validation accuracy curve rose sharply and then plateaued at around 96% after about 10 epochs, with little deviation from the training curve; the loss curve decreased smoothly, consistent with good regularisation and strong generalisation.

LSTM (

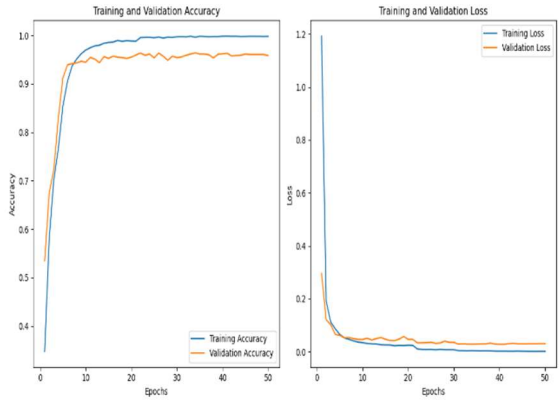


Figure 13) displayed similar stability, with smoothly decreasing losses and only minor variance between training and validation accuracy, suggesting limited overfitting. These patterns reflect the contribution of early stopping, Dropout and learning-rate scheduling to reliable optimisation.

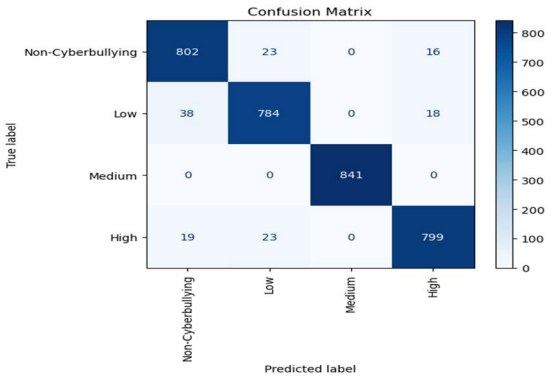


Figure 10. Confusion Matrix for CNN.

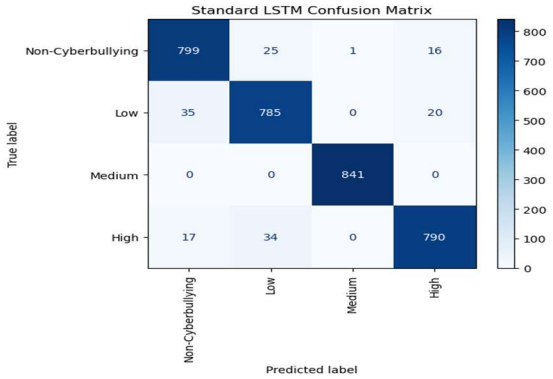
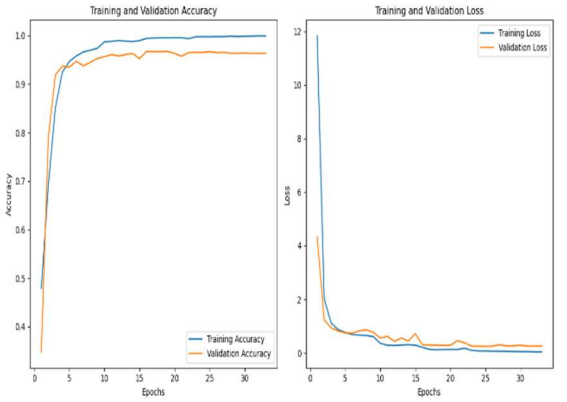


Figure 11. Confusion Matrix for LSTM.

Figure 12. CNN Training & Validation Curves.



