Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 2, No. 5, February : 2024

ANDROID BASED DISASTER MANAGEMENT APPLICATION USING MACHINE LEARNING

Dr. Naresh Thoutam, **Prof. Pramod Patil,** Associate Professor, Dept. of Computer Engineering, Sandip Institute of Technology and Research Centre, Nashik, Savitribai Phule Pune University, Maharashtra, naresh1060@gmail.com
Rupesh Gujar, Kishor Patil, Prajwal Sananse, Aayush Shah, Student of, Department Computer Engineering, Sandip Institute of Technology and Research Center Nashik, Savitribai Phule Pune University, Maharashtra, rupeshgujar98@gmail.com

Abstract
A good saying" Precaution is better than cure" for emergency situations. Natural disasters in India, many of them related to the climate of India, cause massive losses of Indian life and property. Droughts, flash floods, cyclones, avalanches, landslides brought on by torrential rains and snowstorms pose the greatest threats. Landslides are common in the Lower Himalayas. Parts of the Western Ghats also suffer from low-intensity landslides. Floods are the most common natural disaster in India. Therefore developing a system which can monitor as well as providing instructions on basis of current disaster using Supervised learning, reducing time for quick actions, Google map API for locating nearby locations, Disaster prediction using data analytics. The mobile application will have four tabs: Alert buzzer, weather forecast, the menu tab and the communication forum. Furthermore, the system notifies the users through SMS and even through the forum whenever there are the alert notification posted on the application Using system can help a great deal in the planning and implementation of disaster mitigation schemes.
Keywords: ML-Machine Learning, SVM-Support Vector Machine, NDRF-National Disaster Response Force, SDRF-State Disaster Response Force.

I. Introduction
Disaster management is the act through which communities are connected and encouraged to overcome adversity towards hazards and to contend with disasters. Disaster management is not about addressing or directing threats; instead, it accentuates creating plans to lessen the outcomes of natural or man-made disasters. Disasters have two types, natural and man-made disasters. The disasters took away the lives of many people. And Government of India is constantly trying to help out with disaster situations to decrease the unfavorable effect of disasters
Machine learning is a sub-domain of computer science which evolved from the study of pattern recognition in data, and also from the computational learning theory in artificial intelligence. It is the first-
class ticket to most interesting careers in data analytics today. As data sources proliferate along with the computing power to process them, going straight to the data is one of the most straightforward ways to quickly gain insights and make predictions
As the output of our model is in the format of positive and negative, binary classification is more suitable. There are three types of binary classifiers which we can apply into our model, they are namely
- Support Vector Machine (SVM).
- K Nearest Neighbour (KNN) Classifier.
- Naive Bayes Classifier.

Key points
Creating an Android disaster management application using machine learning involves several key components and considerations. Here are some important keypoints to keep in mind:
1. **Data Collection**:
Gather relevant disaster-related data, such as weather patterns, seismic activity, geographical information, and historical disaster records.

Consider using APIs, sensors, or crowdsourced data to update and enrich your dataset.

2. **Machine Learning Models**:

Implement machine learning models to predict and detect disasters. Common algorithms include:
Natural Language Processing (NLP) for text analysis in social media for early warning signs.
Computer Vision for analyzing satellite images or photos to detect visual cues of disasters.
Time series forecasting for predicting weather patterns or seismic activity.
Ensure that your models are trained on a diverse and representative dataset.

3. User Interface (UI):

Design a user-friendly interface for your Android app.
Provide a map view to display real-time disaster information and user locations.
Include features for reporting disasters or emergencies.

4. Real-time Data Integration:

Integrate real-time data feeds from relevant sources, such as weather APIs, seismic sensors, and government agencies.
Implement push notifications to keep users updated about potential disasters.

5. Geographical Information System (GIS):

Utilize GIS to map disaster-prone areas, evacuation routes, and safe zones.
Integrate GIS data into your app for location-based services.

6. Early Warning System:

Implement an early warning system that can notify users when a potential disaster is imminent in their area.
Use machine learning models to analyze historical data and predict future events.

II. Literature

IoT has emerged as a game-changer in disaster management, revolutionizing how we prepare for and respond to natural and man-made disasters. Numerous studies have explored its multifaceted applications across various disaster phases. In "Internet of Things for Disaster Management: State-of-the-Art and Prospects," Alsaqer et al. (2017) highlight how IoT enables real-time monitoring and data collection, enhancing disaster response by providing critical information to authorities. This real-time data empowers decision-makers with insights into evolving situations, such as flood levels, air quality, or seismic activity, enabling more effective resource allocation and evacuation planning. Moreover, IoT extends its impact beyond response to the entire disaster management lifecycle.

