CYBERBULLYING DETECTION ON TIKTOK USING A DEEP LEARNING APPROACH

Razvan STOLERIU[1], Andrei NASCU[2], Florin POP[3,4,5,*]

Nowadays, the Internet has penetrated all aspects of the humans' lives. Despite providing many advantages and benefits, it can also produce a series of social problems such as spam, Internet crime, and Internet addiction. Unfortunately, negative information has become more common in modern society. In recent years, cyberbullying, one of the representatives of abnormal behavior on the Internet has been prominent in many countries across the globe. Different organizations and institutions tried to formulate and take some measures against it. In this paper, we propose a solution for cyberbullying detection in TikTok videos using a deep learning approach. We employ a Transformer-based model that operates on Convolutional Neural Network (CNN) feature maps. Moreover, we utilize the DenseNet121 model pre-trained on the ImageNet-1k dataset. We evaluated the accuracy of the model and observed we got an accuracy of 94.16%.

Keywords: Cyberbullying, Detection, video, TikTok, Deep learning.

1. Introduction

The best way people interact with each other, and exchange ideas is by physical communication. With the appearance of the Internet technologies that are used more and more in human lives, communication has shifted from face-to-face to online. Social media is the most preferred medium for discussing and interacting with others in virtual environments [1]. Nowadays, over 5 billion people are active users of social media [2]. The latter consists of multiple applications that people can use to create, share, and react to different posts. Most support various media file types like text, images, videos, and GIFs. Moreover, some of them have live-streaming features [3,5]. Entertainment represents the main reason people use social media platforms [4].

[1] Doctoral student, National University of Science and Technology Politehnica Bucharest, Romania; e-mail: raz.stoleriu@gmail.com
[2] Doctoral student, University of Craiova, Craiova, Romania; e-mail: andreinascu3@gmail.com
[3] Professor, National University of Science and Technology Politehnica Bucharest, Romania; e-mail: florin.pop@upb.ro
[4] Researcher, National Institute for Research and Development in Informatics (ICI), Bucharest, Romania; e-mail: florin.pop@ici.ro
[5] Scientist, Academy of Romanian Scientists, Bucharest, Romania
[*] Correspondence: florin.pop@upb.ro; Tel.: +40-723-243-958 (F.P.)

According to a recent study, teenagers spend more than an hour per day on social media [6]. Although there are many advantages it offers, there are also some possible risks [7]. One of them often encountered is cyberbullying or bullying in cyberspace. It occurs when people are aggressed or harassed by others through electronic means [8]. Another study describes that one in five adolescents aged 10 to 18 had a cyberbullying experience [9]. The factors that cause this behavior depend on the school environment, education system, interpersonal relationships, and cultural norms. For instance, herd mentality, cultural background differences, and the stress between students at school or amongst employees at work represent factors that increase the risk of bullying. It has been observed the cyberbullying phenomenon is more and more present among youth on social media, where they make bad jokes or harass somebody [10].

There are six roles the people involved in cyberbullying incidents can play: harasser, victim, assistant, defender, reinforcer, and accuser [11]. They are described in Table 1.

Table 1

Cyberbullying roles

Role	Depiction
Aggressor	The person who bullies across social media platforms.
Victim	An individual harassed by the aggressor.
Assistant	A person who aids the aggressor to bully.
Defender	An individual who tries to save the victim from the aggressor.
Reinforcer	A person who does not bully, but supports the aggressor in continuing to perform malicious actions.
Accuser	An individual who accuses someone of being the aggressor.

Social media via different platforms is omnipresent in human lives. It is used by various entities, organizations, companies, and institutions. A lot of data flows through them and is consumed by the final end-users.

TikTok is a social media application founded in China in 2016 and widely distributed all over the globe. According to [14], it is one of the most popular social networks worldwide. This platform supports just video files and gives users the possibility to create, share, and react to posts, among many other features. As of November 2020, there were reported 800 million users per month, while in 2019, 738 million users installed the app for the first time. This application has some usage restrictions. For instance, its users cannot be under 13, and the direct messaging feature is allowed just for people who are 16 or older. These are measures for preventing grooming. Having a lot of users and billions of videos that are loaded and watched, unfortunately, some of them have inappropriate content (e.g., smoking, drinking, rude language). As a result, the developers took some measurements by developing special filters for unsuitable content for young users.

