Threat Eye - An Application of Safety Alert System for Real Time Threat Detection

Tejas S[1], Shilpa R[2]

[1]Student, Department of Computer Science & Engineering, SDM Institute of Technology, Ujire
Email: *tejasgowdaurt[at]gmail.com*

[2]Assitant Professor, Department of Computer Science & Engineering, SDM Institute of Technology, Ujire
Email: *shilparayee[at]gmail.com*

Abstract: ***The "Emergency Threat Detection Android Application" addresses the critical need for swift and reliable emergency response by leveraging advancements in AI and mobile technology. Using Natural Language Processing (NLP), the app analyses user inputs, such as text and speech, to detect emergencies in real time by identifying distress keywords like "help me," "fire," or "accident." Upon detection, it triggers automated actions, including notifying emergency contacts, sending location-based alerts, and providing assistance options. The app features a user-friendly interface, speech-to-text conversion, and a supervised NLP classifier trained on emergency data for precise threat identification. It integrates GPS for real-time location tracking and cloud-based APIs for advanced functionalities like messaging and speech recognition. Designed with scalability in mind, the app supports future enhancements such as multilingual capabilities, smart device integration, and enhanced privacy measures. By reducing response times and improving outcomes during crises, this project demonstrates the potential of combining NLP and mobile technologies to enhance personal safety and emergency management.***

Keywords: Safety alert system, Real-time threat detection, Speech recognition, Location tracking, Real-time image capturing

1. Introduction

Emergencies, ranging from personal crises to large-scale threats, are unavoidable, but advancements in AI and mobile technology offer promising solutions for timely detection and response. The "Emergency Threat Detection Android Application" aims to address the gap in emergency management systems by detecting and responding to threats in real-time. Traditional emergency services often suffer from delays due to manual communication or reliance on bystanders, but this system integrates Natural Language Processing (NLP) and mobile computing to analyse text and voice inputs for urgent keywords, ensuring accurate and swift threat identification.

The application's architecture includes a user-friendly interface, backend NLP algorithms for threat detection, and integrated communication tools like GPS for location tracking and SOS messaging. It also provides immediate response recommendations based on the detected threat, such as notifying emergency contacts and sending real-time location alerts. The key objectives are real-time threat detection, automated emergency responses, an easy-to-use interface, integration with mobile features like GPS and speech-to-text conversion, and scalability for future enhancements, such as multilingual support and wearable device integration.

This project aims to reduce response times, improve communication with emergency services, and enhance personal safety. By integrating AI and NLP, the application represents a significant step toward smarter emergency management systems that can save lives and mitigate risks. The project's comprehensive approach,

combining intelligent threat detection with automated responses and mobile features, ensures faster and more efficient responses to critical situations. Future chapters will explore the system's design, algorithms, and its potential impact on public safety.

2. Literature Review

Javed and Luo (2020) introduced an SOS Intelligent Emergency Rescue System designed to enable rapid emergency communication through a single-tap mechanism that activates voice input. This system employs real-time voice recognition to improve accessibility and response times. The integration of intelligent algorithms and user-friendly mobile interfaces addresses the shortcomings of traditional emergency systems [1].

Gosavi et al. (2021) developed a system aimed at enhancing women's safety by leveraging GPS and mobile technology for real-time location sharing and emergency contact notifications. This system enables users to send distress signals and their live location to pre-registered contacts, emphasizing the importance of reliable and timely communication for personal security [2].

Uma et al. (2015) created an Android-based application that uses voice recognition technology to detect distress signals and initiate safety mechanisms, such as sending SOS messages or contacting emergency services. This voice-activated system highlights the value of hands-free functionality in high-stress scenarios, facilitating more accessible safety applications [3].

Shinde et al. (2015) proposed a real-time vehicle monitoring and tracking system that combines an embedded Linux board with an Android application. The system provides reliable GPS tracking and real-time communication, offering a robust solution for vehicle monitoring. Its adaptability to broader applications, including personal safety, is notable [4].

Dafallah (2014) designed an accurate real-time GPS tracking system focused on precision and reliability in dynamic environments. By incorporating GPS modules and real-time processing, this system addresses challenges related to tracking accuracy, forming a foundation for advanced GPS-based safety and emergency technologies [5].