The paper "A Survey on Internet of Things for Disaster Management" by Khan et al. (2017) discusses how IoT contributes to preparedness and mitigation by offering predictive capabilities through sensor networks. These networks continuously gather environmental data, facilitating early warning systems for tsunamis, hurricanes, or wildfires. Additionally, IoT's role in recovery cannot be overlooked, as it supports damage assessment and infrastructure restoration efforts. Overall, the literature underscores IoT's potential to reshape disaster management strategies, making them more proactive, data-driven, and responsive to the needs of affected communities.

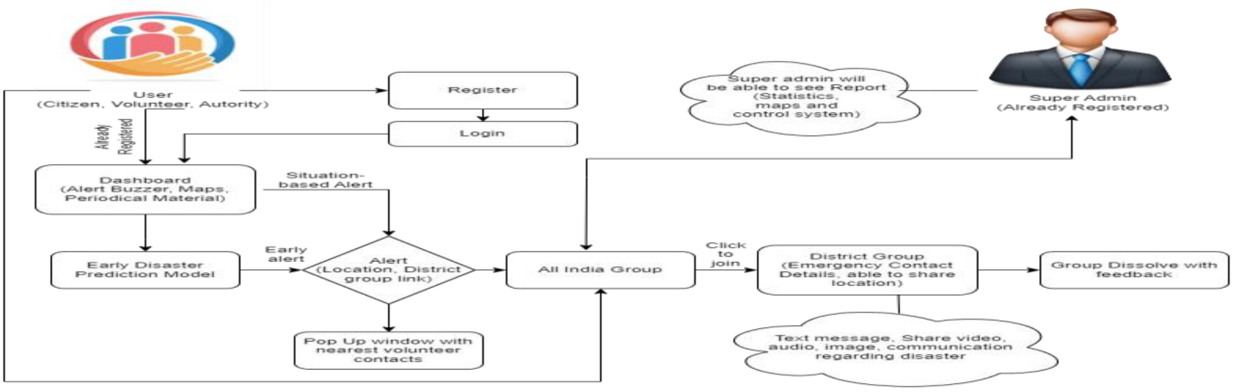
Furthermore, the incorporation of cooperative D2D communication in disaster-recovery scenarios has gained significant attention. D2D communication allows devices to communicate directly with each other without relying solely on infrastructure, which can be critical in disaster-stricken areas with damaged or congested networks. Research in this area focuses on how cooperative D2D communication can enhance network resilience and improve information dissemination during emergencies.

By combining Stackelberg game theory with cooperative D2D communication, researchers aim to develop novel strategies that address the unique challenges of disaster recovery, such as limited resources, dynamic network conditions, and the need for rapid response and adaptive communication protocols. Overall, the literature highlights the potential of this approach to enhance disaster-recovery communications and contribute to more efficient and reliable post-disaster communication networks.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 2, No. 5, February : 2024

Disaster communication systems play a pivotal role in ensuring connectivity during emergencies. In recent years, Device-to-Device (D2D) multi-hop relaying services have emerged as a promising solution to enhance the reliability and reach of such systems. A literature survey reveals that researchers have been actively exploring D2D multi-hop relaying in disaster scenarios. They have investigated various aspects, including routing protocols, energy efficiency, and network resilience. Some studies have proposed dynamic routing algorithms that adapt to changing disaster conditions, while others have focused on optimizing the use of energy-constrained D2D devices. Additionally, research has delved into ensuring network robustness by considering factors such as node mobility and interference mitigation. These efforts collectively contribute to the development of more effective and resilient disaster communication systems.

System Design



SYSTEM ARCHITECTURE

The system architecture is elaborated into two modules:
i. Situation based alert. ii. Prediction based

- Situation based Alert:

In this module the alert will be pressed by the volunteer/citizen whoever is present at the disaster location [6]. The alert will be notified to the nearest volunteers or citizens, consecutively the district wise disaster group will be created where the further discussion of the disaster prevention strategies will be discussed among the disaster prevention authorities and the common citizen, volunteers, etc.

- Prediction based Alert:

In this module we are giving an early alert which will help the disaster authorities to connect with the nearby people more easily and take necessary actions accordingly.

H/W and S/W Requirement

A. H/W Requirement
1.CPU Speed-2.5GHz- Provides the instructions and processing power the computer needs for android development.
2.RAM-4GB- The faster the RAM, the faster the processing speed
B. S/W Requirement
Platform:
1.Operating System: Windows 10 or later (64-bit), x86-64 based.
2.IDE: Microsoft Visual Studio 2019 or Android Studio 10.
3.Programming Language: DART (Flutter development) Python (API development).

