future internet

MDPI

Review

A Review on Deep-Learning-Based Cyberbullying Detection

Md. Tarek Hasan [1], Md. Al Emran Hossain [1], Md. Saddam Hossain Mukta [1], Arifa Akter [1], Mohiuddin Ahmed [2] and Salekul Islam [1,*]

[1] Department of Computer Science and Engineering, United International University, Plot-2, United City, Madani Avenue, Badda, Dhaka 1212, Bangladesh
[2] School of Science, Edith Cowan University, Joondalup 6027, Australia
* Correspondence: salekul@cse.uiu.ac.bd

Abstract: Bullying is described as an undesirable behavior by others that harms an individual physically, mentally, or socially. Cyberbullying is a virtual form (e.g., textual or image) of bullying or harassment, also known as online bullying. Cyberbullying detection is a pressing need in today's world, as the prevalence of cyberbullying is continually growing, resulting in mental health issues. Conventional machine learning models were previously used to identify cyberbullying. However, current research demonstrates that deep learning surpasses traditional machine learning algorithms in identifying cyberbullying for several reasons, including handling extensive data, efficiently classifying text and images, extracting features automatically through hidden layers, and many others. This paper reviews the existing surveys and identifies the gaps in those studies. We also present a deep-learning-based defense ecosystem for cyberbullying detection, including data representation techniques and different deep-learning-based models and frameworks. We have critically analyzed the existing DL-based cyberbullying detection techniques and identified their significant contributions and the future research directions they have presented. We have also summarized the datasets being used, including the DL architecture being used and the tasks that are accomplished for each dataset. Finally, several challenges faced by the existing researchers and the open issues to be addressed in the future have been presented.

Keywords: cyberbullying; machine learning; data representations; deep learning; frameworks

Citation: Hasan, M.T.; Hossain, M.A.E.; Mukta, M.S.H.; Akter, A.; Ahmed, M.; Islam, S. A Review on Deep-Learning-Based Cyberbullying Detection. *Future Internet* **2023**, *15*, 179. https://doi.org/10.3390/fi15050179

Academic Editor: Wei Yu

Received: 19 February 2023
Revised: 18 April 2023
Accepted: 23 April 2023
Published: 11 May 2023

1. Introduction

Bully that occurs through the Internet is called cyberbullying, or cyber harassment [1]. There are different forms of cyberbullying that we can observe nowadays. For example, writing indecent textual content and sharing inappropriate visual content, e.g., memes. Social media platforms such as Facebook, Instagram, Twitter, etc. have made it easier for us to create content, interact with others and connect with others. However, unfiltered exchange of message content and the missing protection of private information can lead to bullying on different social media platforms [2]. Cyberbullies could be in any form, including flames, vitriolic comments, sending offensive emails, humiliating pictures, mean remarks made by comments, and harassing others by posting on blogs or social media. Bullies may bring severe consequences such as depression, which may even lead people to commit suicide [3,4].

Detecting cyberbullying is important to stop the threatening problem. Detection of cyberbullying is a difficult task due to the lack of identifiable parameters and the absence of a quantifiable standard. These contents are short, noisy, and unstructured, with incorrect spelling and symbols. Sometimes users intentionally obfuscate the words or phrases (e.g., b***h, a**, etc.) in the sentence to deceive automatic detection [5]. Researchers use traditional machine learning (ML) algorithms to identify cyberbullying (i.e., text and image format), whereas the majority of the existing solutions are based on supervised learning methods [6]. Due to the subjective nature of bully expressions, traditional ML

models perform lower in detecting cyber harassment than the deep learning (DL)-based approaches [7]. A recent study shows that DL models outperform traditional ML algorithms regarding cyberbullying identification. Deep Neural Networks such as Recurrent Neural Network (RNN), Gated Recurrent Unit (GRU) [8], Long Short-term Memory (LSTM) [9], Bi-LSTM [10] and several other DL models can be used to detect this problem.

Introducing DL-based models for detecting cyberbullying over traditional models has several benefits. When the data size is large, several studies [11–14] have shown that DL algorithms outperform the traditional ML algorithms. Extracting features manually for text and image classification is a tedious and error-prone task. Sometimes exploiting traditional ML models are not reasonable to extract features, whereas in DL-based models, the task is performed automatically in the hidden layers. However, extracting features intelligently is an essential task during cyberbullying detection from text and image [15–18]. In addition, understanding the context of the text or images increases the chance of providing better accuracy [19–23]. When we have minimum domain knowledge, the performance of ML algorithms is prone to deteriorating over time during solving complex problems [24].

Furthermore, conventional ML models suffer in model adaptability and transferability. For instance, if we train a model over a YouTube dataset and reuse the model over a Twitter dataset, using ML will not provide the desired results. DL models outperform ML models when we encounter complex linguistic expressions such as harassment with cyberbullying [25].

Figure 1 shows a typical cyberbullying detection pipeline where different steps from social media data input to cyberbullying detection have been explained. In this pipeline, the input dataset can contain either the text data or the image data, which are collected from social media. The cyberbully image data can be extracted by using two methods: optical character recognition (OCR) and image similarity. On the other hand, the raw text data are sent to data preprocessing for improving the data quality. Various text preprocessing steps including data cleaning, tokenization, stemming, lemmatization, and stop word removal are used for the reduction of dimensionality. After the preprocessing step, feature extraction is carried out to transform the raw data into numerical features, which are more meaningful to a machine learning model. Next, the outcome is sent to a deep-learning-based cyberbully detection module for detecting the cyberbully contents. Finally, the cyberbullying content is classified as bully or non-bully. Consider the case where we need to identify cyberbullying on a social media site such as Twitter. The text of tweets would be analyzed by a text-based model to spot any words or phrases that suggest harassment, aggression, or discrimination. Tweets containing the words “kill yourself”, “ugly” or “stupid”, for instance, might be labeled as cyberbullying. Again, consider that we want to identify cyberbullying on an Instagram-like photo-sharing app. Images with offensive gestures, hate symbols, or violent scenes are just a few examples of visual cues that can be used by an image-based model to analyze the content of images and detect cyberbullying.

In this paper, we have reviewed the research works that focus on automated cyberbullying detection using either DL models [26–29] or supervised learning techniques [30,31]. A few papers focused on the DL models, but none concentrated on the frameworks and area of applications to detect cyberbullying. Prior survey papers do not present an overall ecosystem for cyberbullying detection methods to understand the DL-based solution systems’ comprehensive structure. These studies do not show publicly available datasets for cyberbullying detection, and a few survey papers address the open issues and challenges [26,27]. The absence of a globally acknowledged definition of cyberbullying is one of the major issues in the studied literature on automated cyberbullying detection.

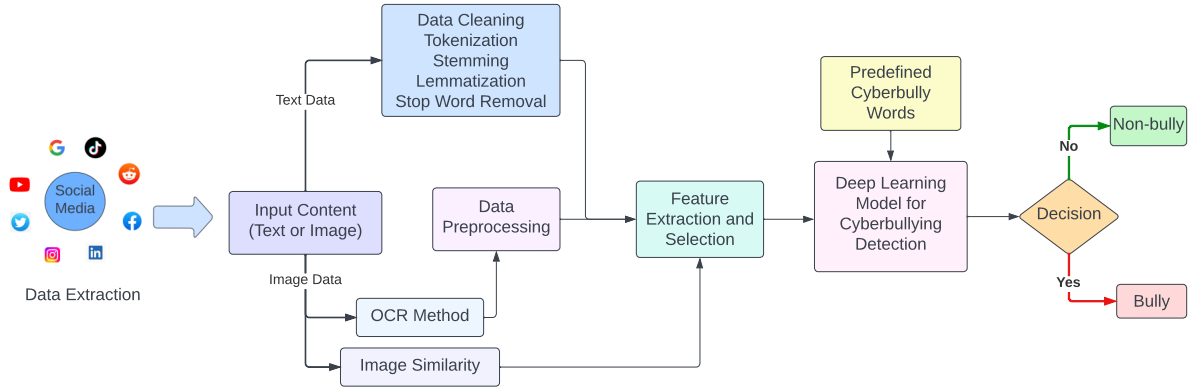

Figure 1. A typical cyberbullying detection pipeline.

In this study, we first develop a clear taxonomy for our DL-based cyberbullying ecosystems. Figure 2 shows the graphical presentation of the detailed taxonomy of our cyberbullying ecosystem. The ecosystem broadly encompasses data representation techniques, DL models, and DL frameworks. To predict cyberbullying behavior, we collect datasets from the Internet where the content can be texts or images. Machine learning algorithms are typically fed vectorized numeric data, but the natural language or images are non-numeric data. To represent the data to be compatible with machine learning algorithms, we use data representation techniques to present the data in a numeric form. To convert the text into numeric form, we generally use two types of word-embedding techniques: pretrained and non-pretrained. Pretrained word-embedding techniques include but are not limited to Word2Vec [32,33], GloVe [34], ELMo [35], fastText [36], and BERT [37] whereas One-hot encoding and TF-IDF [38] are nonpretrained. To convert the image data to numeric data, we use effective, Graph, or ANN-based methods. After converting the non-numerical data into numeric data, we deliberately choose a suitable deep-learning algorithm. If the problem requires a generative model, then we might choose the Boltzmann Machine (BMs) [39], Deep Belief Network (DBN) [40], Deep Autoencoder (DAE) [41], and Generative Adversarial Network (GAN) [42] techniques. If the problem demands a discriminative model, Convolutional Neural Network (CNN) [43] and Recurrent Neural Network (RNN) [44] might be chosen. However, the problem may need to utilize a hybrid model if the dataset is multi-modal (i.e., image, text, speech) or different algorithms may enhance the accuracy by fusing multiple techniques. To simplify the process of building and training Deep Neural Networks by providing pre-built libraries and abstractions, several popular deep learning frameworks (i.e., TensorFlow, Torch, Theano, etc.) have been introduced. Note that Sections 4–7 present the contents of this cyberbullying ecosystem in detail.

Although numerous studies have been conducted on cyberbullying, a limited number of survey papers on DL-based cyberbullying have been found in the literature. We reviewed existing surveys that cover various aspects of cyberbullying. In this paper, we present several applications related to cyberbullying detection, mainly in social media, YouTube, Wikipedia and Q/A discussion forums, using RNN and CNN-based techniques. Users likely communicate with each other through these virtual platforms, and the perpetrators exhibit their creepy nature through digital devices. We also present the datasets that are used in various cyberbullying detection applications, which have different modalities such as text, photographs, collages, memes, etc. Finally, we also discuss the challenges and open issues of detecting cyberbullying, which might be a thought-provoking matter for future researchers. Since cyberbullying has a strong involvement in human psychology, how users respond to this misdemeanor might be exciting due to its multi-modal nature, i.e., image, emotion, culture, language, etc.

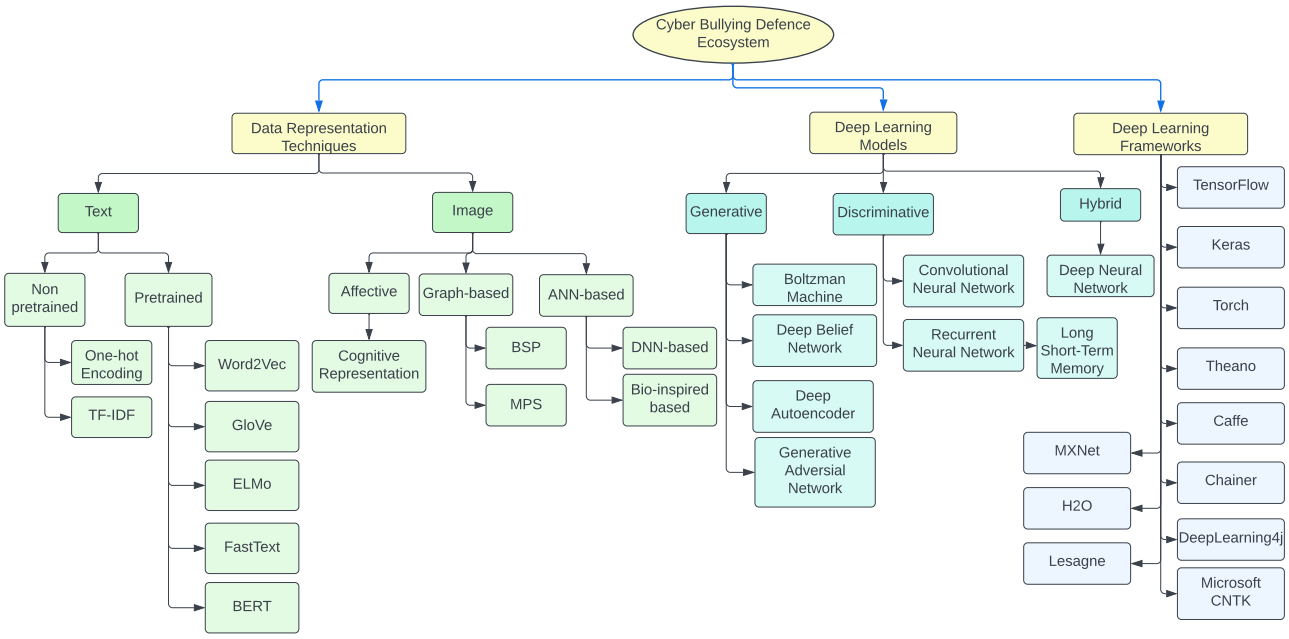

Figure 2. Taxonomy of our DL-based cyberbullying defense ecosystem.

The motivation of this review paper lies in scrutinizing the shortcomings of state-of-the-art approaches to address the automatic detection of cyberbullying. In addition, we have identified the gaps in the existing literature and have filled out the latest improvement in the above aspects. We conduct a complete review of the existing problems, lack of traditional representation and ML models, contemporary frameworks, available datasets and scope of future works. In summary, this paper has the following salient contributions:

- We present a DL-based cyberbullying defense ecosystem with the help of a taxonomy. We also discuss data representation, models and frameworks for DL techniques.
- We compare several RNN, CNN, attention, and their fusion-based cyberbullying detection studies in the existing literature.
- We analyze several text and image datasets extracted from social media and virtual platforms related to cyberbullying detection.
- We identify the challenges and open issues related to cyberbullying.

The organization of the paper is presented graphically in Figure 3. In section 2, we briefly present the existing surveys related to our work. Section 4 discusses the data representation techniques. Sections 5 and 7 present DL-based models and frameworks, respectively. Sections 6 and 8 present applications of DL models in cyberbullying and several popular datasets regarding cyberbullying, respectively. Section 9 presents the challenges and open issues of DL models in cyberbullying.

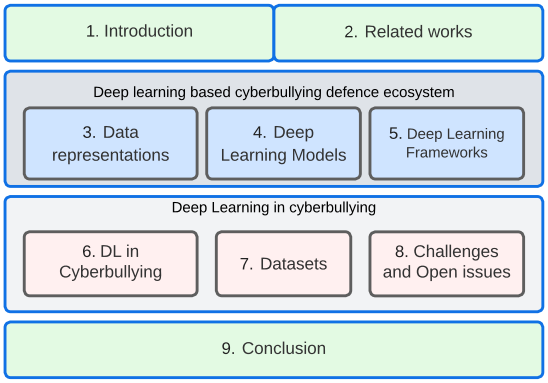

Figure 3. Organization of the paper.

2. Related Works

This section briefly discusses a few notable review papers on machine learning-based cyberbullying detection. We also present a comparison between our work with these existing works to show the novelty of our work. We have mentioned the survey papers according to the year of publication.

Haidar et al. [30] first detected cyberbullying in Arabic. They also offered a brief background on cyberbullying, related technologies, and an exhaustive survey on multilingual cyberbullying detection techniques. They finally proposed a plan to address the problem of Arabic cyberbullying.

Salawu et al. [26] presented a systematic review on cyberbullying detection approaches. They divided the existing approaches into four categories based on their substantial literature review: supervised learning, lexicon-based, rule-based, and mixed-initiative approaches. Supervised learning-based techniques commonly use classifiers such as SVM and naive Bayes to create predictive models for cyberbullying detection. Lexicon-based techniques identify cyberbullying using word lists and the presence of words within the lists. Mixed-initiative approaches combine human-based reasoning with one or more of the above-mentioned approaches to identify bullying. Rule-based approaches compare text to predetermined rules to identify bullying. The authors discovered two significant obstacles in cyberbullying detection research: the shortage of labeled datasets and academics' failure to take a holistic approach to cyberbully while creating detection systems. Their study effectively presents the current state of cyberbullying detection research with traditional ML techniques.

Rosa et al. [27] analyzed the existing research on automatic cyberbullying detection in depth. Their findings revealed that cyberbullying is frequently misinterpreted in the literature, resulting in erroneous systems with limited real-world utility. Furthermore, there is no standard methodology for evaluating these systems, and the natural imbalance of datasets continues to be an issue. They identified the future trend of research on the issue toward a position more consistent with the phenomenon's description and depiction, allowing future systems to be more practical and focused.

Al-Garadi et al. [31] studied existing publications to detect aggressive behavior using ML approaches. They summarized and recognized the critical factors for detecting cyberbullying through ML techniques, especially supervised learning. For this purpose, they have utilized accuracy, precision-recall and f-measure to determine the area under the curve function for modeling the behaviors in cyberbullying.

Elsafoury et al. [29] reveal some challenges and constraints of cyberbullying detection. Their paper represents a systematic literature review on automated cyberbullying detection that wraps all the steps in the ML pipeline. They also demonstrate that utilizing slang-based word embedding improves the detection of cyberbullying.

Kim et al. [28] give a thorough analysis of the past ten years of computational research concentrating on developing ML models for cyberbullying detection. A saturated corpus of 56 papers examined how humans are involved and considered directly or indirectly in building these detection algorithms. The authors focused on current algorithms' congruence with theories of cyberbullying. They then examined if and how current algorithms have incorporated humans. Finally, they shed insight into how academics have envisioned using current detection algorithms. Their evaluation reveals essential gaps in this research area due to the lack of human-centeredness in algorithm creation.

A comparison of automated cyberbullying detection methods, including data annotation, preprocessing, and feature engineering, is presented in the study by Al-Harigy et al. [45]. Emoji use in cyberbullying detection and the application of self-supervised learning to annotation are also covered. Due to the detrimental effects of cyberbullying, particularly on social media where anonymity can foster hate speech and cyberbullying, the paper emphasizes the need for efficient cyberbullying detection.

We have summarized the above-mentioned studies in Table 1 where the existing surveys of machine-learning-based cyberbullying detection are compared with various

features of our work. We have also compared these studies with ours according to their methodology of conducting the systematic review that is illustrated in Table 2.

Table 1. Comparison of our survey with existing surveys (addressed: ✓, not addressed: ✘, not applicable: N/A).

Reference	Deep Learning Models		Method.	Taxnom.	Data Represent. Tech.		Framewrk.	Dataset (Pub. Avail.)	Discussion in Challenges and Future Trends			
	Application in Cyberbullying	Strength and Limitation			Text	Img.			Cultural Diversity	Data Represent.	Multimedia and Multilingual Content	Impact on Mental Health
[30]	✘	✘	✘	✘	✓	✘	N/A	✘	✘	✘	✘	✘
[26]	✓	✓	✓	✘	✓	✘	✘	✓	✘	✘	✘	✓
[27]	✘	✘	✓	✘	✓	✓	N/A	✓	✘	✘	✘	✘
[31]	✘	✘	✓	✘	✓	✘	N/A	✓	✓	✓	✓	✘
[29]	✓	✘	✓	✘	✓	✘	✘	✓	✘	✓	✓	✘
[28]	✓	✘	✓	✘	✓	✓	✘	✘	✘	✘	✘	✘
[45]	✓	✘	✓	✘	✓	✓	✘	✘	✘	✘	✘	✓
Ours	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 2. Comparison of methodology with existing surveys.