Google Developers (2024) highlight the functionality of the Google Speech-to-Text API, which provides advanced real-time speech recognition and transcription capabilities. Supporting multiple languages and offering seamless integration with mobile applications, this API simplifies the development of voice-driven systems. Its adaptability and efficiency make it invaluable for creating accessible safety applications, particularly those requiring quick and reliable voice input [6]. Twilio (2024) emphasizes the features of its Programmable SMS API, which enables the integration of SMS functionalities into various applications. This API ensures reliable and efficient real-time messaging, essential for delivering distress signals and emergency notifications. Its robust architecture supports critical communication systems, ensuring messages are delivered promptly during emergencies [7]. Firebase (2024) details the capabilities of the Firebase Realtime Database, a cloud-hosted solution designed to synchronize data across devices in real-time. The database supports immediate updates, making it an ideal choice for applications requiring continuous data sharing, such as live location tracking and safety status notifications. Its versatility and reliability enhance the responsiveness of emergency systems [8]. Paradkar and Sharma (2015) propose an all-in-one intelligent safety system tailored to women's security needs, integrating GPS tracking, SOS alerts, and wearable devices into a unified platform. This comprehensive system merges various safety mechanisms, providing a holistic solution to tackle personal security challenges effectively. Its design ensures a rapid and coordinated response to potential threats [9].

TensorFlow (2024) offers advanced natural language processing (NLP) tools that facilitate the creation of intelligent systems capable of understanding and processing human language. These tools enable the development of context-aware voice recognition systems, improving user interaction and functionality. This adaptability enhances safety applications, allowing them to respond intelligently in emergency scenarios [10].

Recent literature on ERS highlights a movement away from traditional, rigid systems towards more flexible, interactive platforms capable of adapting to dynamic crisis environments. Since 2020, significant advancements have been made in emergency response systems (ERS), with an emphasis on shifting away from traditional one-way notification methods to more interactive, real-time systems. One notable work by Ma Thomas, Fk Andoh-Baidoo, and S George (2005) introduced the concept of EVResponse, highlighting critical shortcomings in traditional systems. These systems, often non-interactive, are limited by their inability to adapt in real time to dynamic emergencies. The proposed EVResponse system integrates interactive technologies, such as VoiceXML, to facilitate two-way communication between responders and affected individuals. This shift allows for better coordination and information flow, crucial for effective crisis management [11].

In 2017, Shahrah and Al-Mashari explored the limitations of traditional Client/Server (C/S) models and advocated for the adoption of Browser/Server (B/S) models in WebGIS. This shift not only enhances system flexibility but also enables seamless integration with geographic data, vital for emergency management applications. The authors emphasized the importance of scalable, web-based platforms in managing large-scale emergencies, particularly when rapid information dissemination is critical [12]. Furthermore, Vivacqua and Borges (2012) discussed leveraging collective knowledge in ERS, where crowdsourcing and community participation can greatly enhance situational awareness. Their work illustrated that real-time information from individuals involved in or witnessing an emergency can provide valuable insights, aiding in more informed decision-making and faster responses [13]. Another significant contribution comes from Turoff, Chumer, and de Walle (2004), who focused on dynamic emergency response management information systems (DERMIS). Their work underscored the need for flexibility and adaptability in ERS, particularly in unpredictable disaster scenarios. The DERMIS framework was designed to be responsive to a variety of crisis types, providing a more fluid and efficient emergency management process [14]. The transition towards interactive and scalable systems, as demonstrated by these studies, reflects a broader trend in the evolution of emergency response technologies. As highlighted in recent literature, including the work by Bram and Vestergren (2012), the development of more advanced ERS that integrate real-time data, crowdsourced information, and dynamic decision-making capabilities is a necessary step toward more effective crisis management. As technology continues to evolve, future emergency response systems are likely to incorporate even more sophisticated features, such as real-time geospatial data, artificial intelligence, and community-driven contributions, further enhancing their efficiency and effectiveness in handling emergencies [15].

Ghazal et al. (2016) demonstrated the increasing reliance on smartphones and mobile technologies to improve the effectiveness of emergency response. They developed a smart mobile-based system for reporting community technological emergencies, integrating a web portal for authorities to receive notifications and alerts, enabling quick response actions during emergencies [16]. Amailef and Lu (2011) introduced a mobile-based emergency response system (MERS) designed to provide intelligent government services. They highlighted the conceptual and technical frameworks of MERS, discussing how mobile technologies can enhance response efficiency by offering a structured platform for interaction between emergency responders and affected individuals [17]. Gaziel-Yablowitz and Schwartz (2018) assessed mobile-based emergency intervention apps, emphasizing key criteria for evaluating these systems. Their work identified essential factors for assessing mobile emergency systems, such as user accessibility, response time, and integration with emergency management agencies [18].