Proposed Work

The development of an Android-based machine learning disaster management application can greatly improve emergency response activities' efficacy and efficiency. An example overview for such an application is provided below:

1. Design of the User Interface (UI): User Registration/Login: Permit users to safely log in or establish accounts. Dashboard: Show catastrophe notifications in real time. Give emergency contacts' details. Display a map showing the locations of disasters.
2. Disaster Alert System: Using External APIs Integration: To receive updates in real time, integrate with weather APIs. Make use of flood or seismic monitoring systems. Push Notifications: Notify users instantly based on where they are. Permit consumers to alter their choices for notifications.
3. Mdels for Machine Learning:
Catastrophe forecast: Use achine learning models to forecast disasters by analyzing past data. Think about employing models for forecasting floods, earthquakes, etc.
4. Integration of Emergency Services:
Location-Based Services: Let consumers tell emergency services where they are in real time. Provide a map of the impacted locations to emergency services. Allocating Resources: During catastrophes, using machine learning to forecast the need for resources in particular areas.
5. User-Generated Reports Using Crowdsourced Data: Permit users to submit reports in real-time that include text, images, and videos. Use sentiment analysis to rank the most important reports. Social Media Integration: Keep an eye out for posts about disasters on social media and add pertinent information to the system.
6. Communication Channels:
Two-Way Communication: Create a channel for users and emergency services to communicate back and forth.
7. Offline Features:
Data Caching: Provide consumers with offline access to vital information. Use local storage for important files and maps.
8. Community Building:
Community Forums: Establish forums where users may assist, advise, and exchange experiences. Promote participation in disaster preparedness within the community.
9. Instruction and Learning Resources: Manuals and Guides Provide lessons about being prepared for emergencies. Provide interactive instructions on what to do in case of particular disaster scenarios.
10. Accessibility:
Multilingual Support: Make sure that users who speak various languages may utilize the program. Offer voice-activated functionalities to those with impairments.
11. Feedback Mechanism for Continuous Improvement: Set up a feedback system where users may report problems or make suggestions for changes. Update the program often in response to user input and new technological developments.
12. Data transfer security: Use encryption to protect sensitive information. Ascertain that user data is safely maintained and complies with data protection laws.
13. Scalability: Cloud Integration: For scalability and dependability, leverage cloud services. Make sure the program can manage higher user loads during emergencies.

Other Specification

A. Advantages

1. EHRs make patient information readily accessible to authorized healthcare providers, allowing them to quickly access and share critical patient data such as medical history, test results, and medications.
2. EHRs support better decision-making by providing real-time access to patient data. This can lead to more accurate diagnoses and treatment plans, ultimately improving patient care.
3. Electronic records streamline administrative tasks, reducing paperwork, and making it easier to manage appointments, billing, and insurance claims. This can result in time and cost savings for healthcare facilities.

4. EHRs can be designed to communicate and share data with other healthcare systems, enhancing coordination of care across different providers and healthcare settings. This interoperability is vital for modern healthcare networks.
5. HER is very useful because of this we don't need to carry his hospital treatment history.

B. Disadvantages

1. implementing an EHR system can be expensive, involving costs related to software, hardware, training, and data migration. Smaller healthcare practices, in particular, may find the upfront expenses burdensome.
2. Transitioning from paper-based records to EHRs can be challenging for healthcare providers and staff. Learning to use the system effectively may take time and result in a temporary decrease in productivity.
3. Inputting patient data into EHRs can be timeconsuming, especially for healthcare providers who need to type or dictate information during patient visits. This can lead to longer appointment times and reduced face-to-face interaction with patients.
4. EHR implementation may disrupt existing workflows and require adjustments to accommodate the new system. This can lead to
temporary inefficiencies and staff resistance
5. Some healthcare providers and patients argue that EHRs can lead to a loss of the personal touch in healthcare interactions, as providers may spend more time interacting with the computer
than with patients

C. Applications

1. Clinical Decision Support: Provides alerts, reminders, and recommendations to aid in clinical decision-making.
2. Prescription Management: Facilitates electronic prescribing and checks for drug interactions.
3. Billing and Coding: Automates coding and billing processes for accurate reimbursement.
4. Appointment Scheduling: Helps patients book appointments and providers manage schedules.

III. Conclusion and Future Work

Conclusion

In conclusion, electronic health records (EHRs) have become integral to modern healthcare, offering numerous advantages while presenting certain challenges. They serve as a central hub for patient information, facilitating better patient care, clinical decision support, and efficient management of healthcare operations. EHRs enhance patient engagement, improve data security, and contribute to research and public health efforts. However, their implementation comes with an initial cost and a learning curve, and they can introduce new challenges related to data security, interoperability, and alert fatigue. Despite these challenges, EHRs have transformed healthcare by streamlining processes, improving information accessibility, and supporting quality care. To maximize the benefits of EHRs, healthcare organizations must carefully plan their implementation, provide adequate training, and address issues related to data security and privacy.