TikTok is available for both iPhone and Android mobile devices. Humans that utilize it can create videos, share them with others, comment on them, or send

a reaction to them (e.g., Like). The high number of users in a very short period made TikTok one of the most popular social media platforms all over the globe. However, being widely used, it also attracted some criticism since there were discovered some issues concerning privacy and data protection [12]. Moreover, some studies highlight how this platform is used for spreading hate and engendering cyberbullying [13].

From our knowledge, at the moment of this writing, there are no other proposed solutions in the literature that treat the problem of cyberbullying content in the videos from TikTok. As this platform is one of the most popular across the world, a system for malicious content identification should be developed. This research proposes a cyberbullying detection solution for TikTok videos using a deep learning approach. We employ a Transformer-based model that operates on Convolutional Neural Network (CNN) feature maps. Moreover, we utilize the DenseNet121 model pre-trained on the ImageNet-1k dataset. We evaluated the model accuracy and observed we got 94.16%. This paper brings more contributions that are stated below.

- Created a TikTok videos dataset with both non-bullying and bullying categories.
- Proposed a Transformer-based model that operates on Convolutional Neural Network (CNN) feature maps. It takes the videos as input and classifies them into bullying or non-bullying.

The rest of the paper is structured as follows: in the *Introduction*, we present a short overview of the detection of cyberbullying content. In *Related Work*, we present a critical analysis of similar papers that deal with the detection of cyberbullying material. Moreover, in *Solution Design and Implementation*, we present our proposed solution, and in *Experimental Results and Analysis*, we present the experimental setup and analyze the obtained experimental results. Finally, in *Conclusions and Future Work*, we conclude the results of the solution and identify future research opportunities.

2. Related Work

Cyberbullying belongs to the category of malicious Internet behavior, and it can be of many types, such as trolling, hate speech, cyber-aggression, and offensive language [15,16]. This section discusses various scientific articles concerning cyberbullying detection. Table 2 presents some of the potential works for cyberbullying detection. In [17], the authors propose a solution for cyberbullying detection in Vine. Their dataset consists of 969 media sessions. Each of them contains both the video and the associated comments. Their dataset was initially larger, but the authors selected the videos with at least 15 comments. CrowdFlower was used for labeling. Several steps were followed for the data

processing stage, such as whitespace removal and handling of characters that are not recognized. The researchers utilized Python for sentiment analysis. They considered a comment related to bullying if it contained at least one word that belongs to that category. To better identify the bullying words, the authors have used a dictionary of negative terms. For classification, they employed several ML algorithms. The best results were obtained with AdaBoost: 89% accuracy, 90% precision, and 88% recall.

The authors of [18] developed a solution for cyberbullying detection in videos from Vine. Their dataset contains 969 media sessions. Each of them consists of a video and the associated comments. All the videos in the dataset have at least 15 comments. The researchers used CrowdFlower for labeling. Each media session was reviewed by five different persons. The authors determined four types of features that were considered when building the classification models. They are related to comments, videos, the profile of the person, and N-grams. To extract the comment features, they used sentiment analysis techniques. As regards the features from videos, they looked at the number of associated comments and how many people watched and liked the video. The features for the person's profile refer to the number of humans that follow and are followed by that individual and how many videos were posted from that profile. Several Machine Learning algorithms have been employed for classification. The highest accuracy, 76.39%, was obtained by the AdaBoost classifier.

In [19], the authors propose a multimodal cyberbullying detection solution for media sessions on Vine. Their system considers both the text from comments and the visual information from the frames of a video. Their dataset consists of 839 media sessions. 269 files belong to the bullying category, while the rest are part of non-bullying. Each media session has one video and 15 comments. The authors considered an architecture based on Transformer Encoder for text processing. They employed the Residual-BiLSTM model for text classification. The authors leveraged a Recurrent Convolution Neural Network model for visual feature extraction. It extracts features from 30 frames of a video. The highest accuracy was obtained for the two models' fusion used in the text and video classification. The resulting model is Residual-BiLSTM-RCNN and has an F-measure of 0.75 for the bullying class.

