

Development of an Android-Based Terrorist Messages Detection and Profile Capturing System on Social Media

Azemobor Daniel¹, I. I. Achi², Nwamini Bartholomew Tochukwu³,
Chibuike Ezeocha Madubuike⁴, Anyanwu Chinyere Ihuoma⁵, Julius Michael
Sunday⁶

¹(ICT Unit/ Evangel University, Akaeze, Nigeria)

²(Computer Science, Physical Sciences/ Alex Ekwueme Federal University Ndufu-Alike Ikwo, Nigeria)

³(ICT Unit / Alex Ekwueme Federal Teaching Hospital, Abakaliki, Nigeria)

⁴(Computer Science / Akanu Ibiam Federal Polytechnic Unwana Afikpo, Nigeria)

⁵(Computer Science / Evangel University Akaeze, Nigeria)

⁶(ICT Unit / Evangel University, Akaeze, Nigeria)

Abstract:

Background: Social media platforms such as Facebook, WhatsApp, Twitter, and Instagram are widely used for communication and sharing information. Unfortunately, terrorists have begun misusing these platforms to spread violent content, recruit members, and plan attacks. This growing problem presents serious challenges for governments and security agencies. Current systems struggle to detect harmful content early, understand coded language, send real-time alerts, verify user identity, or keep proper records of suspicious activities.

Materials and Methods: To solve these problems, an Android-based system was developed using Natural Language Processing (NLP) to detect harmful or suspicious messages, even if they are written in slang or secret codes. The system uses the Termii SMS API to send instant alerts via SMS to security agents when threats are found. It also includes a facial recognition feature that allows users to upload real-time photos, which helps verify identity. All data, including messages and user details, are stored in a central database for future reference. The system was built using PHP, JavaScript, HTML, CSS, jQuery, and converted into an Android application using a Web-to-APK tool. The Object-Oriented Analysis and Design Methodology (OOADM) was applied to support system flexibility and ease of maintenance.

Results: The system was successfully able to detect and report harmful messages, send real-time alerts to authorities, verify users through facial recognition, and store detailed records. It worked effectively on mobile devices and addressed the key issues found in existing systems.

Conclusion: This research shows that combining NLP, facial recognition, and instant messaging technologies can improve the early detection and tracking of terrorist activities on social media. The developed system provides a practical and efficient tool for enhancing national security efforts.

Keywords: Cybersecurity, Social Media, Terrorist Detection, Natural Language Processing (NLP), Facial Recognition, SMS Alert, Android Application, OOADM.

Date of Submission: 07-07-2025

Date of Acceptance: 17-07-2025

I. Introduction

In today's digital world, social media platforms like Facebook, WhatsApp, Telegram, and X (formerly Twitter) have become important tools for communication, sharing information, building connections, and promoting businesses. While these platforms offer many benefits, they also create serious risks, especially for public safety and national security. One major concern is how terrorist groups now use social media to spread harmful messages, recruit new members, and plan attacks. These activities are often hidden in normal-looking posts, images, and videos, making them hard to detect in time [9].

Terrorism means using violence or threats, usually against civilians, to push political, religious, or ideological agendas. In the past, these actions happened mostly in physical locations, using secret networks. But now, many of these groups operate online, especially on social media, because it offers fast communication and access to large audiences [6]. Terrorists often use vague or hidden language to avoid being detected. For example, they might use words like "gift" instead of "bomb," or "cleanse" instead of "kill" [7].

Security agencies face a big challenge because of the huge amount of data shared every second on social media. Millions of posts, videos, and images make it hard to spot harmful content early. Traditional

methods like keyword searches and user reports are not enough. People who spread harmful content now use code words, emojis, or symbols to hide their real meaning, which makes these older systems less effective [12].

To address this problem, a smart Android-based system is needed. This system should be able to detect suspicious messages and collect user profile data automatically. For example, Natural Language Processing (NLP) tools like ChatGPT can check if a message is written in English, fix grammar issues, and identify dangerous keywords such as “bomb,” “kill,” or “attack” even when they appear in slang or coded language [1].

The system will also use a face recognition API to study the profile pictures of users posting the suspicious content. When the system finds a threat, it will save the message and profile image, send an alert to a secured dashboard, and also send an SMS to the mobile phone of the assigned security officer. This allows faster action and helps prevent attacks before they happen [8]. Using both message analysis and facial recognition helps in identifying the sender of the threat more accurately. Research shows that this combined approach improves the ability to detect online terrorism [3].

As more people use social media every day, the risk of online radicalization is also increasing. Recent studies show that extremist groups are now focusing on young people by sharing messages that look harmless but actually contain hidden ideas meant to influence or mislead them [10]. These messages can appear in videos, images, or simple text posts. Because of this growing danger, there is a strong need to create smart tools that can detect and report such harmful content directly from mobile phones. One important solution is building an Android-based system that can find terrorist messages and gather user profile details. This kind of system can use Natural Language Processing (NLP), face recognition, and mobile technology to help security agencies quickly find and respond to possible threats, helping to keep the public safe [5]. For example, Dataminr is a system that uses AI to monitor social media and other online sources. It sends real-time alerts when it finds information that could signal an emerging security risk [4]. Another organization, Cyabra, uses AI and machine learning to find fake social media accounts and disinformation campaigns. These fake accounts are often used by extremists to spread harmful beliefs [2].

Terrorism is changing, and it now includes digital threats from social media. To stay ahead of these dangers, security agencies must use new and smart technologies. By combining NLP, and facial recognition in Android apps, they can detect threats faster and more accurately. Developing an Android-based system to detect terrorist content and identify suspicious users is a major step forward. It will help create a safer online space and protect communities from digital threats.

The increasing misuse of social media by terrorists to share extremist messages, recruit new members, and plan violent attacks have become a major threat to public safety. One of the main challenges is the inability of current systems to detect harmful messages early. Often, these messages spread widely before any action is taken. In addition, terrorists use coded language, slang, emojis, and secret words that make it difficult for traditional systems to recognize threats. Another issue is the lack of an effective notification system that alerts security agents immediately when suspicious content is detected. This delay in communication reduces the chances of stopping the threat on time. Many systems also fail to verify whether the user’s identity is real. Terrorists can easily register with fake photos, cartoons, or symbols that hide their true identity. Furthermore, the absence of a centralized storage system means there is no reliable way to keep records of suspicious messages and user profiles for future investigation. These problems create serious gaps in the efforts to track, analyze, and respond to terrorist activities on social media.