Reference	Collection Sources	Keywords	Timeline	Initial Paper Count	Final Paper Count
[30]	-	-	-	-	-
[26]	Scopus, the ACM Digital Library, and the IEEE Xplore digital library	Cyberbully or cyberbullying detection, detecting cyberbully or cyberbullying, electronic or online bullying detection, detecting electronic or online bullying, cyberbullying prevention tool, cyberbullying prevention software, cyberbullying software, anti cyberbullying detecting electronic or online harassment	2008–2016	89	46
[27]	Google Scholar, Research Gate, ACM Digital Library, Arxiv, Scopus, Mendeley	-	2011–2018	71	22
[31]	Scopus, Clarivate Analytics' Web of Science, DBLP Computer Science Bibliography, ACM Digital Library, ScienceDirect, SpringerLink, and IEEE Xplore, Qatar University's digital library	Cyberbullying, aggressive behavior, big data, and cyberbullying models	-	-	-
[29]	Google Scholar, IEEE Xplore, Science Direct, ACM Digital Library and Wiley online databases	Cyberbullying detection	2008–2020	106	65
[28]	The ACM Digital Library, IEEE Xplore Digital Library, and Springer Link databases	Cyberbullying detection, Cyberbullying detection algorithm	2010–2020	118	56
[45]	Google Scholar, IEEE, Springer, ACM, and others	Abuse, offensive or hate speech, sarcasm, and irony	2012–2020	70	45
Ours	IEEE Xplore, ScienceDirect, ACM Digital Library, Wiley, Springer Link, Taylor & Francis, MDPI, etc.	Cyberbullying and deep learning, cyberbullying detection, cyberharassment and deep learning, social media and cyberbullying, deep fake and cyberbullying	2017–Jan 2023	1331	63

The limitations of the existing survey in the area of detecting cyberbullying using deep learning are shown in Table 1. To the best of our knowledge, there is no survey of deep-

learning-based cyberbullying detection in existence because the majority of survey papers in the field are outdated. Although there is very little discussion about the applicability of deep learning models for solving this problem, as shown in Table 1, the majority of the papers did not discuss the strengths and weaknesses of the models in the context of classifying cyberbullying. Since it was not the primary focus of the existing studies, the taxonomy of deep-learning-based cyberbullying classification is not covered in any existing surveys. Taxonomy helps in organizing and adding clarity to complex ideas by categorizing them into practical categories. When complex concepts are broken down into smaller, more manageable parts, it is easier to understand and communicate ideas. A thorough discussion of taxonomy is crucial for that purpose. The majority of the existing survey papers omitted discussing image-based data representation techniques, but each paper briefly discussed text-based data representation techniques. However, since it is currently necessary to detect cyberbullying from images, we discuss them in our paper. Selecting an appropriate framework from the wide choice is also crucial in order to implement the models robustly while dealing with the problem of classifying cyberbully. In contrast to the existing studies, which lack a discussion of the framework, our study explicitly states the applicability of various frameworks based on the problem. Another crucial factor is the accessibility of the datasets mentioned in the studies, without which it would be challenging for the researchers to assess the viability of their research hypothesis. We also discuss cultural diversity, data representation, multimedia and multilingual content, and the impact on mental health as part of the discussion of challenges and future trends. The majority of existing studies did not go into detail about these issues. Therefore, we include this in our study because it is essential to fully understand the difficulties and potential future trends before beginning any work.

Table 2 depicts the comparison of methodology with the existing surveys. From the keywords of each existing survey, it is clear that no existing surveys have focused on deep-learning-based cyberbullying detection, which is a necessity nowadays, as deep learning models surpass the traditional machine learning models. Additionally, the most recent year of the surveys in use is 2020, but the current year is 2023. Three years of time between survey papers is significant. Note that deep-learning-based ideas in detecting cyberbullying have emerged during this period.

3. Methodology

We are particularly interested in relevant English-language articles: reputed journals and conferences published between January 2017 and January 2023 in academic databases (e.g., IEEE Xplore, ScienceDirect, ACM Digital Library, Wiley, Springer Link, Taylor & Francis, MDPI, etc.) and patents.

Figure 4 shows that we conducted a comprehensive search for related articles on Google Scholar, using various combinations of initial keywords such as “cyberbullying” and “deep learning”, “cyberbullying” and “detection”, “cyberharassment” and “deep learning”, and “social media” and “cyberbullying”. After screening 1331 article titles, we removed duplicate content and subsequently excluded low-tier journals and conferences. In the third round, we excluded articles that did not align with our research content, and finally, we shortened our list further by excluding contributions that were deemed insignificant.

We have selected 63 relevant articles for inclusion in this paper, as they closely align with the focus of our study. We have exclusively included primary research in our review. To further enhance our search, we have conducted an additional search using keywords such as “deepfake” and “cyberbullying”, focusing on the subfields of title, abstract, and keywords, spanning the period of January 2017 to January 2023.

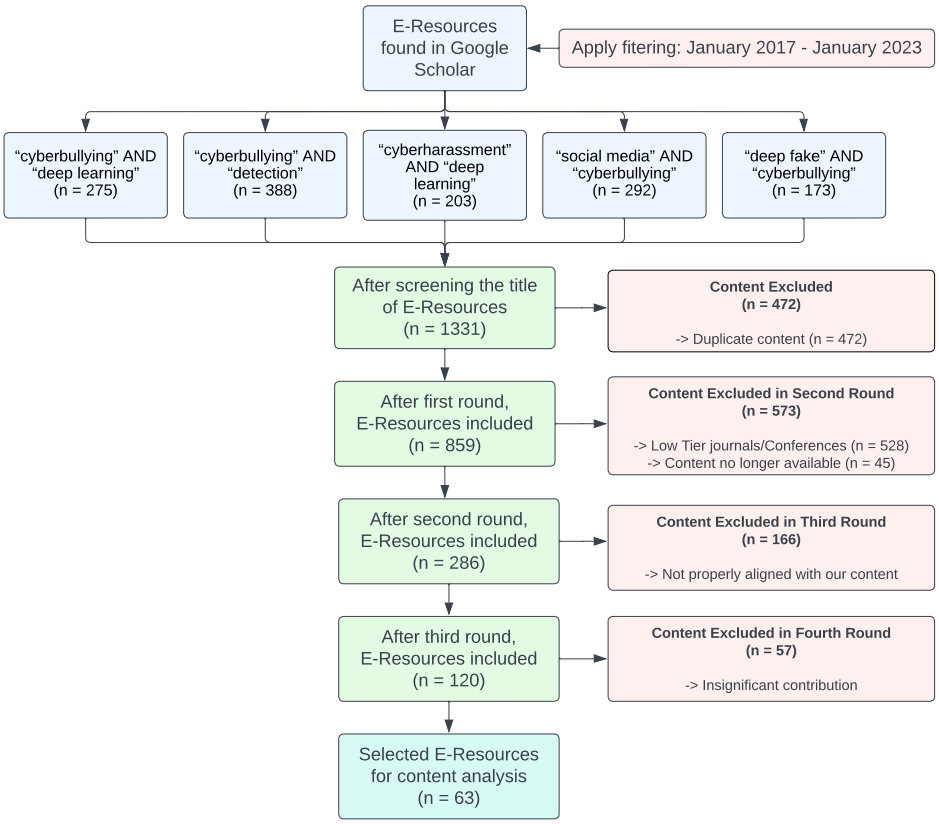

Figure 4. Online resources inclusion and exclusion process flowchart.

4. Data Representation Techniques

In many situations, we use an independent representation of words or images as input to the DL network. If these words or images are better understood by these representations, then it is expected that the predictive performance improves. Thus, exploiting a better representation technique is important since it affects the overall performance of the DL model. In this section, we mainly present major data representation techniques (i.e., text and meme) by which we experience prominent cyberbullying attacks. Note that data representation techniques are shown as the left-most branch of our taxonomy shown in Figure 2.

In the following sections, we first discuss different word-embedding techniques to represent text data: One-hot encoding, TF-IDF, Word2Vec, GloVe, ELMo, fastText and BERT.

4.1. Text Data Representation

4.1.1. One-Hot Encoding

One-hot encoding is a technique for converting categorical input (i.e., words) to integers so that ML algorithms can use it. The majority of ML algorithms cannot deal with categorical data directly. This technique transforms a categorical variable into a set of binomials, or a binary vector with a value of 0 or 1. The number of columns in this method is equal to the number of classes in the category. This approach is useful for converting data such that it may be utilized for ML. However, the approach has been criticized because it simply adds more columns. As a result, the dataset becomes massive, and the algorithm that has many columns might decrease the accuracy.

4.1.2. TF-IDF

Term Frequency Inverse Document Frequency (TF-IDF) [38] is used to determine how relevant a term is in a document, with word relevance referring to the quantity of information provided about the term's context. Term frequency (TF) is a metric that

quantifies how frequently a term appears in a document. If a term appears more frequently in a text than other terms, it is more relevant to the content than other terms. In addition, the inverse document frequency (IDF) score is calculated by dividing the total number of documents by the total number of documents in the collection that contains them. The approach aids in reducing the weight of terms that appear often across a collection of papers. Overall, TF-IDF, which is essentially the multiplication of TF and IDF scores, is used to identify the relevant needs for a text so that the most significant and informative words may be readily found. In our context, we found few cyberbullying detection works using TF-IDF [46,47].

4.1.3. Word2Vec

Word2Vec [32,33] is a method for recreating word linguistic contexts. The method has a neural network with two layers. A vast corpus of words is used as input, and the result is a vector space with hundreds of dimensions. A matching vector space is allocated to each unique word in the corpus. Word vectors in the corpus are arranged in such a way that words with similar contexts or nearly identical meanings are clustered together in the space. Word2Vec is a computationally fast approach for learning word embeddings from raw text. Word2vec uses two separate methods: the Continuous Bag-of-Words (CBOW) model and the Skip-Gram model. The architecture of these two methods has been shown in Figure 5.

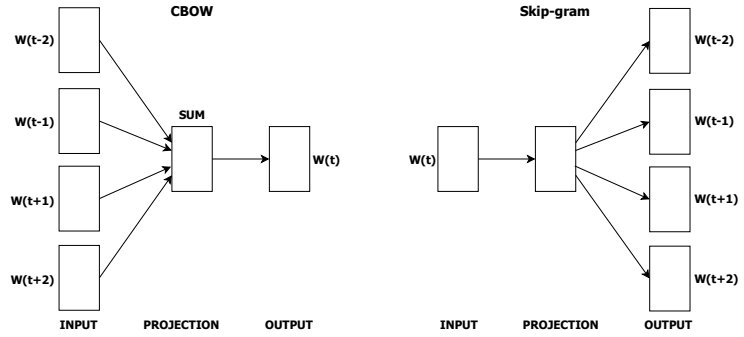

Figure 5. Word2Vec model.

4.1.4. GloVe

GloVe [34] is an unsupervised ML technique that stands for Global Vector for Word Representation. Stanford created GloVe to construct word embedding by aggregating a corpus's global word to word co-occurrence matrix. The outcome of embedding in vector space reveals intriguing linear substructures of the word.

4.1.5. ELMo

The acronym ELMo [35] stands for Embeddings from Language Model. This word-embedding approach is used to represent a series of words as a corresponding sequence of vectors. Character-level tokens are used as inputs to construct word-level embeddings in a bi-directional LSTM. ELMo is a sophisticated computer model for converting words to numbers.

4.1.6. FastText

The Facebook research team created FastText [36] as a library. It has two uses. The first is efficient word representation learning, and the second is sentence categorization. The method supports both supervised and unsupervised representations of words and sentences. On Facebook, if anyone puts a status update on their Facebook timeline about purchasing a bike, after a few moments, they may see an ad related to bikes. Facebook uses the text data to serve you better ads by using FastText. Figure 6 shows the word embedding for 3-gram in FastText.

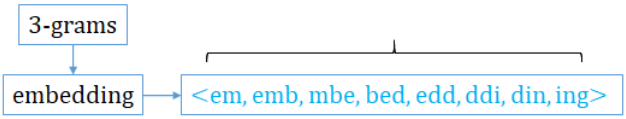

Figure 6. FastText word embedding for 3-gram.

4.1.7. BERT

Bidirectional Encoder Representation from Transformers (BERT) [37] is based on the transformer architecture. Wikipedia (2500 million words) and Book Corpus (800 million words) are part of a vast corpus of unlabeled text that has been pre-trained. The success of BERT mainly lies in the pre-trained step, which has been trained with a large number of texts. The BERT model gathers information from both the left and right sides of a sentence context. Figure 7 shows an example of bi-directionality. If we forecast the nature of a word by choosing other words to its left or right sides, by selecting both sides of this term, BERT precisely predicts the exact meaning.

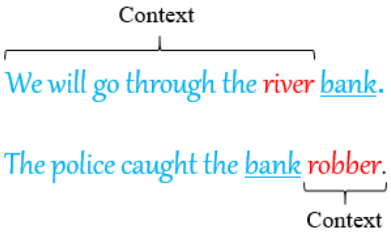

Figure 7. Capturing context by BERT of two sentences.

The transformer is the foundation of the BERT architecture. BERT has two variants: BERT base and BERT large. BERT base has 12 layers of transformer blocks, 12 attention heads, and 110 million of parameters. BERT large, on the other hand, has 24 transformer layers, 16 attention heads, and 340 million parameters. Figure 8 shows the architecture of BERT base and BERT large. Figure 9 and 10 shows the input representation of BERT model and output as the embedding of BERT base respectively.

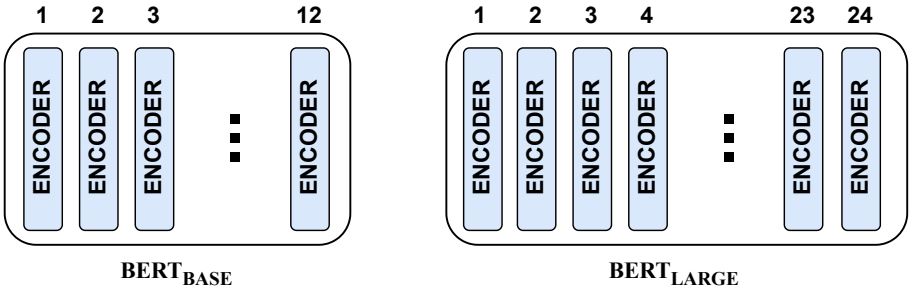

Figure 8. BERT architecture (BERT base and BERT large).

BERT has been pre-trained on two natural language challenges. The first is Masked Language Modeling (MLM), which studies word relationships. The second is Next Sentence Prediction (NSP), which is necessary to comprehend how sentences relate to one another. There are some variations of BERT that are also used for the cyberbullying detection problem.

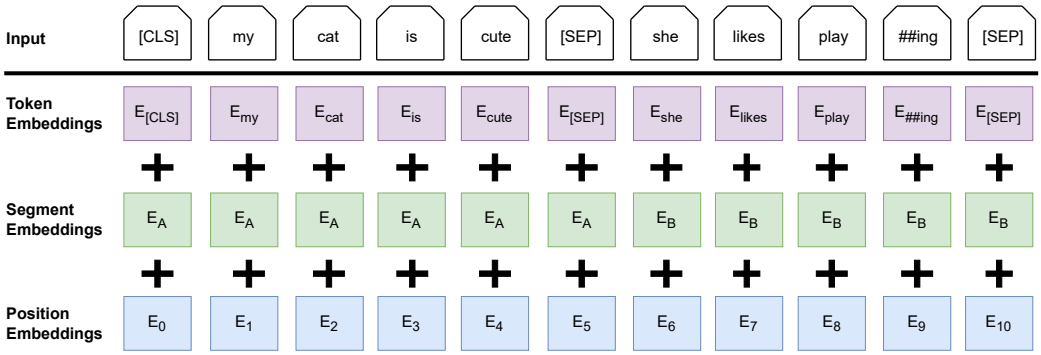

Figure 9. Input representation of BERT model. The input embeddings are the sum of the token embeddings, the segmentation embeddings and the position embeddings.

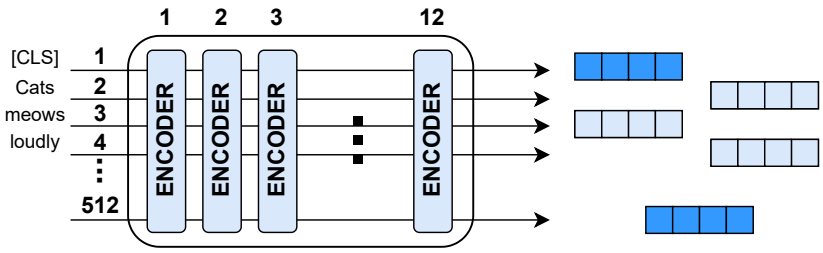

Figure 10. BERT base output as embeddings.

RoBERTa (Robustly Optimized BERT) [48]: Liu et al. found that BERT was significantly undertrained, and they proposed an improved version for training BERT models, namely RoBERTa. RoBERTa has the following modifications: (1) train the model over more data, extend the training time and consider larger batches; (2) remove the target object of the next sentence; (3) train on longer sequences; and (4) dynamically change the masking pattern over the training data. The authors used a novel dataset, CCNEWS, and suggested that if more data are used during pre-training, downstream tasks can be improved further. Yani et al. [49] utilized RoBERTa to detect cyberbullying on the popular social media platform Twitter. After experimental analysis, they obtained an accuracy score of 86.9% and an F1 score of 77.5%.

ALBERT (A Lite BERT) [50]: Improving the model performance is not always possible due to GPU/TPU memory limitations and longer training times. To mitigate the issue, the authors reduced two parameters to lower the memory consumption and to increase the training speed of BERT. A number of studies show that ALBERT presents better performance compared to BERT over GLUE, RACE, and SQuAD benchmarks. Tripathy et al. [51] used an ALBERT-based fine-tuning model for cyberbullying detection, as it does not require large amounts of data for fine-tuning. The experimental results show that their proposed method outperformed the current approaches CNN + word2Vec, CNN + GRU, and BERT implementations in terms of an F1 Score of 95%.

ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacements Accurately) [52]: BERT corrupts the input by replacing some token with MASK and by training a model to reconstruct the original model. The authors corrupted the tokens with plausible alternatives sampled from a small generator network that improves the model performance significantly.

DistilBERT (Distilled BERT) [53]: DistilBERT is a pre-trained smaller general-purpose language representation model that is a faster variant of the BERT model. The version is fine-tuned with good performance on a wide range of tasks and is designed for low-resource environments. It achieves similar performance to the original BERT model while using fewer resources. The approach leveraged knowledge distillation during the pre-training phase and reduced the size of general BERT by 40%, yet it retained a language understanding capability of 97%. The model is faster, smaller and lighter to pre-train.

Several studies [54–56] used DistilBERT for cyberbullying detection in social networks. Their experimental results show that they obtained a promising performance while using DistilBERT as the word-embedding technique as well as the fine-tuned classifier.

MobileBERT [57]: BERT suffers from heavy model sizes and high latency, which cannot be applied to limited resource devices such as mobile phones. Sun et al. developed MobileBERT for compressing and accelerating the BERT model, which is task-agnostic and can be applied to various downstream NLP tasks by simple fine-tuning. The model is carefully designed to create a balance between self-attentions and feed-forward networks. To train MobileBERT, first the authors trained a specially designed teacher model, and then, they transferred the knowledge from the teacher to MobileBERT, which is 4.3× smaller and 5.5× faster than the general BERT model.

On a variety of natural language understanding tasks, Bidirectional Encoder Representation from Transformers (BERT) has produced outstanding results. In numerous studies, BERT outperformed conventional machine learning algorithms, achieved cutting-edge performance, and demonstrated promising results in the detection of cyberbullying. Studies [22,58–60] show that BERT achieved high accuracy and F1 scores to classify cyberbullying in various types of online content, such as tweets and comments. The performance of BERT may vary depending on the dataset and task. Here are a few examples of state-of-the-art works that demonstrate the extensive recent research on using BERT for cyberbullying detection.

Using a pre-trained BERT model along with deep learning (DL) models, Mazari et al. [58] proposed a multi-aspect hate speech detection approach based on text classification in multiple labels. Bidirectional Long-Short Term Memory (Bi-LSTM) and/or Bidirectional Gated Recurrent Unit (Bi-GRU) are stacked on GloVe and FastText word embeddings to create the DL models that are used. The proposed approach, which detects hate speech on social media, received a ROC-AUC score of 98.63%. BERT was used by Coban et al. [59] to detect Turkish-language Facebook activity content. They reported BERT as the best classifier for the problem after conducting a thorough experimental analysis because it produces the most cutting-edge results when compared to other conventional machine learning and deep learning methods, with a macro F1 score of 92.8.

BERT was applied to three real-world datasets: Formspring, Twitter, and Wikipedia by Paul et al. [22]. According to experimental findings, BERT performs significantly better than conventional machine learning algorithms such as CNN, RNN + LSTM, and Bi-LSTM, with attention in terms of F1 scores. In this study, they solved the cyberbullying detection and classification problem with state-of-the-art performance on three widely used datasets. To better represent the meanings of the semantics of the words, Feng et al. [61] suggested a BHF model that makes use of BERT and a Hierarchical Attention Network (HAN). According to their experimental findings, the BERT and HAN (also known as BHF) combination provides a more precise semantic representation of each word, leading to higher accuracy scores.

A domain-specific BERT model for identifying hate speech that is posted online is being developed by Ishaq et al. [62]. The suggested method introduces "HateSpeechBERT (HSBERT)", a domain-specific language representation model based on BERT and pre-trained on substantial datasets of hate speech. They demonstrate in their study that HSBERT provides state-of-the-art results when compared to other models by comparing its performance against the general-domain BERT through extrinsic and intrinsic evaluations. To assess the effectiveness of BERT in detecting cyberbullying in a social media context, Mozafari et al. [60] used BERT to categorize cyberbullying in two social media datasets. In terms of precision, recall, and F1 scores, their experimental findings are encouraging when compared to the prior research in this area. Overall, these studies demonstrate the effectiveness of using BERT and the combination of BERT with other models for cyberbullying detection, and they highlight the potential for future research in this area.