Al-Khafajiy et al. (2019) presented the Smart Hospital Emergency System, a mobile-based platform that connects patients with emergency services. This system allows for quicker and more efficient responses during medical emergencies, highlighting how mobile technologies can streamline healthcare services during crises [19]. Tian et al. (2014) explored the use of mobile-based systems in supporting emergency evacuation decision-making. They developed an agent-based simulation model that gathers medical and temporal data during emergencies, providing valuable insights for effective evacuation planning [20]. Dzulkarnain et al. (2016) developed a mobile-based mitigation and emergency system for landslides in East Java, Indonesia. Their system aims to provide real-time information to local authorities and residents, helping them prepare and respond to landslide threats more effectively [21]. Aydin et al. (2016) focused on improving disaster resilience through mobile-based disaster management systems. They emphasized the role of mobile applications in enhancing citizen participation in emergency response efforts, thus promoting community involvement and strengthening disaster resilience [22]. These studies illustrate the growing importance of mobile-based systems in enhancing emergency response capabilities. Mobile technologies provide real-time data, improve communication between emergency responders and affected individuals, and enable quicker, more informed decision-making. As mobile technology continues to advance, these systems are likely to play an increasingly critical role in managing emergencies and mitigating disaster impacts.

Singh and Roy (2018) focused on analyzing crowd sentiment to improve emergency response services. They used sentiment analysis on Twitter data and applied a change-point detection algorithm to identify shifts in public sentiment during disasters. By fine-tuning crowd perception through sentiment detection, their work demonstrated the potential for more adaptive and responsive emergency services [23]. Nagy and Stamberger (2012) explored the accuracy of crowd sentiment detection during crises, emphasizing its value in emergency management. They argued that sentiment analysis could provide critical insights into public concerns, perceptions, and behaviours during disasters, ultimately guiding better decision-making for crisis responders. Their work underscored the importance of social media and sentiment detection in emergency response systems [24]. Beigi et al. (2016) provided an overview of sentiment analysis in social media, specifically its application in disaster relief efforts. They discussed the state-of-the-art techniques used to analyze social media trends and sentiments, helping emergency teams track the evolving public mood and gather valuable data for decision-making. Their review suggested that sentiment analysis could accelerate trend detection and improve the speed and effectiveness of disaster response [25]. Huang et al. (2015) analyzed the performance of convolutional neural networks (CNNs) for speech recognition tasks, focusing on the Aurora 4 and Kinect distant speech recognition tasks. Their study demonstrated that CNNs, through learned convolutional kernels, could effectively capture relevant features for speech recognition, improving performance on both noisy and clean speech datasets. This work contributed to the growing use of deep learning techniques in speech recognition by showing that CNNs could handle complex acoustic patterns and environmental variations in speech data [26]. Anusuya and Katti (2011) reviewed the front-end analysis of speech recognition, covering various aspects such as spectral representations of the speech signal. They discussed different techniques for extracting features from the speech signal, which are crucial for improving the accuracy and efficiency of recognition systems. Their work highlighted the importance of preprocessing in speech recognition, including filtering, normalization, and feature extraction, to ensure that the system can handle different speech patterns and environments effectively [27]. Palaz and Collobert (2015) explored the use of raw speech signals as input for CNN-based speech recognition systems. They demonstrated that CNNs, when provided with raw speech, could learn important features without relying on traditional feature extraction methods like Mel-frequency cepstral coefficients (MFCCs). Their results indicated that CNN-based systems could outperform conventional methods in certain continuous speech recognition tasks, thus showing the potential of raw audio input in enhancing system performance and flexibility [28]. Hansen (1996) investigated speech recognition under stressful conditions such as noise and environmental stress. His work emphasized the importance of compensating for variations in speech, especially in challenging environments, to maintain robust recognition performance. By analyzing various speech distortion models and proposing solutions, Hansen provided valuable insights into improving the robustness of speech recognition systems, especially for real-world applications where environmental noise and stress factors affect speech quality [29]. Gaikwad et al. (2010) conducted a review on speech recognition techniques, focusing on error analysis and the enhancement of knowledge-based systems. They discussed the evolution of speech recognition systems, including the challenges posed by human speech variability and the limitations of early recognition models. Their work highlighted the importance of improving the accuracy of speech recognition systems through better feature extraction, error correction mechanisms, and adaptive models that can learn from real-world data [30].

