A Mobile Application for the Immediate Detection and Response to Violent Attacks in Urban Environments Utilizing Speech-to-Text and Natural Language Processing

Andres Doig
Software Engineering Program, Universidad Peruana de Ciencias Aplicadas, Lima, Peru
u201712256@upc.edu.pe

Nicolas Abanto
Information Systems Engineering Program, Universidad Peruana de Ciencias Aplicadas, Lima, Peru
u20201a835@upc.edu.pe

Lenis Wong
Software Engineering Program, Universidad Peruana de Ciencias Aplicadas, Lima, Peru
pcsilewo@upc.edu.pe (corresponding author)

Received: 5 July 2025 | Revised: 27 August 2025 | Accepted: 2 September 2025

 DOI: https://doi.org/10.48084/etasr.13189

ABSTRACT

The increase in violent attacks in urban environments has generated growing social concern about improving emergency response speed, considering that a large portion of the population has experienced some form of violence in public spaces. This study aims to build a mobile application for the immediate detection and response to violent situations using advanced technologies such as Speech-to-Text (STT) and Natural Language Processing (NLP). The development was carried out in four phases: (1) selection of STT service, (2) selection of NLP service, (3) Implementation of STT and NLP, and (4) application development. Additionally, an experiment was conducted in two simulated scenarios representing armed assaults, evaluating reaction times, Help Request Time (HRT), and total response time, both with and without the use of the application. The results showed a significant reduction in total response time, 50.35% in the first scenario and 43.92% in the second. Moreover, the application was evaluated by users and security experts, obtaining highly favorable scores in effectiveness, efficiency, and overall satisfaction. It is concluded that this technological solution represents an effective and practical tool for substantially improving speed and safety in critical situations, enabling more timely and efficient intervention.

Keywords-Speech-to-Text (STT); speech recognition; mobile application; public safety; violent attacks; Natural Language Processing (NLP)

I. INTRODUCTION

The rise in violence and targeted attacks against individuals in urban environments represents a critical issue that demands effective technological solutions. In Peru, the situation is particularly alarming, with a 25% increase in reported assault cases in Metropolitan Lima over the past year [1]. Furthermore, a study conducted by the National Institute of Statistics and Informatics (INEI) revealed that 65% of the urban population has been a victim of some form of violence in public spaces [2]. As cities continue to grow, emergency response capacity is increasingly challenged by the need for rapid and precise action. These figures highlight the urgent need for tools that enable the immediate detection and response to violent situations in order to effectively protect citizens.

Recent studies have explored the use of technologies such as speech recognition and cloud computing to enhance safety across various applications. For instance, systems like SOS Perú and Noonlight have demonstrated the effectiveness of integrating mobile and geolocation technologies to provide emergency assistance. Additionally, authors in [3] introduced a system based on Natural Language Processing (NLP) for identifying risk situations through real-time text analysis. Similarly, authors in [4] proposed a solution using neural networks to improve the accuracy of detecting violent events from acoustic signals.

Recent advances in Internet of Things (IoT)-based emergency response systems have shown significant promise in integrating multiple domains of safety monitoring. Authors in [5] developed a comprehensive Smart Emergency Response System (SERS) that combines advanced sensors, communication technologies, and intelligent algorithms to provide real-time monitoring and emergency management across vehicles, homes, and healthcare settings. Their system achieved server response times of 3 ms and accuracy rates exceeding 99%, demonstrating the potential for IoT technologies in creating faster, more reliable emergency response mechanisms. Similarly, authors in [6] introduced a wearable IoT-based smart glove for elderly care that integrates fall detection, vital sign monitoring, and gesture-based emergency alerts. Although designed for healthcare, their system highlights the applicability of IoT and real-time monitoring to emergency response, demonstrating how rapid detection and discreet alerts can save lives. One promising direction is the use of Intentional Voice Command Detection (IVCD), which enables systems to distinguish purposeful voice commands from ambient speech in real-world environments [7]. This approach has been successfully applied to hands-free control of home appliances and could be adapted to emergency response systems for rapid, voice-activated alerts. However, these approaches often lack fast and discreet mechanisms for triggering alerts without requiring extensive manual interaction, an essential feature in dangerous situations.

Complementary to voice-based approaches, computer vision techniques have also advanced violence detection capabilities. Authors in [8] developed a novel approach that integrates subgroup tracking into violence detection systems, enabling real-time identification and localization of violent events in surveillance footage with accuracies of 91.3% and 87.2% on standard datasets. Their system can process footage in real-time (0.561 s for 2-5 s videos) and provides interpretable outputs by identifying specific groups involved in violent incidents. While such video-based systems excel in monitored environments, they require existing surveillance infrastructure and may not provide immediate assistance to individuals in unmonitored areas.

In recent advancements, Long Short-Term Memory (LSTM) neural networks have significantly improved the recognition of emotional nuances in speech by capturing both temporal dynamics and structural speech features effectively. This approach enhances emotion detection accuracy by leveraging attention mechanisms on both time and feature dimensions, thus providing task-specific information more efficiently. Such methods demonstrated an accuracy of up to 96.81%, indicating their robustness for practical emotion recognition applications [9]. The use of Speech-to-Text (STT) and NLP has been explored in several security-focused studies. Authors in [10] discussed the implementation of voice commands in mobile applications to quickly and effectively activate emergency alerts. Authors in [11] explored the capabilities of automatic transcription, which could be adapted to improve emergency systems that require rapid voice command capture during crisis situations. Similarly, authors in [12] demonstrated the successful integration of Automatic Speech Recognition (ASR) and NLP techniques for controlling Unmanned Aerial Vehicles (UAVs) through real-time voice commands, achieving a transcription confidence rate of 91.86% with a word error rate of 0.021. Their work highlights the robustness of voice-command systems in mission-critical applications, supporting their applicability in emergency response technologies. Authors in [13] emphasized the importance of NLP for correctly interpreting help requests and ensuring a quick and appropriate response. Additionally, IoT has enabled greater interconnection of devices capable of monitoring and detecting risky situations. Authors in [14] explored the use of IoT to detect criminal activities in real time, enhancing the responsiveness of law enforcement agencies. This technology facilitates data collection through connected sensors, which are processed and analyzed to enable rapid and effective intervention.