The authors of [20] designed a solution for cyberbullying detection in Vine media sessions. They use the same dataset utilized by the authors of [18]. The researchers introduced two new components in their detection system: an incremental classifier and a dynamic priority scheduler. The former brings advantages in the scalability and efficiency of the system. Its use case can be highlighted in the following situation: if the features of n comments were already extracted, and other k comments are coming for processing and feature extraction, then this component will retain the already computed feature values and will

calculate and add the ones of the new k comments to the overall feature vector that consists of n+k comments, for classification. The other component, the dynamic priority scheduler monitors the media sessions with high priority and throws away the lower ones. The obtained results highlight the proposed solution is more than 200 times faster than AdaBoost and almost 50 times more rapid than the Logistic Regression classifier when classifying 50.000 media sessions.

In [21], the authors propose a solution for cyberbullying detection in Vine using multimodal features: text, audio, and video. They take the dataset used by the authors of [18]. After applying the necessary filters, they get 733 media sessions where 165 of which belong to the cyberbullying category. Each media session consists of a video and the associated comments. The researchers selected videos that had at least 15 comments. Some textual features they considered are the number of words, uppercase characters, and the prevalence of punctuation. The presence of nudity, gore, and drugs are a few of the visual features. The audio ones also look for content related to speech, music, and silence. The authors employed several Machine Learning algorithms for classification. The highest performance has been obtained with the Logistic Regression classifier: an accuracy of 0.814, a precision of 0.56, a recall of 0.904, and an F1-score of 0.691.

Table 2

Summary of potential works for cyberbullying detection

Study	Machine Learning algorithm(s)	Dataset type / Social Media platform	Best result
Rafiq et al. [17]	Random Forest, Support Vector Machine (SVM), AdaBoost, Extra Trees Classifier	Vine / Video and text	AdaBoost Accuracy - 89% Precision - 90%
Rafiq et al. [18]	Naive Bayes, Decision Tree, AdaBoost, Random Forest	Vine / Video and text	AdaBoost Accuracy - 76.39% Precision - 71.38%
Paul et al. [19]	ResBiLSTM-RCNN	Vine / Video and text	ResBiLSTM-RCNN Precision and Recall for bully - 75% Precision and Recall for non-bully - 87%
Rafiq et al. [20]	Logistic Regression	Vine / Video and text	Logistic Regression Precision - 71% Recall - 66% Time - 44.42 (s)
Soni and Singh [21]	K-Nearest Neighbors (KNN), Logistic Regression, Support Vector Machine (SVM), Random Forest, Gaussian Naive Bayes (GNB)	Vine / Video and text	Logistic Regression Accuracy - 80% Recall - 86.5%
Our proposed solution	Transformer-based model that operates on CNN feature maps	TikTok / Video	Transformer-based model Accuracy – 94.16%

3. Solution Design and Implementation

In this section, we describe the data we collected for this study first, and then, we present how we designed and implemented the proposed approach.

3.1. Data collection and labelling

The most watched TikTok videos belong to categories like entertainment, dance, and sports [31]. As a result, we gathered video files pertaining to those classes to have the highest user coverage. To create our dataset, we collected videos from TikTok that belong to the following categories: *Basketball*, *Football*, *Playing cello*, *Playing guitar*, *Shoot* and *Kick*. We have labeled the last two categories as bullying since they have content that may threaten, terrify, or worry somebody who receives and watches such a video. The remaining categories were labeled as non-bullying. We have manually downloaded all the videos using the SnapTik platform [22]. All the videos in our dataset have between 5.5 and 6.5 seconds. If they were initially larger, we split them into media files of that range of lengths by using the Clipchamp website [25]. Table 3 presents the number of videos in each of the above-mentioned categories.

Table 3

The number of videos for each category

Category	No. of media files
Basketball	100
Football	102
Playing cello	100
Playing guitar	103
Shoot	103
Kick	100

Table 4 describes the number of media files for each labeled class.

Table 4

The number of media files for each class

Class	No. of media files
Non-bullying	405
Bullying	203
Total	608

80% of the data was used for training, while the rest was utilized for testing purposes. Figures 1 and 2 present some videos with non-bullying content.

Fig. 1. Video with non-bullying content (Basketball).

Fig. 2. Video with non-bullying content (Playing cello).

Figures 3 and 4 highlight some examples of media files with bullying content.

Fig. 3. Video with bullying content (Kick).

Fig. 4. Video with bullying content (Shoot).

3.2. Data pre-processing

The data pre-processing steps are described below:

1. We load each video file using the OpenCV library and extract up to 30 frames from each video. We know each video has at least 150 frames since all videos have over 5 seconds.