This paper aims to develop an Android-based system that can detect terrorist messages and capture user profiles on social media to help security agencies respond more effectively. The system will be designed to monitor social media platforms and identify suspicious content, especially those written in slang or hidden language. It will use Natural Language Processing (NLP) tools, such as the ChatGPT API, to understand and analyze the meaning behind dangerous words and phrases. The system will also include a real-time alert feature that uses the Termii SMS API to notify registered security personnel as soon as a threat is found. To ensure that users are genuine, the system will require facial recognition during registration to verify their real identity. All flagged messages and related user data will be stored in a centralized database, which can be used later for tracking and investigations. The overall goal is to provide a smart and efficient solution that strengthens the fight against terrorism online.

II. Material And Methods

Methodology is a step-by-step plan or process that guides how a system is studied, designed, and built. It helps developers stay organized and works in a structured way. Using a good methodology ensures that the final system solves the real problem, meets the user's needs, and can be improved in the future if necessary. Methodologies are very important in software development because they reduce confusion, save time, and help avoid errors during the system development process.

For this paper, the chosen methodology is the Object-Oriented Analysis and Design Methodology (OOADM). This methodology is widely used for building complex and flexible systems. OOADM focuses on

identifying objects in the system. An object is anything that has data and can perform actions. For example, in this project, an object can be a "user," a "message," or a "notification." These objects have their own information (called attributes) and things they can do (called methods).

OOADM is made up of two main stages: Object-Oriented Analysis (OOA) and Object-Oriented Design (OOD). In the analysis phase (OOA), we study the problem we want to solve. We look at what the system should do and try to understand all the important parts. For this terrorist message detection system, we identify the key parts such as users, social media messages, terrorist content, SMS alerts, and administrators. We also define how these parts relate to each other. For example, when a message is flagged as dangerous, the system must send an alert to security agents.

In the design phase (OOD), we take the results of the analysis and plan how the system will be built. Each object is turned into a class in the system. A class is a template that describes the object's attributes (such as name, photo, or content of a message) and its actions (such as detect message, analyze content, or send alert). These classes work together to carry out the system's full job.

OOADM is very useful because it allows for modular design, meaning each part of the system is built separately and can be changed or fixed without affecting the whole system. For instance, if we want to improve the facial recognition feature later, we can update just that part without rewriting the whole software. It also allows code to be reused across different parts of the project. This approach also makes it easier to detect and fix problems. Because each object has a specific task, it is simple to track down where a problem is coming from. In this project, OOADM will be used to design and develop a terrorist message detection system that works on social media platforms. The system will include:

- i. A message detection unit using Natural Language Processing (NLP)
- ii. A user identity checker using Facial Recognition
- iii. A real-time SMS alert system for notifying security agents
- iv. A database for storing user and message information

Each of these components will be treated as objects with specific roles in the system.

Data gathering is the process of collecting useful information that helps to understand a problem and find a solution. In this research, the goal was to build a system that can detect terrorist messages on social media using natural language processing, facial recognition, and Android-based technology. To collect the right information, both primary and secondary methods of data collection were used.

Primary Data Gathering: The primary data gathering involved directly engaging with people through interviews and observing online activities. The Interviews were held with individuals who are familiar with social media or work in security-related fields. These included social media users and law enforcement officers. During the interview, the participants were asked about their experiences with suspicious or harmful posts online, how often they see such messages, what they believe should be done about it, and whether they report such posts when they come across them. Most participants admitted that dangerous content is not always easy to identify because it often comes in coded forms like slang and secret symbols. They also said that many fake accounts are created using false profile pictures such as animals, cartoons, or random objects to hide the identity of the real person.

In addition to interviews, online behavior was also observed directly. Social media platforms like Facebook, Instagram, and X (formerly Twitter) were monitored over several days to watch how users communicate and how harmful messages appear. It was noticed that some posts had threatening language mixed with emojis, and others used fake names and images. These posts often stayed online for hours before they were taken down, which shows a delay in detection and action. The observation also revealed that users rarely report such content, making the situation worse.

Secondary Data Gathering: For secondary data, information was gathered using the internet. Online research helped to understand what other researchers, institutions, and developers have done in the area of detecting harmful messages on social media. This included reading blogs, technical reports, academic papers, and news articles related to terrorism, fake identities, social media threats, natural language processing, and artificial intelligence. The internet was used to access and review several existing studies, practical solutions, and case reports that highlight the challenges of detecting coded language and the delay in response by authorities. It also provided insight into how technology is currently being used in other countries and what improvements can be made.

By combining all the information from the interviews, real-time observation, and internet-based findings, a clear picture was developed of the current challenges in detecting terrorist content online. The data collected from people gave firsthand knowledge, while the internet provided a broader understanding of global approaches and

the technical possibilities available. This helped in identifying the weaknesses of the existing systems and in designing a better solution that can detect harmful content early, recognize fake user identities, and send fast alerts to security agencies.

III. Result

The new system is developed to improve the way terrorist messages and fake social media profiles are detected and reported. It aims to fix the serious problems found in the current system, such as delays in discovering dangerous content, failure to recognize hidden or coded language, lack of fast alerts to security officials, and the inability to confirm the true identity of users who may be using fake profile pictures.

In the design of this system, the main goal is to create a smart and reliable platform that can quickly identify harmful messages posted on social media and take action before they spread or cause harm. The system will automatically monitor all messages on the platform using intelligent software that can understand and interpret human language. This is achieved through a process called Natural Language Processing (NLP), which helps the system understand the meaning behind the words used in posts or messages. NLP also helps the system detect unusual patterns, coded language, and slang that terrorists often use to hide their true intentions.