Despite these advancements, a gap remains in providing individuals with a discreet, rapid, and universally accessible tool for activating emergency assistance in unmonitored urban environments. This paper proposes the development of a mobile application that utilizes advanced STT and NLP technologies, integrated within a cloud computing environment, to immediately detect and respond to violent attacks. The proposed solution focuses on allowing users to activate emergency alerts through predefined voice commands, thereby streamlining the request for assistance and reducing response times in critical situations.

II. MATERIALS AND METHODS

The phases of the proposed approach to develop a mobile application are focused on the immediate detection and response to violent situations using advanced technologies such as STT and NLP. The phases of this approach are as follows: (1) selection of an STT service with high transcription accuracy and multilingual support, (2) selection of an NLP service that enables semantic analysis and contextual interpretation of commands, (3) implementation of STT and NLP within the mobile application for efficient detection and response, and (4) development of the mobile application, ensuring an intuitive interface and a robust architecture for deployment.

A. Selection of Speech-to-Text Service

The selection of the STT service was conducted through benchmarking, considering four services provided by different companies: Google Cloud STT (SE1), AWS Transcribe (SE2), Azure Cognitive Services STT (SE3), and AssemblyAI (SE4). Six criteria were considered in the evaluation: word accuracy rate (C1), information preservation rate (C2), latency (C3), price (C4), supported languages (C5), and additional services (C6). The first criterion, word accuracy rate (C1), measures the accuracy of word transcription from audio files. The second criterion, information preservation rate (C2), represents the proportion of words correctly preserved in the hypothesis compared to the reference. The third criterion, latency (C3), is important to determine the time taken to send and receive information. The fourth criterion, price (C4), was estimated based on an approximate usage of 1440 min of audio per user per month. The fifth criterion, supported languages (C5), ensures functionality in Peru, with Spanish as the primary language. Finally, the sixth criterion, additional services (C6),

includes the different extra services provided by each solution. Table I presents a pairwise comparison matrix, where a "1" indicates that the criterion in the row is equally or more important than the criterion in the column, and a "0" indicates it is less important. Diagonal cells were omitted. The results showed that latency (C3) and additional services (C6) were the most important criteria, each with a weight of 22.73%. Table II summarizes the benchmarking results of the four STT services, showing a score (S) and average (A) for each. Google Cloud STT (SE1) achieved the highest score of 3.59 and was thus selected for the application's development.

TABLE I. CRITERIA PAIRWISE COMPARISON MATRIX

Criteria	C1	C2	C3	C4	C5	C6	Total	Weight
C1	—	1	1	1	0	1	4	18%
C2	0	—	1	1	0	1	3	14%
C3	1	1	—	1	1	1	5	23%
C4	1	0	0	—	0	1	2	9%
C5	1	1	0	0	—	1	3	14%
C6	1	1	1	1	1	—	5	23%
Total							22	100%

TABLE II. STT SERVICE BENCHMARKING

Criterion	Weight	SE1		SE2		SE3		SE4	
		S	**A**	**S**	**A**	**S**	**A**	**S**	**A**
C1	18%	3	0.55	2	0.39	4	0.73	5	0.91
C2	14%	3	0.41	2	0.27	4	0.55	5	0.68
C3	23%	4	0.91	3	0.68	2	0.45	5	1.14
C4	9%	2	0.18	3	0.27	3	0.27	5	0.45
C5	14%	3	0.41	4	0.55	4	0.55	1	0.14
C6	23%	5	1.14	4	0.91	4	0.91	1	0.23
Total	100%		3.59		3.05		3.45		3.55

B. Selection of Natural Language Processing Service

The selection of the NLP service was also conducted through benchmarking, considering four services: Google Cloud Natural Language AI (N1), Amazon Comprehend (N2), Microsoft Azure Cognitive Services - Text Analytics (N3), and IBM Watson Natural Language Understanding (N4). Six criteria were evaluated: accuracy and effectiveness (K1), speed and performance (K2), multilingual capabilities (K3), ease of implementation and integration (K4), scalability and flexibility (K5), and cost-effectiveness (K6). Table III shows the pairwise comparison matrix, using the same methodology as in the STT evaluation. The most important criteria were speed and performance (K2) and ease of implementation and integration (K4), each with a weight of 31.25%. Table IV summarizes the benchmarking results of the NLP services, with Google Cloud Natural Language AI (N1) achieving the highest score of 3.50 and selected for application development.

TABLE III. CRITERIA PAIRWISE COMPARISON MATRIX

Criteria	K1	K2	K3	K4	K5	K6	Total	Weight
K1	—	0	0	0	1	1	2	13%
K2	1	—	1	1	1	1	5	31%
K3	1	0	—	0	1	0	2	13%
K4	1	1	1	—	1	1	5	31%
K5	0	0	0	0	—	1	1	6%
K6	0	0	1	0	0	—	1	6%
Total							16	100%

TABLE IV. NLP SERVICE BENCHMARKING

Criterion	Weight	N1		N2		N3		N4	
		S	**A**	**S**	**A**	**S**	**A**	**S**	**A**
K1	13%	4	0.50	4	0.50	4	0.50	4	0.50
K2	31%	5	0.63	4	0.50	5	0.63	4	0.50
K3	13%	5	0.63	4	0.50	4	0.50	4	0.50
K4	31%	5	0.63	5	0.63	5	0.63	4	0.50
K5	6%	5	0.63	5	0.63	5	0.63	4	0.50
K6	6%	4	0.50	4	0.50	4	0.50	3	0.38
Total	100%		3.50		3.25		3.38		2.88

C. Implementation of Speech-to-Text and Natural Language Processing

To convert speech into real-time text, the application uses the Google STT service (Figure 1), which can be easily integrated with Flutter via packages. Initially, the service is configured for continuous listening, allowing voice command capture. This is achieved by initializing the speech_to_text library, which checks device compatibility and requests necessary permissions. Once active, the service processes incoming audio and converts speech into text. The resulting text is compared to a predefined activation word, and if it matches, an alert is triggered. Once the alert is activated, the recognized text is sent to the Google Cloud Natural Language API (Figure 2) for deeper analysis. This service interprets the intent behind the recognized speech, offering additional capabilities such as key entity detection or sentiment analysis. Integration is performed via HTTP requests, sending the text to the API and receiving a processed response. The combined use of STT and the Cloud Natural Language API enables not only transcription but also understanding of the alert's context. This information is stored in the Firestore real-time database along with GPS location, date, and time, enabling a rapid emergency response.