2. Then, we crop from each frame a central square whose side length is 500 pixels.

3. As OpenCV reads images in BGR format, we convert the loaded images to the much more popular RGB format.

After all the above steps are done, the data is sent to the DenseNet-121 model for feature extraction.

3.3. Overview of proposed approach

Our deep learning-based approach consists of the following two main processes: feature extraction and classification. Concerning the feature extraction process, we first split each video into frames. Then, every frame is taken by our system and treated as an image *I*. A kernel *K* which is like a filter, is then applied for the feature extraction process. It goes through the entire image from left to right. The result of the convolution product between image *I* and kernel *K* is a feature map *F*. It is defined by the relation below [28]:

$$F(i,j) = I(i,j) * K(i,j) = \sum_x \sum_y I(x,y)K(i-x,j-y)$$

The above equation can also be written with integrals as:

$$F(i,j) = I(i,j) * K(i,j) = \int \int I(x,y)K(i-x,j-y)\, dxdy$$

Since filters focus more on the image center, padding is employed where rows and columns are filled with zeros. Afterward, the *ReLU* activation function is utilized. It has an important contribution in producing non-linearity so that the model can establish a relationship between the different values from the input. The formula is:

$$ReLU[z] = max(0, z).$$

The pooling layer that follows in the scheme is applied. It is used to reduce the dimension of the future map by highlighting and taking just the dominant features from the image. So, a matrix *P* is obtained from the pooling layer with the help of a pooling function that takes the value computed from the layer of convolution as an argument. It is defined below:

$$P = f_p(Convolution(I, K))$$

The convolution and pooling layers were used for the image's feature extraction process. Further, the classification process starts, and the result of the previous steps, which is a vector, is transmitted as input to the fully connected layer for classification. At this step, an activation function is applied, where different weights are combined with pooling values, and a bias is finally added. The math formula is:

$$result = activation(pooling\_values * weights + bias)$$

Ultimately, the probability of each class is obtained in a vector, and the classification can be accomplished. The activation function we used for the last layer is *softmax*. Its main role is to ensure the sum of all probabilities from that layer is 1.

Figure 5 presents the system's general architecture.

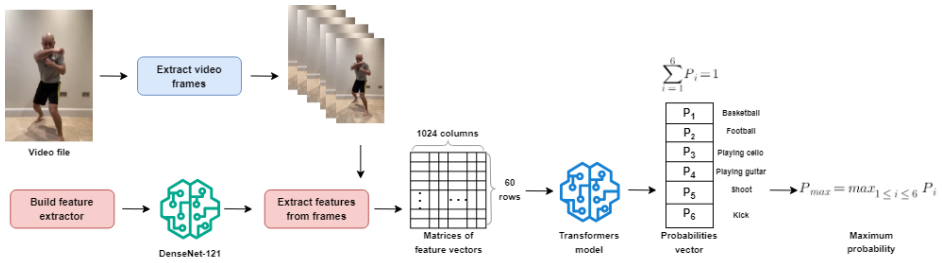

Fig. 5. System's general architecture.

After loading the frames of the videos, we employ a convolutional neural network (CNN) to extract the features. These features represent how we codify each frame to a vector in a vectorial space. Here, similar frames are grouped closer. To achieve this task, we have chosen a DenseNet architecture, which stands for Densely Connected convolutional network and is part of the Keras.applications module in Python. It was designed to solve the problem of vanishing gradients that appear in deep CNNs (they have a lot of layers) [26]. In our case, the input for the DenseNet model, namely DenseNet121, are images of size 500x500, and the output for each image is a 1024 feature vector. The latter will be the input for the transformer model. To obtain the output of the CNN, we didn't include the classifier, and we only used the feature extractor. The number associated with the model denotes how many layers the model has, in this case, 121 layers. These layers are commonly grouped into the following two categories: (1) Dense blocks composed of several convolutional layers that relate to all the other subsequent layers from the same block; (2) Transition layers which are the layers between two dense blocks. They perform operations of convolution and pooling to reduce the dimensionality of the image.

After obtaining the feature data from DenseNet121, we employed a transformer architecture to create our bullying detector model. The transformer model is an architecture developed by Google that relies on the attention mechanism to determine global dependencies between input and output without using recurrence like RNNs or LSTM that are slow. Transformers are state-of-the-art in multiple domains of artificial intelligence, such as language translation and question management chatbots. They are used in applications like GPT Chat 3. Initially, transformers were employed for translation where it needed two components since the model had to analyze two sequences: 1. The original sentence that was processed by the encoder; 2. The translated sequence which was processed by the decoder.

