KidGuardScan: A System for Identifying Harmful TikTok Content for Child Safety

Vo Hoang An[1,2,3], Tran Thai Hoa[1,2,3], Tran Quang Duy[1,2,3], Tran Thi My Duyen[1,2,3], and Do Trong Hop[1,2,4]

[1] Faculty of Information Science and Engineering
[2] University of Information Technology, VNU-HCM, Vietnam
[3] {21520555,21522082,21522013,21522017}@gm.uit.edu.vn
[4] hopdt@uit.edu.vn

Abstract

TikTok, a short-form video sharing platform, has become a global phenomenon with immense popularity among children. However, the platform also presents potential dangers, as some content can be harmful to a child's well-being. In study we introduces KidGuardScan, a system designed to identify harmful TikTok content and promote child safety. KidGuardScan leverages a two-phase approach to achieve its goal. In the offline phase, a video classification model is trained using a custom dataset. This model is then integrated into the online phase, where Apache Kafka and Apache Spark are employed for real-time video streaming and classification. Videos streamed through the system are analyzed by the pre-trained model, allowing for the detection and classification of potentially harmful content. To facilitate this research, we developed TikHarm, a novel dataset comprising 3,948 videos categorized into Safe, Adult Content, Harmful Content, and Suicide. Our system demonstrated robust performance, achieving a classification accuracy of 87.39% with an F1 score. This system addresses the growing concern surrounding the negative impacts of TikTok on children. By proactively identifying harmful content, KidGuardScan can help create a safer online environment for young users.

1 Introduction

"Harmful Video" is a term commonly used to describe videos containing content detrimental to viewers, particularly children and adolescents. This content may include violence, incitement, pornography, offensive speech, suicide, and horror. Such videos often feature violent scenes, inappropriate sexual content, suicide, or self-harm, negatively impacting viewers' mental health. Additionally, they may involve speech or actions that insult nationality, religion, or gender, or use vulgar language.

Harmful video content can have numerous adverse social consequences. These videos can induce stress, anxiety, or psychological harm, especially in children and adolescents. They may also incite

violent or negative behaviors, spread misinformation, cause misunderstandings, and erode trust in reliable news sources.

Currently, harmful videos remain a significant issue on the Internet, particularly on short video platforms like TikTok. In the first quarter of 2024, each short video on TikTok averaged 18,173 views [1], significantly surpassing other platforms like Facebook and Instagram, with a large number of teenagers using the platform. Videos containing violent, hate-inciting, and terrorist content persist, negatively impacting viewers' mental health and promoting societal violence. The spread and accessibility of pornographic content and child abuse videos can severely harm children's mental health. Furthermore, videos with false or misleading information cause informational disorder and affect viewers' trust.

Online platforms, particularly social media, facilitate the rapid dissemination of harmful videos, making it challenging to control and prevent such content. To address this issue, we propose the development of a dataset for classifying harmful videos targeted at children and the creation of a system to limit harmful trends for children.

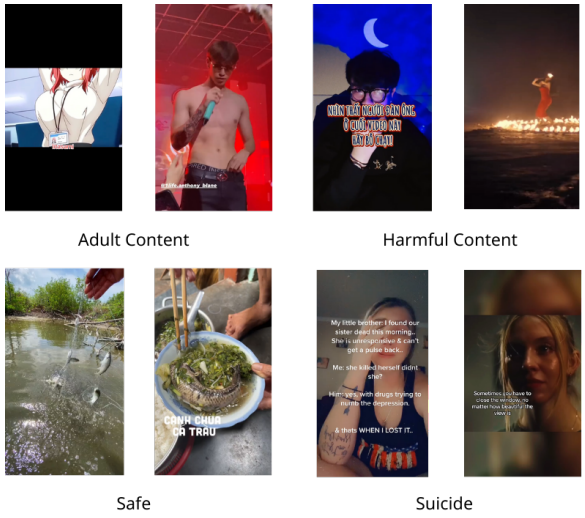

Figure 1: An examples of the Harmful detect task in **TikHarm** dataset.

To bridge this gap, this study introduce the TikHarm dataset and the KidGuardScan system. TikHarm provides a comprehensive resource for training and evaluating harmful content detection models, while KidGuardScan offers a real-time solution for identifying and mitigating harmful TikTok trends.

1. **Dataset Construction:** We developed **TikHarm**, a dataset comprising 3,948 videos categorized into Safe, Adult Content, Harmful Content, and Suicide (See Figure 1). Each video was meticulously verified to ensure a solid foundation for both research and practical applications.

2. **Model Testing:** We fine-tuned several state-of-the-art video processing models on the **TikHarm** dataset to evaluate their effectiveness in classifying harmful content for children.