D. Application Development

The application development follows a multilayer architecture (Figure 1) organized into five main layers: User Layer, Devices Layer, Connectivity Layer, Frontend Layer, and Backend Layer, facilitating system organization and responsibility separation:

- User Layer: This layer includes two user types: Citizens and Observers. Citizens are users generating emergency alerts through the mobile app, while Observers monitor and respond to alerts, typically from PCs or laptops.
- Devices Layer: Users access the app via different devices depending on their role. Citizens use smartphones to generate alerts and provide real-time GPS locations, whereas Observers use PCs or laptops to monitor alerts and view event details.
- Connectivity Layer: Communication occurs through mobile data, WiFi, or Ethernet. Mobile data and WiFi are common for smartphones, whereas Observers use WiFi or Ethernet to ensure a stable connection.
- Frontend Layer: The frontend includes the mobile and web applications. The mobile app, developed with Dart and Flutter, provides an intuitive interface for Citizens,

following the Model-View-ViewModel (MVVM) pattern to separate UI from business logic. It includes speech recognition, emergency contact management, and alert history features. Figure 2 shows interface examples: main menu (Figure 2(a)), alert history (Figure 2(b)), contact management (Figure 2(c)), and user profile (Figure 2(d)). The web app, developed with Angular, allows Observers to monitor alerts with detailed event visualization.

- Backend Layer: This layer hosts cloud services: Google STT, Cloud Natural Language API, and Firestore. STT converts voice into real-time text to detect emergency keywords. The Natural Language API analyzes message content for intent or entity extraction. Firestore stores alert-related data, including GPS, date, and time, enabling instant data synchronization between mobile and web applications.

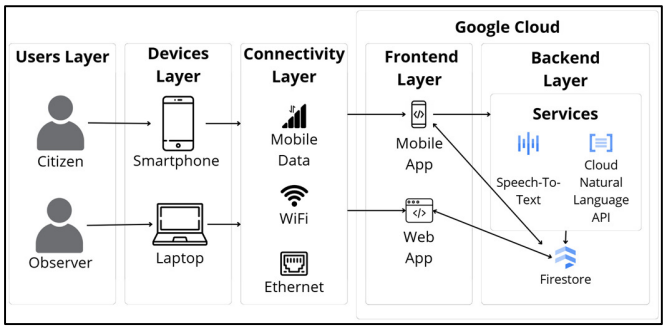
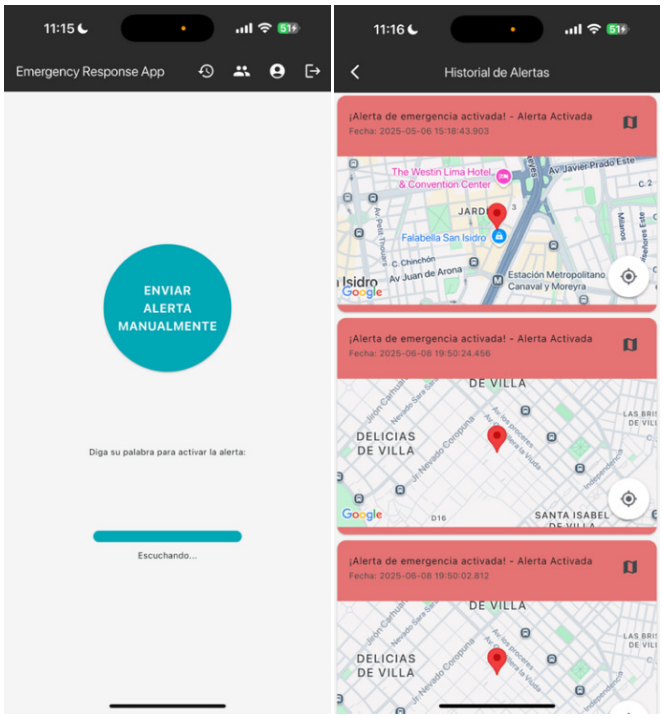
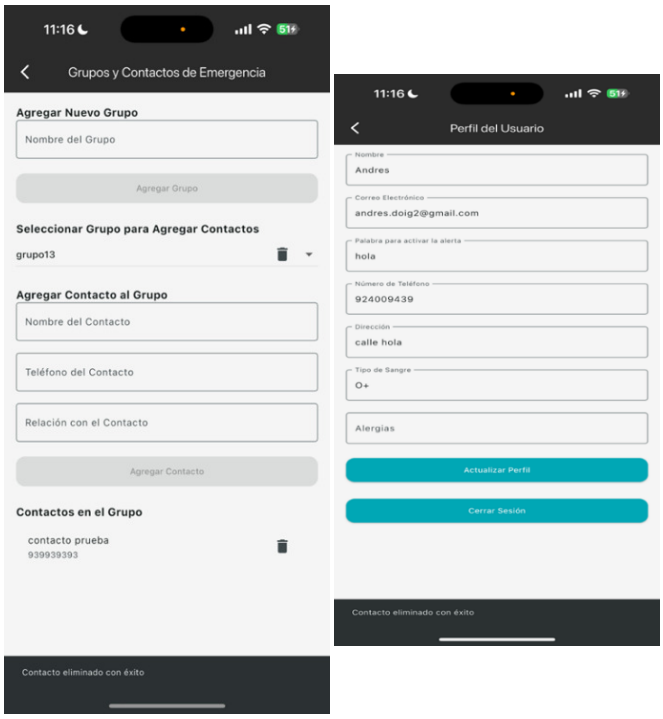


Fig. 1. Solution architecture.



(a) (b)



(c) (d)

Fig. 2. Application interfaces: (a) main menu, (b) alert history, (c) contact management, (d) user profile.