In our implementation of the transformer model, we don't have two sequences because the input, which is a sequence of frames transformed into one of the embedded feature vectors by the DenseNet121, is mapped with a label from the set {Basketball, Football, Playing cello, Playing guitar, Shoot and Kick} as output,

and not with another sequence of embedded features vectors. Since we only have one sequential data, to properly function, the model no longer needs the second part, the decoder. Concerning the implementation, we create two classes: one that deals with the Positional Encoding and another that creates the transformer model. The transformer does not sequence the data. Instead, it uses parallelization to speed up the process of training. Therefore, since there is no sequential structure, we use a mechanism employed by the Positional Encoding class. It informs the network about the position of each frame or vector of features returned by the DenseNet121 network [27]. The parallelization is created by grouping the feature vectors of a video into matrices that have as many rows as the video's sequence length, in our case, 60, and as many columns as the feature vector's size, in our case, 1024. Concerning the implementation of the second class, the transformer encoder, we used the Multi-Head Attention layer with a dropout level of 0.3 so as not to reach an overfitting situation. The number of heads was 1. The feedforward part of the encoder was made using densely connected layers with an activation function of Gaussian Error Linear Units (GELU) to add a non-linearity component to the model. Before and after the feedforward subcomponent of the Encoder, we added a normalization layer. Finally, after the Encoder, we added three more layers: a GlobalMaxPooling1D for our sequential output data, a Dropout to further try to avoid overfitting the architecture, and a Dense with six neurons, one for each class.

4. Experimental Results and Analysis

In this section, we discuss various results obtained while classifying the videos into bullying and other non-bullying classes. The proposed approach employs a Transformer-based model that operates on Convolutional Neural Network (CNN) feature maps. It takes the video file as input, prepares it, extracts the features set, and classifies it. Our dataset has 405 non-bullying media files and 203 bullying ones. We took 80% of the samples for training, and the rest were utilized for testing. We used accuracy as a performance metric to evaluate the model.

4.1. Experimental setup

The program we created for video classification using a deep learning approach is based on the following code from GitHub [29]. The contributions and improvements we brought to the code are detailed below:

- We have implemented a functionality at the beginning of the program that randomly chooses videos from the dataset and puts them in the training and test directories with a ratio of 80% for the former and 20% for the latter.

- We have changed the maximum number of frames taken from a video (i.e., the *MAX_SEQ_LENGTH* parameter). Initially, it was 20, but we increased it to 30.
- We have increased the image size (i.e., the *IMG_SIZE* parameter) from a square with a side of 128 pixels to one of 500.
- We have increased the number of epochs (i.e., the *EPOCHS* parameter) from 5 to 20.

Our program's code is available on GitHub [30]. The pseudocode of our solution is displayed in Algorithm 1.

```
1.  MAX_SEQ_LENGTH = 30
2.  NUM_FEATURES = 1024
3.  IMG_SIZE = 500
4.  featureExtractor = DenseNet121()
5.  trainData = [] //initialize an empty list
6.  trainLabels = [] //initialize an empty list
7.  for each videoPath in videoPaths:
8.  video= loadVideo(videoPath)
9.  trainLabel.append(extractName(videoPath))
10. frames = [] //initialize empty lists
11. videoFeatures = []          //initialize empty lists
12. for each frame in video:
            frames.append(cropCenterImage(IMG_SIZE, IMG_SIZE))
            if(length(frames) == MAX_SEQ_LENGTH)
                        break
13. if(length(frames) < MAX_SEQ_LENGTH)
            frames.concatenate(blackFrame,MAX_SEQ_LENGTH-
            length(frames))
14. for each frame in frames:
            videoFeatures.append(featureExtractor(frame))
15. trainData.append(videoFeatures)
16. model = PositionalEmbedding(MAX_SEQ_LENGTH, NUM_FEATURES)
    + TransormerEncoder(NUM_FEATURES)
17. model.fit(trainData,trainLabel)
```

Algorithm 1. The pseudocode of our solution.

For the experimental setup, we employed a virtual machine hosted on a physical device with the specifications described in Table 5. On the Windows virtual machine, we set up PyCharm IDE [23]. There, we developed our system based on a deep learning model. We used TensorFlow and Keras, as other authors utilize in their implementations [24]. TensorFlow is a platform for developing and training deep neural networks (NN). Keras is leveraged to implement NN and is written in Python. To capture the frames of a video, we used the *VideoCapture()* method from OpenCV.