[1] https://www.statista.com/statistics

3. **System Development:** We built KidGuardScan, a system based on Apache Kafka and Apache Spark, used for real-time streaming and classification of videos.

2 Related works

2.1 Task: Detecting Harmful Content for Children in Short Video

Detecting Harmful Content for children utilizes deep learning algorithms to identify and filter inappropriate segments. This process employs advanced neural networks for analyzing video data, recognizing elements such as violence, explicit material, and hate speech. The methodology necessitates extensive annotated datasets, robust feature extraction techniques, and precise classification models to effectively distinguish between benign and potentially harmful content across diverse video contexts.

Input: Input for detecting harmful content for children in video consists of video clips or frames that need to be analyzed.

Output: The output of the system is a classification label for each analyzed video clip or frame. The classification labels are:

- **Safe:** The content is appropriate and does not contain any harmful elements.
- **Adult Content:** The content contains explicit material not suitable for children.
- **Harmful Content:** The content includes violent, inciting, or otherwise damaging material.
- **Suicide:** The content depicts or promotes self-harm or suicide.

2.2 Previous Datasets

In this section, I present the characteristics and details of various datasets used in harmful content detection (Shown in Table 1).

Table 1: Comparative overview of typical open-domain harmful detect datasets. *Potentially occational phrases or short conversations in additional languages.

Datasets	Samples	Feature		Languages	Children
		Video	Text		
VidHarm [1]	3,589	✔	✗	37 Languages	✗
XD-Violence [2]	4,754	✔	✗	English*	✗
YouTube for Kids [3]	4,797	✔	✔	English	✔
HateMM [4]	1,083	✔	✔	English*	✗
TikHarm (Ours)	3,948	✔	✗	Vietnamese + English	✔

XD-Violence [2] includes 4,754 untrimmed videos totaling 217 hours, with both audio signals and weak labels from movies and YouTube. It covers six violence types: abuse, car accidents, explosions, fighting, riots, and shootings, supporting multimodal violence detection under weak supervision.

VidHarm [1] consists of 3,589 non-overlapping 10-second clips from 351 film trailers, labeled for harmful content by two professional classifiers. It includes original clips, converted versions with images, audio files, and spectrograms, supporting multimodal harmful content detection research.

HateMM [4] offers 43 hours of BitChute videos, manually annotated as hate or non-hate, with specific frame spans explaining the labels. Using hate lexicons, the authors observed various cues in images and audio of hateful videos.

YouTube for Kids [3] includes 12,097 unique seed videos, gathered through keyword extraction from subreddits and random video sampling via the YouTube Data API. It also provides metadata such as titles, descriptions, thumbnails, tags, and video statistics, facilitating the analysis and classification of disturbing videos targeting children.

Previous datasets primarily focus on a broad range of content domains. In contrast, YouTube for Kids [3] and **TikHarm** are centered on the children's content domain, with **TikHarm** being the first Vietnamese dataset dedicated to harmful content detection. Compared to earlier datasets, **TikHarm** provides detailed annotations for harmful content, including labels such as Safe, Adult Content, Harmful Content, and a particularly sensitive label for children, Suicide.

2.3 Previous Methods

Recent advancements in harmful video detection leverage deep learning to improve the identification of inappropriate content. Notably, KidsGUARD employs an LSTM-based autoencoder with a VGG16 CNN to detect child unsafe content with high recall and precision, even when such content is sparsely distributed [5]. Similarly, Fudan-Huawei's approach in MediaEval 2015 used a two-stream CNN and LSTM models to capture both static and dynamic features for violent scene detection, achieving commendable performance [6]. Fudan-NJUST's method in MediaEval 2014 utilized a regularized DNN to fuse visual and audio features, outperforming traditional SVMs [7]. ACORDE's architecture integrates CNNs and LSTMs to detect adult content, significantly reducing false positives and negatives [8]. These approaches underscore the potential of deep learning in enhancing the accuracy and reliability of harmful video detection.

To further address the harmful video detection problem, encoder-decoder architectures offer significant potential, as demonstrated by recent advancements. VideoMAE [9] and its scaled variant, VideoMAE-V2 [10], highlight the efficacy of masked autoencoders in self-supervised video pretraining, achieving impressive results even on small datasets by leveraging high masking ratios and dual masking strategies. The TimeSformer [11], which relies solely on space-time attention without convolution, underscores the benefits of divided attention for high-accuracy video classification, outperforming traditional 3D convolutional networks in both efficiency and performance. These methods collectively underscore the promise of encoder-decoder architectures in enhancing the detection of harmful video content through improved spatiotemporal feature learning and scalability. Consequently, our approach leverages VideoMAE [9] and TimeSformer [11] to capitalize on these advancements, aiming to improve the robustness and accuracy of harmful video detection.