III. EXPERIMENTATION

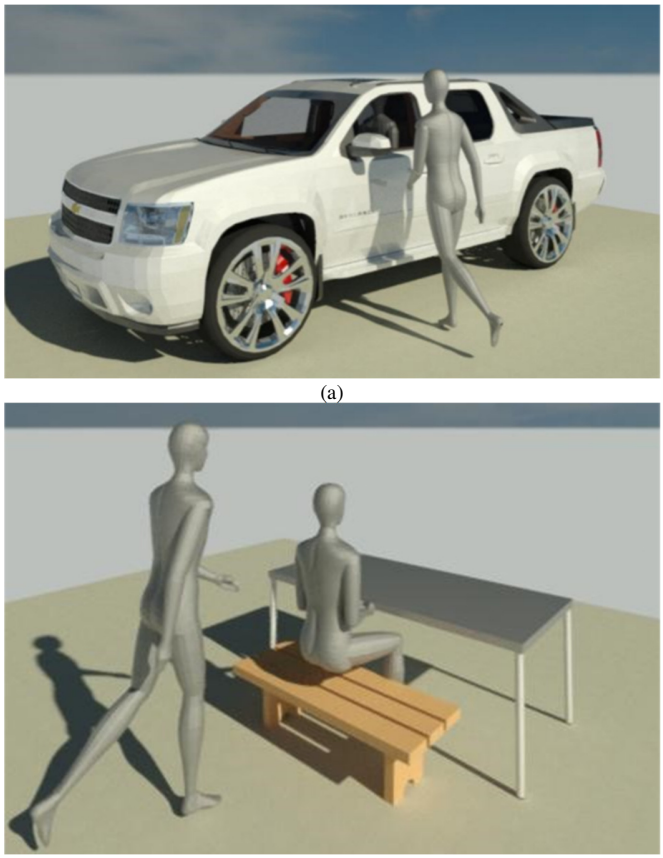
The experimentation phase was conducted at a shooting range located in Piura, Peru, involving 20 participants, including police officers, military personnel, security agents (both active and retired), and entrepreneurs who have either been victims of or are at risk of threats and extortion.

Two simulated scenarios were defined to validate the effectiveness of the proposed system. In Scenario 1 (S1), the victim is inside a vehicle, and an attacker simulates a robbery by forcing the door open (Figure 3(a)). In Scenario 2 (S2), the victim is seated at a table inside a venue or restaurant while an attacker attempts to steal their belongings (Figure 3(b)). In both scenarios, voice commands are used to trigger alerts via the mobile application. Additionally, a participant fires shots toward a secure area to generate noise, testing the effectiveness of speech recognition under adverse conditions. Monitoring is performed from a control table, where Observers record real-time response times through the web platform.

Table V presents the experimentation process. Each scenario will be evaluated both with and without the proposed system to compare efficiency in terms of Help Request Time (HRT) and Information Collection Time (ICT).

TABLE V. EXPERIMENTATION PROCESS

Experiment	Scenarios	Participants	Metrics
Without system	S1, S2	10 victims, 1 attacker, 1 observer	HRT, ICT
With system	S1, S2	10 victims, 1 attacker, 1 observer	HRT, ICT



(a) (b)

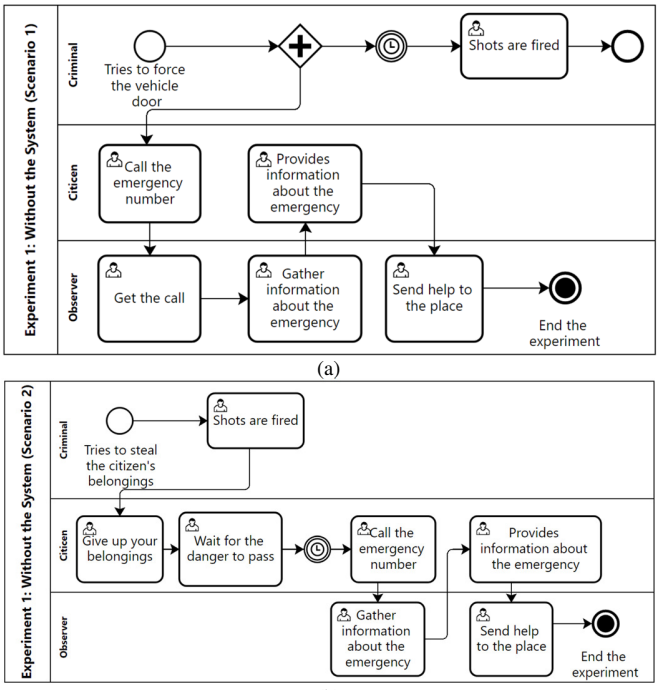
Fig. 3. 3D modeling of the experimentation scenarios: (a) Scenario 1, (b) Scenario 2.

A. Experiment 1: Without the System

Figure 4 shows Experiment 1 without the system, divided into two scenarios: an armed assault with the victim inside the vehicle (Figure 4(a)) and an armed assault with the victim seated at a table (Figure 4(b)).

In Scenario 1 without the mobile application (Figure 4(a)), the Citizen must manually make an emergency call when the assailant attempts to force open the vehicle door. After making the call, the citizen provides information about the situation. The Observer receives the call, gathers the necessary details, and sends help to the location. This process, requiring direct interaction with the device, may consume critical time and increase the risk to the Citizen.

In Scenario 2 without the mobile application (Figure 4(b)), the assailant attempts to steal the Citizen's belongings. After a waiting period, during which a gunshot may occur, the Citizen chooses to hand over their belongings and wait for the danger to pass. Once the situation de-escalates, the Citizen manually calls the emergency number and reports the incident. The Observer receives the call, collects the information, and dispatches help to the location, concluding the experiment. This process depends on the Citizen's ability to make a manual call, which may be difficult in high-risk situations.