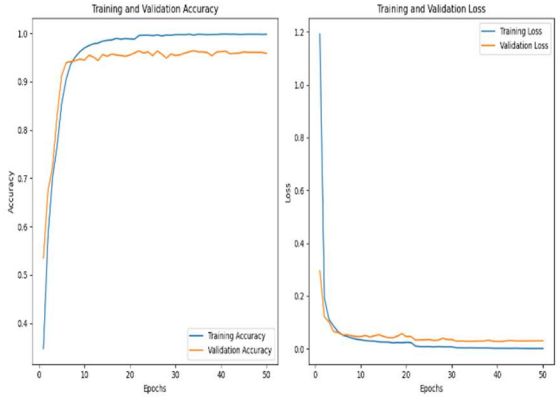


Figure 13. LSTM Training & Validation Curves.

In general, the results show the importance of architecture selection and the role of preprocessing and balancing strategies design in enhancing the performance in Arabic cyberbullying severity detection. The best-performing models used combinations that enhanced linguistic normalisation.

5.6 Discussion

5.6.1 Deep learning model

The model-wise comparison results show that CNN achieves the highest accuracy of 95.92%. The combination of multiple convolutional kernels and pooling layers effectively captures n-gram-level features of Arabic texts, even when accompanied by spatial dropout and other regularisation methods. In addition, combining CAMeL with stop-word removal contributed to this outcome, plausibly because CAMeL handles the morphological richness of Arabic effectively.

The LSTM performed strongly, with its best accuracy of 95.59% under random oversampling and NLTK preprocessing, a configuration that couples linguistic normalisation with balanced class exposure. Its capacity to model long-range context helps with longer, context-heavy tweets on X. Although slightly below CNN in peak accuracy, the LSTM maintained high recall across all severity levels, indicating a strong ability to detect cyberbullying severity.

BiLSTM showed consistent behaviour and practical interpretability across different configurations. Encoding context in both forward and backward directions uses cues from surrounding tokens, which is an advantage for Arabic, where word order is relatively flexible. Although BiLSTm failed to outperform CNN or LSTM in peak accuracy, its results remained high with only minor differences. With a top accuracy of 94.61% under random oversampling with CAMeL, BiLSTM proved effective at recognising cyberbullying severity in the Saudi dialect.

Although GRU remained generally competitive, its results were slightly lower than those of other models. The highest accuracy of 94.53% was achieved with random oversampling and CAMeL preprocessing. Although having a simpler structure than LSTM and BiLSTM, GRU maintained strong performance in terms of precision and F1-score, especially with balanced data conditions. These findings indicate that GRU could be a practical choice in environments with limited computational resources; however, it proved effective at recognising cyberbullying severity in the Saudi dialect.

The evaluation criteria used in this study, accuracy, F1-score, precision, and recall, are consistent with standard practice in similar Arabic cyberbullying and hate speech detection research [4-11], allowing for direct comparison. These metrics are especially important in a four-class, imbalanced scenario, where F1-score and recall

help ensure that minority severity classes, such as "High" and "Medium," are not overshadowed by the majority class. Compared to similar studies, Tashtoush et al. [8] reported 89.5% multi-class accuracy, Albayari et al. [11] achieved their best results with hybrid architectures, and Shaziya et al. [35] reached 95% accuracy, although only in binary classification. The current study attains 95.92% on a more challenging four-class problem, ranking these results among the strongest reported for Arabic cyberbullying detection.

5.6.2 Impact of hyperparameter tuning

This study emphasised the impact of hyperparameter tuning on model performance. Learning rate and epoch combinations were tested across CNN, LSTM, BiLSTM, and GRU to balance convergence speed with generalisation. The study tested rates of 0.009, 0.005, 0.007 and "auto", and applied ReduceLROnPlateau to lower the rate when validation loss plateaued. Lower rates were generally better, especially for deeper models such as CNN and BiLSTM. The best were around 0.0005.

For a number of training epochs, tests were conducted with 100, 200, and 300, across multiple trials, but these longer training schedules did not outperform 50. For the final experiments we fixed 50 as the number of epochs, and used early stopping with patience of 8-10, so the training could halt as soon as the performance on the validation set stabilises, without affecting the generalisation capability.