3 Dataset

3.1 Dataset Collection

The primary objective of our dataset construction is to gather a comprehensive and representative sample of TikTok[2] videos, with a focus on content accessible to children. To achieve this, we utilized unofficial TikTok's API[3] to scrape videos from popular and trending hashtags, which are more likely to be seen by a younger audience. Additionally, we employed keyword-based searches targeting terms frequently associated with content children might encounter, such as "sad", "challenge", "adult", and "kids".

Given the vast amount of content on TikTok, we implemented a stratified sampling approach to ensure diversity in our dataset, capturing a wide range of video types and themes. We also took care to balance the dataset across various time periods to account for temporal trends and shifts in content popularity.

To enhance the relevance and quality of the collected data, we applied filters to exclude videos with low engagement metrics, such as views, likes, and comments, under the assumption that highly engaging content is more likely to be seen by children. Furthermore, we included videos from accounts with substantial followings, as these influencers often have a significant impact on their young audience.

Once the data was collected, we anonymized all user information to adhere to ethical guidelines and privacy regulations. This step ensured that while the content of the videos was analyzed, the identities of the creators and viewers were protected. Our final dataset comprises thousands of videos, providing a robust foundation for subsequent labeling and analysis to detect harmful content effectively.

3.2 Dataset Labeling

The data labeling phase is crucial for the effectiveness of our harmful content detection system. We manually labeled the collected data to categorize each video into predefined categories: safe content, harmful content, adult content, and suicide. The harmful content category was further divided into subcategories such as violence, dangerous actions that children might imitate, and sexual content.

A team of trained annotators reviewed each video, ensuring that the labeling was consistent and accurate. To maintain high inter-rater reliability, we conducted multiple rounds of training and calibration sessions. Annotators were provided with detailed guidelines and examples for each category to minimize subjectivity and bias.

This rigorous labeling process resulted in a rich, annotated dataset that serves as the backbone for training and validating our machine learning models aimed at detecting harmful content on the TikTok platform.

[2] tiktok.com
[3] github.com/davidteather/TikTok-Api

3.3 Dataset Validation

After completing the main annotation phases, we implemented a strategie to ensure the quality and consistency of the dataset. Metric For Inter-Annotator Agreement: Fleiss Kappa is widely used to evaluate inter-annotator agreement (IAA) in several tasks and is considered a benchmark for such measurements [12]. Consequently, we utilized the Fleiss Kappa metric [13] to assess inter-annotator agreement, thus ensuring quality assurance in human annotation.

We randomly selected 10% of the claims (n = 400) from the labeled dataset, assigning them to a group of three annotators. These claims, originally authored by different individuals, were relabeled without revealing the existing annotations. The inter-rater agreement was then calculated using the Fleiss Kappa measure. We achieved an agreement level of 81.25%, indicative of a very high level of agreement among annotators, which confirms the high quality and reliability of our dataset.

3.4 Dataset Statistic

Table 2 shows that the number of samples for each label is fairly consistent, ranging from 977 samples (adult) to 997 samples (safe). However, the total duration for each label is uneven, with the safe label having the highest total duration at 18.1 hours, nearly twice that of the adult (9.84 hours) and harmful (9.88 hours) labels, and almost four times that of the suicide label (4.63 hours). This disparity suggests that while the number of samples is fairly uniform, the duration of suicide-labeled content is significantly lower, indicating that TikTok has effectively filtered out a substantial amount of suicide-related content, leaving fewer videos in this category. Consequently, the system's ability to detect or classify suicide-related content may not be as robust as for other labels due to the limited duration of training data. Additionally, the longer duration of safe-labeled videos may reflect that these videos often provide educational or informative content, which tends to be longer in length.

Table 2: Distribution of Video Samples and Duration by Label Category in the TikHarm Dataset.

	Duration				
	samples	min	max	avg	total (h)
Safe	997	5,04	568,8	65,36	18,1
Adult	977	1,95	600	36,25	9,84
Harmful	990	4,8	600	35,92	9,88
Suicide	984	3,88	181,23	16,96	4,63

The table 3 illustrates that the number of samples for the train, dev, and test sets are 2762, 790, and 396, respectively. The average duration across the three sets is fairly consistent, with values of 38.71, 38.57, and 38.77, respectively. The total duration for the train set is 29.71 hours, the dev set is 4.24 hours, and the test set is 8.51 hours. This indicates that the data has been reasonably divided to ensure consistency and accuracy during model training and evaluation.