Future Work

To implement the proposed system on multiple peer to peer network, with computing which reduce the transactional data processing time

References

[1] Rifi, Nabil, Elie Rachkidi, Nazim Agoulmine, and Nada Chendeb Taher ."Towards using blockchain technology for eHealth data access management." In 2017 Fourth International Conference on Advances in Biomedical Engineering (ICABME), pp. 1-4. IEEE, 2017
[2] Pramod P, Pratyush Kumar Tripathy, Harshit Bajpai, Manjunath R Kounte," Role of Natural Language Processing and Deep Learning in Intelligent Machines", IEEE International Conference on Electrical, Communication,

Electronics, Instrumentation and Computing (ICECEIC), Kanchipuram, India, 30-31 Jan 2019
[3]Kotsiuba, Igor, Artem Velvkzhanin, Yury Yanovich, Iuna Skarga Bandurova, Yuriy Dyachenko, and Viacheslav Zhygulin.
"Decentralized e-Health Architecture for Boosting Healthcare Analytics." In 2018 Second World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4), pp. 113-118. IEEE, 2018.
[4]Mikula, Tomas, and Rune Hylsberg Jacobsen.
"Identity and Access Management with Blockchain in Electronic Healthcare Records." In 2018 21st Euromicro Conference on Digital System Design (DSD), pp. 699-. IEEE, 2018.
[5]Soumyalatha Naveen,
Manjunath R Kounte, Machine Learning-based Fog Computing as an Enabler of IoT, International Conference on New Trends in Engineering and Technology (ICNTET), Tiruvalur, Tamil Nadu, India, 7-8 Sep 2018
[6]Kamau, Gabriel, Caroline Boore, Elizaphan Maina, and Stephen Njenga." Blockchain Technology: Is this the Solution to EMR
Interoperability and Security Issues in Developing Countries?." In 2018 IST-Africa Week.
[7]Alhadhrami, Zainab, Salma Alghfeli, Mariam Alghfeli, Juhar Ahmed Abella, and Khaled Shuaib.
" Introducing blockchains for healthcare." In 2017 International Conference on Electrical and Computing Technologies and Applications (ICECTA), pp. 1-4. IEEE, 2017
[8]Chintarlapallireddy Yaswanth Simha, Harshini V M, L V S Raghuvamsi, Manjunath R Kounte, "Enabling Technologies for Internet of Things Its Security issues", Second International Conference on Intelligent Computting and Control Systems (ICICCS 2018), Madurai, India, 14-15 June 2018, pp 1849-1852
[9] Wehbe, Youssef, Mohamed Al Zaabi, and
Davor Svetinovic."Blockchain AI Framework for Healthcare Records Management: Constrained Goal Model." In 2018 26th Telecommunications Forum (TELFORD), pp. 420-425. IEEE, 2018.
[10] Novikov, Sergey P., Oleg D. Kazakov, Natalya A. Kulagina, and Natalya Yu Azarenko.
"Blockchain and Smart Contracts in a Decentralized Health Infrastructure." In the 2018 IEEE International Conference" Quality Management, Transport, and Information Security, Information Technologygies" (ITQMIS), pp. 697-703. IEEE, 2018.
[11] Jiang, Shan, Jiannong Cao, Hanqing Wu, Yanni Yang, Mingyu Ma, and Jianfei He. "Blochie: a blockchain-based platform for healthcare Electronics, Information Communication Technology (RTEICT), pp. 2145-2148. IEEE
[12] Rahman, Md Abdur, et al. "A Natural User Interface and Blockchain-based in-home Smart Health Monitoring System.", 2020, IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT). IEEE, 2020.
[13] Choi, Jungyong, et al. &Random Seed Generation For IoT Key Generation and Key Management System Using Blockchain.& 2020 International Conference on Information Networking (ICOIN). IEEE, 2020.
[14] Mohanta, Bhabendu Kumar, et al. &Addressing security and privacy issues of IoT using blockchain technology IEEE Internet of Things Journal (2020).
[15] Abraham, Misha, Himajit Aithal, and Krishnan Mohan. &Blockchain and Collaborative Intelligence based next generation Smart Toll Application. & 2020 2nd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS). IEEE, 2020.
[16] Azaria, Asaph, Ariel Ekblaw, Thiago Vieira,
and Andrew Lippman." Medrec: Using blockchain for medical data access and permission management." In 2016 2nd International Conference on Open and Big Data (OBD), pp. 25-30. IEEE, 2016.