All models applied regularisation Dropout, SpatialDropout1D and L1/L2 penalties together with BatchNormalization and LayerNormalization, collectively stabilising training for deeper architectures and complex inputs. The CNN architecture used multi-kernel convolutional blocks and stacked dense layers under strong regularisation. The BiLSTM and GRU models also employed three recurrent layers with Gaussian noise and structured dropout to manage temporal dependencies.

Overall, the precisely tuned training strategy adjusting learning rate, epoch length, and regularisation was essential to each model's performance, especially when combined with the balanced, preprocessed dataset. This combination yielded excellent accuracy, consistently strong recall across all classes, and efficient training across all 28 scenarios.

5.6.3 Effect of preprocessing and data balancing

Across 28 scenarios, CNN, LSTM, BiLSTM, and GRU performance depended on architecture, preprocessing, class balance and tuning when detecting cyberbullying severity in Saudi-dialect tweets. Class imbalance reduced performance across all models, most noticeably for the "medium" and "low" classes, whereas augmentation with random insertion, random oversampling, and synonym replacement improved accuracy, precision, recall, and F1-score. Random oversampling yielded the most consistent gains, particularly with CAMeL or NLTK preprocessing. These results underscore the value of balanced data in limiting bias towards majority classes and supporting reliable severity classification.

5.6.4 Performance of Arabic NLP tools

The selection of Arabic NLP tool was an important factor. The comparison between CAMeL, NLTK and Araby shows that CAMeL consistently performs better when used in conjunction with stop word removal. The superiority of CAMeL is most likely a result of its ability to handle diacritics, affixes and morphological variants, which are common problems in Arabic processing. Stop word removal was not consistently beneficial, but in several cases it led to small, consistent improvements in accuracy and recall, especially for CNN and LSTM. In comparison to CAMeL and NLTK, Araby generally achieved slightly lower results, most likely due to differences in morphological analysis.

5.6.5 Comparison with machine learning

Furthermore, the same 28 scenarios were previously evaluated with SVM and Naïve Bayes, attaining a top accuracy of 92.23%. In the present results, the deep learning models exceeded this benchmark 95.92% for CNN and 95.59% for LSTM, indicating an advantage in modelling Arabic text's complex, context-dependent structure (Table 8. Comparison Result with the pervious study ML. The improvement points that deep learning architectures learn richer, context-sensitive representations that support finer discrimination among severity levels in dialectal Arabic and greater robustness under class imbalance.

Table 8. Comparison Result with the pervious study ML.

Model	Scenario	Highest Accuracy
BoW+NB	Random Oversampling balanced Data+CAMeL	86.41%
BoW+SVM	Random Oversampling balanced Data+NLTK+ stop words removal	92.23%
CNN	Random Oversampling balanced Data+CAMeL+ stop words removal	95.92%
LSTM	Random Oversampling balanced Data+NLTK	95.59%
BiLSTM	Random Oversampling balanced Data+CAMeL	94.61%
GRU	Random Oversampling balanced Data+CAMeL	94.53%

5.7 Statistical Significance Analysis

To strengthen the evidence for performance differences, comprehensive statistical significance tests were conducted across all 28 experimental scenarios [48]. This thorough analysis determines whether observed variations in model performance, preprocessing effectiveness, and data balancing techniques represent genuine effects or are due to random variation.

5.7.1 Deep Learning architecture comparison

A Friedman test confirmed the statistical significance of the differences between the four models in F1-score, χ^2 (3) = 47.51, $p < 0.001$. As shown in Post-hoc pairwise comparisons using Wilcoxon signed-rank tests with the Bonferroni correction ($\alpha = 0.0083$), CNN was significantly better than all other models (Table 9).

Table 9. Model Performance Statistics.