Table 3: Distribution of Video Samples and Duration by Data Split in the TikHarm Dataset.

	Duration				
	samples	min	max	avg	total (h)
Train	2762	3,88	600	38,71	29,71
Dev	790	5,04	600	38,57	4,24
Test	396	1,95	600	38,77	8,51

4 Methods

TimeSformer [11] a novel video classification architecture that employs only space-time self-attention, eliminating the need for convolutional layers. TimeSformer demonstrates state-of-the-art performance on various action recognition benchmarks, including Kinetics-400 and Kinetics-600, while being significantly faster to train and requiring fewer resources for inference. It also excels in long-term video modeling, demonstrating its capability for handling videos of over a minute in duration, a significant departure from conventional methods.

Despite its larger number of parameters, TimeSformer offers faster inference and training compared to 3D convolutional networks, such as SlowFast and I3D, making it suitable for large-scale learning scenarios. TimeSformer learns to attend to relevant regions in the video during classification, as demonstrated by visualizations of its attention patterns, suggesting its capability for spatiotemporal reasoning.

VideoMAE [9] is an extension of Masked Autoencoders (MAE) to video. The architecture of the model is very similar to that of a standard Vision Transformer, with a decoder on top for predicting pixel values for masked patches.

Videos are presented to the model as a sequence of fixed-size patches (resolution 16x16), which are linearly embedded. One also adds a [CLS] token to the beginning of a sequence to use it for classification tasks. One also adds fixed sinus/cosinus position embeddings before feeding the sequence to the layers of the Transformer encoder.

By pre-training the model, it learns an inner representation of videos that can then be used to extract features useful for downstream tasks: if you have a dataset of labeled videos for instance, you can train a standard classifier by placing a linear layer on top of the pre-trained encoder. One typically places a linear layer on top of the [CLS] token, as the last hidden state of this token can be seen as a representation of an entire video.

5 Experimental & Results

5.1 Results

In accordance with the configuration parameters delineated in Appendix A, we present our primary findings in Table 4. The efficacy of VideoMAE and TimeSformer models was assessed using two distinct clip durations: 1.13 seconds and 2.13 seconds.

Table 4: Performance metrics of VideoMAE and TimeSformer models across different clip durations.

Model	Time inference	Clip Duration	Precision	Recall	$F1_{score}$
VideoMAE	31.3s	1.13s	0.8646	0.8633	0.8625
	28.74s	2.13s	0.8648	0.8646	0.8638
TimeSformer	31.43s	1.13s	0.8712	0.8709	0.8700
	32.08s	2.13s	**0.8758**	**0.8747**	**0.8739**

For the VideoMAE model, inference times of 31.3 seconds and 28.74 seconds were observed, corresponding to F1 scores of 0.8625 and 0.8638, respectively. In contrast, the TimeSformer model exhibited inference times of 31.43 seconds and 32.08 seconds, yielding F1 scores of 0.8700 and 0.8739, respectively.

Comparative analysis reveals that the TimeSformer model consistently outperformed VideoMAE, demonstrating marginally superior precision, recall, and F1 scores across both clip duration settings. This suggests an enhanced overall performance of the TimeSformer model in the context of our experimental framework.

5.2 Dicussion

To gain deeper insights into the model's behavior, we conducted an error analysis using the confusion matrix of the best model (TimeSformer with 2.13-second clips) on the test set, as illustrated Figure 2. By analyzing the confusion matrix, we identify the error cases where the predicted labels are incorrect and perform error analysis, focusing on the cases where harmful labels (adult content, suicide, and harmful content) are classified as safe content. The analysis revealed several key patterns of misclassification, particularly in cases where harmful content (adult content, suicide, and harmful content) was erroneously labeled as safe. For more details, please refer to the Appendix B.

- **Confusion due to Overlap between Labels**: The model frequently confuses categories with overlapping visual elements, leading to harmful content being misclassified as safe. This confusion is exacerbated by the absence of contextual information like audio or text, which could provide additional cues for accurate classification.
- **Hidden or Ambiguous Actions**: Harmful actions that are partially obscured or visually ambiguous pose a challenge for the model, often leading to misclassification as safe content. The model's reliance on visual features alone limits its ability to interpret actions that are not fully visible or easily discernible.
- **Highly Imitative Actions**: Videos showing highly imitative actions can lead to misclassifications due to the potential for unforeseen harmful consequences. The model might classify these actions as safe if the dangerous outcomes are not immediately apparent.