A summary of different word-embedding techniques used in cyberbullying detection is shown in Table 3. We have compared these techniques based on context-sensitive, traditional ML-based, RNN-based, transformer-based, and transfer-learning-based models. We can observe that several word-embedding methods depend on the context. Since

cyberbullying detection is a complex task and it is largely context-dependent, context-sensitive word embedding performs better than context-insensitive word embedding. On the other hand, except for one-hot encoding and TF-IDF [38], which are a kind of mathematical calculation-based vectorization technique, the majority of other approaches are traditional ML-based. However, we find that in some studies, the authors use ELMo [35] to feed data to RNN models.

Table 3. Comparison among different data-representation techniques.

Word-Embedding Technique	Context Sensitive Embedding	ML Based	RNN Based	Transformer Based	Pretrained	Used in Cyberbullying Application
One-hot Embedding	No	No	No	No	No	YouTube Bengali text [18]
TF-IDF	No	No	No	No	No	Chinese Weibo dataset and English tweets [5], Twitter English text [63], YouTube Bengali text [18]
Word2Vec	No	Yes	No	No	Yes	Twitter Indonesian text [64], Twitter English text [2,63], Social media text [65,66]
GloVe	No	Yes	No	No	Yes	Twitter English text [2,7,67], Formspring, Twitter, and Wikipedia posts [25,68], YouTube English text [25], Social media text [65]
ELMo	Yes	Yes	Yes	No	Yes	Social media text [65], Formspring English text [69–71], MySpace English text [69,71]
fastText	No	Yes	No	No	Yes	Formspring English text [70], Social media text [72]
BERT	Yes	Yes	No	Yes	Yes	Arabic Social media text [73], Formspring, Twitter, Wikipedia English posts [22]

Currently, the state-of-the-art word embedding is BERT [37], which provides satisfactory outcomes during model building in cyberbullying-related problems. In our study, we have noticed that BERT, the only transformer-based method, is the most potential word-embedding technique used to deal with text-based cyberbullying problems. We have also observed that all the ML-based word-embedding techniques are pre-trained. BERT is also pre-trained by using unlabeled data collected from the English Wikipedia and BooksCorpus, each of which contains 800 million words. One-hot [74] and TF-IDF approaches are utilized in a few studies in the identification of cyberbullying. However, these techniques perform weaker since they tend to be context insensitive. The study mostly made use of Word2Vec, GloVe, and ELMo. However, BERT has lately seen sharp growth and outstanding outcomes in its use as a word embedding approach.

4.1.8. Efficacy of Various Embeddings for Detecting Cyberbullying

In this subsection, we will provide a concise overview of the efficacy of identifying cyberbullying behaviors. The ability of Word2Vec to grasp the semantic meaning of words is crucial in detecting cyberbullying. Aldhyani et al. [75] demonstrates a comparative analysis of different embeddings along with Word2Vec. For example, the word "ugly" can be used in cyberbullying to insult someone's physical appearance and cause emotional harm. However, it can also be used metaphorically in a harmless context, such as expressing dislike for a piece of clothing by saying "That shirt is so ugly". Word2Vec can differentiate harmful and harmless messages in cyberbullying detection. The mapping between the target word to its context word implicitly builds the relationship into the vector space of words, which can be inferred by these word vectors. GloVe uses global matrix factorization

to generate word vectors, which can be particularly useful in examining the context of words. One of the advantages of using GloVe is that it can effectively grasp the associations between words and their co-occurrence patterns, allowing for a more subtle understanding of the meaning behind the text. For example, a model can be trained to categorize a text as cyberbullying or non-cyberbullying based on the presence or absence of certain word clusters identified by GloVe.

FastText is similar to Word2Vec but uses subword information to generate embeddings. The main advantage of FastText is its capability to handle morphological variations in text, which are common in cyberbullying messages, and thus, it can make precise predictions. Below are some studies that have compared the effectiveness of various word embedding techniques.

Pericherla et al. [76] conducted a study to evaluate the performance of different word-embedding techniques, including Bag of Words (BoW), TF-IDF, word2vec, GloVe, FastText, and several language models (ALBERT, ELECTRA, GPT-2, XL-NET, and RoBERTa), in detecting cyberbullying using a Twitter dataset labeled for sexism and racism. The study found that the majority of language models achieved high F1 scores compared to traditional word embeddings such as BoW and TF-IDF, as well as semantic word embeddings such as word2vec, GloVe, and FastText.

Eronen et al. [77] investigated the efficacy of linguistically backed word embeddings in detecting cyberbullying. They trained Word2Vec Skip-Gram embeddings with encoded linguistic information, as well as using dependency structure-based contexts. Their findings suggested that lemmatization can be an effective preprocessing method for increasing detection efficacy with pre-trained word embeddings.

Alhloul et al. [78] conducted a study using lemmatization to extract the roots of each word and utilized the TF-IDF embedding technique. They used a UNICEF dataset of tweets categorized into six classes: age, ethnicity, gender, religion, type of cyberbullying, and non-cyberbullying. Their study found an accuracy of 97.10% and an F1 score of 97.12% in classifying tweets of cyberbullying.

Overall, different embedding techniques can be effective for cyberbullying detection, particularly when it is used with deep learning algorithms. However, similar to any machine learning model, its efficacy depends on the data quality and the selection of hyperparameters.

4.2. Image Data Representation

In the following subsections, we describe several techniques to represent two-dimensional image data such as cognitive image representation, BSP representation, Bio-inspired model representation, MPS representation, and Deep Neural Networks-based image representation.

4.2.1. Cognitive Image Representation

Cognitive image representation [79] is based on the notion that humans recognize images by making successive approximations with increasing resolution for specific regions of interest. Such an image format is appropriate for creating the learning models for the objects, which should be retrieved from picture databases. This method is based on the inverse spectrum pyramid (ISP) decomposition method for image representation, which is a novel way of encoding digital pictures. The picture is decomposed into successive approximations based on any type of 2D orthogonal transform (DCT, WHT, etc.). The obtained transform coefficients are used to construct the spectrum pyramid's successive tiers. This technique enables the creation of interactive systems in which the user may create numerous types of questions. Image archiving, image coding, image transmission systems, remote medical diagnostics, and patient monitoring are only a few examples of important application fields.

4.2.2. BSP Representation

Different images can be represented using a Binary Space Partitioning (BSP) [80] tree. First, using the Binary Quaternion Moment-Preserving (BQMP) thresholding approach, the entire image is binarized. Second, a dividing line is chosen to split the output image into two sections, at least one of which is reasonably homogeneous. Finally, a color is assigned to each region to reflect the portion of the input image. The element values of the representative color are computed as the mean of the red, green, and blue components of all the pixel colors in the region. Finally, these color values are stored along with the dividing line parameters and are utilized as the picture representation at the first partition level. The method is continued until no more areas can be partitioned or a set number of iterations has been achieved. As a result, at the end of the jth iteration, one has a j number of hierarchical picture representations. Figure 11 is a BSP tree representation of an image.

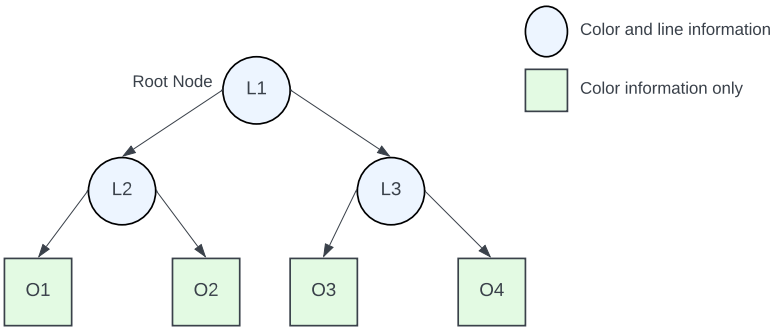

Figure 11. BSP tree representation example.

4.2.3. Bio-Inspired Model Representation

The bio-inspired model [81] is an image representation model based on a non-classical receptive field (nCRF) and reverse control mechanisms offered by biological systems for inspiration. Using a multi-layer neural network based on the human visual system, the model is utilized for image representation and image analysis. The neural model simulates a ganglion cell's non-classical receptive field and its local feedback control circuit, and it can self-adaptively and consistently depict images beyond the pixel level. Experiments on image reconstruction, distribution, and contour detection show that this technique can accurately represent images at a cheap cost while also producing a compact and abstract approximation that may be used for further image segmentation and integration. This representation schema excels at extracting spatial relationships from various image components and emphasizing foreground information. This representation schema is very effective in extracting spatial connections from various components of images and highlighting foreground items from the background, particularly in natural images with complex scenarios.

4.2.4. MPS Representation

In addition to being an efficient image coding scheme, MPS [82] provides a flexible semantics-driven image representation that enables many typical operations in visual computing and communications. The MPS is made up of edges that are retrieved and sorted from fine to coarse scales in order. MPS is a type of picture representation that is intermediate in complexity. Many popular image operations, such as classification, restoration, detection, and content-based information extraction, can be performed directly in the MPS framework without first transforming the coded image back to the spatial domain because the representation consists of high-level semantic primitives such as edges of various scales and types. It has usage in compression, scene categorization, and other areas.

4.2.5. Deep Neural Networks-based Image Representation [83]

In several computer vision applications, DNNs have demonstrated strong image representation performance. Three common building blocks for DNNs are the Restricted Boltzmann Machine (RBM), Auto-Encoder (AE), and Convolutional Neural Nets (ConvNet). Some task-specific DNN designs, such as Convolutional Deep Belief Networks (CDBN), Reconstruction Independent Component Analysis (RICA), and Deconvolutional Networks (DN), are suggested based on these building blocks. Many computer vision tasks, such as handwritten digit identification and object recognition, benefit from these approaches. The obvious conclusion drawn from this research is that successive layers of DNNs extract different characteristics at different scales, ranging from low-level features to higher-level features.

4.2.6. Optical Character Recognition (OCR)

Optical character recognition (OCR) [84] enables computers to read printed or handwritten text and to turn it into digital text that can be edited, searched, and analyzed. OCR can be used to analyze text-based content on social media sites, online forums, and messaging services in order to detect cyberbullying.

Studies [85,86] proposed a multimodel cyberbullying detection framework where they applied OCR to detect cyberbullying from image data. In addition to that, they employed another method named Image Similarity to classify cyberbullying from image data. Kumari et al. [87] utilized OCR to extract text from the images to classify cyberbullying in image data. For instance, Instagram uses OCR to find bullying in pictures and captions. The program looks for offensive language in the captions and images, and if it finds any, it notifies the user that their post may be offensive [88]. Similarly, Facebook employs OCR technology to find offensive material such as cyberbullying, hate speech, and graphic images [89]. Gao et al. [90] proposed a novel method for identifying cyberbullying on Chinese social media platforms. The system extracts text from images using a combination of OCR and image processing techniques and then uses deep learning algorithms to categorize the text as either normal or abusive. Borah [91] identifies cyberbullying on Indian social media platforms. OCR was used by the system to extract text from images, and machine learning algorithms were then used to determine whether any of the text was threatening or offensive.

OCR technology can instantly analyze text-based content, spotting behavioral patterns and identifying offensive language. Message tonality analysis and detecting sarcasm and other subtle forms of bullying can also be performed using technology. These systems can detect threatening or offensive language that might otherwise go unnoticed by conventional text-based analysis techniques by extracting text from images using OCR. These systems can also learn and adapt over time with the help of machine learning algorithms, which enhance their precision and efficiency. Social media platforms and online forums can promote a safer and more positive online environment by utilizing OCR technology for cyberbullying detection.

5. Deep-Learning-Based Models

For cyberbullying detection, many DL-based models have been applied over different applications. A few popular models are Deep Neural Network (DNN), Boltzmann machines, deep belief network, deep autoencoder, etc. Note that DL-based models are shown as the middle branch of our taxonomy shown in Figure 2. Table 4 presents high-level characteristics of different deep learning models and how these models are suitable to handle cyberbullying-related textual and image-based identification. In addition, it depicts the applications of cyberbullying for each deep-learning-based model along with its limitations. We briefly describe the popular models.

5.1. Deep Neural Network (DNN)

Deep Neural Networks (DNN) [92] are artificial neural networks with numerous hidden layers between the input and output layers. When an ML system employs multiple layers of nodes to extract high-level functions from input data, it is referred to as a Deep Neural Network. It entails translating facts into a more abstract and creative component. Similar to other neural network architectures, it has synapses, biases, neurons, functions, and weights. DNNs can represent complex non-linear connections.

As DNN is a type of ANN with multiple hidden layers so that if the model needs to learn more complex non-linear functions in that case, DNN can be used instead of ANN. DNNs are feed forward networks that transfer data from the input layer to the output layer without looping back. As a result, DNN does not perform well in the field of text classification or computer vision. Backpropagation of error is used to update weights and biases such that the latent neurons are activated at appropriate values. DNN is thought to be the key to a solution when the pattern utilized for discriminating is so complicated that standard statistical and numerical techniques fail.

Many difficulties can be developed with naively trained DNNs, just as they might with ANNs. Overfitting and computation time are two typical problems. To overcome the overfitting problem, a dropout [93] layer between the hidden layers can be used, and another approach is early stopping [94], and these are both regularization techniques. Figure 12 shows the Deep Neural network architecture.

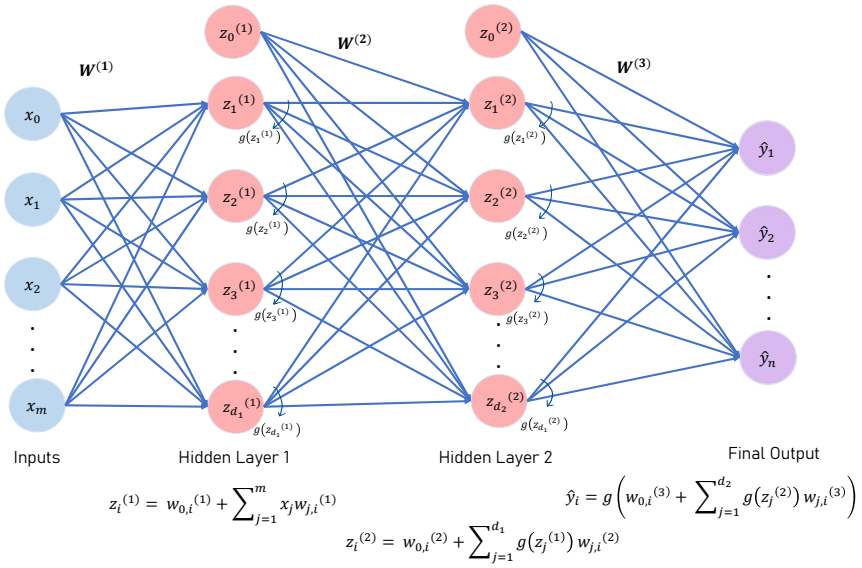

Figure 12. Deep Neural Network [95].

5.2. Boltzmann Machines (BMs)

A Boltzmann machine [39] is a symmetrically linked network of neuron-like units that make stochastic decisions on whether to turn on/off. Boltzmann machines use a basic learning technique used to uncover interesting characteristics in the training data that indicate complicated regularities.

In networks with multiple layers of feature detectors, the learning process is sluggish, but in "restricted Boltzmann machines" with a single layer of feature detectors, it works faster. By building limited Boltzmann machines and using the feature activations of one as the training data for the next, several hidden layers may be learned quickly.

There are different types of Boltzmann machines: Restricted Boltzmann machine [96], Deep Boltzmann machine [97], and Spike-and-slab RBMs [98]. The Boltzmann machine is a relatively broad computing medium in theory. For example, if the machine is trained on images, it may hypothetically model the pattern of images and use that model to finish an incomplete photograph. Figure 13 shows an example of a Boltzmann machine with two hidden units and three visible units.

Boltzmann machines are normally used to tackle diverse computational issues; for example, for an inquiry issue, the loads present on the associations can be fixed and are utilized to address the expense capacity of the improvement issue [39].

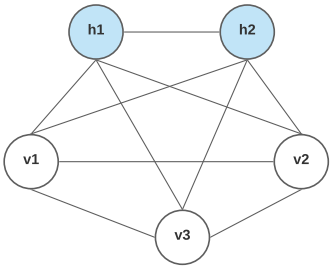

Figure 13. Boltzmann machine [99] (2 hidden units, 3 visible units).

5.3. Deep Belief Network (DBN)

Deep belief networks [40] are probabilistic generative models composed of several layers of stochastic, latent variables. Latent variables with binary values are referred to as hidden units or feature detectors. Undirected, symmetric connections link the top two layers, providing an associative memory. Directed connections are sent down from the higher layer to the lower layers. The states of the units in the lowest tier make up a data vector.

A DBN may learn to probabilistically recreate its inputs when trained on a collection of instances without supervision. Then, the layers serve as feature detectors. After completing this learning step, a DBN can be taught to perform a classification task under supervision. The procedure of training a DBN model consists of two parts. Each RBM layer is trained unsupervised, the input should be mapped into distinct feature spaces, and as much information as possible should be maintained. As a supervised classifier, the LR layer is then put on top of the DBN [100]. Figure 14 shows the architecture of a deep belief network (DBN).

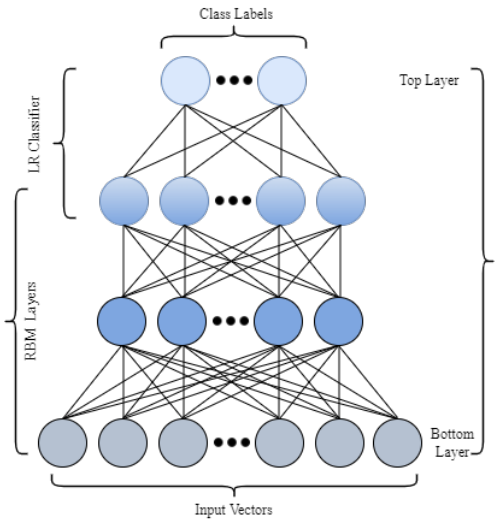

Figure 14. Architecture of a deep belief network (DBN) [101].

5.4. Deep Autoencoder (DAE)

A deep autoencoder (DAE) [41] comprises two symmetrical deep-belief networks: one with four or five shallow layers for encoding and the other with four or five layers for decoding. In image search and data compression, the deep autoencoder is commonly utilized. In the case of image compression, deep autoencoders are beneficial for semantic

hashing [102]. Topic modeling, or statistically modeling abstract subjects that are scattered over a collection of texts, is where deep autoencoders are useful.

Many autoencoders are trained using a single-layer encoder and decoder; however, utilizing multiple (deep) encoders and decoders gives several benefits. The computational cost of modeling some functions can be reduced by an order of magnitude when using depth. Depth can reduce the quantity of training data required to learn some functions tremendously [103]. Deep autoencoders produce superior compression than shallow or linear autoencoders [104].

Autoencoders are most commonly used for dimensionality reduction and information retrieval, although recent variants have been used for a variety of other tasks. Principal component analysis, dimensionality reduction, retrieval of information, detection of anomalies, processing of images, drug development, popularity forecasting, and machine translation are the major tasks where deep autoencoders are used [103]. Figure 15 shows the architecture of a deep autoencoder.

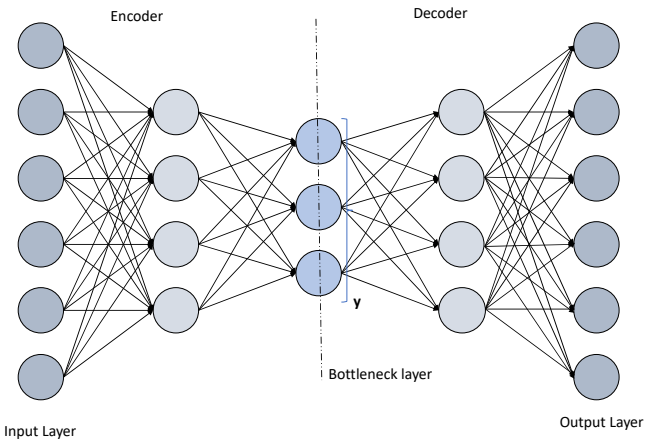

Figure 15. Deep autoencoder (DAE) [105].

5.5. Generative Adversarial Network (GAN)

Goodfellow et al. [42] proposed a model GAN that uses minimax game theory to train the generation model. GANs are a type of generative modeling that uses DL techniques.

In its training phase, GAN presents the challenges as a supervised learning problem with two sub-modals. The generator model creates new instances, but the discriminator model attempts to categorize them. It tries to figure out if the object is genuine from the domain or a forgery (generated). The two models are trained in an adversarial zero-sum game until the discriminator model is tricked roughly half of the time, indicating that the generator model is producing believable instances.

The applications of GAN is increasing rapidly in the sectors of fashion, art and advertising, science, video games, malicious applications, and transfer learning. Inverse methods such as bidirectional GAN (BiGAN) [106] and adversarial autoencoders [107] learn a mapping from a latent space to the data distribution, whereas the conventional GAN model learns a mapping from a latent space to the data distribution. Semi-supervised learning, interpretable ML, and neural machine translation are some of the applications of bidirectional models. Figure 16 shows the actual form of GAN.