Model	Mean F1-score	SD	Min	Max	Median
CNN	0.929	0.029	0.870	0.959	0.938
BiLSTM	0.920	0.029	0.852	0.946	0.933
LSTM	0.909	0.060	0.729	0.955	0.937
GRU	0.902	0.043	0.786	0.948	0.915

n the pairwise comparison between CNN and GRU, the effect size (Cohen's d = 0.70, $p < 0.001$ was of medium magnitude and CNN's mean F1-score was 3.4% higher than GRU's. The difference between CNN and BiLSTM was smaller ($\Delta = 0.009$), but remained statistically significant ($p < 0.001$, Cohen's d = 0.30), ruling out random variation as an explanation for CNN's superior performance.

5.7.2 Data balancing techniques analysis

Pairwise comparisons post hoc tests revealed that random oversampling was significantly better than the other three methods (H (3) = 91.08, $p < 0.001$, Kruskal-Wallis test). As expected, random oversampling (ROS) had the highest F1-score (0.948 ± 0.006), significantly outperforming synonym replacement (0.932 ± 0.011, $p < 0.001$, d = 1.83), random insertion (0.927 ± 0.013, $p < 0.001$, d = 2.11), and the imbalanced baseline (0.853 ± 0.040, $p < 0.001$, d = 3.37). ROS provided an 11.2% improvement (large effect size) over the imbalanced baseline, demonstrating the importance of correcting class imbalance for this task (Table 10).

Table 10. Balancing Technique Performance.

Technique	Mean F1-score	SD	Improvement vs. Baseline	Effect Size
Random Oversampling	0.948	0.006	+0.096 (+11.2%)	Very Large (d = 3.37)
Synonym Replacement	0.932	0.011	+0.080 (+9.3%)	Large (d = 2.72)
Random Insertion	0.927	0.013	+0.074 (+8.7%)	Large (d = 2.52)
Imbalanced (baseline)	0.853	0.040	—	—

5.7.3 Preprocessing toolkit analysis

Comparison of all pre-processed scenarios (CAMeL, NLTK or Araby; N = 96) with the non-pre-processed scenarios (N = 16) revealed a small but reliable benefit of preprocessing. F1-score averaged 0.916 (SD = 0.042) for the pre-processed configurations, and 0.908 (SD = 0.049) for the non-preprocessed data. This corresponds to an overall 0.9% improvement. The effect was small, but the consistently high performance for the pre-processed settings (F1 ≥ 0.91) indicates that Arabic NLP preprocessing reliably supports

effective deep learning, and that CAMeL, NLTK and Araby provide broadly similar normalisation benefits (Table 11).

Table 11. Preprocessing Impact.

Condition	N	Mean F1-score	SD	vs. No Preprocessing
Preprocessed (Combined)	96	0.916	0.042	+0.008 (+0.9%)
• **CAMeL**	32	0.919	0.036	+0.011 (+1.2%)
• **NLTK**	32	0.918	0.040	+0.010 (+1.1%)
• **Araby**	32	0.912	0.049	+0.004 (+0.4%)
No preprocessing	16	0.908	0.049	—

5.7.4 Stop-word removal effect

The Wilcoxon signed-rank tests (N = 48) did not show a significant effect of stop-word removal on either F1-score (W = 471.5, p = 0.784, Cohen's d = 0.004) or accuracy (W = 535.0, p = 0.587, Cohen's d = 0.040). The mean F1-score stayed at 0.916 with or without stop-word removal. Deep learning models trained can apparently learn to weight stop words properly without explicit removal, preserving potentially useful discourse.

It is important to recognise that stop-word removal remains a key preprocessing step in Arabic NLP applications, where reducing dimensionality can greatly improve computational efficiency. These findings are specific to this study and should not be taken as evidence that stop-word removal is always ineffective in Arabic NLP.

5.7.5 Summary of statistical findings

Statistical analyses reveal the following findings: (1) CNN significantly outperforms others ($p<0.001$), (2) random oversampling is the main factor ($d=3.37$), (3) preprocessing toolkits perform similarly, and (4) the optimal configuration setup (CNN + ROS + CAMeL) achieves an F1 score of 0.959.