Apart from analyzing text, the proposed system will also examine user profile pictures using facial recognition technology. This is important because many terrorists use fake images like, flowers, or random objects to hide who they really are. The facial recognition feature checks whether a profile picture shows a real human face or something fake. If it finds anything suspicious, the system will flag that profile for further investigation.

Another important part of the proposed system is its ability to send real-time alerts. When the system detects a harmful message or a suspicious user profile, it automatically sends a detailed warning to the appropriate security agents. This is done through an SMS gateway that delivers fast text messages directly to the phones of these agents. These messages include information such as the content of the suspicious message, the username of the sender, and the exact time it was posted. This quick alert system helps ensure that security teams can respond immediately and stop the threat from growing.

The system is designed to support different types of users. There are regular users who register and post content on the platform, and the system continuously monitors their activities to ensure safety. The administrator manages the system's overall operation, reviewing flagged messages, managing user accounts, and making decisions about which users may need to be suspended or investigated. The system also works in the background as an intelligent engine, automatically scanning content without human input. Lastly, the security agents receive notifications from the system and are responsible for taking action when a possible threat is detected.

The technologies used in this system work together to provide a complete solution. The natural language processing feature helps understand complex messages. The facial recognition system helps verify identity. The SMS gateway sends alerts quickly, and the database stores important information like messages, user data, and system activity.

By combining these features, the proposed system improves the speed and accuracy of identifying terrorist activity. It reduces the need for human staff to manually check every message or user, which saves time and energy. It also makes it harder for terrorists to use fake identities or hidden messages, because the system can now detect these tricks. By sending alerts as soon as a threat is found, the system gives security agents enough time to act and stop problems before they become dangerous.

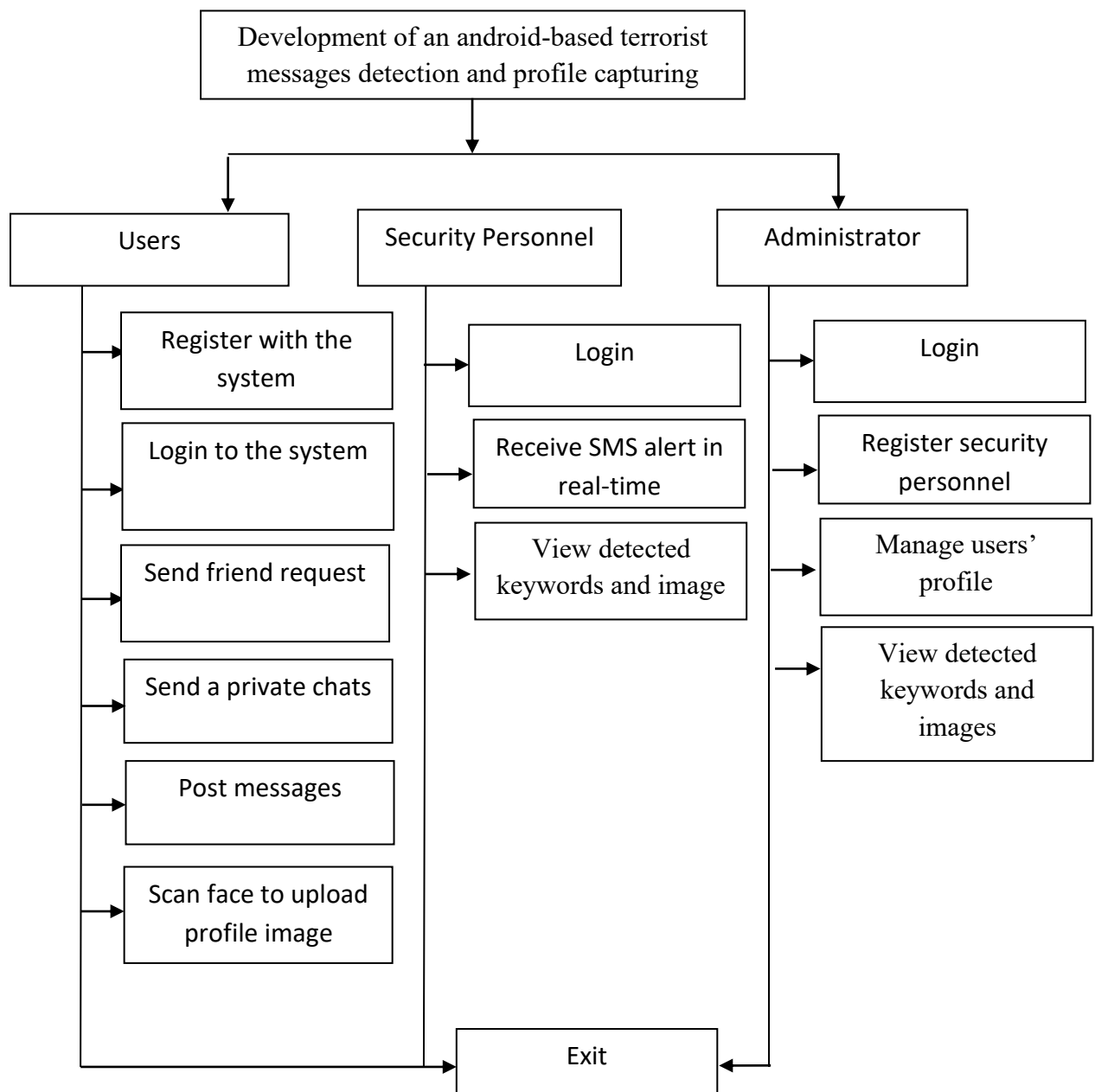


Figure 1: High Level Model of the New System

Figure 1 describes the high level model of the proposed system and the various users of the system. Here there are three users to the system which are users, security personnel and the administrator. The users register and login to send friend request and start send messages not knowing that there is a background check on their conversation and also scan and upload their profile picture. The security personnel will receive a real-time notification alerting them to login that a threat has been detected, and then view the detected messages and the captured profile images. Finally the administrator will login to register security personnel, view captured detected messages and profile images and manage users profiles.