These findings underscore the importance of incorporating multimodal information, such as audio and textual cues, to enhance the model's understanding of context and improve its classification

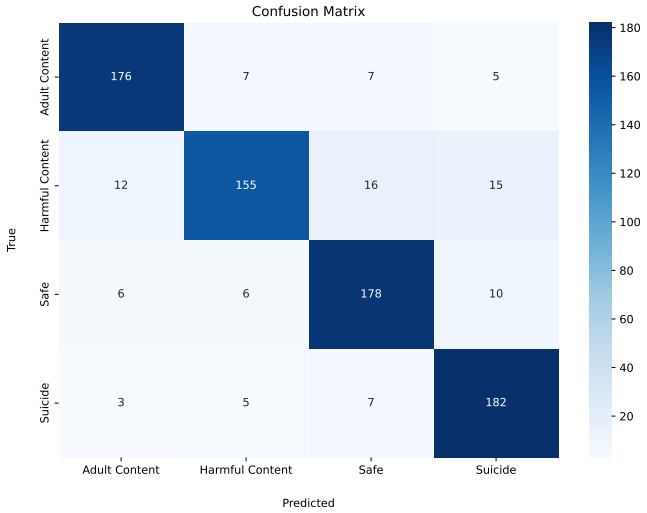

Figure 2: Confusion matrix of TimeSformer model on the test set.

accuracy. Additionally, refining the model's ability to detect subtle or hidden harmful actions and predict the potential consequences of seemingly innocuous actions would significantly bolster its effectiveness in safeguarding children from harmful content on TikTok. The promising performance of the TimeSformer model, coupled with the insights gained from the error analysis, lays the groundwork for the development of a robust and effective system for real-time TikTok video classification, as detailed in the subsequent section.

6 KidGuardScan proposed System

Our proposed system for TikTok video classification leverages trending hashtags and is structured to provide a scalable and efficient solution for real-time video classification, as illustrated in Figure 3. This system integrates advanced streaming, data processing, and deep learning techniques to provide a scalable and efficient solution for real-time TikTok video classification.

We utilize a dedicated server to fine-tune a classifier using the Hugging Face library and the TikHarm dataset. This server also facilitates future retraining to continually improve the model's accuracy in detecting harmful content through pre-training and transfer learning.

To deploy the system in a real-world environment, we collect and process data in real-time using a combination of advanced technologies. Trending hashtags and metadata are gathered through the Unofficial TikTok API and managed with Apache Airflow. Data is streamed via Apache Kafka and processed continuously with Spark Streaming. Videos are downloaded using the TikTokDownloader API, followed by preprocessing steps such as noise removal, normalization, and feature extraction. These preprocessed videos are then classified using the fine-tuned model. The metadata and videos, along with their predicted labels, are stored in a MongoDB database, as detailed in Appendix D.

To ensure ease of use for non-expert users, we designed an accessible user interface with Streamlit.

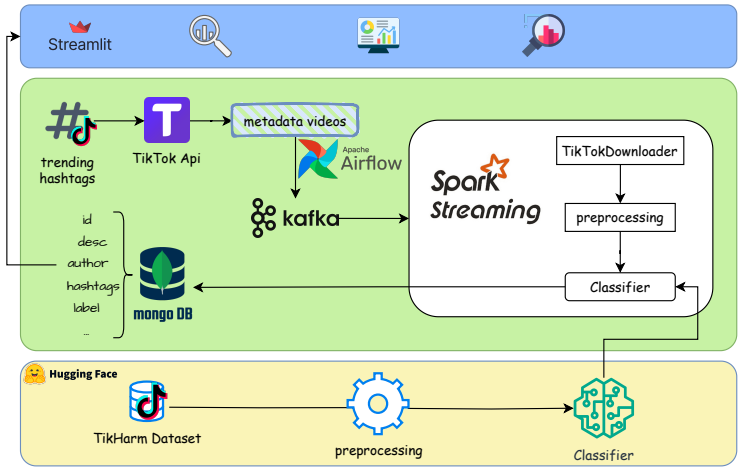

Figure 3: Architecture of KidGuardScan system.

This interface allows users to query the database, view classification results, and provide feedback. The feedback mechanism enables continuous improvement of the system's accuracy by incorporating new classification results into the MongoDB database based on user input. This user-centric approach ensures the system remains accurate and relevant, adapting to new trends and user needs seamlessly.

7 Conclusion & Future works

7.1 Conclusion

In this research, we introduced TikHarm, the first Vietnamese dataset specifically designed for classifying harmful content on TikTok aimed at children. This dataset encompasses 3,948 videos meticulously labeled into four distinct categories: Safe, Adult Content, Harmful Content, and Suicide. Our experiments, leveraging state-of-the-art video processing models like VideoMAE and TimeSformer, demonstrated the feasibility of automated harmful content detection on TikTok. Notably, the TimeSformer model, when processing 2.13 (second) video clips, achieved a commendable F1 score of 87.39% with an inference time of 32.08 seconds, indicating its potential for real-time application. The insights gained from our error analysis, particularly the identification of common misclassification patterns, underscore the need for incorporating multimodal information and refining the model's ability to detect subtle or hidden harmful actions.