6. Conclusion

Cyberbullying in Saudi- dialect Arabic social media poses a growing threat that automated detection systems have struggled to tackle. Current deep learning studies on Arabic cyberbullying mainly use binary classification and general Arabic corpora, leaving severity-aware, dialect- specific detection largely unexplored. This study aimed to address that gap by investigating whether deep learning models could reliably identify four levels of cyberbullying severity- high, medium, low, and non- cyberbullying- in Saudi- dialect tweets.

A systematic evaluation of CNN, LSTM, BiLSTM, and GRU across 28 experimental scenarios, using three Arabic NLP toolkits and three data- balancing strategies, leads to several clear conclusions. First, deep learning models are highly suitable for this task: all four architectures outperformed the previous ML baseline of 92.23%, with CNN reaching 95.92% and LSTM 95.59%, showing that hierarchical feature learning offers significant benefits over conventional methods for dialect- specific, multi-class severity detection. Second, data balancing is crucial: random oversampling greatly improved results over the imbalanced baseline (d = 3.37), highlighting the importance of addressing class imbalance before choosing architectures in low-resource dialectal environments. Third, dialect-aware preprocessing is important: CAMeL consistently outperformed NLTK and Araby, confirming that morphologically informed tools are better suited to Saudi- dialect text than general- purpose options.

These findings have immediate practical implications. The ability to identify severity levels rather than just flag content as harmful or not allows content moderation systems to respond proportionately- ranging from educational interventions for low- severity cases to urgent action for high- severity threats. This is especially relevant in Saudi Arabia, where cyberbullying among students ranges from 18% to 42.8% [18-20].

Limitations of this research include reliance on a single- platform dataset (Twitter/X) and evaluating individual architectures separately. Future research should investigate hybrid architectures such as CNN- LSTM and CNN-BiLSTM- GRU, combining convolutional layers for local feature extraction with recurrent layers for sequential understanding, to further enhance severity detection. Expanding the framework to cover other Gulf and Arabic dialects would also be a valuable step toward developing broader dialect- aware cyberbullying detection systems.