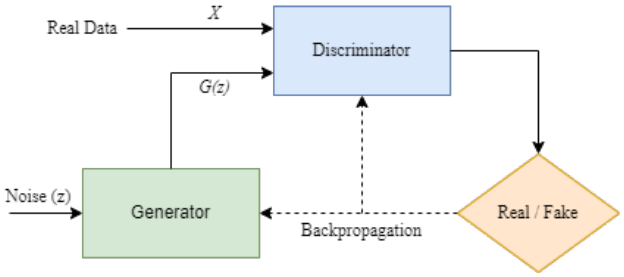

Figure 16. Architecture of a generative adversarial network (GAN) [108].

5.6. Recurrent Neural Network (RNN)

RNN [44] stands for Recurrent Neural Network, which is used for the sequential text data as input, for example, if there is a sentence and there needs to be a prediction of whether this sentence contains positive context or negative context. In such a situation, we can use RNN. Spam classifiers, time-series data, sales forecasting, stock forecasting, and many more problems can be addressed with a better accuracy by using RNN. For other models, when the input is given as a sequence of text data by using text preprocessing techniques (such as Word2Vec, TF-IDF, BagOfWord, etc.), we need to preprocess the raw data and convert them into vectors. For applying ML algorithms over the sequential data, we need to convert them into vectors. When a sentence is converted into a vector, the sequence information is discarded. Once the sequence information is discarded, the accuracy will decrease. We will also discuss text representation. Since we may analyze cyberbullying from textual content, RNN is used for controlling this sequence information. RNN has an internal memory that helps it to control the sequence information. In other neural networks, all the input is basically the vector, which is totally independent, but in RNN, every input is dependent on its previous output and current input. In this way, RNN restores the context of the whole sentence.

In Figure 17, the current state input is $h_t = f(h_{t-1}, X_t)$. Then, we have to apply activation functions such as sigmoid, ReLU, or tanh, and then, the output will be, $y_t = W_{ht}h_t$ where W_{ht} is the weight of the output. RNN has some problems:

1. The training of an RNN is very difficult.
2. It cannot process with a very long sequence of sentences.
3. RNN does not support long-term memory storage.

For solving these problems, LSTM was introduced.

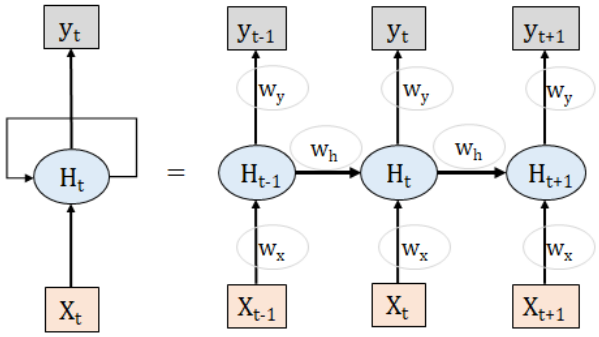

Figure 17. Recurrent Neural Network.

5.7. Long Short-Term Memory

LSTM stands for Long Short-Term Memory [109] network, which is basically the modified version of RNN. LSTM is used for remembering the past data for a long period, which is mainly possible for backpropagation during the training period. As we can see in Figure 18, three gates: Forget, Input, and Output gate, represent the LSTM network.

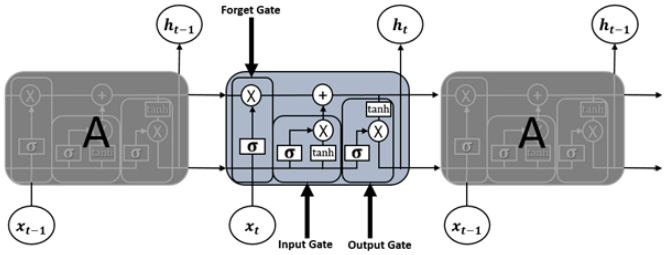

Figure 18. LSTM Cell.

Thus, the LSTM cell contains the following components:

1. Forget Gate "f";
2. Cell State "C";
3. Input Gate "i" ;
4. Output Gate "o";
5. Hidden state "h";
6. Memory state "C".

Here is the diagram for a LSTM cell at the time step t.

Here,
⊗—element wise multiplication;
⊕—element wise addition;
C_t = current cell memory;
C_{t-1} = previous cell memory;
o_t = output gate;
f_t = forget gate;
σ = sigmoid function;
w, b = weight vectors;
h_{t-1} = previous cell output;
x_t = input vector;
h_t = current cell output.

Forget Gate: From the previous hidden state, we obtain some information. The forget gate decides which information is important and which is not based on the previous state information. It basically passes current input x_t and previous state output h_{t-1} into a sigmoid function, which gives the value between 0 and 1. If the value is important, then the sigmoid output gives the value closer to 1. Then, this output is passed to the cell state and will be multiplied with previous cell state values. The equation of forget gate.

$$f_t = \sigma\left(w_{f'}[h_{t-1}, x_t] + b_f\right) \tag{1}$$

Input Gate: The current x_t and the previous h_{t-1} are passed into a sigmoid activation function, which transform the value between 0 and 1, and these values are stored into a vector. In this case, 0 indicates important and 1 indicates not important. Again, the same x_t and h_{t-1} are passed into a tanh activation function, which transforms the value between -1 and 1. A vector is created with all these possible values of the tanh function. Finally, the output of both the sigmoid function and tanh function will be multiplied and passed to the call state.

$$i_t = \sigma(w_i * [h_{t-1}, x_t] + b_i) \tag{2}$$

$$\sim C_t = \tanh(w_{c'}[h_{t-1'} x_t] + b_c) \tag{3}$$

Cell State: Now the network has both input and forget gate information, which is required in the cell state to decide and store the information from the new state. After that, the previous cell state and the output of the forget gate will be multiplied. Then, the values

are dropped if the output of the multiply is 0. Next, the result of this multiplication will perform addition with the input gates' result and will generate a new cell state.

$$C_t = f_t * C_{t-1} + i_t * \sim C_t \tag{4}$$

Output Gate: The gate finds the value of the next state, which also includes the information of the previous state's input. Here, again the current x_t and the previous h_{t-1} are passed into another sigmoid function. On the other side, the new cell state value is passed into a tanh function. Then, the output of the tanh function and the sigmoid function are multiplied and the final result is generated, which is passed into the next hidden state.

$$o_t = \sigma(w_o * [h_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = o_t * \tanh(C_t) \tag{6}$$

Other Recurrent Neural Network-based architectures are: Hopfield [110], Bi-LSTM [10], GRU [111] etc.

5.8. Convolutional Neural Network (CNN)

Convolutional Neural Networks (ConvNets or CNNs) [43] are one of the most common types of neural networks used to recognize and classify images. CNNs are commonly utilized in domains such as object detection, facial recognition, and so on. The convolution layer, pooling layer, activation layer, and fully connected layer are the major layers of CNN architecture.

There are multiple layers in CNN that process and extract features from the data. To perform the convolution operation, there are several filters in the convolution layer. To perform operations on elements, there is a ReLU layer in CNN, and the rectified feature map is the output from this layer. Then, the rectified feature map passes to the pooling layer. Pooling reduces the dimension of the feature map. Then, the pooling layer converts the two-dimensional vector space into single-dimensional vector space by flatting it. The flattened vector space then passes to the fully connected layer and then classifies the input image.

The applications of CNN are in the field of image recognition, video analysis, natural language processing, anomaly detection, drug discovery, health risk assessment and biomarkers of aging discovery, checkers game, computer go, time series forecasting, cultural heritage, and 3d datasets.

5.9. Hybrid Models (LSTM-CNN, CNN-LSTM)

The LSTM-CNN architecture for cyberbullying detection is a Deep Neural Network model that combines the advantage of both LSTM and CNN to detect cyberbullying. Processing sequential data, such as text, is where the model performs well.

The architecture consists of three main components: an embedding layer, an LSTM layer, and a CNN layer as shown in Figure 19. The embedding layer converts the input text into a vector representation. After this, dropout can be applied to prevent overfitting. Then, the main structure of this architecture is built with a Bidirectional LSTM layer followed by a CNN layer, which is an extension of traditional LSTMs that can improve model performance on sequence classification problems that allow the model to capture both local and global context information. The LSTM-CNN architecture can be trained to identify messages or posts as cyberbullying or non-cyberbullying. The model takes in a sequence of words and outputs a probability score for each class.

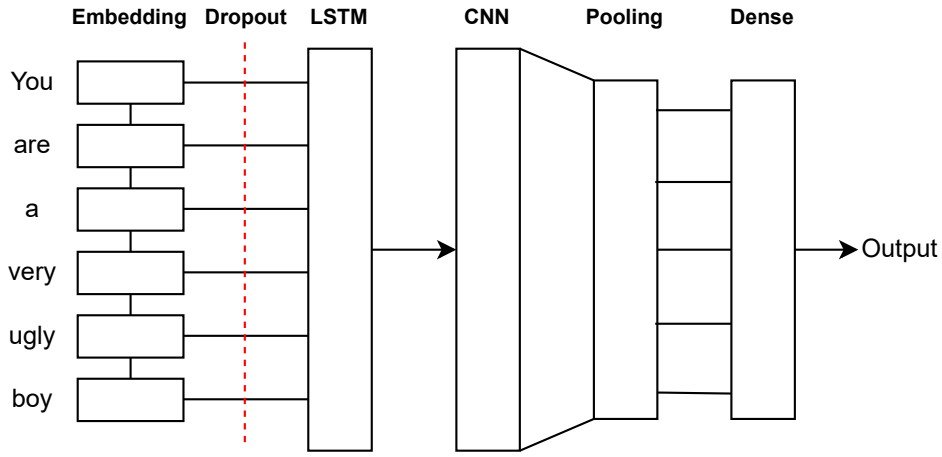

Figure 19. LSTM-CNN Model Architecture

The choice between LSTM-CNN and CNN-LSTM in the context of cyberbullying detection may rely on the type of input data. For instance, LSTM-CNN may be more appropriate if the input data are textual, such as social media postings or chat logs, because it can model the temporal dependencies in the data. On the contrary, CNN-LSTM might be a better fit if the input data are visual, such as images or videos, because it can simulate the spatial and temporal relationship in the data.

In [63], we found both CNN-LSTM and LSTM-CNN experiments, and they showed that LSTM-CNN performs better than CNN-LSTM because the CNN layer would receive the word embeddings as input, which will further be pooled to a smaller dimension, and then, the LSTM layer will use the ordering of said features to learn about the input's text ordering.

5.10. Attention-Based Model

By increasing the accuracy of automated systems, attention-based deep learning models have made a significant contribution to the field of cyberbullying detection [112]. These models can identify various data types, such as text, images, and videos, and can capture contextual information. Attention mechanisms are useful for this goal because they provide interpretability and resilience against noise, in which non-bullying content usually obscures cyberbullying behavior. Attention-based deep learning models have been successful in identifying and categorizing cyberbullying behavior on different platforms. We briefly explain some widely used attention-based deep learning models below.

5.10.1. Transformers

Transformers are deep learning models based on attention that have shown effectiveness in detecting cyberbullying [113–115]. Since its primary application is machine translation, the transformer model has been used for various natural language processing tasks, including the detection of cyberbullying. Transformers are made to process text sequences by utilizing mechanisms for self-attention to record the connections between every element in the input sequence, enabling them to track long-range dependencies in the data. Therefore, transformer models are suitable for detecting cyberbullying in social media content, which frequently contains long and intricate messages. Researchers obtained state-of-the-art performance in detecting and preventing cyberbullying behavior on social media content by using fine-tuned pre-trained transformer models.

5.10.2. BERT (Bidirectional Encoder Representations from Transformers)

BERT (Bidirectional Encoder Representations from Transformers) has demonstrated outstanding results in detecting cyberbullying since it is an extensive language model that can extract contextual information from a given text. The transformer model can capture intricate associations between words and contexts. BERT can accurately detect bullying behavior in social media messages when the cue is subtle or indirect [22,116,117].

5.10.3. Hierarchical Attention Networks (HAN)

Hierarchical Attention Networks (HAN) have demonstrated promising results at both the document and sentence levels, especially in detecting cyberbullying. HANs are neural networks that concentrate on significant portions of the input text by using an attention mechanism that enables them to collect the most relevant features for classification. HANs have been used to understand the common tone of a social media message and the presence of certain bullying behaviors at the sentence level [61,118,119].

5.10.4. Convolutional Neural Networks with Attention (CNN-Att)

Convolutional Neural Networks with Attention (CNN-Att) have also presented exciting results in the detection of cyberbullying. To extract the most significant features, CNN accomplishes the tasks from the input text. These features are then carefully weighted by an attention mechanism to understand how relevant they are to the classification task. By utilizing the patterns and textual structures in social media messages, CNNs with Attention have been used to detect cyberbullying in social media messages [78].

5.10.5. Long Short-Term Memory Networks with Attention (LSTM-Att)

Long Short-Term Memory Networks with Attention (LSTM-Att): By fusing the capacity of LSTMs to capture long-term dependencies in sequential data with the interpretability of attention mechanisms, the model has demonstrated promising results in the detection of cyberbullying. Recurrent neural networks of LSTM variants can deal with sequential data of different lengths, making them suitable for modeling text data. By incorporating attention mechanisms with LSTMs, the model can focus on the most crucial portions of the input text, improving its ability to understand and categorize social media bullies. Therefore, LSTM-Att might be a powerful tool for enhancing the precision of automated systems to detect cyberbullies [120]. Attention-Based Bi-LSTM (AB-LSTM) is also an effective neural network model for detecting cyberbullying on social media sites such as Twitter [115,121,122].

5.10.6. Gated Recurrent Units with Attention (GRU-Att)

The gated Recurrent Units with Attention (GRU-Att) model has also demonstrated exciting results in detecting bullying behavior in a social media text. The model can capture long-term dependencies in sequential data faster than the LSTM model while capturing the important part of the input text by combining Gated Recurrent Units (GRUs) with attention mechanisms. This makes it possible for the model to more accurately interpret and categorize the behavior of cyberbullying on social media platforms [71,123,124].

Attention-based deep learning models have demonstrated promising outcomes in the detection and prevention of cyberbullying behavior on social media platforms. These models, including the transformer, GRU with Attention, BERT, HAN, and CNN with Attention, have outperformed conventional machine learning methods and are capable of capturing complex relationships in text data. However, the quality and quantity of the training data, the selection of the hyperparameters, and the unique characteristics and design decisions of the model are all important factors that affect how well these models perform. As a result, even though attention-based deep learning models offer a promising method for identifying cyberbullying, careful assessment and validation of the models are required before applying them to real-world situations to ensure their efficacy, dependability, and moral implications.

As shown in Table 4, LSTM, Bi-LSTM, and CNN are frequently used models for the identification of cyberbullying. The LSTM and CNN models have recently been used in a variety of natural language processing (NLP) applications, because these models produce better results. Convolutional layers as well as maximum pooling or max-overtime pooling layers are used in CNN models to extract higher-level features. CNNs may be trained to extract character-level embeddings and n-grams, which are crucial for finding instances of cyberbullying in text. A CNN is an effective technique for detecting cyberbullying because its filters may be used to identify various patterns and elements in the text at various levels of abstraction, but CNN has some limitations such as capturing long-term dependencies, which is challenging, and requiring fixed size input, and it is significantly slower due to high operations. On the other hand, long-term dependencies between word sequences can be captured by LSTM models, which is a vital requirement in the context of cyberbullying detection [125]. Long-term dependencies can be captured by LSTM in text data, which is crucial for spotting abusive language or behavior patterns over time. Additionally, it can process variable-length input sequences, which is helpful for handling text data with a range of durations, such as comments or posts on social media. GRU is another RNN-based model that is employed to detect cyberbullying problems. In terms of time and space complexity, GRU is more effective than LSTM [111], although LSTM can produce more accurate results while working with datasets that contain longer sequences. As the focus point is cyberbullying detection and the texts are generally too long, GRU is not frequently used for this purpose.

A Deep Belief Network (DBN) is rarely used for cyberbullying detection. In the majority of cases, the network is used as one of the components of hybrid models [126]. DBN is an unsupervised learning method, as opposed to perceptron and backpropagation neural networks. The noise in the input data can be reduced by using autoencoders, which greatly increases the effectiveness of deep learning models. In addition, autoencoders are frequently employed to address the issues with unsupervised learning and to spot anomalies. However, the drawbacks of autoencoders are what make them ineffective for the goal of cyberbullying detection. The limitations of autoencoders include imperfect decoding, misinterpreting important variables, and using too much lossy compression [127].

The application of attention models in cyberbullying detection tasks helps the models perform better by helping them to concentrate on the most important sections of the input text. When identifying delicate or nuanced instances of cyberbullying, attention mechanisms can assist the model to recognize the keywords or phrases in the text. The issue of vanishing gradients, which can prevent recurrent neural networks such as LSTMs and GRUs from accurately capturing long-term relationships in the text, is another issue that attention models can assist in solving. Attention models can aid in reducing this problem and in enhancing the model's overall performance by enabling the model to selectively attend to certain areas of the input text.

The remaining models are not typically employed in cyberbullying detection tasks. Rather, they are primarily utilized in the development of hybrid models that serve to enhance overall model performance in the realm of cyberbullying detection. In a nutshell, CNN and RNN-based models (i.e., LSTM, Bi-LSTM) outperform the other deep learning models (i.e., GRU, DBN, MLPs, BMs, etc.) in the context of cyberbullying detection. This is why these models are widely used to perform the detection of cyberbullying.

Table 4. DL models with applications in cyberbullying detection along with their strengths and weaknesses.

DL Models	Used in Cyberbullying Applications	Area of Applications	Limitations
Deep Neural Network (DNN) [92]	Chats and Tweets [14], Social networks' text and image [128]	Speech Recognition, Image recognition and the natural language processing	Requires large amount of data, expensive to train, and issues of overfitting
Boltzmann Machines (BMs) [39]	Offline content [129], Image content [130], Arabic content [74]	Emotion recognition from thermal images, estimation of music similarity, extracting the structure of explored data	Training is challenging, and weight adjustment is hard
Deep Belief Networks (DBN) [40]	Arabic content [74], Social media text [131], Social media image [132]	Image classification, natural language understanding, speech recognition to audio classification	Expensive to train because of the complex data models, huge data is required, and needs classifiers to grasp the output
Deep Autoencoder (DAE) [41]	Chats and Tweets [14], Social media content [73]	Image search and data compression, dimensionality reduction, image denoising	The bottleneck layer is too narrow, lossy, and requires large amount of data
Generative Adversarial Networks (GAN) [42]	Web-application for detecting cyberbullying [133]	Improve astronomical images, gravitational lens simulation for dark matter exploration, excellent low resolution, generate realistic images and cartoon characters	Non-convergence, mode collapse, and diminished gradient
Recurrent Neural Networks (RNN) [44]	Social Commentary [21], Cyberbert: Bert for cyberbullying identification [22], Identification and classification from social media [134]	Image captioning, time-series analysis, natural language processing, handwriting recognition,and machine translation	The gradation disappears and the problem explodes, difficult to train, and unable to handle very long sequences when tanh or ReLU is used as the activation function
Long Short-Term Memory (LSTM) [109]	Social media content [7,21,68], Wikipedia, Twitter, Formspring and YouTube [25], CyberBERT [22], Bangla text [18], Indonesian language [64], Twitter [2,63]	Time-series prediction, speech recognition, music composition, and pharmaceutical development	Training takes time, training requires more memory, easy to overfit, and Dropouts are much more difficult to implement in LSTMs
Bidirectional LSTM (Bi-LSTM)	Social media content [6,7,21,68,134], Visual contents [6], Wikipedia, Twitter, Formspring and YouTube [25], CyberBERT [22], Bangla text [18], Indonesian language [64], Text and emoji data [135], Facebook [136], Twitter[2,136]	Text classification, speech recognition, and forecasting models	Costly as double LSTM cells are used, takes longer to train, and easy to overfit
Convolutional Neural Networks (CNN) [43]	Social media content [5–7,21,68,115,134], Visual contents [6], Twitter [2,14,25,63,67,136,137], Formspring.me [25,137], Facebook [136], Chats [14], YouTube and Wikipedia [25]	Image processing, and object detection	Significantly slower due to an operation such as maxpooling, large datasets are required to process, and train neural networks [138]
Radial Basis Function Networks (RBFNs) [139]	Youtube content [140], Formspring.me, MySpace, and YouTube content [141]	Classification, regression and time-series prediction	Classification is slow because every node in the hidden layer needs to compute the RBF function
Multilayer Perceptrons (MLPs)	Text and emoji data [135]	Speech recognition, image-recognition, and machine translation	As it is fully connected, there are too many parameters, each node is connected to another node in a very dense network, which creates redundancy and inefficiency
Self-Organizing Maps (SOMs) [142]	Social media content [143]	Data visualization for high dimensional data	Requires sufficient neuron weight to cluster inputs [144]

Table 4. *Cont.*

DL Models	Used in Cyberbullying Applications	Area of Applications	Limitations
Restricted Boltzmann Machines (RBMs) [96]	Turkish social media contents [145], Arabic content [74]	Dimensionality reduction, classification, regression, feature learning, topic modeling, and collaborative filtering	Training is more difficult because it is difficult to calculate the energy gradient function, the CD-k algorithm used in RBM is not as well known as the backpropagation algorithm, weight adjustment
Gated Recurrent Units (GRU) [146]	Social Commentary [21], Facebook and Twitter aggressive speech [115], Bangla text [18], Formspring.me, MySpace and YouTube content [135]	Sequence learning, Solved Vanishing–Exploding gradients problem	Slow convergence and low learning efficiency
Attention-based model [147]	Twitter bullied text identification [78], social media text analysis [112], online textual harassment detection [71], contextual textual bullies [148], Instagram bullied text identification [118], Abusive Bangla Comment detection [121], Trait-based bullying detection [114]	The method provides a simple and efficient architecture with a fixed length vector to pay attention of a sentence's high-level meaning	The model requires more weight parameters, which results in a longer training time

5.11. Performance Comparison of DL Models in Cyberbullying Detection

It is important to investigate the performance of a deep learning model for a classification problem such as cyberbullying detection. Training accuracy, validation accuracy, learning curves, and early stopping are crucial metrics that can be used to assess the model during the training and testing phases. Training loss and validation loss are also important metrics to measure the performance of a deep learning model. When we train our model, we usually evaluate the performance of a deep learning model using a test dataset.