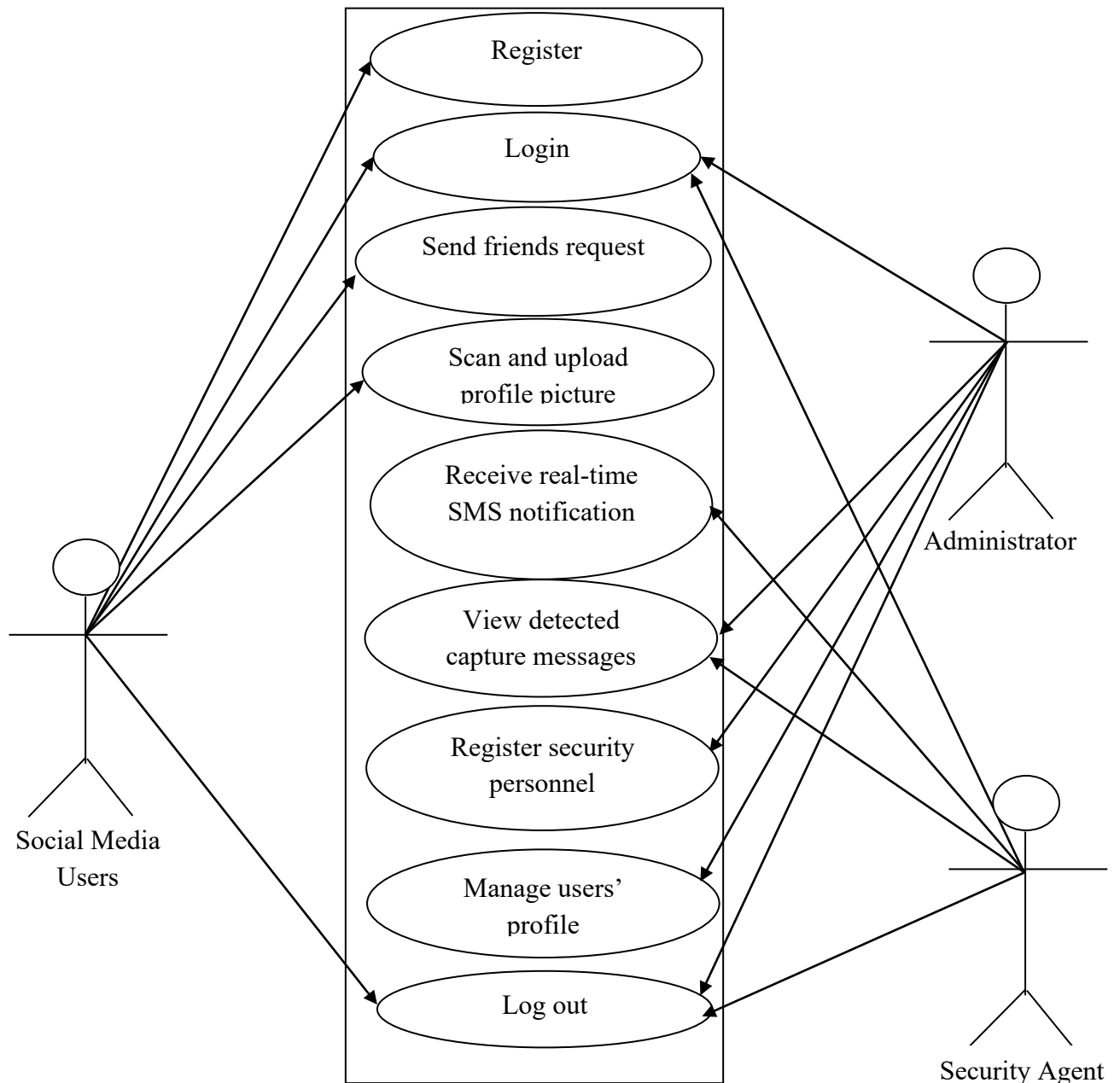


Figure 2: Use Case Diagram of the New System

Figure 2 is the use case diagram of the new system, this diagram describe the function of various users to the system.

A use case diagram is a visual tool used in software design to show how users (also called actors) interact with a system. It describes the different actions that each type of user can perform in the system. In this proposed system, which is designed to detect terrorist messages on social media using natural language processing, facial recognition, and SMS gateway, there are several key users. These include the Admin, Social Media User, and Security Agent. Each of these users performs different tasks in the system, and the use case diagram helps to show their roles clearly.

The Social Media User is the person who uses the application to log in, create a profile, and post messages. The system monitors the content of their messages. If a message contains harmful or suspicious language, it is flagged automatically by the NLP module. Also, their profile picture is scanned using the facial recognition module to detect whether it is a real human face or a fake image like a cartoon, flower, or animal.

The Admin manages the system. The Admin can view flagged messages, update the detection rules, manage user accounts, and control how the system works. The Admin also has access to reports and system logs that show how many messages have been flagged and what actions were taken.

The Security Agent receives notifications when a harmful message is detected. The system sends them an instant alert through SMS or in-app notification. This allows them to act quickly and prevent any possible damage. The Security Agent can log in, view flagged messages, track suspicious accounts, and investigate further if needed.