REFERENCES

[1] R. Luttrell, *Social media: How to engage, share, and connect*. Bloomsbury Publishing USA, 2025.
[2] S. R. Department. "Number of users of twitter in Saudi Arabia 2019-2028." https://www.statista.com/statistics/558404/number-of-twitter-users-in-saudi-arabia/ (accessed 17/9, 2024).
[3] M. Campbell and S. Bauman, "Cyberbullying: Definition, consequences, prevalence," in *Reducing cyberbullying in schools*: Elsevier, 2018, pp. 3-16.
[4] R. Alshalan and H. Al-Khalifa, "A deep learning approach for automatic hate speech detection in the saudi twittersphere," *Applied Sciences,* vol. 10, no. 23, p. 8614, 2020.
[5] K. O. Aljuhani, K. H. Alyoubi, and F. S. Alotaibi, "Detecting Arabic offensive language in microblogs using domain-specific word embeddings and deep learning," *Tehnički glasnik,* vol. 16, no. 3, pp. 394-400, 2022.
[6] F. Y. A. Anezi, "Arabic hate speech detection using deep recurrent neural networks," *Applied Sciences,* vol. 12, no. 12, p. 6010, 2022.
[7] H. Mohaouchane, A. Mourhir, and N. S. Nikolov, "Detecting offensive language on arabic social media using deep learning," in *2019 sixth international conference on social networks analysis, management and security (SNAMS)*, 2019: IEEE, pp. 466-471.
[8] Y. Tashtoush, A. Banysalim, M. Maabreh, S. Al-Eidi, O. Karajeh, and P. Zahariev, "A Deep Learning Framework for Arabic Cyberbullying Detection in Social Networks," *Computers, Materials & Continua,* vol. 83, no. 2, 2025.
[9] B. Haidar, M. Chamoun, and A. Serhrouchni, "Arabic cyberbullying detection: Using deep learning," in *2018 7th international conference on computer and communication engineering (iccce)*, 2018: IEEE, pp. 284-289.
[10] B. A. Rachid, H. Azza, and H. H. B. Ghezala, "Classification of cyberbullying text in Arabic," in *2020 International Joint Conference on Neural Networks (IJCNN)*, 2020: IEEE, pp. 1-7.
[11] R. Albayari, S. Abdallah, and K. Shaalan, "Cyberbullying detection model for arabic text using deep learning," *Journal of Information & Knowledge Management,* p. 2450016, 2024.
[12] C.-T. L. Bader Azi Alanazi, "Severity Detection of Cyberbullying in Saudi-Dialect Tweets: A Machine-Learning Approach," *International Journal of Engineering Trends and Technology* 2025.
[13] A. König, M. Gollwitzer, and G. Steffgen, "Cyberbullying as an act of revenge?," *Journal of Psychologists and Counsellors in Schools,* vol. 20, no. 2, pp. 210-224, 2010.
[14] P. K. Smith, J. Mahdavi, M. Carvalho, S. Fisher, S. Russell, and N. Tippett, "Cyberbullying: Its nature and impact in secondary school pupils," *Journal of child psychology and psychiatry,* vol. 49, no. 4, pp. 376-385, 2008.
[15] D. t. Label. "Cyberbullying Statistics: What They Tell Us." May 15, 2017. https://www.ditchthelabel.org/cyber-bullying-statistics-what-they-tell-us (accessed 06/07/2023, 2023).
[16] D. Goebert, I. Else, C. Matsu, J. Chung-Do, and J. Y. Chang, "The impact of cyberbullying on substance use and mental health in a multiethnic sample," *Maternal and child health journal,* vol. 15, pp. 1282-1286, 2011.
[17] T. Beran and Q. Li, "The relationship between cyberbullying and school bullying," *The Journal of Student Wellbeing,* vol. 1, no. 2, pp. 16-33, 2007.
[18] N. Alrasheed, S. Nishat, A. B. Shihah, A. Alalwan, and H. Jradi, "Prevalence and risk factors of cyberbullying and its association with mental health among adolescents in saudi arabia," *Cureus,* vol. 14, no. 12, 2022.
[19] G. Gohal *et al.*, "Prevalence and related risks of cyberbullying and its effects on adolescent," *BMC psychiatry,* vol. 23, no. 1, p. 39, 2023.
[20] N. A. Alissa and R. A. Shryei, "CYBERBULLYING AMONG FEMALE COLLEGE STUDENTS IN SAUDI ARABIA," *International Journal of Child, Youth and Family Studies,* vol. 16, no. 1, pp. 52-66, 2025.
[21] R. Kumar and A. Bhat, "A study of machine learning-based models for detection, control, and mitigation of cyberbullying in online social media," *International Journal of Information Security,* vol. 21, no. 6, pp. 1409-1431, 2022.
[22] D. Maher, "Cyberbullying: An ethnographic case study of one Australian upper primary school class," *Youth Studies Australia,* vol. 27, no. 4, pp. 50-57, 2008.