Four fundamental concepts are utilized to assess the performance of a model of classification task: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). In a binary classification problem, TP refers to the cases where the model correctly identifies a positive instance, TN refers to the cases where the model correctly identifies a negative instance, FP refers to the cases where the model incorrectly identifies a negative instance as positive, and FN refers to the cases where the model incorrectly identifies a positive instance as negative.

A TP would be a circumstance in which a model correctly recognizes a piece of content as cyberbullying in the case of cyberbullying classification, while a TN would be a circumstance in which a model correctly identifies non-cyberbullying content. On the other hand, an FP would be a situation where the model incorrectly identifies non-cyberbullying content as cyberbullying, while an FN would be a situation where the model incorrectly identifies cyberbullying content as non-cyberbullying.

Based on the four fundamental concepts (TP, TN, FP, FN), accuracy, precision, recall, F1 score, MCC, and area under the receiver operating characteristic (AUC-ROC) curve are frequent performance analysis evaluation metrics for the classification task.

Accuracy: This metric counts the percentage of all predictions made by the model that came true, both positively and negatively. A higher accuracy means that more instances of cyberbullying have been correctly classified by the model.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN}$$

Precision: The precision metric calculates the ratio of true positives to true positives plus false positives. A higher precision means the model is more accurate at classifying instances of cyberbullying and has fewer false positives.

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall: This metric is the ratio of true positives to true positives plus false negatives. A higher recall means the model is more accurate at spotting instances of cyberbullying and has fewer false negatives.

$$\text{Recall} = \frac{TP}{TP + FN}$$

F1 score: This metric, which gives an overall assessment of the performance of the model, is the harmonic mean of precision and recall. A higher F1 score means that the model is more accurate at classifying instances of cyberbullying and has balanced precision and recall.

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

Matthews Correlation Coefficient (MCC): The metric is a balanced metric that penalizes false positives and false negatives while accounting for both true positives and true negatives. Model performance is better when the MCC is higher.

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

Area Under the Receiver Operating Characteristic Curve (AUC-ROC): This metric measures the trade-off between the true positive rate and the false positive rate and is a graphical representation of the performance of the model. Better model performance is indicated by a higher AUC-ROC.

Several deep-learning-based cyberbullying detection methods have been conducted by using individual models and presenting the performance of the models with popular evaluation metrics [21,134,149,150]. Raj et al. [149] compared different deep learning and traditional machine learning approaches for cyberbullying classification tasks. They utilized LSTM, Bi-LSTM, GRU, and Bi-GRU for the cyberbullying classification. They employed the Wikipedia Attack Dataset and Wikipedia Web Toxicity Dataset for this purpose. The experimental results show that Bi-LSTM and Bi-GRU outperform other deep-learning-based models in this context on accuracy and F1 score. They obtained the best accuracy and F1 scores of 96.98% and 98.56%, respectively, by using Bi-GRU on the Wikipedia Attack Dataset. On the other hand, they achieved the best accuracy and F1 scores of 96.5% and 98.69%, respectively, using Bi-LSTM over the Wikipedia Web Toxicity Dataset. Bharti et al. [150] applied different deep learning models over the Twitter dataset and found that Bi-LSTM outperformed other deep learning models with accuracy, precision, and F1 scores of 92.60%, 96.60%, and 94.20%, respectively. Iwendi et al. [21] used Bi-LSTM, GRU, LSTM, and RNN models for cyberbullying detection over the DISCo Kaggle dataset where Bi-LSTM outperformed other models with an accuracy score of 82.18%. Agarwal et al. [134] utilized Bi-LSTM with attention layers over the Wikipedia dataset and compared their study with some existing works. Experimental results show that their proposed method outperformed the existing works in terms of precision, recall, and F1 score of 89%, 86%, and 88%, respectively.

In the literature, there are several studies that also present the performance evaluation of different hybrid deep-learning-based approaches [66,115,151,152]. Alotaibi et al. [115] proposed a multichannel deep learning framework where they proposed a combination of transformer block, Bi-GRU, and CNN to detect cyberbullying in Twitter comments where the model outperforms the individual model with an accuracy score of 88%. Bu et al. [66] applied a combined CNN and LRCN model to detect cyberbullying from SNS comments

where the model obtained an AUC-ROC and accuracy scores of 88.54% and 87.22%, respectively. Murshed et al. [151] proposed a hybrid method named DEA-RNN that combines Elman-type Recurrent Neural Networks (RNN) with an optimized Dolphin Echolocation Algorithm (DEA) that optimizes the training time to detect cyberbullying on Twitter. They also applied Bi-LSTM and RNN models for the performance comparison, and the experimental results show that their proposed method outperformed the models in terms of accuracy, precision, recall, and F1 score of 90.45%, 89.52%, 88.98%, and 89.25%, respectively. Similarly, Raj et al. [152] combined CNN and Bi-LSTM to classify cyberbullying in real-time posts on Twitter. For the experimental analysis, they employed two other combinations of deep learning models: CNN + Bi-GRU, and Bi-LSTM + Bi-GRU. Experimental results show that their proposed method outperformed the other two combinations in terms of an accuracy score of 95%. Beniwal et al. [153] proposed a hybrid model that combines CNN and Bi-GRU to detect cyberbullying from the Kaggle Toxic comment classification dataset. The proposed model obtained the best accuracy and F1 scores of 98.39% and 79.91%, respectively. Table 5 shows the performance comparison of deep-learning-based cyberbullying-detection systems on different datasets where it is classified whether the model the study used is a hybrid model or not. In addition, the best-performing model in that experiment is reported along with the scores of performance matrices.

Table 5. Performance comparison of deep learning models on different datasets

Study	Dataset	Hybrid Model	Experimental Models	Best Performing Model	Performance Metrics
Raj et al. [149]	Wikipedia Attack Dataset	No	LSTM, Bi-LSTM, GRU, Bi-GRU	Bi-GRU	Accuracy: 96.98%, F1 Score: 98.56%
Raj et al. [149]	Wikipedia Web Toxicity Dataset	No	LSTM, Bi-LSTM, GRU, Bi-GRU	Bi-LSTM	Accuracy: 96.5%, F1 Score: 98.69%
Bharti et al. [150]	Tweets	No	Bi-LSTM	Bi-LSTM	Accuracy: 92.60%, Precision: 96.60%, F1 Score: 94.20%
Iwendi et al. [21]	DISCo dataset	No	Bi-LSTM, GRU, LSTM, RNN	Bi-LSTM	Accuracy: 82.18%
Agarwal et al. [134]	Wikipedia dataset	No	Bi-LSTM with attention layers	Bi-LSTM with attention layers	Precision: 89%, Recall: 86%, F1 Score: 88%
Singh et al. [154]	Twitter dataset	No	LSTM, GRU, traditional ML algorithms	GRU	F1 Score: 92%
Alotaibi et al. [115]	Twitter comments	Yes	Transformer block, Bi-GRU, CNN	Proposed model	Accuracy: 88%
Bu et al. [66]	SNS comments	Yes	CNN, LRCN	Proposed model	AUC-ROC score: 88.54%, Accuracy: 87.22%
Murshed et al. [151]	Twitter dataset	Yes	Bi-LSTM, RNN, DEA-RNN (proposed model)	DEA-RNN	Accuracy: 90.45%, Precision: 89.52%, Recall: 88.98%, F1 Score: 89.25%
Raj et al. [152]	Real-time posts on Twitter	Yes	CNN + Bi-GRU, Bi-LSTM + Bi-GRU, CNN + Bi-LSTM (proposed model)	Proposed model	Accuracy: 95%
Beniwal et al. [153]	Toxic Comment Classification Challenge	Yes	CNN + Bi-GRU	Proposed model	Accuracy: 98.39%, F1 Score: 79.91%

According to the most recent studies in the area of cyberbullying detection, hybrid deep learning models show more promising results than individual deep learning models.

This decision is supported by the growing understanding that classifying cyberbullying is a challenging task that calls for a combination of methods and approaches in order to produce accurate and trustworthy results. In independent models, the methods may have some limitations, which are complemented when we use hybrid models. These models present exciting results in terms of enhancing the overall effectiveness and stability of cyberbullying classification systems.

6. DL in Cyberbullying Detection

Several studies [21,64,115,134] have been conducted on the automatic identification of cyberbullying by using different independent ML techniques. A few studies [21,134] exploit RNN-based techniques, i.e., LSTM, BiLSTM, RNN, etc., while some other studies [5,137] use CNN-based techniques (i.e., CNN, PCNN, Char-CNNS, etc.) to detect cyberbullying from different sources. However, we also observe that some authors [63] perform integration of RNN-CNN-based techniques. In this section, we briefly discuss different applications of DL models in cyberbullying detection. In Table 6, we have organized the papers based on three main themes: improvements of DL models, optimization of model performance, and improving data capabilities. Furthermore, we have included a comprehensive group-wise analysis of the significant contributions made by these papers, as well as their potential impact on future research directions.

Iwendi et al. [21] applied three DL models: Bi-LSTM, LSTM, and RNN, to investigate the performance of DL algorithms in identifying bullying (i.e., insults) in social media. They discovered that Bi-LSTM outperforms other models in terms of accuracy and F1 scores after extensive testing. They also asserted that DL is the most effective method for detecting cyberbullying and related cyber challenges. Anindyati et al. [64] constructed a DL-based model employing three common text classification algorithms: LSTM, Bi-LSTM, and CNN, to detect bullying on Twitter in Indonesia.

Marwa et al. [2] applied a DL technique on a large human-labeled dataset to categorize cyberbullying. LSTM, Bi-LSTM, and CNN were the DL models employed in their tests compared to other algorithms. Agarwal et al. [134] developed an RNN-based technique to identify and categorize cyberbullying posts. To decrease data imbalance and remove ambiguities in classification, they employed a Tomek Link approach to accomplish under-sampling. Their classification model was Max-Pooling combined with a Bi-LSTM network and an attention layer. To test their model, they utilized Wikipedia datasets.

Alotaibi et al. [115] offered automation to identify violent cyberbullying act. The approach uses multichannel DL-based on BiGRU, transformer blocks, and CNN models to determine whether a Twitter comment is hostile. They also integrated three well-known hate speech datasets to assess the model performance. Luo et al. [135] presented a BiGRU-CNN sentiment classification model for cyberbullying identification. The BiGRU layer, attention mechanism layer, CNN layer, fully connected layer, and classification layer are different parts of the model. They trained and tested their proposed model using the Kaggle text dataset and the emoji dataset scraped from social networks, which outperforms traditional algorithms. Lu et al. [5] presented the Char-CNNS (Character-level Convolutional Neural Network with Shortcuts) model for detecting cyberbullying in social media discourse. Since the content available on social media is short, noisy, and unstructured with wrong spellings and symbols, they chose the character as the smallest unit of learning to overcome spelling mistakes and purposeful obfuscation. They conducted experiments over the Chinese Weibo dataset and the English Tweet dataset. Results of their experiment show that outstanding performance on the cyberbullying detection task is competitive with the state-of-the-art approaches. Zhang et al. [137] proposed a new pronunciation-based convolutional neural network (PCNN) to handle the difficulty of noise and distortion in social media postings and messages in detecting cyberbullying. They also used three strategies in their model to solve the problem of class imbalance: threshold-moving, cost function modifying, and a hybrid solution. Ahmed et al. [18] built a model to identify cyberbullying in Bangla and Romanized Bangla writings by using ML and DL methods.

In their experiment, they discovered that for one dataset, CNN, a DL algorithm outperforms other ML and DL models, whereas ML models outperform DL models for the other two datasets.

Buan et al. [7] introduced a neural network design for cyberbullying detection that is based on an existing design, which combines convolution layers with LSTM layers. They also introduced a novel activation mechanism known as SVM-like activation, which is accomplished by using L2 weight regularization. They evaluated their suggested model using the bullying traces dataset to classify the challenge between open aggressiveness, covert aggression, and non-aggression in social media writings. Gada et al. [63] suggested an LSTM-CNN model for text-based cyberbullying detection that captures sentence semantics. In addition, they developed a web application for their suggested paradigm. Bu et al. [66] suggested an approach that combines two DL models, one of which is a character-level CNN and the other a word-level LRCN. The first model extracts low-level syntactic information from a character sequence. It is also noise-resistant. The second model, which works in tandem with the CNN model, gathers high-level semantic information from a series of words. They also demonstrated that their suggested ensemble technique outperforms the state-of-the-art algorithms for detecting cyberbullying in comments of social networking sites.

Agrawal et al. [68] developed models by implementing DL models. They used three real-world datasets: Formspring (https://spring.me/ accessed on 18 April 2023), Twitter (https://github.com/zeeraktalat/hatespeech/ accessed on 18 April 2023), and Wikipedia (https://figshare.com/articles/dataset/Wikipedia_Talk_Corpus/4264973 accessed on 18 April 2023), to conduct comprehensive tests. The study gives some interesting insights on the detection of cyberbullying, such as that swear words are not sufficient for detecting cyberbullying. Powerful models that are used for detecting cyberbullying are not expected to depend on such handcrafted features. Dadvar et al. [25] found that DL-based models outperform traditional ML models. They used Wikipedia, Twitter, and Formspring datasets. Al-Ajlan et al. [67] suggested optimized Twitter cyberbullying detection based on DL (OCDD), a unique technique to the cyberbullying detection. To preserve the meaning of the words, their suggested approach encodes a tweet as a set of word vectors rather than collecting characteristics from tweets and feeding them into a classifier. For the classification phase, they employed DL, and for parameter tuning, they used a metaheuristic optimization approach.

Golem et al. [136] offered classical ML, DL, and a mixture of both approaches. To test their algorithms, they used data from Twitter and Facebook (https://sites.google.com/view/trac1/shared-task accessed on 18 April 2023). They ensembled classic ML with DL algorithms by using a voting mechanism. Yadav et al. [73] suggested a unique strategy to identify cyberbullying on social media platforms that improve on current findings by combining a pre-trained BERT model with a single linear neural network layer as a classifier. Their algorithm trains and tests on two manually labeled social media datasets: Formspring (a Q&A forum) and Wikipedia, using a consolidated DL approach. Paul et al. [22] demonstrates a unique use of BERT for detecting cyberbullying. They claim that a simple classification model based on BERT can obtain state-of-the-art results in the three real-world corpora of Formspring, Twitter, and Wikipedia. They discovered that their model outperforms prior studies when compared to slot-gated or attention-based Deep Neural Network models. Paul et al. [6] suggested a DL-based early identification framework that predicts whether a post is classified as bully/nonbully, and they analyzed data for each of the modalities (both separately and fusion-based). Furthermore, the frameworks perform outstandingly.

A DL algorithm needs to understand the pattern from the data, such that it requires a huge amount of data. The performance of the DL model improves when it has a huge amount of training data; otherwise, it does not perform that well. Another reason is that DL models can learn more complex, non-linear functions. It reduces the hassle of feature engineering, as it is performed by the DL algorithm itself. DL models perform well when it

comes to complicated problems such as natural language processing, speech recognition, and image classification.

Table 6. Major contributions and prospective future works of cyberbullying detection.

References	Theme	Major Contributions	Future Research Directions
[7,64,137]	Improvement of DL models	These studies show improvement of cyberbullying detection by using CNN, LSTM, and BiGRUA-CNN models. These models show enhancement of the classification problem by adjusting activation function, weight regularization, and dropout configuration.	• Investigating the stacked ConvLSTM with SVM activation method, if it can perform faster than the SoftMax activation for cyberbullying detection [7]. • Adding some customized layers with the CNN model to enhance the effectiveness of the conversion and features [137]. • Using an SVM classifier to replace Softmax lost function for optimized prediction score [64].
[5,115,135]	Performance optimization of the models	Studies applied char-CNN, BiGRU, and transformer models. They largely optimize weights, number of layers, combination of models during cyberbullying detection in social media discourse.	• Increasing the number of layers with different weights to improve the performance. [5] • Adding many channels with the current DL model and exploring combination of models. It can help to optimize the weights and other parameters of deep and large neural networks. [115] • Improving the stability of the model by optimizing the overfitting problem through methods such as dropout and regularization. [135]
[18,63,134]	Improving data capability	LSTM-CNN and RNN-based models have been applied in text, randomized and wikipedia datasets. The authors proposed several techniques to improve the capacity of the dataset.	• Adding more semantically meaningful classes to the datasets to improve the accuracy [18]. • Extending the work to various social media platforms such as Instagram, Reddit, Facebook, etc [63]. • Besides image classification, as Not Safe For Work (NSFW) content, adding video and audio classification [63] • Implementing cyberbullying detection methods in various languages such as French, Spanish, Russian, etc. [63]. • Exploring attention mechanisms for imbalanced classification [134].

7. Deep Learning Frameworks

DL frameworks provide a high-level programming interface for building blocks of designing, training and validating Deep Neural Networks. Note that DL frameworks are shown as the right-most branch of our taxonomy shown in Figure 2. In Table 7, we briefly explain 13 different DL frameworks, their strengths, and their limitations. In addition, we have added the supported DL algorithms of these frameworks. Moreover, we specify the usage of these frameworks in the classification of cyberbullying.

Table 7. Strengths, limitations, suitability of DL algorithms, and application in cyberbullying of DL frameworks.