Table 5

System properties

System type	Operating system	Architecture	CPU	Memory
Virtual Machine	Win. 10	64-bit	8 cores	8 GB RAM
Host	Win. 10	64-bit	Intel-i9	32 GB RAM

4.2. Classification results

We proposed a system for cyberbullying detection in TikTok videos using a deep-learning approach. Our solution employs a Transformer-based model that operates on Convolutional Neural Network (CNN) feature maps. Table 6 describes the experiments we conducted with our deep learning-based model and the obtained accuracy. The *MAX_SEQ_LENGTH* field refers to the maximum number of frames we take from each video. We pad the video with zeros when a video frame count is lower than this field value. The *NUM_FEATURES* field represents the number of features the CNN model (i.e., DenseNet-121) extracts from each video. The *IMG_SIZE* field refers to the matrix dimensions that are cropped from the center of each frame. In our case, that will be 500x500 pixels.

Table 6

Classification results

Keras Application	MAX_SEQ_LENGTH	NUM_FEATURES	IMG_SIZE	EPOCHS	Accuracy
DenseNet-121	30	1024	500	20	94.16%

The confusion matrix represents another way we can measure the system's performance. Figure 6 presents the confusion matrix for the proposed solution where all classes in the dataset were implied. In this case, we can observe two basketball videos were classified as *football*, one football video was classified as *basketball*, one kick video was classified as *playingGuitar*, while the other two from this category were classified as *shoot*. Finally, one shoot video was classified as *football*.

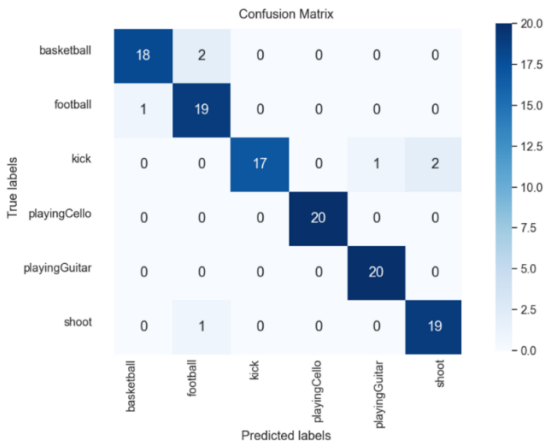

Fig. 6. Confusion matrix

When testing our solution against a video whose content is about shooting, we got the following remarkable results depicted in Table 8. Figure 7 shows one frame from that video.

Fig. 7. Test video with non-bullying content (Shoot).

Table 8

Test video results (Shoot)

Class	Probability
Shoot	99.99%
Football	0.01%
Basketball	0.00%
Kick	0.00%
Playing cello	0.00%
Playing guitar	0.00%

5. Conclusions and Future Work

In this paper, we propose a novel cyberbullying detection solution for TikTok videos using a deep learning-based model. From our knowledge, at the moment of this writing, there is no other solution in the literature that detects cyberbullying content in media files from TikTok. This research brings the following important contributions. Firstly, we create a dataset of videos from TikTok with both bullying and non-bullying content. We manually downloaded them with the help of the SnapTik platform. All the videos in our dataset have between 5.5 and 6.5 seconds. If they were initially larger, we split them into media files of that range of lengths by using the Clipchamp website. Secondly, we create a system that employs a Transformer-based model for video classification, which operates on Convolutional Neural Network (CNN) feature maps. We evaluated our model against the created dataset and got an accuracy of 94.16%.

We can conclude that our solution is one step forward in the research and development of security systems, especially for the mitigation of cyberbullying attacks via videos from TikTok. Our design outcomes most classical video classification systems since we use a Transformer-based model that operates on CNN feature maps. Possible use cases that might benefit from our proposal include online bullying or harassment in private and public environments such as schools,

workplaces, and playgrounds. This work brings important benefits to Internet users by offering advanced capacities for detecting cyberbullying in media files from TikTok.

Our proposed system can be integrated into browsers or mobile apps to detect cyberbullying in TikTok videos and make a safer Internet for users. Once our solution is deployed, we can employ a transformer-based federated learning approach where each user's model will evolve independently based on the access history of videos from the TikTok platform. This approach comes in the context of giving the best performance for each individual and not using a holistic approach that uses general and universal available data. After an established period, the new weights of each model are sent back to us, the solution's developers, and integrated into a new model that will be deployed to users through software updates.