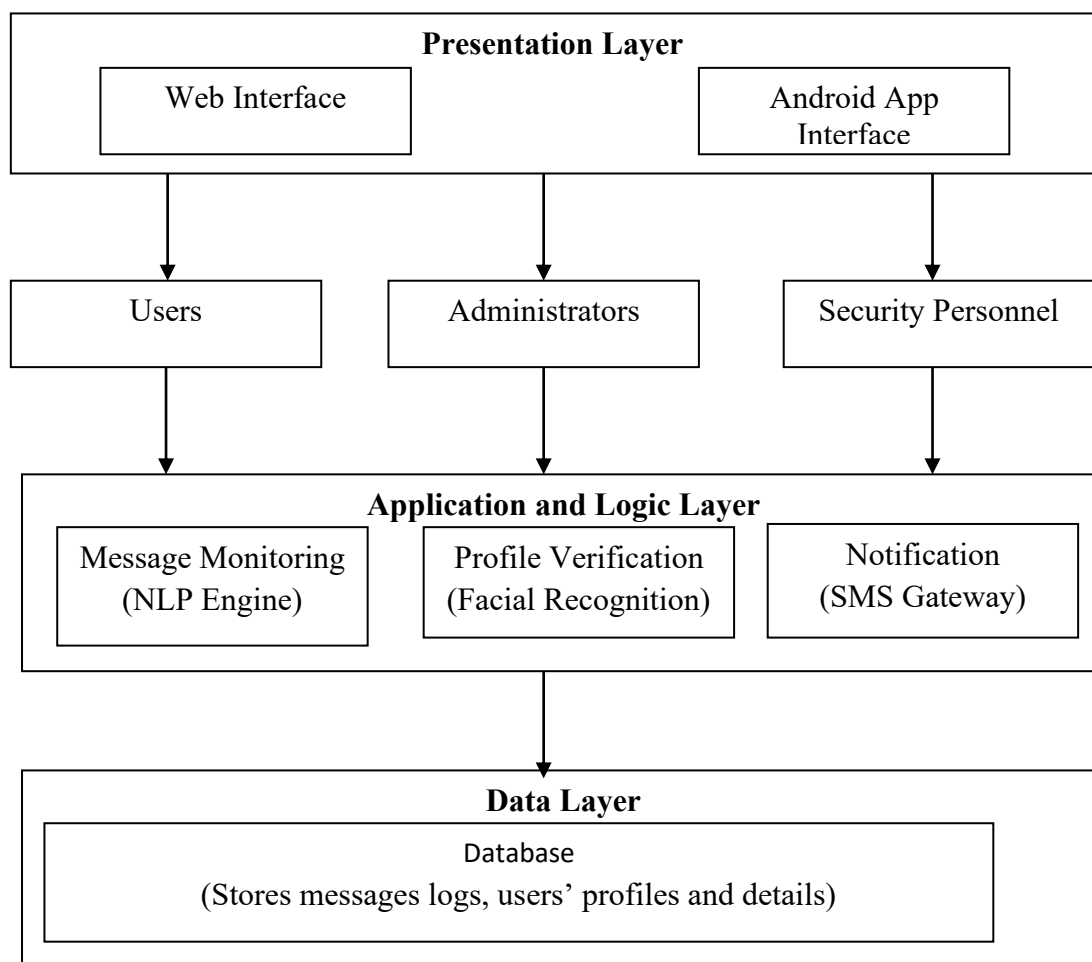


Figure 3: System Architecture

Figure 3 describe the architecture of the system, how the layers are interconnected and how the users interact with the system. The system architecture uses a client-server model, where the Android device acts as the client and communicates with the server through the internet. The server handles all the processing, checking, and alerting. This makes the system fast and reliable, even with many users.

The system architecture is designed to support detection of terrorist messages using powerful tools like NLP and facial recognition. It ensures that users can easily interact with the system, threats can be quickly analyzed, and alerts can be sent out without delay. Each part of the architecture plays a vital role in making the system effective, secure, and easy to maintain.

The implementation stage is a crucial phase in the Software Development Life Cycle, where the system design is translated into code. This stage includes key activities such as testing, documentation, training, and conversion to ensure the application functions as intended. This section explains how the new system was built and how its different parts work together to detect and report suspicious terrorist messages on social media. The system was designed as an Android-based application that uses Natural Language Processing (NLP), facial recognition, and an SMS alert system. The front-end part of the system was developed using HTML, CSS, JavaScript, and jQuery. These tools helped create a simple and user-friendly interface that allows users to post

messages, upload their profile pictures, and view flagged content. The design makes it easy for both security agents and regular users to use the app without much training. The back-end was built using PHP. It handles important processes like storing user information, analyzing messages with NLP tools, checking images with facial recognition, and sending SMS alerts when harmful messages are found. The app uses a centralized database to save records of users and suspicious posts for future investigation. The database also helps with tracking repeat offenders and managing security reports. To make the app mobile-friendly, it was converted from web to Android format using a Web-to-APK converter. This allowed the system to run on Android phones without needing to use a browser. The facial recognition part of the app checks if a profile picture is a real human face; if not, the account is flagged for review. The SMS notification feature was added using the Termii SMS API. When a message is flagged as dangerous, the system sends a text alert to security agents immediately, helping them respond faster. The Object-Oriented Analysis and Design Methodology (OOADM) were used to build the system. This helped in organizing the system into different parts or "objects" like users, messages, and alerts. OOADM made the system easier to design, test, and update in the future. Overall, the system implementation brought together different tools and technologies to build a smart, responsive, and easy-to-use solution for detecting and stopping terrorism on social media.