Frameworks	Strengths	Limitations	Supported DL Algorithms	Used in Cyberbullying
TensorFlow	• Provides one of the best numeric libraries for dataflow programming for DL research and development. • Works effectively with mathematical expressions that include multi-dimensional arrays.	• Does not support any other GPU except NVIDIA. • TensorFlow releases frequent updates every 2–3 months, which increases the overhead of users to install. • Difficult to use lower-level API.	Wide range of models including CNN, RNN, GAN, Transformer, etc. [155]	Chats and Tweets [14], Bangla Text [18], Offline Content [129], Social Media text analysis [112], Comments and Toxicity [156], Multilingual Tweets and Hate speech [157], Wikipedia talk page [158], Post of Social Network platform Gab [159,160]

Table 7. *Cont.*

Frameworks	Strengths	Limitations	Supported DL Algorithms	Used in Cyberbullying
Keras	• High-level API available for DL with good documentation. • Wraps all the back-end libraries and hides their complexity from users.	• Experimenting with new architectures is not that optimal. • In some advanced topics, documentation is not clear enough. • Code modifications are not allowed.	Wide range of models including CNN, RNN, GAN, Transformer, etc. [161]	Twitter [2,75,115,162], Bully, Sentiment, Emotion and Sarcasm from Twitter and Reddit [124], Social media content[68,115,163], Twitter and Wikipedia [164], Chats and Tweets [14], Wikipedia, Twitter, Formspring and YouTube [25], Social networks' text and image [25], online textual harassment [71]
Torch/ PyTorch	• Depends on the Tensor library and thus implementation cost is zero. • It is flexible and readable for developers. Its modularity and speed are high, and codes are easy to reusable.	• It is no longer under development, and the last version is Torch7. • Lua is not popular, and LuaJIT creates problems in integration.	Majority of the DL Models including CNN, RNN, GAN, Transformer, etc. [165]	Social Network platform Gab [159], Twitter, Wikipedia, Formspring [22], Harmful meme of COVID-19 [166], Memes of US politics [167], Image from online [168], Cyberbert: BERT for cyberbullying identification [22], Social media content [73]
Theano	• An open-source and cross-platform project. • As a powerful library, it helps to reduce both development and execution time. • To make RNN implementation easier, the scan API supports looping control.	• Although wrappers exist such as Keras, because of lower-level API, it is difficult to use for creating DL models directly. • No longer under active development.	Majority of the DL Models [169]	Twitter and Formspring.me [137], Twitter [137,170], Comments and posts from YouTube, Instagram and Twitter [171], Twitter and Facebook [172], Social media image [132], Online textual harassment [71]
Caffe	• Efficient for image processing with convolutional neural networks. • Some pre-trained models are available. • Since Python and MATLAB are very high-level languages, it is very easy to code with Caffe.	• Development is not active so that it loses its effectiveness day by day. • Many RNN applications need variable-sized input because the static model graph does not fit. • Custom layers of Caffe must be written in C++ language.	Initially designed for CNNs [173]	No Works Found
Chainer	• Based on the Define-by-Run principle, Chainer has a dynamic computational graph. • It provides libraries for industrial applications. • It has some strong investors such as Toyota, NTT, FANUC, etc.	• Does not support higher-order gradients. • For fixed networks, Dynamic Computational Graph (DCG) is generated every time.	For CNNs, Dynamic Computational Graph [174]	No Works Found
Deep-Learning4j	• Java ecosystem is the great advantage of DL4J. • Popular Big Data tools such as Spark, Apache Hadoop, and Kafka with an arbitrary number of GPUs and CPUs can be implemented. • Java ecosystem is dominated by many commercial industries where DL4J is a very popular and useful framework.	• Python is a very popular language for DL/ML research, whereas Java or Scala is not due to a lack of DL libraries. • Relatively less popular compared to TensorFlow and PyTorch.	DL models that are used in NLP tasks [175]	No Works Found

Table 7. *Cont.*

Frameworks	Strengths	Limitations	Supported DL Algorithms	Used in Cyberbullying
DyNet	• Optimized C++ backend and lightweight representation. • It is faster than other static declaration toolkits and significantly faster than Chainer.	• Does not support model parallelism, although it supports execution on a single CPU or GPU.	RNNs [176]	No Works Found
MXNet	• Portable and lightweight framework supported in mobile devices. • Faster computational scalability with multiple CPUs and GPUs. • Supports Python, Julia, C++, JavaScript, R, Go, Matlab, Perl, Scala, and Wolfram. • Supports ONNX format such as Microsoft CNTK, for transforming models between MXNet, Caffe, CNTK, and other DL models.	• Due to its powerful and flexible interface, MXNet can have a steep learning curve for beginners. • Limited or difficult documentation to navigate according to some MXNet's users. • Sometimes, APIs are not user-friendly.	CNNs, RNNs, GANs [177]	Wikipedia talk pages [178]
Lasagne	• Easy to understand, use, and expand. • Does not hide the abstractions behind Theano and directly processes. • Supports many optimization methods such as Nesterov momentum, ADAM, and RMSprop. • Makes common use cases easier and does not overrate the uncommon cases.	• Reliability and validity is not well established. • Smaller resources available	Feed-Forward Networs such as CNNs, Recurrent Networks including LSTM, and any combination thereof [179].	No Works Found
H_2O	• For Big Data processing and analytics, it is an optimized framework. • It provides a generic set of ML algorithms that leverage Hadoop/Spark engines for large-scale dataset processing. • Supports ONNX format such as Microsoft CNTK, for transforming models between MXNet, Caffe, CNTK, and other DL models.	• Spark can not interact with the web-based UI of H2O.	Variety of DL models including CNNs RNNs [180].	No Works Found
Google JAX	• Ability to accelerate computations on hardware accelerators such as GPUs and TPUs. • Easy implementation of complex mathematical models • Relatively simple and intuitive API • Support for distributed computing.	• Smaller user community compared to other deep learning frameworks. • Relatively new framework; thus, stability issues or bugs may be present. • As designed primarily for numerical computing, it may not be the best choice for tasks that require a lot of data preprocessing or manipulation.	Variety of DL models including CNNs and autoaggressive models.	No Works Found

Table 7. *Cont.*

Frameworks	Strengths	Limitations	Supported DL Algorithms	Used in Cyberbullying
Mind-Spore	• Designed with a focus on performance and scalability for large-scale DL. • Supports distributed training across multiple devices and platforms. • Supports multiple programming languages, including Python and C++.	• The documentation and resources are still being developed and may be less comprehensive than other frameworks. • Some advanced features and functionality may be missing or not yet implemented. • May not be the best choice for small-scale projects.	CNNs and RNNs with a focus on distributed training and image processing [181].	No Works Found

Applicability of Different DL Frameworks

We observed different DL frameworks to pinpoint their suitability of using in cyberbully detection and prediction problems. Some frameworks are flexible and easy to implement while others are quick to implement in any deep-learning-based prototypes. In some cases, a few frameworks are fast to deploy different machine learning models, while other frameworks have quick processing speed.

In order to define layers, network types (CNNs, RNNs), and standard model designs for image or natural language processing, these frameworks provide easy accessibility to their respective libraries. We find that several studies used TensorFlow [14,129,182,183] in their experiments, because the framework is flexible, easy to use and suits natural language processing tasks. A rich collection of libraries make Tensorflow easily usable for the researchers in the field of natural language processing and in cyberbullying detection. The TensorFlow interface is not difficult to understand and use and the framework does not make the platform complicated for the beginners.

However, Keras is also a popular framework [20,25,171,184] for cyberbullying detection since the framework is built to provide a simple interface for quick prototyping by building active neural networks that can work upon TensorFlow. Another widely used framework for cyberbullying detection [19,73,168,185] is PyTorch, which is developed based on the Torch [186] library of Lua programming language. PyTorch is fast and more Pythonic than the rest of the frameworks.

Theano is another heavily used framework in cyberbullying-related tasks [137,170,171] due to its quick processing speed. We have also observed other research that examined cyberbullying by using different frameworks, such as Caffe [187], Deeplearning4j [188], and MXNet [178]. Furthermore, other well-known frameworks such as Chainer, DyNet, Lasagne, H2O, etc., can be used for handling large amounts of data; although in the current literature, researchers hardly use these frameworks in cyberbullying detection problems.

8. Datasets for Experiments

Researchers have conducted several studies to identify cyberbullying over the years. People may encounter cyberbullying by different form of contents such as text, images, collage, meme and others. In this section, we present different datasets, which are relevant to cyberbullying, DL architecture, and tasks that have been conducted in previous studies in Table 8. Most datasets are collected from social media, i.e., Twitter, YouTube, and Wikipedia. Users are likely to interact on social media such as in real-life society. Thus, they might experience different behavior from others, including bullying. On the other hand, we find a limited amount of cyberbullying datasets with images and text.

Table 8. Cyberbullying-related research dataset.

Dataset	DL Architectures	Major Tasks
Textual Content		
Impermium [189]	Bi-LSTM, GRU, LSTM, and RNN	Intimidation detection on social media platforms [21]
Formspring (a Q&A forum)	Single Linear Neural Network Layer and Transformer	Cyberbullying detection [73]
	CNN, LSTM, Bi-LSTM, Bi-LSTM with Attention	Systematically analyzes cyberbullying detection [68]
	PCNN	Handle the difficulty of noise and distortion in social media postings and messages in detecting cyberbullying [137]
Wikipedia	Single Linear NN, Transformer [73], MLP [158]	Cyberbullying detection [68,73,158]
	CNN, LSTM, Bi-LSTM, Bi-LSTM with Attention [68]	Systematically analyzes cyberbullying detection
Twitter [190]	CNN, LSTM, Bi-LSTM, Bi-LSTM with Attention [68]	Systematically analyzes cyberbullying detection [68]
	Char-CNNS	Cyberbullying detection [5]
Text		
Twitter [191]	PCNN	Handling the difficulty of noise and distortion in social media postings and messages in detecting cyberbullying [137]
Twitter (combination of 3 datasets) [115,190,192]	Bi-GRU, Transformer Block, and CNN	Detecting Aggressive Behavior
Twitter (Indonesian Language) [64]	LSTM, Bi-LSTM, and CNN	Cyberbullying Detection
YouTube [193]	Bi-LSTM with attention	Cyberbullying Detection [25]
Bangla and Romanized Bangla [18]	CNN, LSTM, Bi-LSTM, and GRU	Comparative analysis [18]
Toxic Comment Classification challenge [194]	LSTM-CNN [63]	Cyberbullying Detection [63]
The bullying traces dataset [195]	SVM activated stacked convolution LSTM network [7]	
Textual and Visual		
Vine [196,197]	ResidualBiLSTM-RCNN	Cyberbullying Detection [6]

9. Challenges, Open Issues, and Future Trends

Detecting cyberbullying is a problem connected with human psychology and emotional response to how an individual reacts toward it due to different factors (i.e., image, emotion, culture, etc.). In the following subsections, we discuss different challenging and open issues with cyberbullying with DL.

9.1. Issues in DL

- Require a large amount of dataset: Large volumes of labeled data are required for DL. For example, the creation of self-driving cars involves millions of photos and hundreds of hours of video [198]. It is commonly known that data preparation consumes 80–90% of the time spent on ML development. Furthermore, even the strongest DL algorithms will struggle to function without good data and present weak performance to handle biased and unclean data during model training [199].

- High computational power: DL takes a lot of computational power. The parallel design of high-performance GPUs is ideal for DL. When used in conjunction with clusters or cloud computing, this allows development teams to cut DL network time for training from weeks to hours or less [198].
- Reasoning of prediction unexplainable: DL result prediction follows the Black-Box testing approach. Thus, it is not capable of making any explainable predictions. Since DL's hidden weight and activation are non-interpretable, its predictions are considered as non-explainable [200].
- Security issue: Preventing the DL models from security attacks is the biggest challenge nowadays. Based on the occurring time, there are two types of security attacks. One is *poisoning* attack, which occurs during the training period, and another one is *evasion* attack, which occurs during interference (after training). By corrupting the data with malicious examples, poisoning attacks compromise the training process. On the other hand, evasion attacks use adversarial examples to confuse the entire classification process [201].
- Models are not adaptive: In the present world, data are very dynamic. Data are changing due to various factors, which may be constantly changing, such as location, time, and many other factors. However, DL models are built using a defined set, which is called the training dataset. Later, the performance of the model is measured by the data, which also comes from the same distribution of the training data, and eventually, the model performs well. Later, the same model may start performing poorly due to the changing the characteristics of the data, which are not entirely different, but have some variations from the training data. This is difficult to manage in DL to retrain the old models.

9.2. Challenges in Cyberbullying detection

- Cultural diversity for cyberbullying: Language is one of the important parts of the culture of a nation. Since cyberbullying has become a common problem among different nations, we may not expect a good prediction model by using a dataset of one nation and testing over the dataset of another culturally varied nation.
- Language challenge: Capturing context and analyzing the sentiment from different types of sentences is a difficult task and challenging work for cyberbullying detection. For example, "The image that you have sent so irritated me and I would rather not contact with you any longer!" is not easy to detect as cyberbullying without investigating from a rationale factor, albeit that model shows negative sentiment [26].
- Dataset challenge: Retrieving data from social media is not an easy task, as it relates to private information. Moreover, social media sites do not share user data publicly. Due to these issues, it is hard to gather quality data from social sites, which causes the lack of quality data to improve learning. Another challenging task is to annotate or label the data because they require a domain expert to label the corpus [202].
- Data representation challenge: Setting up an effective cyberbullying-detection system is difficult due to the need for human interaction and the nature of cyberbullying. Furthermore, the nature of cyberbullying is challenging to identify in the cyberbullying detection problem. The vast majority of the exploratory works directly identified bullying words in social media. However, separating content-based features have their own difficulties. For the absence of appropriate information, the performance of the model might decay [203].
- Natural Language Processing (NLP) challenges: The biggest challenge in natural language processing is understanding the meaning of the text. The relevant task is to build the right vocabulary, link the various components of the vocabulary, establish context, and extract semantic meaning from the data [204]. Misspelling and ambiguous expressions are other challenges that are very difficult to solve for the machine.
- Reusability of pre-trained model for sentiment analysis and cyberbullying: Although cyberbullying detection and sentiment analysis are related tasks, these two tasks have

significant differences from each other; therefore, the pre-trained model of one task is likely to be difficult to use to predict another task. Sentiment analysis involves determining the overall emotional tone of a text, where the sentence is positive, negative, or neutral. On the contrary, cyberbullying detection involves identifying specific patterns of harmful words.

Yet, there are some sentiment analysis approaches that can be used to identify cyberbullying. Atoum et al. [205] proposed an approach for detecting cyberbullying using sentiment analysis techniques. Nahar et al. [206] presented a novel method for identifying online bullying on social media sites from sentiment analysis. Dani et al. [207] presented a novel framework for supervised learning that uses sentiment analysis to identify cyberbullying.

Overall, while sentiment analysis models may be helpful for cyberbullying detection, they cannot be directly reused without significant modifications and additional training. Cyberbullying detection (i.e., yes/no classes) largely needs to identify negative words, which are used to harass a person, while sentiment analysis has three different classes (i.e., negative, positive, and neutral) where negative patterns are part of the problem. In this case, positive and neutral categories are also dominant class labels. Since the nature of the outputs is different in two different problems, we cannot completely reuse one pre-trained model for other cases.

9.3. Future Trends

Challenges and issues of technology may unveil the opportunity to conduct further research. There are many avenues to extend the above issues for deploying concrete research. We mainly discuss a few possible aspects as future trends.

- Multilingual and multimedia content: In current times, social media and other virtual platforms are widely used among different levels of users in terms of age group, culture, language, taste, education, etc. Since social media is a vital platform for propagating cyber harassment, users may use multilingual and multimedia content; therefore, we may put more attention on building efficient cyberbullying detection systems for multilingual and multimedia content.
- Cyberbullying detection-specific word embedding: In recent times, researchers are introducing different domain specific word-embedding techniques, because these platforms produce accurate results for relevant sets of vocabularies. For example, Med-BERT is used for health-domain-based BERT-aware embedding systems. In this connection, researchers may propose a specialized word-embedding system for cyberbullying detection problems.
- Cyberbullying detection in SMS and email: Users are concerned with combating cyberbullying problems, which largely propagate through social media platforms. However, future researchers may put more attention on investigating Short Message Service (SMS)- and email-based cyberbullying detection methods.
- Cyberbullying impact on mental health: Cyberbullying may leave a long-term impact on the mental status of an individual. Some may take a life-threatening step or commit self-injury to curb the severity of the harassment and take death for granted. Therefore, mental health researchers can consider this issue as a timely topic and introduce different methods to fight against cyber harassment.
- Use of cutting-edge deep learning: With the advancement of deep-learning-based methods, we may introduce more subtle and delicate techniques to detect cyberbullying problems. For example, stacked and multi-channel CNN or Bi-LSTM-based cyberbullying-based frameworks or their advanced version or hybridization of these models may produce more sophisticated solutions to counter the problems.

10. Conclusions

Cyberbullying is a kind of harassment using digital technologies, which might take place on smartphones, social media sites, messaging applications, etc. The targeted indi-

viduals will likely become agitated by repeated behavior, angering and shaming from the rouge users. This can affect the victim mentally and physically and may lead to severe trauma or mental disorder. In this study, we have thoroughly investigated cyberbullying detection-related existing studies that are based on DL techniques. We also conducted a holistic review to identify the strength and future direction of these works. Future researchers will benefit from this timely review since they can find the existing datasets, the research challenges, and the open issues in this area.

We plan to thoroughly investigate hybrid deep learning models used for the detection of cyberbullying in the future. The research on the identification of cyberbullying in texts and images has been explored in this paper; however, the classification of cyberbullying in speech, videos, or deep fakes is hardly found. In addition, we are interested in performing an extensive analysis of the personalized behavior (i.e., personality, values, etc.) of online users. In the literature, we could not find significant research work on the association between cyberbullying behavior and perpetrators' mental health issues, which could be an interesting part of the research. Additionally, a review of a recommender system can be beneficial for future research in this area because it will be extremely helpful in recognizing patterns in cyberbullying. The research could be associated with the link prediction research because a user can be monitored well ahead by observing his/her day-to-day online behavior so that he/she cannot be turned into a bullier in the course of time. There are several domain-specific word-embedding models in the literature (i.e., Med-BERT for the health domain). We suggest that future enthusiasts on cyberbullying research may plan for cyberbully BERT so that the pre-trained model easily predicts the bully behavior online.

Author Contributions: Conceptualization, M.T.H., M.A.E.H. and M.S.H.M.; methodology, M.T.H.; formal analysis, M.T.H.; resources, M.S.H.M.; writing—original draft preparation, M.T.H. and M.A.E.H.; writing—review and editing, M.S.H.M., A.A., M.A. and S.I.; supervision, M.S.H.M., M.A. and S.I.; project administration, M.S.H.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research is funded by the Institute for Advanced Research Publication Grant of United International University, Ref. No.: IAR-2023-Pub-013.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Feinberg, T.; Robey, N. Cyberbullying. *Educ. Dig.* **2009**, *74*, 26.
2. Marwa, T.; Salima, O.; Souham, M. Deep learning for online harassment detection in tweets. In Proceedings of the 2018 3rd International Conference on Pattern Analysis and Intelligent Systems (PAIS), Tebessa, Algeria, 24–25 October 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–5.
3. Nikolaou, D. Does cyberbullying impact youth suicidal behaviors? *J. Health Econ.* **2017**, *56*, 30–46. [PubMed] [CrossRef]
4. Brailovskaia, J.; Teismann, T.; Margraf, J. Cyberbullying, positive mental health and suicide ideation/behavior. *Psychiatry Res.* **2018**, *267*, 240–242. [CrossRef] [PubMed]
5. Lu, N.; Wu, G.; Zhang, Z.; Zheng, Y.; Ren, Y.; Choo, K.K.R. Cyberbullying detection in social media text based on character-level convolutional neural network with shortcuts. *Concurr. Comput. Pract. Exp.* **2020**, *32*, e5627. [CrossRef]
6. Paul, S.; Saha, S.; Hasanuzzaman, M. Identification of cyberbullying: A deep learning based multimodal approach. *Multimed. Tools Appl.* **2020**, *81*, 26989–27008. [CrossRef]
7. Buan, T.A.; Ramachandra, R. Automated cyberbullying detection in social media using an svm activated stacked convolution lstm network. In Proceedings of the 2020 the 4th International Conference on Compute and Data Analysis, Silicon Valley, CA, USA, 9–12 March 2020; pp. 170–174.
8. Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv* **2014**, arXiv:1406.1078.
9. Gers, F.A.; Schmidhuber, J.; Cummins, F. Learning to forget: Continual prediction with LSTM. *Neural Comput.* **2000**, *12*, 2451–2471. [CrossRef]
10. Huang, Z.; Xu, W.; Yu, K. Bidirectional LSTM-CRF models for sequence tagging. *arXiv* **2015**, arXiv:1508.01991.