Furthermore, we proposed KidGuardScan, a comprehensive framework for real-time TikTok video classification. This system leverages trending hashtags to identify potentially harmful content, utilizing Apache Kafka and Spark Streaming for efficient data processing. The classified videos, along with their metadata and predicted labels, are stored in a MongoDB database, facilitating user interaction and feedback through a Streamlit-based interface. This user feedback loop enables continuous improvement of the system's accuracy, ensuring its adaptability to evolving trends and user needs.

7.2 Future works

Building upon the promising results of this study and current limitations identified in 5.2, future work will focus on several key areas to enhance the system's effectiveness.

We aim to incorporate multimodal data, including audio and textual information, to address the limitations of relying solely on visual features. This will enable the system to better discern harmful content in cases where visual cues are ambiguous or insufficient. We will investigate advanced feature extraction techniques and temporal modeling approaches to improve the detection of hidden or ambiguous harmful actions. This will involve developing algorithms that can identify subtle visual cues, patterns of motion, and temporal dependencies that may indicate harmful behavior.

Finally, we will explore methods to predict the potential consequences of highly imitative actions, enabling the system to identify videos that may lead to dangerous imitations, even if the immediate visual content appears innocuous. By addressing these challenges, we envision a future where KidGuardScan can effectively protect children from harmful content on TikTok, fostering a safer online environment for young users.

References

[1] Johan Edstedt et al. "Vidharm: A clip based dataset for harmful content detection". In: *2022 26th International Conference on Pattern Recognition (ICPR)*. IEEE. 2022, pp. 1543–1549.

[2] Peng Wu et al. "Not only Look, but also Listen: Learning Multimodal Violence Detection under Weak Supervision". In: *European Conference on Computer Vision (ECCV)*. 2020.

[3] Kostantinos Papadamou et al. "Disturbed YouTube for kids: Characterizing and detecting inappropriate videos targeting young children". In: *Proceedings of the international AAAI conference on web and social media*. Vol. 14. 2020, pp. 522–533.

[4] Mithun Das et al. "Hatemm: A multi-modal dataset for hate video classification". In: *Proceedings of the International AAAI Conference on Web and Social Media*. Vol. 17. 2023, pp. 1014–1023.

[5] Shubham Singh et al. "KidsGUARD: fine grained approach for child unsafe video representation and detection". In: *Proceedings of the 34th ACM/SIGAPP symposium on applied computing*. 2019, pp. 2104–2111.

[6] Qi Dai et al. "Fudan-Huawei at MediaEval 2015: Detecting Violent Scenes and Affective Impact in Movies with Deep Learning." In: *MediaEval*. Vol. 1436. 2015.

[7] Qi Dai et al. "Fudan-NJUST at MediaEval 2014: Violent Scenes Detection Using Deep Neural Networks." In: *MediaEval*. 2014.

[8] Jônatas Wehrmann et al. "Adult content detection in videos with convolutional and recurrent neural networks". In: *Neurocomputing* 272 (2018), pp. 432–438. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2017.07.012. URL: https://www.sciencedirect.com/science/article/pii/S0925231217312493.

[9] Zhan Tong et al. “Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training”. In: *Advances in neural information processing systems* 35 (2022), pp. 10078–10093.

[10] Limin Wang et al. “Videomae v2: Scaling video masked autoencoders with dual masking”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 14549–14560.

[11] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. “Is space-time attention all you need for video understanding?” In: *ICML*. Vol. 2. 3. 2021, p. 4.

[12] ML McHugh. “Interrater reliability: the kappa statistic”. In: *Biochem Med (Zagreb)* 22.3 (2012), pp. 276–282.

[13] Joseph L Fleiss. “Measuring nominal scale agreement among many raters”. In: *Psychological bulletin* 76.5 (1971), p. 378.

Appendix

In this appendix, we provide more details of our proposed system from the following aspects:

- The configuration setting of training and evaluation phase is in § A.
- The error analysis details are in § B.
- Multiple screenshots of the system running are in § C.
- The structure of sample data sent to MongoDB is in § D.

A Configuration Setting

This section outlines the configuration settings used for fine-tuning our models. The Table 5 below presents detailed information about the optimizer, learning rate, batch size, training epochs, sampling strategy, augmentation techniques, inference protocol, and pre-trained models employed in our experiments. These settings are critical to replicating the results and ensuring the reproducibility of our study. Each configuration parameter has been carefully selected to optimize performance on the TikHarm dataset.