(a) (b)

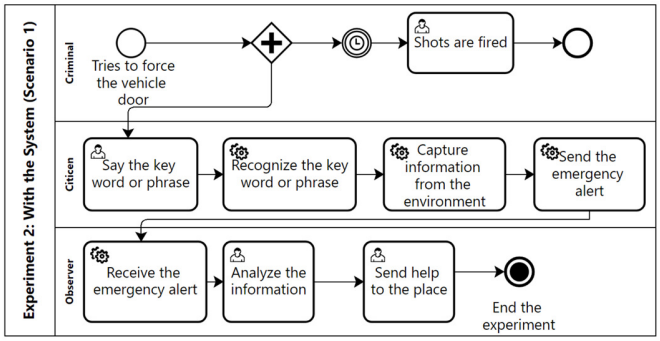
Fig. 4. Process diagram of Experiment 1: (a) Scenario 1, (b) Scenario 2.

B. Experiment 2: With the System

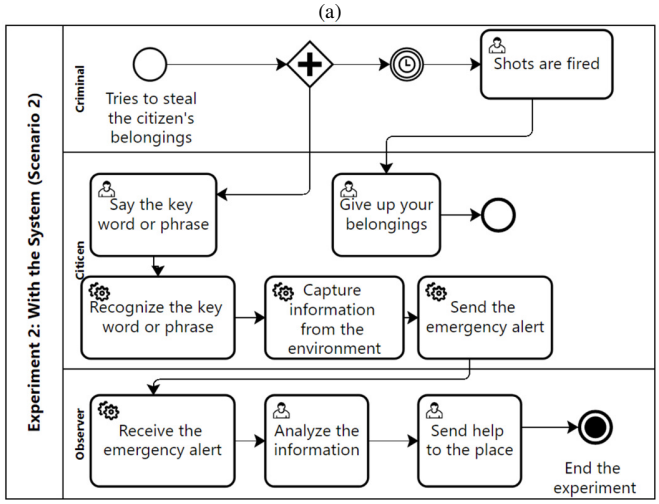
Figure 5 illustrates Experiment 2 with the system, also divided into two scenarios: (a) an armed assault with the victim inside a vehicle and (b) an armed assault with the victim seated at a table.

In Scenario 1 with the mobile application (Figure 5(a)), an assailant attempts to force open a vehicle door. Upon detecting the threat (Figure 6(a)), the Citizen activates the system by saying a predefined keyword. The application recognizes the keyword or key phrase (Figure 6(b)), captures relevant contextual information, and sends an emergency alert to an Observer (e.g., a security agent or authority). The Observer receives the alert, analyzes the information, and dispatches help to the specified location, concluding the experiment. This automated process enables a rapid response without requiring the user to manipulate a mobile device.

In Scenario 2 with the mobile application (Figure 5(b)), when the assailant attempts to steal the Citizen's belongings, the Citizen activates the emergency system using a predefined keyword or phrase. The mobile application promptly recognizes the command, gathers relevant contextual information, and automatically sends an alert to the Observer. Upon receiving the alert, the Observer reviews the information, evaluates the situation, and coordinates the dispatch of assistance to the Citizen's location. The process concludes when assistance is delivered, ending the experiment. The application allows for a quicker response without direct manual interaction, enabling the Citizen to obtain immediate support in emergency situations.

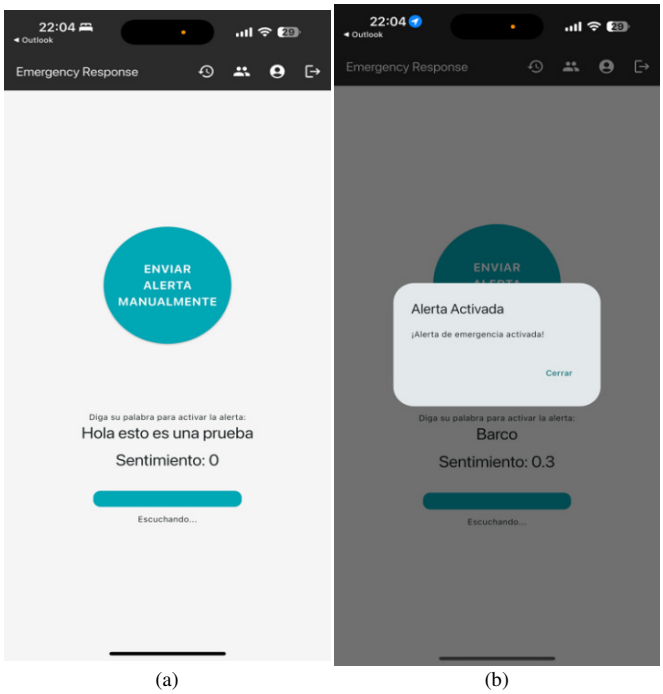


(a)



(b)

Fig. 5. Process diagram of Experiment 2: (a) Scenario 1, (b) Scenario 2.



(a) (b)

Fig. 6. Application interfaces during experimentation: (a) threat detection, (b) emergency alert activation.

C. Survey Design

To assess the usability of the proposed solution, a 10-question survey was administered to users (Table VI), including both Citizen users and Observer users. Each question was rated using a 5-point Likert scale (1 = strongly disagree, 2 = disagree, 3 = neutral, 4 = agree, and 5 = strongly agree). In addition, to evaluate the acceptance of the emergency mobile application, a survey was conducted with experts (Table VII). Questions were rated on a 5-point Likert scale (1 = very poor, 2 = poor, 3 = neutral, 4 = good, and 5 = very good).

TABLE VI. USER SURVEY QUESTIONS

Category	Question	
Effectiveness	Q1	I found the system's various features to be well integrated.
	Q2	I felt very secure using the system.
Efficiency	Q3	I found the system unnecessarily complex.
	Q4	I think I would need technical support to use this system.
	Q5	I felt there was too much inconsistency in the system.
	Q6	I found the system very awkward to use.
	Q7	I needed to learn many things before I could get going with this system.
Satisfaction	Q8	I think I would like to use this system frequently.
	Q9	I thought the system was easy to use.
	Q10	I imagine that most people would learn to use this system very quickly.