11. Caroppo, A.; Leone, A.; Siciliano, P. Comparison between deep learning models and traditional machine learning approaches for facial expression recognition in ageing adults. *J. Comput. Sci. Technol.* **2020**, *35*, 1127–1146. [CrossRef]
12. Yilmaz, A.; Demircali, A.A.; Kocaman, S.; Uvet, H. Comparison of Deep Learning and Traditional Machine Learning Techniques for Classification of Pap Smear Images. *arXiv* **2020**, arXiv:2009.06366.
13. Finizola, J.S.; Targino, J.M.; Teodoro, F.G.S.; Moraes Lima, C.A.d. A comparative study between deep learning and traditional machine learning techniques for facial biometric recognition. In Proceedings of the Ibero-American Conference on Artificial Intelligence, Trujillo, Peru, 13–16 November 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 217–228.
14. Banerjee, V.; Telavane, J.; Gaikwad, P.; Vartak, P. Detection of cyberbullying using Deep Neural Network. In Proceedings of the 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India, 15–16 March 2019; IEEE: PIscataway, NJ, USA, 2019; pp. 604–607.
15. Kamath, C.N.; Bukhari, S.S.; Dengel, A. Comparative study between traditional machine learning and deep learning approaches for text classification. In Proceedings of the ACM Symposium on Document Engineering 2018, Halifax, NS, Canada, 28–31 August 2018; pp. 1–11.
16. Wang, P.; Fan, E.; Wang, P. Comparative analysis of image classification algorithms based on traditional machine learning and deep learning. *Pattern Recognit. Lett.* **2021**, *141*, 61–67. [CrossRef]
17. Naufal, M.F.; Kusuma, S.F.; Prayuska, Z.A.; Yoshua, A.A.; Lauwoto, Y.A.; Dinata, N.S.; Sugiarto, D. Comparative Analysis of Image Classification Algorithms for Face Mask Detection. *J. Inf. Syst. Eng. Bus. Intell.* **2021**, *7*, 56–66. [CrossRef]
18. Ahmed, M.T.; Rahman, M.; Nur, S.; Islam, A.; Das, D. Deployment of machine learning and deep learning algorithms in detecting cyberbullying in bangla and romanized bangla text: A comparative study. In Proceedings of the 2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), Bhilai, India, 19–20 February 2021; IEEE: PIscataway, NJ, USA, 2021; pp. 1–10.
19. Rezvani, N.; Beheshti, A.; Tabebordbar, A. Linking textual and contextual features for intelligent cyberbullying detection in social media. In Proceedings of the 18th International Conference on Advances in Mobile Computing & Multimedia, Chiang Mai, Thailand, 30 November–2 December 2020; pp. 3–10.
20. Al-Ajlan, M.A.; Ykhlef, M. Deep learning algorithm for cyberbullying detection. *Int. J. Adv. Comput. Sci. Appl* **2018**, *9*, 199–205. [CrossRef]
21. Iwendi, C.; Srivastava, G.; Khan, S.; Maddikunta, P.K.R. Cyberbullying detection solutions based on deep learning architectures. *Multimed. Syst.* **2020**, 1–14. [CrossRef]
22. Paul, S.; Saha, S. CyberBERT: BERT for cyberbullying identification. *Multimed. Syst.* **2022**, *28*, 1897–1904. [CrossRef]
23. Akhter, M.P.; Jiangbin, Z.; Naqvi, I.R.; AbdelMajeed, M.; Zia, T. Abusive language detection from social media comments using conventional machine learning and deep learning approaches. *Multimed. Syst.* **2022**, *28*, 1925–1940. [CrossRef]
24. Picon, A.; Alvarez-Gila, A.; Irusta, U.; Echazarra, J. Why deep learning performs better than classical machine learning? *Dyna Ing. Ind.* **2020**, *95*, 119–122. [CrossRef]
25. Dadvar, M.; Eckert, K. Cyberbullying detection in social networks using deep learning based models; a reproducibility study. *arXiv* **2018**, arXiv:1812.08046.
26. Salawu, S.; He, Y.; Lumsden, J. Approaches to automated detection of cyberbullying: A survey. *IEEE Trans. Affect. Comput.* **2017**, *11*, 3–24. [CrossRef]
27. Rosa, H.; Pereira, N.; Ribeiro, R.; Ferreira, P.C.; Carvalho, J.P.; Oliveira, S.; Coheur, L.; Paulino, P.; Simão, A.V.; Trancoso, I. Automatic cyberbullying detection: A systematic review. *Comput. Hum. Behav.* **2019**, *93*, 333–345. [CrossRef]
28. Kim, S.; Razi, A.; Stringhini, G.; Wisniewski, P.J.; De Choudhury, M. A Human-Centered Systematic Literature Review of Cyberbullying Detection Algorithms. *Proc. ACM Hum.-Comput. Interact.* **2021**, *5*, 1–34. [CrossRef]
29. Elsafoury, F.; Katsigiannis, S.; Pervez, Z.; Ramzan, N. When the Timeline Meets the Pipeline: A Survey on Automated Cyberbullying Detection. *IEEE Access* **2021**, *9*, 103541–103563. [CrossRef]
30. Haidar, B.; Chamoun, M.; Yamout, F. Cyberbullying detection: A survey on multilingual techniques. In Proceedings of the 2016 European Modelling Symposium (EMS), Pisa, Italy, 28–30 November 2016; pp. 165–171.
31. Al-Garadi, M.A.; Hussain, M.R.; Khan, N.; Murtaza, G.; Nweke, H.F.; Ali, I.; Mujtaba, G.; Chiroma, H.; Khattak, H.A.; Gani, A. Predicting Cyberbullying on Social Media in the Big Data Era Using Machine Learning Algorithms: Review of Literature and Open Challenges. *IEEE Access* **2019**, *7*, 70701–70718. [CrossRef]
32. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *arXiv* **2013**, arXiv:1301.3781.
33. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed representations of words and phrases and their compositionality. *Adv. Neural Inf. Process. Syst.* **2013**, *26*.
34. Pennington, J.; Socher, R.; Manning, C.D. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1532–1543.
35. Peters, M.E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; Zettlemoyer, L. Deep contextualized word representations. *arXiv* **2018**, arXiv:1802.05365. https://doi.org/10.48550/ARXIV.1802.05365.
36. Joulin, A.; Grave, E.; Bojanowski, P.; Mikolov, T. Bag of tricks for efficient text classification. *arXiv* **2016**, arXiv:1607.01759.
37. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.

38. Ramos, J. Using tf-idf to determine word relevance in document queries. In *Proceedings of the First Instructional Conference on Machine Learning*; Rutgers University: Piscataway, NJ, USA, 2003; Volume 242, pp. 29–48.
39. Ackley, D.H.; Hinton, G.E.; Sejnowski, T.J. A learning algorithm for Boltzmann machines. *Cogn. Sci.* **1985**, *9*, 147–169. [CrossRef]
40. Hinton, G.E. Deep belief networks. *Scholarpedia* **2009**, *4*, 5947. [CrossRef]
41. Baldi, P. Autoencoders, unsupervised learning, and deep architectures. In Proceedings of the ICML Workshop on Unsupervised and Transfer Learning, Bellevue, WA, USA, 2 July 2011; pp. 37–49.
42. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. *Adv. Neural Inf. Process. Syst.* **2014**, *27*.
43. Albawi, S.; Mohammed, T.A.; Al-Zawi, S. Understanding of a convolutional neural network. In Proceedings of the 2017 International Conference on Engineering and Technology (ICET), Antalya, Turkey, 21–23 August 2017; pp. 1–6.
44. Mikolov, T.; Karafiát, M.; Burget, L.; Cernockỳ, J.; Khudanpur, S. Recurrent neural network based language model. In Proceedings of the Interspeech, Makuhari, Japan, 26–30 September 2010; Volume 2, pp. 1045–1048.
45. Al-Harigy, L.M.; Al-Nuaim, H.A.; Moradpoor, N.; Tan, Z. Building toward Automated Cyberbullying Detection: A Comparative Analysis. *Comput. Intell. Neurosci.* **2022**, *2022*, 4794227. [CrossRef] [PubMed]
46. Riadi, I.; Widiandana, P. Mobile Forensics for Cyberbullying Detection using Term Frequency-Inverse Document Frequency (TF-IDF). *J. Eng. Sci. Technol.* **2019**, *5*, 68–76. [CrossRef]
47. Rahman, S.; Talukder, K.H.; Mithila, S.K. An Empirical Study to Detect Cyberbullying with TF-IDF and Machine Learning Algorithms. In Proceedings of the 2021 International Conference on Electronics, Communications and Information Technology (ICECIT), Khulna, Bangladesh, 14–16 September 2021; pp. 1–4.
48. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
49. Yani, M.A.A.; Maharani, W. Analyzing Cyberbullying Negative Content on Twitter Social Media with the RoBERTa Method. *JINAV J. Inf. Vis.* **2023**, *4*.
50. Lan, Z.; Chen, M.; Goodman, S.; Gimpel, K.; Sharma, P.; Soricut, R. Albert: A lite bert for self-supervised learning of language representations. *arXiv* **2019**, arXiv:1909.11942.
51. Tripathy, J.K.; Chakkaravarthy, S.S.; Satapathy, S.C.; Sahoo, M.; Vaidehi, V. ALBERT-based fine-tuning model for cyberbullying analysis. *Multimed. Syst.* **2022**, *28*, 1941–1949. [CrossRef]
52. Clark, K.; Luong, M.T.; Le, Q.V.; Manning, C.D. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv* **2020**, arXiv:2003.10555.
53. Sanh, V.; Debut, L.; Chaumond, J.; Wolf, T. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *arXiv* **2019**, arXiv:1910.01108.
54. Harbaoui, A.; Benaissa, A.R. Cost-Sensitive PLM-based Approach for Arabic and English Cyberbullying Classification. Available at Research Square. 2023. Available online: https://www.researchsquare.com/article/rs-2524732/v1 (accessed on 6 May 2023).
55. Sokolová, Z.; Staš, J.; Juhár, J. Review of Recent Trends in the Detection of Hate Speech and Offensive Language on Social Media. *Acta Electrotech. Inform.* **2022**, *22*, 18–24.
56. Warke, O.; Jose, J.M.; Breitsohl, J. Utilising Twitter Metadata for Hate Classification. In Proceedings of the Advances in Information Retrieval: 45th European Conference on Information Retrieval, ECIR 2023, Dublin, Ireland, 2–6 April 2023; Proceedings, Part II; Springer: Berlin/Heidelberg, Germany, 2023; pp. 676–684.
57. Sun, Z.; Yu, H.; Song, X.; Liu, R.; Yang, Y.; Zhou, D. Mobilebert: A compact task-agnostic bert for resource-limited devices. *arXiv* **2020**, arXiv:2004.02984.
58. Mazari, A.C.; Boudoukhani, N.; Djeffal, A. BERT-based ensemble learning for multi-aspect hate speech detection. *Clust. Comput.* **2023**, 1–15. [CrossRef]
59. Coban, O.; Ozel, S.A.; Inan, A. Detection and cross-domain evaluation of cyberbullying in Facebook activity contents for Turkish. In *ACM Transactions on Asian and Low-Resource Language Information Processing*; Association for Computing Machinery: New York, NY, USA, 2023; Volume 22.
60. Mozafari, M.; Farahbakhsh, R.; Crespi, N. A BERT-based transfer learning approach for hate speech detection in online social media. In Proceedings of the Complex Networks and Their Applications VIII: Volume 1 Proceedings of the Eighth International Conference on Complex Networks and Their Applications COMPLEX NETWORKS 2019, Lisbon, Portugal, 10–12 December 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 928–940.
61. Feng, Z.; Su, J.; Cao, J. BHF: BERT-based Hierarchical Attention Fusion Network for Cyberbullying Remarks Detection. In Proceedings of the 2022 5th International Conference on Machine Learning and Natural Language Processing, Hangzhou, China, 23–25 September 2022; pp. 1–7.
62. Ishaq, A.; Malik, K.M.; Zafar, A. Hatespeechbert: Retraining Bert for Automatic Hate Speechdetection. Available online: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4329716 (accessed on 5 April 2023).
63. Gada, M.; Damania, K.; Sankhe, S. Cyberbullying Detection using LSTM-CNN architecture and its applications. In Proceedings of the 2021 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 27–29 January 2021; pp. 1–6.

64. Anindyati, L.; Purwarianti, A.; Nursanti, A. Optimizing deep learning for detection cyberbullying text in indonesian language. In Proceedings of the 2019 International Conference of Advanced Informatics: Concepts, Theory and Applications (ICAICTA), Yogyakarta, Indonesia, 20–21 September 2019; pp. 1–5.
65. Al-Hashedi, M.; Soon, L.K.; Goh, H.N. Cyberbullying detection using deep learning and word embeddings: An empirical study. In Proceedings of the 2019 2nd International Conference on Computational Intelligence and Intelligent Systems, Bangkok Thailand, 23–25 November 2019; pp. 17–21.
66. Bu, S.J.; Cho, S.B. A hybrid deep learning system of CNN and LRCN to detect cyberbullying from SNS comments. In Proceedings of the International Conference on Hybrid Artificial Intelligence Systems, Oviedo, Spain, 20–22 June 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 561–572.
67. Al-Ajlan, M.A.; Ykhlef, M. Optimized twitter cyberbullying detection based on deep learning. In Proceedings of the 2018 21st Saudi Computer Society National Computer Conference (NCC), Riyadh, Saudi Arabia, 25–26 April 2018; pp. 1–5.
68. Agrawal, S.; Awekar, A. Deep learning for detecting cyberbullying across multiple social media platforms. In Proceedings of the European Conference on Information Retrieval, Springer: Berlin/Heidelberg, Germany, 2018; pp. 141–153.
69. Azumah, S.W.; Elsayed, N.; ElSayed, Z.; Ozer, M. Cyberbullying in Text Content Detection: An Analytical Review. Available online: https://arxiv.org/pdf/2303.10502.pdf (accessed on 5 April 2023).
70. Bhatt, J. Using Hybrid Deep Learning and Word Embedding Based Approach for Advance Cyberbullying Detection. Ph.D Thesis, National College of Ireland, Dublin, Ireland, 2020.
71. Kumar, A.; Sachdeva, N. A Bi-GRU with attention and CapsNet hybrid model for cyberbullying detection on social media. *World Wide Web* **2022**, *25*, 1537–1550. [CrossRef]
72. Wang, K.; Cui, Y.; Hu, J.; Zhang, Y.; Zhao, W.; Feng, L. Cyberbullying detection, based on the fasttext and word similarity schemes. *ACM Trans. Asian Low-Resour. Lang. Inf. Process. (TALLIP)* **2020**, *20*, 1–15. [CrossRef]
73. Yadav, J.; Kumar, D.; Chauhan, D. Cyberbullying detection using pre-trained BERT model. In Proceedings of the 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2–4 July 2020; pp. 1096–1100.
74. Haidar, B.; Chamoun, M.; Serhrouchni, A. Arabic cyberbullying detection: Using deep learning. In Proceedings of the 2018 7th International Conference on Computer and Communication Engineering (ICCCE), Kuala Lumpur, Malaysia, 19–20 September 2018; pp. 284–289.
75. Aldhyani, T.H.; Al-Adhaileh, M.H.; Alsubari, S.N. Cyberbullying identification system based deep learning algorithms. *Electronics* **2022**, *11*, 3273. [CrossRef]
76. Pericherla, S.; Ilavarasan, E. Performance analysis of word embeddings for cyberbullying detection. In Proceedings of the IOP Conference Series: Materials Science and Engineering, Sanya, China, 12–14 November 2021; IOP Publishing: Bristol, UK, 2021; Volume 1085, p. 012008.
77. Eronen, J.; Ptaszynski, M.; Masui, F. Comparing Performance of Different Linguistically Backed Word Embeddings for Cyberbullying Detection. *arXiv* **2022**, arXiv:2206.01950.
78. Alhloul, A.; Alam, A. Bullying Tweets Detection Using CNN-Attention. *Int. J. Cybern. Inform. (IJCI)* **2023**, *12*, 65–78. [CrossRef]
79. Kountchev, R.; Rubin, S.; Milanova, M.; Todorov, V.; Kountcheva, R. Cognitive image representation based on spectrum pyramid decomposition. In Proceedings of the WSEAS International Conference on Mathematical Methods and Computational Techniques in Electrical Engineering (MMACTEE), Athens, Greece, 29–31 December 2008; pp. 230–235.
80. Sudirman, S.; Qiu, G. Colour image representation using BSP Tree. *Proc. CGIP* **2000**, 1–4. Available online: http://www.cs.nott.ac.uk/~pszqiu/Online/CGIP2000.pdf (accessed on 5 April 2023).
81. Wei, H.; Zuo, Q.; Lang, B. A bio-inspired model for image representation and image analysis. In Proceedings of the 2011 IEEE 23rd International Conference on Tools with Artificial Intelligence, Boca Raton, FL, USA, 7–9 November 2011; pp. 409–413.
82. Xue, X.; Wu, X. Directly operable image representation of multiscale primal sketch. *IEEE Trans. Multimed.* **2005**, *7*, 805–816.
83. Gao, S.; Duan, L.; Tsang, I.W. DEFEATnet—A deep conventional image representation for image classification. *IEEE Trans. Circuits Syst. Video Technol.* **2015**, *26*, 494–505. [CrossRef]
84. Mori, S.; Suen, C.Y.; Yamamoto, K. Historical review of OCR research and development. *Proc. IEEE* **1992**, *80*, 1029–1058. [CrossRef]
85. Pradheep, T.; Sheeba, J.; Yogeshwaran, T.; Pradeep Devaneyan, S. Automatic Multi Model Cyber Bullying Detection from Social Networks. In Proceedings of the International Conference on Intelligent Computing Systems (ICICS 2017), Irbid, Jordan, 15–16 September 2017; Sona College of Technology: Tamil Nadu, Idia, 2017.
86. Sheeba, J.; Devaneyan, S.P. Impulsive intermodal cyber bullying recognition from public nets. *Int. J. Adv. Res. Comput. Sci.* **2018**, *9*.
87. Kumari, K.; Singh, J.P.; Dwivedi, Y.K.; Rana, N.P. toward Cyberbullying-free social media in smart cities: A unified multi-modal approach. *Soft Comput.* **2020**, *24*, 11059–11070. [CrossRef]
88. Verge, T. Instagram's Bullying Comment Filter Hides Mean Comments. 2018. Available online: https://www.theverge.com/2018/5/1/17307980/instagram-bullying-comment-filter-machine-learning (accessed on 8 March 2023).
89. Facebook. Inside Feed: Fighting Abuse. 2018. Available online: https://about.fb.com/news/2018/05/inside-feed-fighting-abuse/ (accessed on 8 March 2023).
90. Gao, X.; Han, Y.; Tang, Q.; Liu, Z.; Ma, J. Chinese cyberbullying detection with OCR and deep learning. *J. Intell. Fuzzy Syst.* **2021**, *40*, 4731–4743.

91. Borah, N.; Borah, P. Detecting cyberbullying on social media using OCR and machine learning. In Proceedings of the 4th International Conference on Intelligent Computing and Control Systems (ICICCS 2020), Madurai, India, 13–15 May 2020; pp. 300–305.
92. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Netw.* **2015**, *61*, 85–117. [PubMed] [CrossRef]
93. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
94. Prechelt, L. Early stopping-but when? In *Neural Networks: Tricks of the Trade*; Springer: Berlin/Heidelberg, Germany, 1998; pp. 55–69.
95. Das, H.S.; Roy, P. A deep dive into deep learning techniques for solving spoken language identification problems. In *Intelligent Speech Signal Processing*; Elsevier: Amsterdam, The Netherlands, 2019; pp. 81–100.
96. Hinton, G.E. A practical guide to training restricted Boltzmann machines. In *Neural networks: Tricks of the Trade*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 599–619.
97. Salakhutdinov, R.; Hinton, G. Deep Boltzmann Machines. In Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics, Clearwater Beach, FL, USA, 16–18 April 2009; van Dyk, D., Welling, M., Eds.; Proceedings of Machine Learning Research; PMLR, Hilton Clearwater Beach Resort: Clearwater, FL, USA, 2009; Volume 5, pp. 448–455.
98. Courville, A.; Bergstra, J.; Bengio, Y. A spike and slab restricted Boltzmann machine. In Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, Fort Lauderdale, FL, USA, 11–13 April 2011; JMLR Workshop and Conference Proceedings; pp. 233–241.
99. Hinton, G.E. Boltzmann machine. *Scholarpedia* **2007**, *2*, 1668. [CrossRef]
100. Bengio, Y.; Lamblin, P.; Popovici, D.; Larochelle, H. Greedy layer-wise training of deep networks. *Adv. Neural Inf. Process. Syst.* **2006**, *19*.
101. Li, C.; Wang, Y.; Zhang, X.; Gao, H.; Yang, Y.; Wang, J. Deep belief network for spectral–spatial classification of hyperspectral remote sensor data. *Sensors* **2019**, *19*, 204. [CrossRef]
102. Salakhutdinov, R.; Hinton, G. Semantic hashing. *Int. J. Approx. Reason.* **2009**, *50*, 969–978. [CrossRef]
103. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
104. Hinton, G.E.; Salakhutdinov, R.R. Reducing the dimensionality of data with neural networks. *Science* **2006**, *313*, 504–507. [CrossRef]
105. Vachhani, B.; Bhat, C.; Das, B.; Kopparapu, S.K. Deep Autoencoder Based Speech Features for Improved Dysarthric Speech Recognition. In Proceedings of the Interspeech, Stockholm, Sweden, 20–24 August 2017; pp. 1854–1858.
106. Donahue, J.; Krähenbühl, P.; Darrell, T. Adversarial feature learning. *arXiv* **2016**, arXiv:1605.09782.
107. Makhzani, A.; Shlens, J.; Jaitly, N.; Goodfellow, I.; Frey, B. Adversarial autoencoders. *arXiv* **2015**, arXiv:1511.05644.
108. Feng, J.; Feng, X.; Chen, J.; Cao, X.; Zhang, X.; Jiao, L.; Yu, T. Generative adversarial networks based on collaborative learning and attention mechanism for hyperspectral image classification. *Remote Sens.* **2020**, *12*, 1149. [CrossRef]
109. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
110. Hopfield, J.J. Hopfield network. *Scholarpedia* **2007**, *2*, 1977. [CrossRef]
111. Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv* **2014**, arXiv:1412.3555.
112. Rezvani, N.; Beheshti, A. toward Attention-Based Context-Boosted Cyberbullying Detection in social media. *J. Data Intell.* **2021**, *2*, 418–433. [CrossRef]
113. Pericherla, S.; Ilavarasan, E. Transformer network-based word embeddings approach for autonomous cyberbullying detection. *Int. J. Intell. Unmanned Syst.* 2021, *ahead-of-print*.
114. Ahmed, T.; Ivan, S.; Kabir, M.; Mahmud, H.; Hasan, K. Performance analysis of transformer-based architectures and their ensembles to detect trait-based cyberbullying. *Soc. Netw. Anal. Min.* **2022**, *12*, 99. [CrossRef]
115. Alotaibi, M.; Alotaibi, B.; Razaque, A. A multichannel deep learning framework for cyberbullying detection on social media. *Electronics* **2021**, *10*, 2664. [CrossRef]
116. Yafooz, W.M.; Al-Dhaqm, A.; Alsaeedi, A. Detecting Kids Cyberbullying Using Transfer Learning Approach: Transformer Fine-Tuning Models. In *Kids Cybersecurity Using Computational Intelligence Techniques*; Springer International Publishing: Berlin, Germany, 2023.
117. Guo, X.; Anjum, U.; Zhan, J. Cyberbully Detection Using BERT with Augmented Texts. In Proceedings of the 2022 IEEE International Conference on Big Data (Big Data), Osaka, Japan, 17–20 December 2022; pp. 1246–1253.
118. Cheng, L.; Guo, R.; Silva, Y.; Hall, D.; Liu, H. Hierarchical attention networks for cyberbullying detection on the instagram social network. In Proceedings of the 2019 SIAM International Conference on Data Mining, Calgary, AB, Canada, 2–4 May 2019; pp. 235–243.
119. Cheng, L.; Guo, R.; Silva, Y.N.; Hall, D.; Liu, H. Modeling temporal patterns of cyberbullying detection with hierarchical attention networks. *ACM/IMS Trans. Data Sci.* **2021**, *2*, 1–23. [CrossRef]
120. Sreelakshmi, K.; Premjith, B.; Soman, K.P. Detection of Hate Speech Text in Hindi-English Code-mixed Data. *Procedia Comput. Sci.* **2020**, *171*, 737–744. [CrossRef]
121. Aurpa, T.T.; Sadik, R.; Ahmed, M.S. Abusive Bangla comments detection on Facebook using transformer-based deep learning models. *Soc. Netw. Anal. Min.* **2022**, *12*, 24. [CrossRef]