Table 5: Fine-tuning Settings for TikHarm Dataset.

Configuration	**TikHarm Dataset**
Optimizer	AdamW
Learning Rate	5e-5
Batch Size	12
Training Epochs	10
Sampling Strategy	Uniform Temporal Subsampling
Augmentation	Normalize() RandomShortSideScale(min_size=256, max_size=320) RandomCrop((224,224)) RandomHorizontalFlip(p=0.5)
Inference Protocol	2 Clips x 3 Crops
Pre-trained Models	VideoMAE (MCG-NJU/videomae-base), num_frames=16 TimeSformer(facebook/timesformer-base-finetuned-k400), num_frames=8

B Error Analysis Details

In this section, we provide detailed analysis of the error cases identified from the confusion matrix. Each error case was carefully examined to understand the underlying reasons for the incorrect predictions. Specific examples of misclassifications are presented, highlighting patterns or commonalities that may inform future improvements. This detailed examination aims to guide further refinement of the model to enhance its accuracy and robustness.

Figure 4: Examples of Confusion due to Overlap between Labels. From left to right: Misclassification of Suicide as Safe, Adult Content as Safe, and Suicide as Safe.

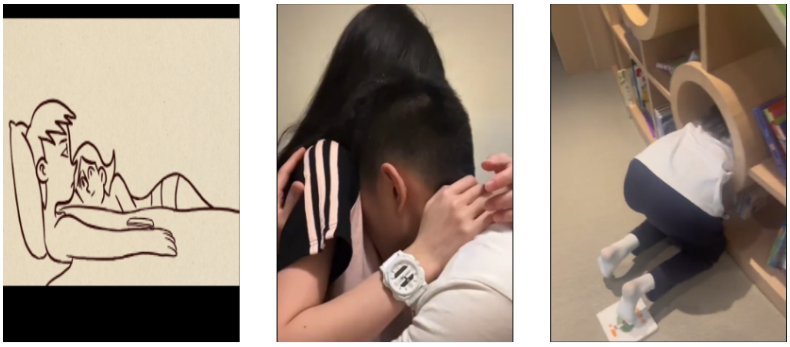

Figure 5: Examples of Hidden or Ambiguous Actions. From left to right: Misclassification of Adult Content as Safe, Adult Content as Suicide, and Adult as Safe.

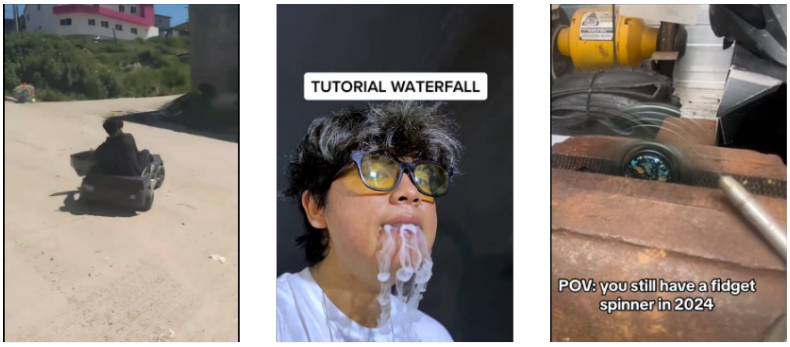

Figure 6: Examples of Highly Imitative Actions. Misclassification of Harmful Content as Safe.

C Screenshots of System Running

This section presents a collection of screenshots depicting the functionalities and interfaces of various systems used in our project. These images illustrate the practical application and integration of different tools within the workflow. Each figure emphasizes a specific aspect of the system, providing a visual representation to complement the detailed descriptions in the main text. Below are the screenshots along with brief explanations:

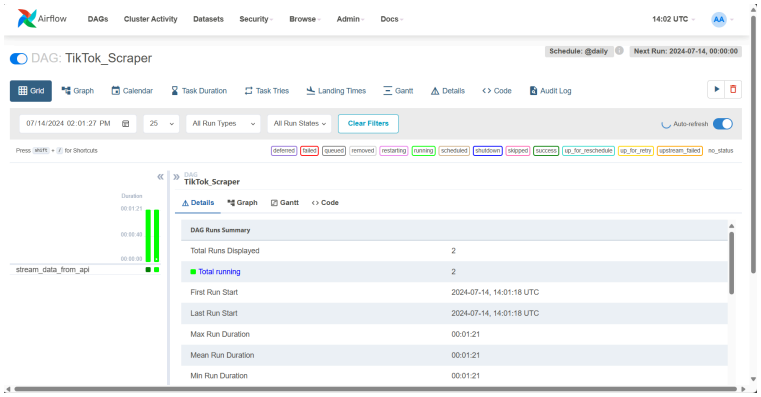

Figure 7: Screenshot of the Airflow interface showcasing workflow management.