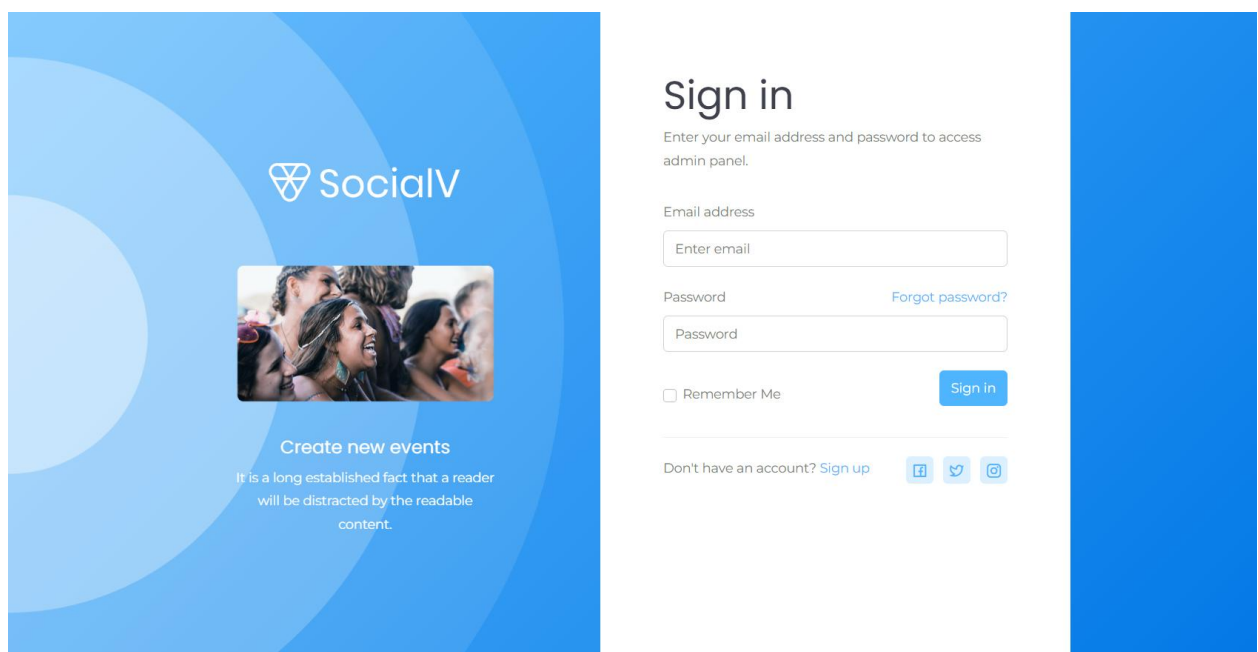


Figure 4: Login Interface Implementation

Figure 4 is the login implementation; this is the interface that displays once a user wants to login to his or her dashboard

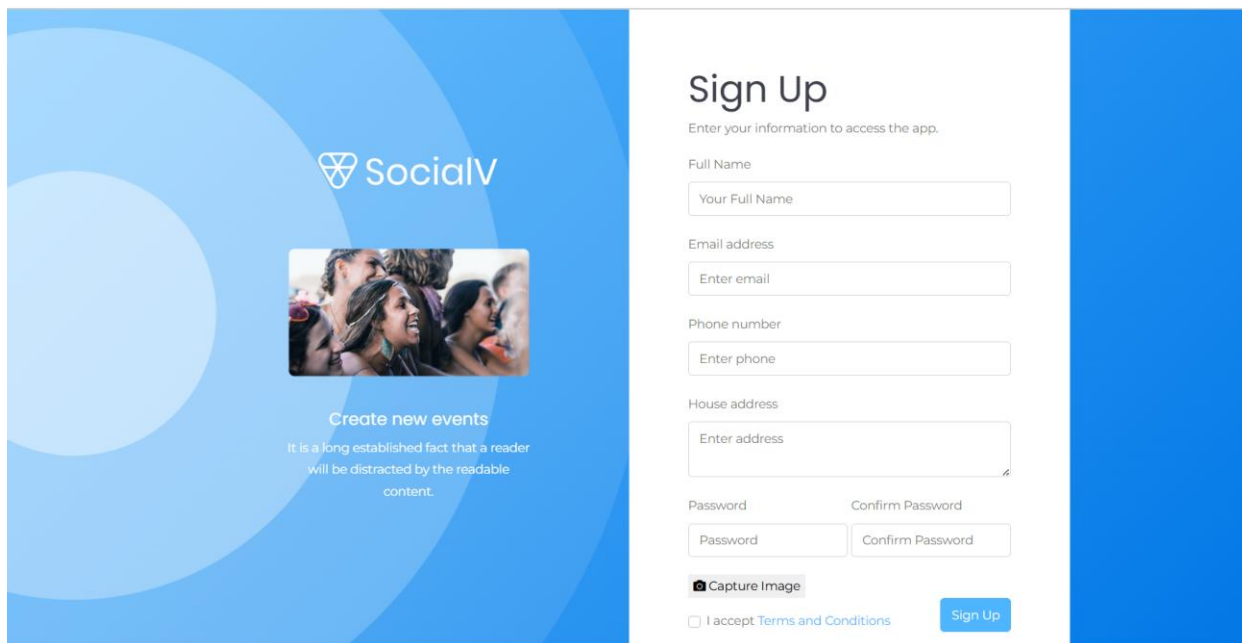
The image shows a web-based sign-up interface for an application named "SocialIV". On the left, there is a blue sidebar with the "SocialIV" logo and a section titled "Create new events" with a placeholder image and text. The main area is white and contains the "Sign Up" heading and a subtext "Enter your information to access the app.". Below this, there are input fields for "Full Name", "Email address", "Phone number", and "House address". There are also fields for "Password" and "Confirm Password". At the bottom, there is a "Capture Image" button, a checkbox for "I accept Terms and Conditions", and a "Sign Up" button.

Figure 5: Sign Up Interface Implementation

Figure 5 is the registration interface implementation, this is the interface that enables users to register and scan their face to be uploaded to the system