We propose to implement the following enhancements for our solution in the future. First, we intend to expand the database with new classes from the bullying category (e.g., sexual harassment, intimidation) and the non-bullying one (e.g., water polo, running, horse riding). Second, we propose training several Machine Learning models for action detection in videos. In this case, the final decision is based on a metric that will consider the accuracies obtained in the training process for each model. Third, we propose the use of three detection models: one that analyzes the action, one that handles the sound, and another that processes the text in the video. The final decision will be represented by a weighted average that will consider the action in the video, the duration of the sound part (time in seconds), the quantity (number of words), importance (keywords considered to be vulgar) and font size of the text within the video.

Acknowledgment

The work presented in this paper was supported by the Core Program within the National Research Development and Innovation Plan 2022-2027, financed by Ministry of Research, Innovation and Digitalization of Romania, project no 23380601. This work is partially supported by the Research Grant no. 94/11.10.2023 Modern Distributed Platform for Educational Applications in Cloud Edge Continuum Environments GNAC-ARUT-2023.

We would also like to thank the reviewers for their time and expertise, constructive comments and valuable insight.

R E F E R E N C E S

[1] *Zulian Fikry*, *Gumi Rizal*, and *Willy Sintia*, “The Impact of Empathy towards Cyberbullying Behavior among Adolescents Who Accessed TikTok in Indonesia,” International Conference of Mental Health, January 1, 2021.

[2] *Simon Kemp*, "Digital 2024: Global Overview Report — DataReportal – Global Digital Insights," DataReportal – Global Digital Insights, January 31, 2024, https://datareportal.com/reports/digital-2024-global-overview-report.
[3] *Andreas M. Kaplan* and *Michael Haenlein*, "Users of the World, Unite! The Challenges and Opportunities of Social Media," Business Horizons 53, no. 1 (January 1, 2010): 59–68.
[4] *Leonard Reinecke, Peter Vorderer*, and *Katharina Knop*, "Entertainment 2.0? The Role of Intrinsic and Extrinsic Need Satisfaction for the Enjoyment of Facebook Use," Journal of Communication 64, no. 3 (May 19, 2014): 417–38, https://doi.org/10.1111/jcom.12099.
[5] *O'Dea, Bridianne, and Andrew Campbell*, "Online Social Networking and the Experience of Cyber-Bullying". Studies in Health Technology and Informatics 181 (2012): 212–17.
[6] *Marja Leonhardt* and *Stian Overå*, "Are There Differences in Video Gaming and Use of Social Media Among Boys and Girls?—A Mixed Methods Approach," International Journal of Environmental Research and Public Health/International Journal of Environmental Research and Public Health 18, no. 11 (June 4, 2021): 6085.
[7] *George, Madeleine*. "The Importance of Social Media Content for Teens' Risks for Self-harm." Journal of Adolescent Health 65, no. 1 (July 1, 2019): 9–10.
[8] *N. E. Willard*, "Cyberbullying and Cyberthreats: Responding to the Challenge of Online Social Aggression, Threats, and Distress," Research Press, 2007
[9] *P. J. Parks*, "Cyberbullying (Compact Research: The Internet)," ReferencePoint Press, 2012.
[10] *Zhong, Jinping, Yunxiang Zheng, Xingyun Huang, Dengxian Mo, Jiaxin Gong, Mingyi Li*, and *Jingxiu Huang*. "Study of the Influencing Factors of Cyberbullying Among Chinese College Students Incorporated With Digital Citizenship: From the Perspective of Individual Students." Frontiers in Psychology 12 (March 4, 2021).
[11] *Wan, Ali Wan Noor Hamiza, Fauzi Fariza*, and *Mohd Masnizah*. "Identification of Profane Words in Cyberbullying Incidents Within Social Networks." Journal of Information Science Theory and Practice 9, no. 1 (January 1, 2021): 24–34.
[12] *Montag, Christian, Haibo Yang*, and *Jon D. Elhai*. "On The Psychology of TikTok Use: A First Glimpse From Empirical Findings." Frontiers in Public Health 9 (March 16, 2021).
[13] *Weimann, Gabriel*, and *Natalie Masri*. "Research Note: Spreading Hate on TikTok." Studies in Conflict and Terrorism 46, no. 5 (June 19, 2020): 752–65.
[14] *S. J. Dixon.* (2024, Feb. 2). Biggest social media platforms 2024 — Statista [Online]. Available: https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/ (accessed Feb. 27, 2024).
[15] *Nafis, Nazia, Diptesh Kanojia, Naveen Saini*, and *Rudra Murthy*. "Towards Safer Communities: Detecting Aggression and Offensive Language in Code-Mixed Tweets to Combat Cyberbullying". In The 7th Workshop on Online Abuse and Harms (WOAH), 29–41. Toronto, Canada: Association for Computational Linguistics, 2023.
[16] *Febriana, Trisna*, and *Arif Budiarto*. "Twitter Dataset for Hate Speech and Cyberbullying Detection in Indonesian Language". In 2019 International Conference on Information Management and Technology (ICIMTech), 379–82. Jakarta/Bali, Indonesia: IEEE, 2019.
[17] *Rahat Ibn Rafiq* et al., "Analysis and Detection of Labeled Cyberbullying Instances in Vine, a Video-Based Social Network", Social Network Analysis and Mining 6, no. 1 (December 2016): 88.
[18] *Rahat Ibn Rafiq* et al., '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 2015 (ASONAM '15: Advances in Social Networks Analysis and Mining 2015, Paris France: ACM, 2015), 617–22.
[19] *Paul, Sayanta, Sriparna Saha*, and *Mohammed Hasanuzzaman*. "Identification of cyberbullying: A deep learning based multimodal approach." Multimedia Tools and Applications 81, no. 19 (September 10, 2020): 26989–8.