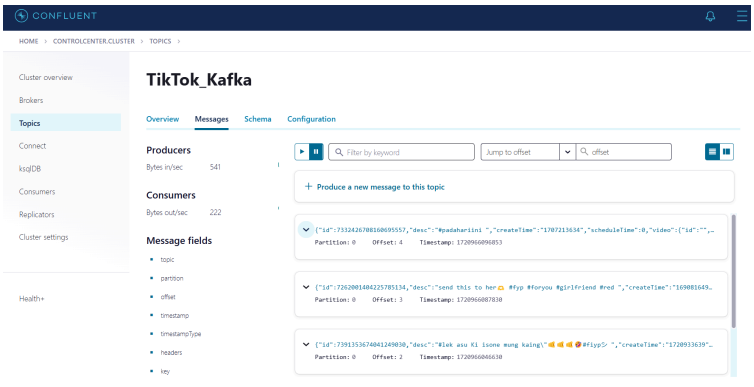

Figure 8: Screenshot of the Kafka dashboard displaying real-time data streaming.

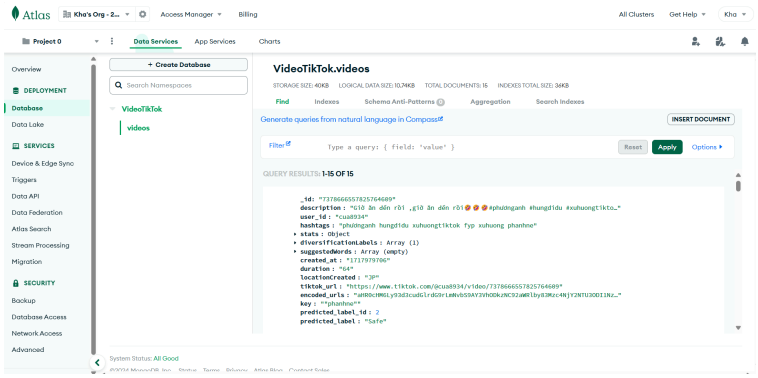

Figure 9: Screenshot of the MongoDB interface showing the structure of sample data.

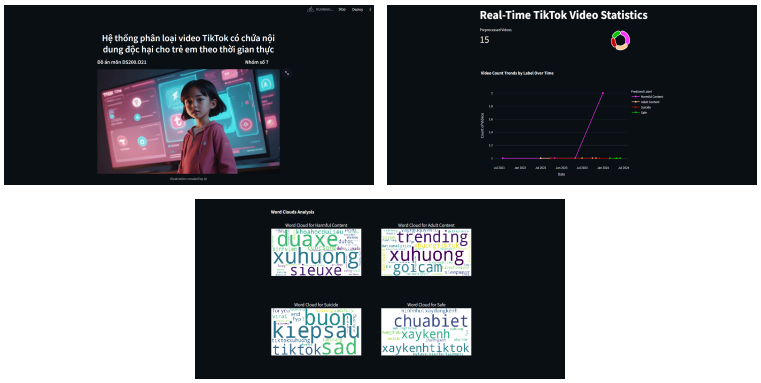

Figure 10: Screenshot of the Streamlit app demonstrating the user interface and analytics.

D Sample Data Examples and Structure

This section provides an example of the data structure used in our project. The following JSON snippet represents a sample data record, detailing various attributes such as the description, user information, statistics, and labels associated with the data.


```
{
  "id": "7323774777028594950",
  "description": "Should I study data science and data analysis? #
     life #dataanalysis #datascience #student #studyabroad #
     studyincanada #whattomajor",
  "user_id": "itstamhn",
  "hashtags": ["life", "dataanalysis", "datascience", "student", "
     studyabroad", "studyincanada", "whattomajor"],
  "stats": {
    "shareCount": 459,
    "playCount": 666800,
    "commentCount": 249,
    "diggCount": 45100,
    "collectCount": 7656
  },
  "diversificationLabels": ["Professional Development", "Education",
      "Technology"],
  "created_at": 1705199200,
  "suggestedWords": [],
  "duration": 34,
  "locationCreated": "CA",
  "tiktok_url": "https://www.tiktok.com/@itstamhn/video/732377477702
     8594950",
  "predicted_label": "Safe",
  "predicted_label_id": 2
}
```


Figure 11: Data examples and structure.