TABLE VII. EXPERT SURVEY QUESTIONS

Category	Question	
Perceived usefulness	Q1	I believe the system improves my job performance.
	Q2	Using this system increases my productivity.
	Q3	This system facilitates my daily tasks.
	Q4	I find this system useful for completing my work tasks.
	Q5	Using this system allows me to work more efficiently.
Perceived ease of use	Q6	Learning to use the system is easy for me.
	Q7	I find it easy to operate the system.
	Q8	Interaction with the system is clear and understandable.
	Q9	The system is flexible to use.
	Q10	It is easy to remember how to use the system.
Intention to use	Q11	If given the opportunity, I would use this system frequently.
	Q12	I intend to use this system in the future.
	Q13	I would like to use this system as part of my work tools.
	Q14	I would recommend this system to colleagues or friends.
Overall satisfaction	Q15	I am satisfied with the system's functionality.
	Q16	The overall user experience with the system is positive.
	Q17	I believe the system meets my expectations.

IV. RESULTS AND DISCUSSION

The objective of the experiment was to evaluate the impact of using the mobile application on reducing response times in emergency situations. Tests were conducted in two representative armed assault scenarios, both with and without the application. Three key metrics were recorded: (1) HRT, which measures how long it takes for the victim to initiate the help request; (2) ICT, which evaluates the time required to complete the alert submission process; and (3) Total Time (TT), which is the sum of the two previous metrics. The experimentation consisted of 40 tests, with 10 repetitions for each combination of scenario and usage condition. Tables VIII and IX present the values obtained in each test, organized by scenario and usage condition, whereas Table X shows the average values per scenario, based on the results of the 40 tests performed.

TABLE VIII. RESULTS OF THE 20 TESTS PERFORMED IN SCENARIO 1

Scenario 1							
With the app				Without the app			
Test No.	HRT (s)	ICT (s)	TT (s)	Test No.	HRT (s)	ICT (s)	TT (s)
1	8.10	9.75	17.85	11	11.33	24.52	35.85
2	10.14	8.26	18.40	12	10.46	26.61	37.07
3	8.93	9.08	18.01	13	10.92	27.95	38.87
4	9.06	8.92	17.98	14	11.90	24.10	36.00
5	9.08	9.07	18.15	15	10.09	22.30	32.39
6	11.49	9.06	20.55	16	9.66	22.25	31.91
7	8.04	8.85	16.89	17	11.23	25.81	37.04
8	7.14	8.92	16.06	18	11.96	27.36	39.32
9	8.90	9.04	17.94	19	12.65	23.14	35.79
10	8.14	8.97	17.11	20	12.07	24.08	36.15

TABLE IX. RESULTS OF THE 20 TESTS PERFORMED IN SCENARIO 2

Scenario 2							
With the app				Without the app			
Test No.	HRT (s)	ICT (s)	TT (s)	Test No.	HRT (s)	ICT (s)	TT (s)
1	11.62	8.11	19.73	11	13.87	25.05	38.92
2	15.15	8.81	23.96	12	14.06	22.07	36.13
3	12.11	11.81	23.92	13	13.28	24.36	37.64
4	12.22	9.01	21.23	14	13.66	27.78	41.44
5	11.49	9.08	20.57	15	16.07	24.60	40.67
6	13.06	8.97	22.03	16	12.31	25.17	37.48
7	11.78	8.88	20.66	17	12.99	23.35	36.34
8	13.71	9.02	22.73	18	14.03	24.98	39.01
9	12.49	9.05	21.54	19	14.09	25.49	39.58
10	9.91	8.95	18.86	20	13.03	23.52	36.55

TABLE X. AVERAGE TIMES PER EVALUATED SCENARIO

Scenario	App usage	HRT (s)	ICT (s)	TT (s)
1	With the app	8.90	8.99	17.89
	Without the app	11.23	24.81	36.04
2	With the app	12.35	9.17	21.52
	Without the app	13.74	24.64	38.38

As shown in Table X, the TT was reduced by 50.35% in Scenario 1 and by 43.92% in Scenario 2. The difference in TT between scenarios is attributed to a slower reaction time in Scenario 2. Nevertheless, the system's low latency helped maintain stable communication times. Overall, the results confirm the system's effectiveness in reducing vulnerability and enabling a timelier response.

An HRT analysis was also performed for each test in both scenarios (Figure 7) to compare the results obtained using the application versus traditional methods.

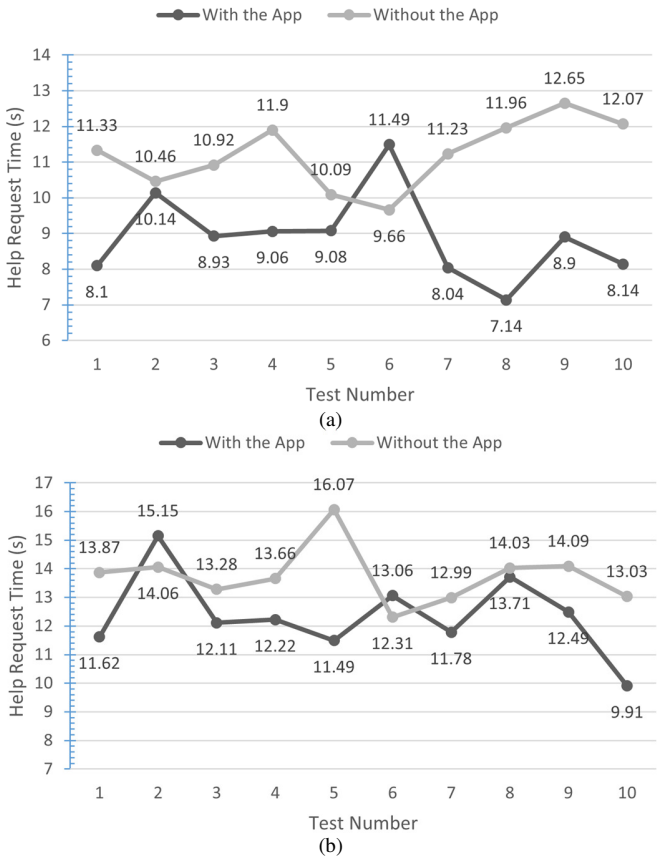


Fig. 7. HRT per test: (a) Scenario 1, (b) Scenario 2.