122. Khan, S.; Fazil, M.; Sejwal, V.K.; Alshara, M.A.; Alotaibi, R.M.; Kamal, A.; Baig, A.R. BiCHAT: BiLSTM with deep CNN and hierarchical attention for hate speech detection. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 4335–4344. [CrossRef]
123. Fang, Y.; Yang, S.; Zhao, B.; Huang, C. Cyberbullying detection in social networks using Bi-gru with self-attention mechanism. *Information* **2021**, *12*, 171. [CrossRef]
124. Maity, K.; Kumar, A.; Saha, S. A Multitask Multimodal Framework for Sentiment and Emotion-Aided Cyberbullying Detection. *IEEE Internet Comput.* **2022**, *26*, 68–78. [CrossRef]
125. Jang, B.; Kim, M.; Harerimana, G.; Kang, S.u.; Kim, J.W. Bi-LSTM model to increase accuracy in text classification: Combining Word2vec CNN and attention mechanism. *Appl. Sci.* **2020**, *10*, 5841. [CrossRef]
126. Jiang, M.; Liang, Y.; Feng, X.; Fan, X.; Pei, Z.; Xue, Y.; Guan, R. Text classification based on deep belief network and softmax regression. *Neural Comput. Appl.* **2018**, *29*, 61–70. [CrossRef]
127. Goroshin, R.; LeCun, Y. Saturating auto-encoders. *arXiv* **2013**, arXiv:1301.3577.
128. Dadvar, M.; Eckert, K. Cyberbullying detection in social networks using deep learning based models. In Proceedings of the International Conference on Big Data Analytics and Knowledge Discovery, Bratislava, Slovakia, 14–17 September 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 245–255.
129. Chandra, N.; Khatri, S.K.; Som, S. Cyberbullying detection using recursive neural network through offline repository. In Proceedings of the 2018 7th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Nordia, India, 29–31 August 2018; pp. 748–754.
130. Elmezain, M.; Malki, A.; Ibrahim, G.; El-Sayed, A. Hybrid Deep Learning Model–Based Prediction of Images Related to Cyberbullying. *Int. J. Appl. Math. Comput. Sci.* **2022**, *32*, 323–334.
131. Chandrasekaran, S.; Singh Pundir, A.K.; Lingaiah, T.B. Deep Learning Approaches for Cyberbullying Detection and Classification on Social Media. *Comput. Intell. Neurosci.* **2022**, *2022*, 2163458.
132. Kumar, A.; Sachdeva, N. Cyberbullying detection on social multimedia using soft computing techniques: A meta-analysis. *Multimed. Tools Appl.* **2019**, *78*, 23973–24010. [CrossRef]
133. Sripada, N.K.; Sirikonda, S.; Areefa; Leena, G.; Sriabhinay, K. Web-application for detecting cyber bullying using machine learning approach. *Proc. AIP Conf. Proc.* **2022**, *2418*, 020082.
134. Agarwal, A.; Chivukula, A.S.; Bhuyan, M.H.; Jan, T.; Narayan, B.; Prasad, M. Identification and classification of cyberbullying posts: A recurrent neural network approach using under-sampling and class weighting. In Proceedings of the International Conference on Neural Information Processing, New Delhi, India, 22–26 November 2021; Springer: Berlin/Heidelberg, Germany, 2020; pp. 113–120.
135. Luo, Y.; Zhang, X.; Hua, J.; Shen, W. Multi-featured Cyberbullying Detection Based on Deep Learning. In Proceedings of the 2021 16th International Conference on Computer Science & Education (ICCSE), Lancaster, UK, 17–21 August 2021; pp. 746–751.
136. Golem, V.; Karan, M.; Šnajder, J. Combining shallow and deep learning for aggressive text detection. In Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018), Santa Fe, NM, USA, 25 August 2018; pp. 188–198.
137. Zhang, X.; Tong, J.; Vishwamitra, N.; Whittaker, E.; Mazer, J.P.; Kowalski, R.; Hu, H.; Luo, F.; Macbeth, J.; Dillon, E. Cyberbullying detection with a pronunciation based convolutional neural network. In Proceedings of the 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), Anaheim, CA, USA, 18–20 December 2016; pp. 740–745.
138. O'Shea, K.; Nash, R. An introduction to convolutional neural networks. *arXiv* **2015**, arXiv:1511.08458.
139. Orr, M.J. *Introduction to Radial Basis Function Networks*; Technical Report; Center for Cognitive Science, University of Edinburgh: Edinburgh, UK, 1996.
140. Çürük, E.; Acı, Ç.; Eşsiz, E.S. The effects of attribute selection in artificial neural network based classifiers on cyberbullying detection. In Proceedings of the 2018 3rd International Conference on Computer Science and Engineering (UBMK), Sarajevo, Bosnia and Herzegovina, 20–23 September 2018; pp. 6–11.
141. Çiğdem, A.; Çürük, E.; Eşsiz, E.S. Automatic detection of cyberbullying in formspring. me, Myspace and YouTube social networks. *Turk. J. Eng.* **2019**, *3*, 168–178.
142. Ritter, H.; Martinetz, T.; Schulten, K. *Neural Computation and Self-Organizing Maps: An Introduction*; Addison-Wesley Reading: Boston, MA, USA, 1992.
143. Desai, A.; Kalaskar, S.; Kumbhar, O.; Dhumal, R. Cyber Bullying Detection on Social Media using Machine Learning. In Proceedings of the ITM Web of Conferences, Navi Mumbai, India, 14–15 July 2021; EDP Sciences: Ulis, France, 2021; Volume 40, p. 03038.
144. Miljković, D. Brief review of self-organizing maps. In Proceedings of the 2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 22–26 May 2017; pp. 1061–1066.
145. Bozyiğit, A.; Utku, S.; Nasiboğlu, E. Cyberbullying detection by using artificial neural network models. In Proceedings of the 2019 4th International Conference on Computer Science and Engineering (UBMK), Samsun, Turkey, 11–15 September 2019; pp. 520–524.
146. Cho, K.; Van Merriënboer, B.; Bahdanau, D.; Bengio, Y. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv* **2014**, arXiv:1409.1259.
147. de Santana Correia, A.; Colombini, E.L. Attention, please! A survey of neural attention models in deep learning. *Artif. Intell. Rev.* **2022**, *55*, 6037–6124. [CrossRef]

148. Zhang, A.; Li, B.; Wan, S.; Wang, K. Cyberbullying detection with birnn and attention mechanism. In Proceedings of the Machine Learning and Intelligent Communications: 4th International Conference, MLICOM 2019, Nanjing, China, 24–25 August 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 623–635.
149. Raj, C.; Agarwal, A.; Bharathy, G.; Narayan, B.; Prasad, M. Cyberbullying detection: Hybrid models based on machine learning and natural language processing techniques. *Electronics* **2021**, *10*, 2810. [CrossRef]
150. Bharti, S.; Yadav, A.K.; Kumar, M.; Yadav, D. Cyberbullying detection from tweets using deep learning. *Kybernetes* **2021**, *51*, 2695–2711. [CrossRef]
151. Murshed, B.A.H.; Abawajy, J.; Mallappa, S.; Saif, M.A.N.; Al-Ariki, H.D.E. DEA-RNN: A hybrid deep learning approach for cyberbullying detection in Twitter social media platform. *IEEE Access* **2022**, *10*, 25857–25871. [CrossRef]
152. Raj, M.; Singh, S.; Solanki, K.; Selvanambi, R. An application to detect cyberbullying using machine learning and deep learning techniques. *SN Comput. Sci.* **2022**, *3*, 401. [CrossRef] [PubMed]
153. Beniwal, R.; Maurya, A. Toxic comment classification using hybrid deep learning model. In Proceedings of the Sustainable Communication Networks and Application: Proceedings of ICSCN 2020, Erode, India, 6–7 August 2020; Springer: Berlin/Heidelberg, Germany, 2021; pp. 461–473.
154. Singh, N.K.; Singh, P.; Chand, S. Deep Learning based Methods for Cyberbullying Detection on Social Media. In Proceedings of the 2022 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 4–5 November 2022; pp. 521–525.
155. Pang, B.; Nijkamp, E.; Wu, Y.N. Deep learning with tensorflow: A review. *J. Educ. Behav. Stat.* **2020**, *45*, 227–248. [CrossRef]
156. Borkan, D.; Dixon, L.; Sorensen, J.; Thain, N.; Vasserman, L. Nuanced metrics for measuring unintended bias with real data for text classification. In Proceedings of the Companion Proceedings of the 2019 World Wide Web Conference, San Francisco, CA, USA, 13–17 May 2019; pp. 491–500.
157. Huang, X.; Xing, L.; Dernoncourt, F.; Paul, M.J. Multilingual twitter corpus and baselines for evaluating demographic bias in hate speech recognition. *arXiv* **2020**, arXiv:2002.10361.
158. Wulczyn, E.; Thain, N.; Dixon, L. Ex machina: Personal attacks seen at scale. In Proceedings of the 26th International Conference on World Wide Web, Perth, Australia, 3–7 April 2017; pp. 1391–1399.
159. Kennedy, B.; Atari, M.; Davani, A.M.; Yeh, L.; Omrani, A.; Kim, Y.; Coombs, K.; Havaldar, S.; Portillo-Wightman, G.; Gonzalez, E.; et al. Introducing the Gab Hate Corpus: Defining and applying hate-based rhetoric to social media posts at scale. *Lang. Resour. Eval.* **2022**, *56*, 79–108. [CrossRef]
160. Gencoglu, O. Cyberbullying detection with fairness constraints. *IEEE Internet Comput.* **2020**, *25*, 20–29. [CrossRef]
161. Gulli, A.; Pal, S. *Deep Learning with Keras*; Packt Publishing Ltd.: Birmingham, UK, 2017.
162. Kargutkar, S.; Chitre, V. Implementation of Cyberbullying Detection Using Machine Learning Techniques. *Int. J. Res. Appl. Sci. Eng. Technol.* **2021**, *9*, 290–294. [CrossRef]
163. Roy, P.K.; Mali, F.U. Cyberbullying detection using deep transfer learning. *Complex Intell. Syst.* **2022**, *8*, 5449–5467. [CrossRef]
164. Jain, V.; Kumar, V.; Pal, V.; Vishwakarma, D.K. Detection of cyberbullying on social media using machine learning. In Proceedings of the 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 8–10 April 2021; pp. 1091–1096.
165. Stevens, E.; Antiga, L.; Viehmann, T. *Deep Learning with PyTorch*; Manning Publications: Shelter Island, NY, USA, 2020.
166. Pramanick, S.; Sharma, S.; Dimitrov, D.; Akhtar, M.S.; Nakov, P.; Chakraborty, T. MOMENTA: A multimodal framework for detecting harmful memes and their targets. *arXiv* **2021**, arXiv:2109.05184.
167. Sharma, S.; Akhtar, M.; Nakov, P.; Chakraborty, T. DISARM: Detecting the victims targeted by harmful memes. *arXiv* **2022**, arXiv:2205.05738.
168. Vishwamitra, N.; Hu, H.; Luo, F.; Cheng, L. toward understanding and detecting cyberbullying in real-world images. In Proceedings of the 2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA), Online, 14–17 December 2020.
169. Brownlee, J. *Deep Learning with Python: Develop Deep Learning Models on Theano and 1678 Tensor Flow Using Keras*; Machine Learning Mastery: Vermont, VIC, Australia, 2016.
170. Sintaha, M.; Satter, S.B.; Zawad, N.; Swarnaker, C.; Hassan, A. Cyberbullying Detection Using Sentiment Analysis in Social Media. Ph.D Thesis, BRAC University, Dhaka, Bangladesh, 2016.
171. Kumar, A.; Sachdeva, N. Multimodal cyberbullying detection using capsule network with dynamic routing and deep convolutional neural network. *Multimed. Syst.* **2021**, *28*, 2043–2052. [CrossRef]
172. Kumar, A.; Sachdeva, N. Multi-input integrative learning using Deep Neural Networks and 1684 transfer learning for cyberbullying detection in real-time code-mix data. *Multimed. Syst.* **2022**, *28*, 2027–2041. [CrossRef]
173. Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S.; Long, J.; Girshick, R.; Guadarrama, S.; Darrell, T. Caffe: Convolutional architecture for fast feature embedding. In Proceedings of the 22nd ACM International Conference on Multimedia, Orlando, FL, USA, 3–7 November 2014; pp. 675–678.
174. Tokui, S.; Okuta, R.; Akiba, T.; Niitani, Y.; Ogawa, T.; Saito, S.; Suzuki, S.; Uenishi, K.; Vogel, B.; Yamazaki Vincent, H. Chainer: A Deep Learning Framework for Accelerating the Research Cycle. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; ACM: NewYork, NY, USA, 2019; pp. 2002–2011.

175. Parvat, A.; Chavan, J.; Kadam, S.; Dev, S.; Pathak, V. A survey of deep-learning frameworks. In Proceedings of the 2017 International Conference on Inventive Systems and Control (ICISC), Coimbatore, India, 19–20 January 2017; pp. 1–7.
176. Neubig, G.; Dyer, C.; Goldberg, Y.; Matthews, A.; Ammar, W.; Anastasopoulos, A.; Ballesteros, M.; Chiang, D.; Clothiaux, D.; Cohn, T.; et al. Dynet: The dynamic neural network toolkit. *arXiv* **2017**, arXiv:1701.03980.
177. Hodnett, M.; Wiley, J.F. *R Deep Learning Essentials: A Step-by-Step Guide to Building Deep Learning Models Using TensorFlow, Keras, and MXNet*; Packt Publishing Ltd.: Birmingham, UK, 2018.
178. Zaheri, S.; Leath, J.; Stroud, D. Toxic comment classification. *SMU Data Sci. Rev.* **2020**, *3*, 13.
179. Dieleman, S.; Schlüter, J.; Raffel, C.; Olson, E.; Sønderby, S.K.; Nouri, D. *Lasagne: First Release* ; Zenodo: Geneva, Switzerland, 2015. [CrossRef]
180. Candel, A.; Parmar, V.; LeDell, E.; Arora, A. *Deep learning with H2O*; H2O ai Inc.: Mountain View, CA, USA, 2016; pp. 1–21.
181. Tong, Z.; Du, N.; Song, X.; Wang, X. Study on MindSpore Deep Learning Framework. In Proceedings of the 2021 17th International Conference on Computational Intelligence and Security (CIS), Chengdu, China, 19–22 November 2020; pp. 183–186.
182. Zhao, Z.; Gao, M.; Luo, F.; Zhang, Y.; Xiong, Q. LSHWE: Improving similarity-based word embedding with locality sensitive hashing for cyberbullying detection. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; pp. 1–8.
183. Ahmed, M.F.; Mahmud, Z.; Biash, Z.T.; Ryen, A.A.N.; Hossain, A.; Ashraf, F.B. Cyberbullying detection using Deep Neural Network from social media comments in bangla language. *arXiv* **2021**, arXiv:2106.04506.
184. Dewani, A.; Memon, M.A.; Bhatti, S. Cyberbullying detection: Advanced preprocessing techniques & deep learning architecture for Roman Urdu data. *J. Big Data* **2021**, *8*, 160. [PubMed]
185. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. Pytorch: An imperative style, high-performance deep learning library. *Adv. Neural Inf. Process. Syst.* **2019**, *32*.
186. Collobert, R.; Bengio, S.; Mariéthoz, J. *Torch: A modular Machine Learning Software Library*; Technical Report; Idiap: Maldini, Switzerland, 2002.
187. Patange, T.; Singh, J.; Thorve, A.; Somaraj, Y.; Vyawahare, M. Detection of Cyberhectoring on Instagram. In Proceedings of the Proceedings 2019: Conference on Technologies for Future Cities (CTFC), Pillai College of Engineering, New Panvel, India, 8–9 February 2019.
188. Shahane, P.; Gore, D. Detection of Fake Profiles on Twitter using Random Forest & Deep Convolutional Neural Network. *Int. J. Manag. Technol. Eng.* **2019**, *9*, 3663–3667.
189. Bhaskaran, J.; Kamath, A.; Paul, S. DISCo: Detecting Insults in Social Commentary. 2017. Available online: http://cs229.stanford.edu/proj2017/final-reports/5242067.pdf (accessed on 5 April 2023).
190. Waseem, Z.; Hovy, D. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In Proceedings of the NAACL Student Research Workshop, San Diego, CA, USA, 13–15 June 2016; pp. 88–93.
191. Kasture, A.S. A predictive Model to Detect Online Cyberbullying. Ph.D Thesis, Auckland University of Technology, Auckland, New Zealand, 2015.
192. Davidson, T.; Warmsley, D.; Macy, M.; Weber, I. Automated hate speech detection and the problem of offensive language. In Proceedings of the International AAAI Conference on Web and Social Media, Montreal, QC, Canada, 15–18 May 2017; Volume 11, pp. 512–515.
193. Dadvar, M.; Trieschnigg, D.; Jong, F.d. Experts and machines against bullies: A hybrid approach to detect cyberbullies. In Proceedings of the Canadian conference on artificial intelligence, Montreal, QC, USA, 6–9 May 2014; Springer: Berlin/Heidelberg, Germany, 2014; pp. 275–281.
194. Toxic Comment Classification Challenge. Available online: https://www.kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge/data (accessed on 5 April 2023).
195. Data and Code for the Study of Bullying. Available online: https://research.cs.wisc.edu/bullying/data.html (accessed on 5 April 2023).
196. Rafiq, R.I.; Hosseinmardi, H.; Han, R.; Lv, Q.; Mishra, S.; Mattson, S.A. Careful what you share in six seconds: Detecting cyberbullying instances in Vine. In Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), Paris, France, 25–28 August 2015; pp. 617–622.
197. Rafiq, R.I.; Hosseinmardi, H.; Mattson, S.A.; Han, R.; Lv, Q.; Mishra, S. Analysis and detection of labeled cyberbullying instances in Vine, a video-based social network. *Soc. Netw. Anal. Min.* **2016**, *6*, 88. [CrossRef]
198. Zohuri, B.; Moghaddam, M. Deep learning limitations and flaws. *Mod. Approaches Mater. Sci* **2020**, *2*, 241–250. [CrossRef]
199. Whang, S.; Lee, J.G. Data Collection and Quality Challenges for Deep Learning; In Proceedings of the VLDB Endowment, Online, 31 August–4 September 2020; VLDB Endowment: Los Angeles, CA, USA, 2020; Volume 13.
200. Pramod, A.; Naicker, H.S.; Tyagi, A.K. Machine learning and deep learning: Open issues and future research directions for the next 10 years. *Computational Analysis and Deep Learning for Medical Care: Principles, Methods, and Applications*; John Wiley & Sons, Ltd.: Chichester, UK, 2021; pp. 463–490.
201. Bae, H.; Jang, J.; Jung, D.; Jang, H.; Ha, H.; Lee, H.; Yoon, S. Security and privacy issues in deep learning. *arXiv* **2018**, arXiv:1807.11655.
202. Raisi, E.; Huang, B. Cyberbullying identification using participant-vocabulary consistency. *arXiv* **2016**, arXiv:1606.08084.

203. Ali, W.N.H.W.; Mohd, M.; Fauzi, F. Cyberbullying detection: An overview. In Proceedings of the 2018 Cyber Resilience Conference (CRC), Putrajaya, Malaysia, 13–15 November 2018; pp. 1–3.
204. Weischedel, R.M.; Bates, M. *Challenges in Natural Language Processing*; Cambridge University Press: Cambridge, UK, 2006.
205. Atoum, J.O. Cyberbullying Detection Through Sentiment Analysis. In Proceedings of the 2020 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 16–18 December 2020; pp. 292–297.
206. Nahar, V.; Unankard, S.; Li, X.; Pang, C. Sentiment analysis for effective detection of cyber bullying. In Proceedings of the Web Technologies and Applications: 14th Asia-Pacific Web Conference, APWeb 2012, Kunming, China, 11–13 April 2012; Proceedings 14; Springer: Berlin/Heidelberg, Germany, 2012; pp. 767–774.
207. Dani, H.; Li, J.; Liu, H. Sentiment informed cyberbullying detection in social media. In Proceedings of the Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2017, Skopje, Macedonia, 18–22 September 2017; Proceedings, Part I 10; Springer: Berlin/Heidelberg, Germany, 2017; pp. 52–67.