[23] N. M. Zainudin, K. H. Zainal, N. A. Hasbullah, N. A. Wahab, and S. Ramli, "A review on cyberbullying in Malaysia from digital forensic perspective," in *2016 International Conference on Information and Communication Technology (ICICTM)*, 2016: IEEE, pp. 246-250.
[24] E. D. Liddy, "Natural language processing," 2001.
[25] R. Egger and E. Gokce, "Natural Language Processing (NLP): An Introduction," *Applied Data Science in Tourism,* 2022.
[26] K. Chowdhary and K. Chowdhary, "Natural language processing," *Fundamentals of artificial intelligence,* pp. 603-649, 2020.
[27] Y. Kang, Z. Cai, C.-W. Tan, Q. Huang, and H. Liu, "Natural language processing (NLP) in management research: A literature review," *Journal of Management Analytics,* vol. 7, no. 2, pp. 139-172, 2020.
[28] M. Abdul-Mageed and M. T. Diab, "AWATIF: A Multi-Genre Corpus for Modern Standard Arabic Subjectivity and Sentiment Analysis," in *LREC*, 2012, vol. 515, pp. 3907-3914.
[29] M. Elmahdy, R. Gruhn, W. Minker, and S. Abdennadher, "Survey on common Arabic language forms from a speech recognition point of view," in *proceeding of International conference on Acoustics (NAG-DAGA)*, 2009, pp. 63-66.
[30] K. Darwish *et al.*, "A panoramic survey of natural language processing in the Arab world," *Communications of the ACM,* vol. 64, no. 4, pp. 72-81, 2021.
[31] M. Abd Elaziz, M. A. Al-qaness, A. A. Ewees, and A. Dahou, "Recent Advances in NLP: The Case of Arabic Language," 2019.
[32] M. Itriq and M. H. M. Noor, "Arabic hate speech detection using deep learning: a state-of-the-art survey of advances, challenges, and future directions (2020–2024)," *PeerJ Computer Science,* vol. 11, p. e3133, 2025.
[33] M. A. Al-Ajlan and M. Ykhlef, "Optimized twitter cyberbullying detection based on deep learning," in *2018 21st Saudi Computer Society National Computer Conference (NCC)*, 2018: IEEE, pp. 1-5.
[34] M. A. Al-Ajlan and M. Ykhlef, "Deep learning algorithm for cyberbullying detection," *International Journal of Advanced Computer Science and Applications,* vol. 9, no. 9, 2018.
[35] H. Shaziya, "Cyberbullying Detection in Arabic Text Using Different Deep Learning Approaches," *Wasit Journal of Computer and Mathematics Science,* vol. 4, no. 2, pp. 45-55, 2025.
[36] F. Husain, "Towards safe digital communication: building an offensive language detection framework for Kuwaiti Arabic," *Social Network Analysis and Mining,* 2026.
[37] B. A. Talpur and D. O'Sullivan, "Cyberbullying severity detection: A machine learning approach," *PloS one,* vol. 15, no. 10, p. e0240924, 2020.
[38] O. Obeid *et al.*, "CAMeL tools: An open source python toolkit for Arabic natural language processing," in *Proceedings of the twelfth language resources and evaluation conference*, 2020, pp. 7022-7032.
[39] E. Loper and S. Bird, "Nltk: The natural language toolkit," *arXiv preprint cs/0205028,* 2002.
[40] T. Zerrouki, "PyArabic: A Python package for Arabic text," *Journal of Open Source Software,* vol. 8, no. 84, p. 4886, 2023.
[41] J. Wei and K. Zou, "Eda: Easy data augmentation techniques for boosting performance on text classification tasks," *arXiv preprint arXiv:1901.11196,* 2019.
[42] A. Glazkova, "A comparison of synthetic oversampling methods for multi-class text classification," *arXiv preprint arXiv:2008.04636,* 2020.
[43] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882,* 2014.
[44] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation,* vol. 9, no. 8, pp. 1735-1780, 1997.
[45] G. Liu and J. Guo, "Bidirectional LSTM with attention mechanism and convolutional layer for text classification," *Neurocomputing,* vol. 337, pp. 325-338, 2019.
[46] M. Zulqarnain, R. Ghazali, M. G. Ghouse, and M. F. Mushtaq, "Efficient processing of GRU based on word embedding for text classification," *JOIV: International Journal on Informatics Visualization,* vol. 3, no. 4, pp. 377-383, 2019.
[47] D. M. Powers, "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation," *arXiv preprint arXiv:2010.16061,* 2020.
[48] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine learning research,* vol. 7, no. Jan, pp. 1-30, 2006.