In Scenario 1 and Scenario 2 (Figures 7(a) and 7(b)), HRT was reduced by 20.71% and 10.08%, respectively. The use of the application showed greater effectiveness in Scenario 1, where the victim felt more protected inside the vehicle, allowing for a calmer reaction. Without the app, the victim must wait for the aggression to end before acting, which increases the response time. In Scenario 2, the difference was smaller due to the victim's direct exposure and the anxiety caused by the surprise attack.

An analysis was also performed on the ICT per test in each scenario (Figure 8) to compare the results with and without the application. In Scenario 1 and Scenario 2 (Figures 8(a) and 8(b)), ICT was reduced by 63.76% and 62.78%, respectively. ICT was consistent between scenarios when the application was used and significantly lower compared to traditional methods. This is due to the automation provided by speech recognition and NLP technologies. Without the application, human intervention increases variability and the time required to complete the process.

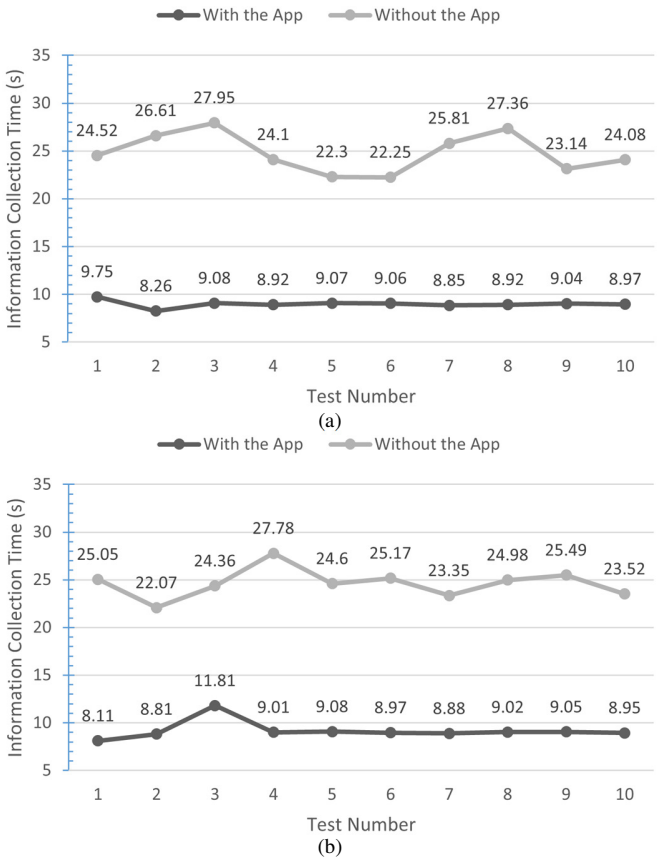


Fig. 8. HRT per test: (a) Scenario 1, (b) Scenario 2.

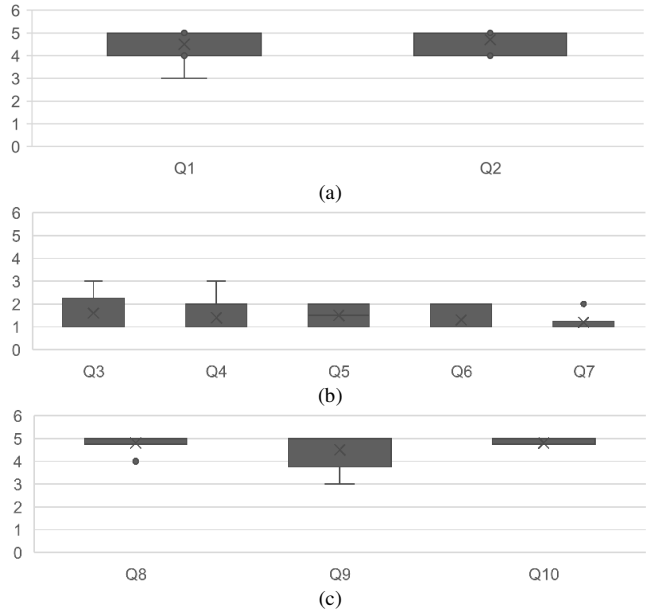


Fig. 9. User survey results: (a) effectiveness, (b) efficiency, (c) satisfaction.

A user survey was conducted, and the results are presented in Figure 9. For effectiveness (Figure 9(a)) and satisfaction (Figure 9(c)), which are based on positive statements, the average scores were 4.6 and 4.7, both interpreted as "Very Good." For efficiency (Figure 9(b)), which includes negatively worded items, the average score was 1.4, also interpreted as "Very Good." An expert survey was also performed, and the results are presented in Figure 10. Perceived usefulness and perceived ease of use received average scores of 4.38 and 4.44, interpreted as "Good", whereas intention to use and overall satisfaction received average scores of 4.58 and 4.57, respectively, both interpreted as "Very Good."

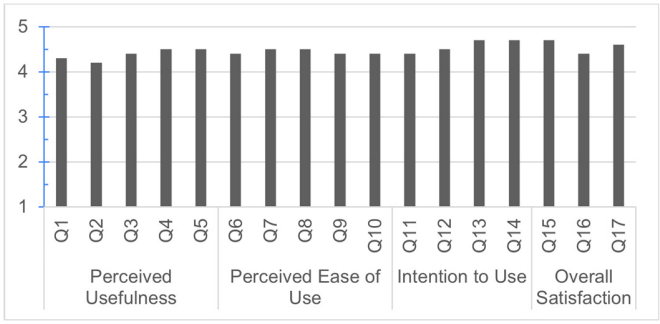


Fig. 10. Expert survey results.