[20] *Rahat Ibn Rafiq* et al., 'Scalable and Timely Detection of Cyberbullying in Online Social Networks', in Proceedings of the 33rd Annual ACM Symposium on Applied Computing (SAC 2018: Symposium on Applied Computing, Pau France: ACM, 2018), 1738–47.

[21] *Soni, Devin*, and *Vivek K. Singh*. "See No Evil, Hear No Evil." Proceedings of the ACM on Human-Computer Interaction 2, no. CSCW (November 1, 2018): 1–26.

[22] snaptik.app. "Tiktok Downloader - Download Video tiktok Without Watermark - SnapTik." Available online: https://snaptik.app/, Accessed: Mar. 04, 2024.

[23] www.jetbrains.com. "PyCharm: the Python IDE for Professional Developers by JetBrains." Available online: https://www.jetbrains.com/pycharm/, Accessed: Mar. 05, 2024.

[24] *Rukiye Savran Kızıltepe, John Q. Gan*, and *Juan José Escobar*, "A Novel Keyframe Extraction Method for Video Classification Using Deep Neural Networks," Neural Computing & Applications 35, no. 34 (August 2, 2021): 24513–24, https://doi.org/10.1007/s00521-021-06322-x.

[25] clipchamp.com. "Home | Clipchamp." Available online: https://app.clipchamp.com/, Accessed: Mar. 06, 2024.

[26] *S. Das*, "Implementing DenseNet-121 in PyTorch: A Step-by-Step Guide — by Shuvam Das — deepkapha notes — Medium." medium.com. https://medium.com/deepkapha-notes/implementing-densenet-121-in-pytorch-a-step-by-step-guide-c0c2625c2a60 (accessed Apr. 04, 2024).

[27] *Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin*. "Attention is all you need." *Advances in neural information processing systems* 30 (2017).

[28] *Xin, Mingyuan, and Yong Wang*. "Research on image classification model based on deep convolution neural network." *EURASIP Journal on Image and Video Processing*2019, no. 1 (2019): 1-11.

[29] github.com. "keras-io/examples/vision/video transformers.py at master · keras-team/keras-io · GitHub." Available online: https://github.com/keras-team/keras-io/blob/master/examples/vision/video transformers.py, Accessed: Apr. 29, 2024

[30] github.com. "Razvan96/TikTok Cyberbullying detection: A deep learning-based solution for cyberbullying detection in TikTok videos." Available online: https://github.com/Razvan96/TikTok Cyberbullying detection, Accessed: Apr. 30, 2024.

[31] thesocialshepherd.com. "25 Essential TikTok Statistics You Need to Know in 2024." Available online: https://thesocialshepherd.com/blog/tiktok-statistics, Accessed: Sep. 11, 2024.