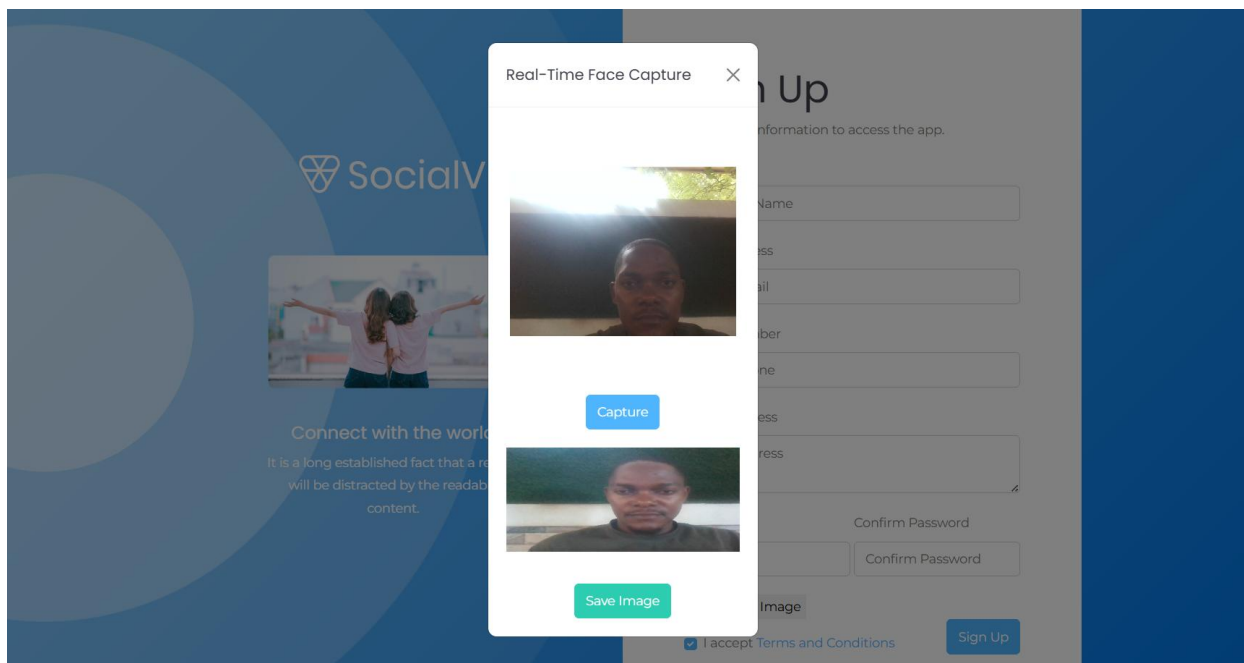
The image shows a web-based face capturing interface. A modal window titled "Real-Time Face Capture" is overlaid on the sign-up form. The modal contains a live video feed of a person's face. Below the video feed, there is a "Capture" button. Below the "Capture" button, there is a thumbnail of the captured face and a "Save Image" button. The background shows the same sign-up form as in Figure 5, but it is dimmed.

Figure 6: Face Capturing Interface Implementation

Figure 6 is the facial recognition capturing interface implementation; this is the interface that enables users to capture their face before the form can be submitted.

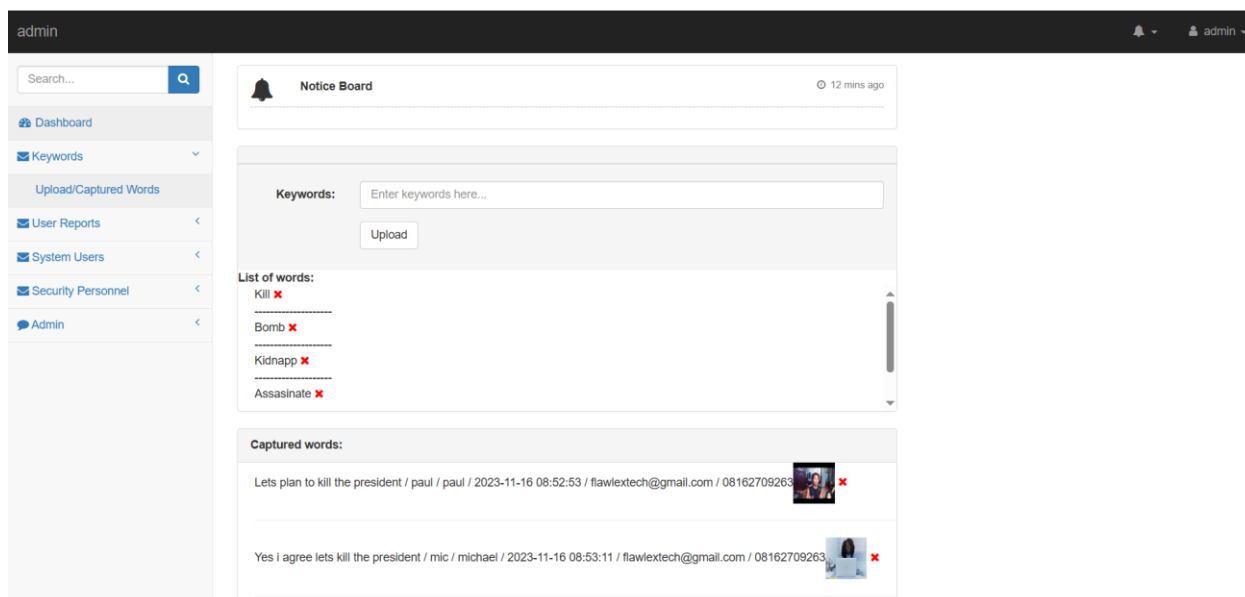


Figure 7: Detected Messages Output Interface Implementation

Figure 7 is the output interface implementation, this is the interface that displays once suspicious messages and keywords are detected.

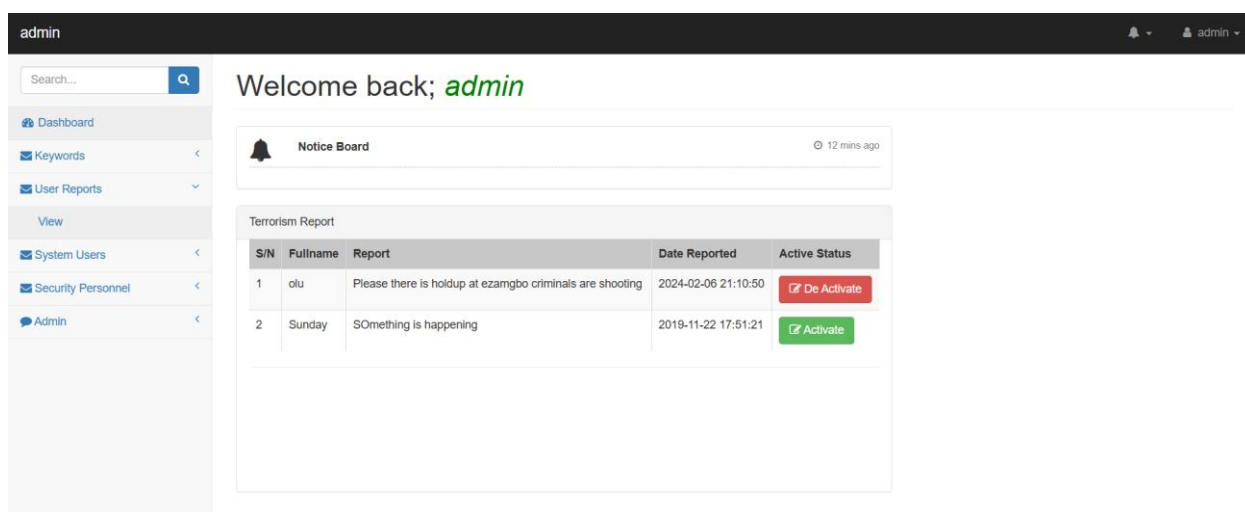


Figure 8: Report Output Interface Implementation

Figure 8 is the output interface implementation; this is the interface that displays all reported messages from users.