3. Problem Identification

The challenges and limitations of the existing systems are examined in detail, along with a discussion of the proposed system. The goal is to address the gaps and shortcomings in emergency threat detection and response mechanisms through a more robust and intelligent solution.

3.1 Existing System

Recent advancements in speech recognition have seen significant contributions from convolutional neural networks (CNNs) and deep learning techniques. Huang et al. (2015) demonstrated the effectiveness of CNNs in capturing relevant speech features for tasks like Aurora 4 and Kinect distant speech recognition, improving performance in both clean and noisy conditions. Anusuya and Katti (2011) reviewed front-end analysis techniques, emphasizing the importance of feature extraction, spectral representation, and preprocessing in enhancing speech recognition systems. Palaz and Collobert (2015) explored the potential of using raw speech signals as input for CNN-based systems, showing that CNNs could outperform traditional methods like MFCCs in continuous speech recognition. Hansen (1996) addressed the challenges of speech recognition under stress and noise, proposing methods to improve robustness in real-world environments. Gaikwad et al. (2010) highlighted the evolution of speech recognition systems and the need for error correction and adaptive models to improve accuracy. Collectively, these studies underscore the transformative role of deep learning and CNNs in advancing speech recognition technologies, demonstrating their ability to handle diverse acoustic conditions and environmental factors.

3.2 Proposed System

The proposed system aims to address the limitations of current emergency response solutions by integrating advanced Natural Language Processing (NLP) with mobile technology, creating a proactive, intelligent, and user-friendly threat detection and response system. Key features include real-time threat detection using NLP algorithms to analyze text and voice inputs for distress signals, automated emergency response with notifications and location sharing, and advanced NLP models like sentiment analysis and speech recognition that adapt to different languages and accents. The system also integrates mobile features such as GPS, accelerometers, and offline functionality for critical tasks. Personalization and context awareness allow the system to learn from past interactions and adapt to individual preferences. Advantages of the system include proactive automation, scalability, flexibility, enhanced accessibility for users with disabilities, and seamless integration with emergency services. The system overcomes existing limitations by automating emergency detection, leveraging AI for real-time analysis, ensuring scalability, and prioritizing inclusivity and multilingual support.

4. Methodology and Architecture

This chapter outlines the step-by-step process used to design and implement the Emergency Threat Detection Android Application. It elaborates on the system design and describes the various modules that constitute the system. The methodology ensures that the app meets its objectives efficiently while providing a scalable and user-friendly solution.

Architecture of proposed Model
The system design for the Emergency Threat Detection Android Application adopts a modular, layered architecture to ensure scalability, maintainability, and seamless integration. Key layers include the User Interface (UI) for interaction, the Application Logic for core functionalities like NLP and decision-making, the Service Layer for integrating external services like GPS and APIs, the Database Layer for secure data storage, and the Communication Layer for real-time connectivity. The workflow involves emergency triggering via voice or manual input, voice recognition and NLP analysis, data processing for action determination, and a robust alert system for notifying emergency contacts with real-time location and evidence. The modular approach comprises distinct functionalities such as voice recognition, location tracking, notification, evidence collection, secure database management, and feedback analysis, streamlining development and enhancing system efficiency. This design addresses critical gaps in emergency response systems while ensuring functionality, scalability, and user-centric operation.