V. CONCLUSIONS AND FUTURE WORK

In this study, a mobile application was developed for the immediate detection and response to violent attacks, using Google Speech-to-Text (STT) and Natural Language Processing (NLP) APIs for speech recognition and emotion analysis, respectively. The solution was tested in simulated scenarios representing urban assaults, allowing analysis of the system's impact in terms of Help Request Time (HRT), Information Collection Time (ICT), and Total Time (TT). The experimentation involved two simulated scenarios: Scenario 1, an assault inside a vehicle, and Scenario 2, an assault in a restaurant. With the participation of one observer, one attacker, and 10 test subjects, a total of 40 trials were conducted at a shooting range in Piura, Peru. The experimental results showed a significant reduction in TT, with improvements of 50.35% and 43.92% in Scenario 1 and Scenario 2, respectively. This demonstrates the operational value of the solution in critical situations where manual phone use is not feasible. The application performed efficiently even under limited mobile signal conditions; during the tests, a 3G network was used, which may produce different results compared to Wi-Fi or 4G/5G networks. Nevertheless, the results confirm that the solution is robust and functional across various connectivity scenarios.

This work presents a novel integration of real-time STT and contextual NLP technologies for hands-free emergency activation, distinguishing it from existing systems that rely on manual interactions during high-stress violent encounters. The key contributions include: (1) a systematic benchmarking framework for optimal STT and NLP service selection considering accuracy, latency, and multilingual capabilities; (2) empirical validation in realistic simulation environments with security professionals, demonstrating significant reductions in response times; and (3) a scalable hybrid cloud-AI architecture combining speech recognition with automated geolocation and

database synchronization. User surveys revealed very high levels of effectiveness and satisfaction, whereas expert surveys indicated strong intention to use and overall satisfaction, highlighting the system as useful, intuitive, and easy to use, with a balanced design that facilitates adoption and enhances the user experience.

As future work, the solution will be tested in real urban environments, involving public security entities for institutional and large-scale validation.

ACKNOWLEDGMENT

The authors extend their gratitude to the Research Department of the Universidad Peruana de Ciencias Aplicadas for their support through the UPC-Expost-2025-2 incentive.

REFERENCES

[1] J. Inquilla Mamani, M. López Cueva, E. Catacora Vidangos, and E. Flores Mamani, "La morfología de la criminalidad urbana en el Perú: un análisis de tendencias, niveles y factores de riesgo," *Andamios*, vol. 21, no. 55, pp. 411–435, Aug. 2024, https://doi.org/10.29092/uacm.v21i55.1110.

[2] "Estadísticas de Criminalidad, Seguridad Ciudadana y Violencia. Julio-Setiembre 2024," Instituto Nacional de Estadística e Informática. [Online]. Available: https://www.gob.pe/institucion/inei/informes-publicaciones/6334643-estadisticas-de-criminalidad-seguridad-ciudadana-y-violencia-julio-setiembre-2024?utm_source=chatgpt.com.

[3] A. Swaminathan *et al.*, "Natural language processing system for rapid detection and intervention of mental health crisis chat messages," *NPJ Digital Medicine*, vol. 6, Nov. 2023, Art. no. 213, https://doi.org/10.1038/s41746-023-00951-3.

[4] A. Bakhshi, J. García-Gómez, R. Gil-Pita, and S. Chalup, "Violence Detection in Real-Life Audio Signals Using Lightweight Deep Neural Networks," *Procedia Computer Science*, vol. 222, pp. 244–251, Jan. 2023, https://doi.org/10.1016/j.procs.2023.08.162.

[5] A. S. M. Mohsin and M. A. Muyeed, "IoT based smart emergency response system (SERS) for monitoring vehicle, home and health status," *Discover Internet of Things*, vol. 4, no. 1, Oct. 2024, Art. no. 22, https://doi.org/10.1007/s43926-024-00073-6.

[6] N. Hasan and M. F. Ahmed, "Wearable Technology for Elderly Care: Integrating Health Monitoring and Emergency Alerts," *Journal of Computer Networks and Communications*, vol. 2024, no. 1, Nov. 2024, Art. no. 5593708, https://doi.org/10.1155/jcnc/5593708.

[7] Y. Obuchi and T. Sumiyoshi, "Intentional Voice Command Detection for Trigger-Free Speech Interface," *IEICE Transactions on Information and Systems*, vol. E93.D, no. 9, pp. 2440–2450, Jan. 2010, https://doi.org/10.1587/transinf.E93.D.2440.

[8] E. Veltmeijer, M. Franken, and C. Gerritsen, "Real-time violence detection and localization through subgroup analysis," *Multimedia Tools and Applications*, vol. 84, no. 7, pp. 3793–3807, Feb. 2025, https://doi.org/10.1007/s11042-024-19144-5.

[9] S. C. Venkateswarlu, S. R. Jeevakala, N. U. Kumar, P. Munaswamy, and D. Pendyala, "Emotion Recognition From Speech and Text using Long Short-Term Memory," *Engineering, Technology & Applied Science Research*, vol. 13, no. 4, pp. 11166–11169, Aug. 2023, https://doi.org/10.48084/etasr.6004.

[10] M. D. Vu, H. Wang, Z. Li, G. Haffari, Z. Xing, and C. Chen, "Voicify Your UI: Towards Android App Control with Voice Commands," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 7, no. 1, Mar. 2023, Art. no. 44, https://doi.org/10.1145/3581998.

[11] H. Tolle *et al.*, "From voice to ink (Vink): development and assessment of an automated, free-of-charge transcription tool," *BMC Research Notes*, vol. 17, no. 1, Mar. 2024, Art. no. 95, https://doi.org/10.1186/s13104-024-06749-0.

[12] C. M. G. Villame and S. Guirnaldo, "Design and implementation of voice-command controller for fixed-wing unmanned aerial vehicles using automatic speech recognition and natural language processing techniques," *Sustainable Engineering and Innovation*, vol. 6, no. 2, pp. 199–212, Oct. 2024, https://doi.org/10.37868/sei.v6i2.id309.

[13] Z. Li, "Leveraging AI automated emergency response with natural language processing: Enhancing real-time decision making and communication," *Applied and Computational Engineering*, vol. 71, pp. 1–6, Aug. 2024, https://doi.org/10.54254/2755-2721/71/20241629.

[14] A. Tundis, H. Kaleem, and M. Mühlhäuser, "Detecting and Tracking Criminals in the Real World through an IoT-Based System," *Sensors*, vol. 20, no. 13, Jul. 2020, Art. no. 3795, https://doi.org/10.3390/s20133795.