Testing and Performance Evaluation

After successful deployment, the system was tested based on data presented. During the testing, the actual and expected results were compared to ensure they produced same result or if there is a difference, it should be slight and negligible. The result is depicted in table 1.

Table 1: Actual Test Result versus Expected Test Result

Test Case Description	Expected Result	Actual Result	Pass/Fail
User posts a terrorist message with threat keywords	Message should be flagged and alert sent to security personnel	Message was flagged and alert sent immediately via SMS	Pass
User uploads a fake profile picture (cartoon or object)	Profile picture should be rejected by facial recognition system	Fake image was rejected and user was asked to upload a real photo	Pass
Normal user posts a harmless message	Message should not be flagged	Message posted successfully without any alert	Pass

Admin logs into the system	Admin should have full access to users, posts, and flagged messages	Admin logged in and had full control as expected	Pass
Security officer logs in to view flagged messages	Security officer should see only flagged messages and alerts	Security officer was able to view flagged messages and respond	Pass
User tries to register with fake data	System should validate and prevent incomplete or suspicious data	System detected missing facial data and rejected registration	Pass
User posts message with coded emojis and slang	System should detect hidden meaning and flag it for review	Message was flagged correctly after NLP analysis	Pass
Harmful message is posted in the chat section	System should scan chat messages and raise alert if necessary	System scanned chat, detected threat, and sent alert	Pass
SMS alert notifies security personnel of danger	SMS should be delivered quickly with message content	SMS was received within seconds by the security personnel	Pass
User tries to access admin features without permission	System should block unauthorized access	Access was denied and user was redirected	Pass

IV. Discussion

This study focused on designing and implementing a system to detect terrorist messages on social media using natural language processing, facial recognition, and SMS API gotten from termii website to send notification to security personnel. The major findings showed that the system can successfully identify harmful messages by analyzing text, and coded language often used to hide dangerous content. It was also found that fake user profiles, which use false or cartoon images, can be detected by the facial recognition feature. This helps in reducing the chances of terrorists hiding behind fake accounts.

Another important finding is that the system can send quick SMS alerts to security personnel when suspicious messages are detected. This fast notification helps security agents respond more quickly to potential threats. During testing, the system showed good accuracy in detecting harmful posts and reducing the time it takes for authorities to act.

However, the study also found some challenges. The system may not detect English word used as coded messages. Network problems can also affect the performance of features like facial recognition and SMS alerts.

Overall, the research demonstrated that combining natural language processing, facial recognition and instant messaging improves the detection and management of terrorist content online. The findings suggest that this system can be a useful tool to support security agencies in monitoring social media and protecting the public from harmful messages. Regular updates and improvements will be necessary to keep the system effective as new threats and methods continue to appear.

V. Conclusion

This paper successfully designed and implemented a system to detect terrorist messages on social media by using natural language processing, facial recognition, and Android-based technology. The system was able to identify harmful content quickly, even when the messages were hidden using slang, or secret codes. It also helped to find fake user profiles by checking profile pictures with facial recognition. This made it harder for dangerous people to hide their real identity online.

The system's ability to send fast SMS alerts to security agents improved how quickly they could respond to threats. This shows that technology can play a strong role in helping keep social media safer and preventing the spread of harmful messages. In conclusion, this system offers a useful tool for detecting and managing terrorist activities on social media. It supports security agencies by providing early warnings and better user identity checks. With further development and regular updating, the system can help reduce online dangers and make social media safer for everyone. This project shows how modern technology can be used to fight crime and protect communities in the digital age.

References

- [1] Ahuja, R., and Rathi, V. (2024). Natural language processing applications in security and surveillance systems. *Journal of Artificial Intelligence and Security*, 11(1), 34–45.
- [2] Cyabra. (2024). Using AI to detect fake accounts and disinformation. <https://www.cyabra.com>
- [3] Daramola, O., and Alabi, T. (2024). Combining NLP and facial recognition for detecting online terrorist activities. *African Journal of Cybersecurity and Information Systems*, 9(2), 66–78.
- [4] Dataminr. (2024). Real-time AI alerts for crisis detection. <https://www.dataminr.com>
- [5] Department of Homeland Security (DHS). (2024). Emerging technologies for threat detection on digital platforms. <https://www.dhs.gov>
- [6] Dreher, M., and Zimmerman, J. (2023). The transformation of terrorism in the digital age. *Security and Global Affairs Review*, 8(1), 15–29.

- [7] Mahmood, S., Ahmad, R., and Latif, M. (2023). Coded language and the challenge of detecting extremist messages online. *Journal of Cyber Intelligence*, 6(3), 101–112.
- [8] Musa, A., Salihu, A., and Eze, T. C. (2024b). Automated surveillance and SMS alert systems for public safety. *International Journal of Mobile Computing and Security*, 12(2), 59–70.
- [9] Salem, A., Khan, M., and Adepoju, A. (2023). Social media and the rise of digital terrorism. *Journal of Social Media and Terrorism Studies*, 10(4), 88–102.
- [10] United Nations Interregional Crime and Justice Research Institute (UNICRI). (2024a). Preventing youth radicalization through AI-powered monitoring. <https://www.unicri.it>
- [11] United Nations Interregional Crime and Justice Research Institute (UNICRI). (2020b). *Artificial intelligence and robotics for law enforcement*. <https://unicri.it>
- [12] Zhou, L., Wang, H., and Lin, Q. (2024b). Limitations of traditional keyword filtering for social media threat detection. *Computers & Security*, 136, Article 103250.