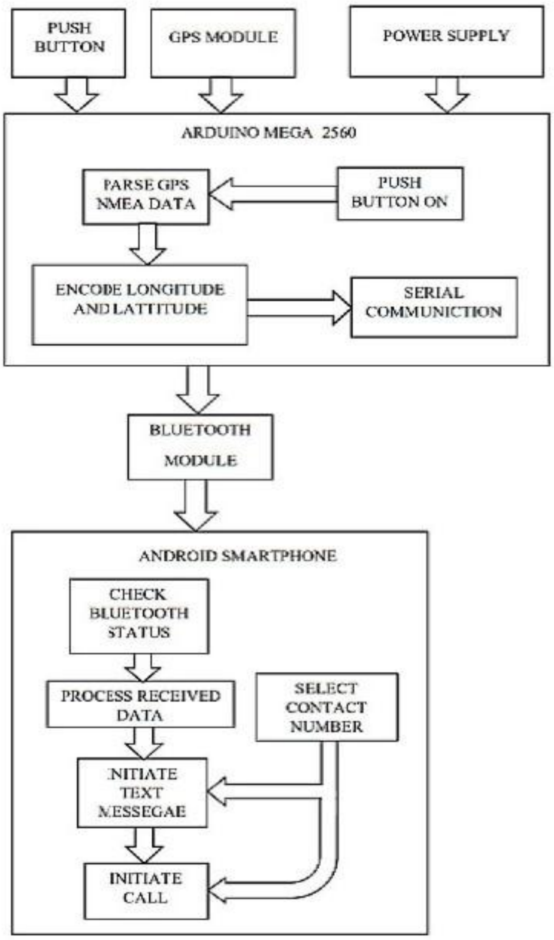

Figure 1: Architecture of the proposed system

5. Testing and Analysis

Testing is an essential phase in the software development lifecycle, ensuring the application performs as expected and meets the defined requirements. This chapter highlights the testing process for the Emergency Threat Detection Android Application, focusing on critical functionalities such as voice recognition, location tracking, and alert mechanisms. Each test case is meticulously designed to evaluate specific components and validate their functionality, ensuring reliability and accuracy. The successful execution of all test cases demonstrates the application's compliance with functional and non-functional requirements, confirming its readiness for deployment. The voice recognition module was tested for its ability to detect distress keywords accurately, while the location tracking module was validated for precision in capturing and sharing real-time user locations. Additionally, the alert mechanisms were examined to ensure timely and reliable notifications to emergency contacts. While the application passed all tests, further testing is recommended. Stress testing will evaluate the app's performance under high loads, and user testing will provide insights into its usability and effectiveness in real-world scenarios. These steps are crucial for enhancing robustness, ensuring seamless performance, and addressing any potential challenges users may encounter.

5.1 Sample Test Case Details

The following table provides a detailed overview of the test cases executed during the testing phase:

Sample Test Case Details
Test Case ID: TC01
Test Case Name: Voice Command Recognition
Objective: Verify that the system correctly recognizes and processes the distress phrase "Help me."

Preconditions:
The app is installed and configured with permissions for microphone access.

Steps:
1) Launch the app.
2) Speak the distress phrase "Help me."
3) Observe the system's response.

Expected Result:
- The app correctly identifies the phrase and triggers the SOS process.

Actual Result:
- The phrase was identified, and SOS was successfully triggered.

Status: Passed

Test Case ID: TC02

Test Case Name: Emergency Call Initiation

Objective: Verify that the app automatically places a call to the configured emergency service.

Preconditions:
The app has telephony permissions enabled.

Steps:
1) Activate the SOS function.
2) Monitor the app for automated call initiation.

Expected Result:
- The app places a call to the pre-configured emergency number (e.g., 911).

Actual Result:
- The call was successfully placed.

Status: Passed

6. Conclusion

The Emergency Threat Detection Android Application marks a significant advancement in leveraging technology to enhance personal safety and emergency response. By integrating Natural Language Processing (NLP) for voice recognition and location-based services, it addresses challenges such as delayed responses, manual intervention needs, and the lack of comprehensive situational data. The project achieved its goals by developing a reliable, scalable, and user-friendly application that automates threat detection, alerts emergency contacts, and provides real-time assistance. Key features include real-time threat detection using advanced NLP algorithms to recognize distress signals, multi-channel notifications via SMS, email, and calls to share location and context, and enhanced contextual awareness through features like photo capture. Its user-centric design offers customization, making it accessible to diverse users. Challenges included ensuring NLP accuracy, optimizing performance for mobile devices, and addressing data privacy, which were mitigated through pre-trained machine learning models, efficient coding, and robust encryption.

Future enhancements could include integration with wearable devices for broader accessibility, multilingual NLP support for global usability, AI-driven insights to predict emergencies, and cloud-based scalability for robust data storage and analytics. The application's ability to combine intelligent detection, automation, and real-time communication establishes a new standard for emergency response systems.

This project highlights the potential of modern technologies to address societal challenges. With continued refinement, the application can further enhance safety and responsiveness, bridging the gap between technological advancements and human security to create a safer environment for all.

References

[1] M. Javed and X. Luo, "SOS Intelligent Emergency Rescue System: Tap Once to Trigger Voice Input," in *Proceedings of the 4th International Conference on Computer Science and Artificial Intelligence (CSAI)*, 2020, pp. 150–156.
[2] A. P. Gosavi, Y. S. Chavan, H. P. Ghadi, H. D. Bandekar, and A. S. Dhuri, "Emergency Contact and Location Sharing System for Women Safety," *International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)*, vol. 5, no. 1, pp. 466–471, May 2021.
[3] D. Uma, V. Vishakha, R. Ravina, and B. Rinku, "An Android Application for Women Safety Based on Voice Recognition," *International Journal of Computer Science and Mobile Computing (IJCSMC)*, vol. 4, no. 3, pp. 216–220, Mar. 2015.
[4] P. A. Shinde, Y. Mane, and P. H. Tarange, "Real-Time Vehicle Monitoring and Tracking System Based on Embedded Linux Board and Android Application," in *Proceedings of the International Conference on Circuits, Power, and Computing Technologies (ICCPCT)*, IEEE, 2015, pp. 1–7.
[5] H. A. A. Dafallah, "Design and Implementation of an Accurate Real-Time GPS Tracking System," in *Proceedings of the Third International Conference on e-Technologies and Networks for Development (ICeND)*, IEEE, 2014, pp. 183–188.
[6] Google Developers, "Google Speech-to-Text API," [Online]. Available: https://cloud.google.com/speech-to-text/. [Accessed: Oct. 15, 2024].
[7] Twilio, "Programmable SMS API Documentation," [Online]. Available: https://www.twilio.com/docs/sms. [Accessed: Oct. 15, 2024].

[8] Firebase, "Firebase Realtime Database," [Online]. Available: https://firebase.google.com/products/realtime-database. [Accessed: Oct. 15, 2024].

[9] R. Paradkar and D. Sharma, "All-in-One Intelligent Safety System for Women Security," *International Journal of Computer Applications*, vol. 130, no. 11, pp. 33–40, Nov. 2015.

[10] TensorFlow, "Natural Language Processing with TensorFlow," [Online]. Available: https://www.tensorflow.org/text. [Accessed: Oct. 15, 2024].

[11] Rich and Conn (1995) assessed “call down” systems for emergency notifications, demonstrating their efficiency in public communication without causing panic. Their foundational insights underline the value of automated risk communication in public safety.

[12] Lee and Mihailidis (2005) developed an intelligent fall detection system using image-based sensors, tested with 21 subjects. Their work emphasizes supporting independent living and emergency management, marking a significant contribution to telemedicine.

[13] Ranganathan et al. (2007) proposed a game-theoretic optimization-based system for urban emergency management, effectively allocating response units to minimize risks in critical scenarios. Their study underscores advancements in emergency response optimization.

[14] Poulymenopoulou et al. (2012) explored mobile computing and cloud services for automating emergency healthcare processes, particularly triaging. Their work demonstrates improved efficiency in prioritizing emergency cases and showcases the potential of technology in healthcare automation.

[15] Ghazal, M., Ali, S., Al Halabi, M., & Ali, N. (2016). Smart mobile-based emergency management and notification system. *2016 IEEE 4th International Conference on Communications and Signal Processing (ICCSP)*. IEEE. Retrieved from ieeexplore.ieee.org

[16] Amailef, K., & Lu, J. (2011). A mobile-based emergency response system for intelligent m-government services. *Journal of Enterprise Information Management*. Emerald.

[17] Gaziel-Yablowitz, M., & Schwartz, D. G. (2018). A review and assessment framework for mobile-based emergency intervention apps. *ACM Computing Surveys (CSUR)*. ACM.

[18] Al-Khafajiy, M., Kolivand, H., Baker, T., Tully, D., & others. (2019). Smart hospital emergency system: Via mobile-based requesting services. *Multimedia Tools and Applications*. Springer.

[19] Tian, Y., Zhou, T. S., Yao, Q., Zhang, M., & Li, J. S. (2014). Use of an agent-based simulation model to evaluate a mobile-based system for supporting emergency evacuation decision-making. *Journal of Medical Systems*. Springer.

[20] Huang, J. T., Li, J., & Gong, Y. (2015). An analysis of convolutional neural networks for speech recognition. *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. Retrieved from ieeexplore.ieee.org

[21] Anusuya, M. A., & Katti, S. K. (2011). Front end analysis of speech recognition: a review. *International Journal of Speech Technology*. Springer.

[22] Hansen, J. H. L. (1996). Analysis and compensation of speech under stress and noise for environmental robustness in speech recognition. *Speech Communication*. Elsevier.

[23] Gaikwad, S. K., Gawali, B. W., & Yannawar, P. (2010). A review on speech recognition technique. *International Journal of Engineering and Technology*. Academia.edu.

Author Profile

Tejas S, Student, Department of Computer Science & Engineering. SDM institute of Technology, Ujire

Shilpa R, Assistant Professor, Department of Computer Science & Engineering, SDM Institute of Technology, Ujire