Robust Deep Learning Model for Accurate Fall Detection using Smartphone Sensor Data

MCS

Author

SAMINA ARSHED

Registration Number
00000320922

Supervisor
DR. AYESHA MAQBOOL

A thesis submitted to the faculty of Department of Computer Software Engineering, Military College of Signals (MCS), National University of Sciences and Technology (NUST) Islamabad, in partial fulfilment of the requirement for the degree of MS in Computer Software Engineering

August 2023

THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS Thesis written by **NS Samina Arshed**. Registration No. **00000320922** of **Military College of Signals** has been vetted by undersigned, found complete in all respects as per NUST Statutes/Regulations/MS Policy, is free of plagiarism, errors, and mistakes and is accepted as partial fulfillment for award of MS degree. It is further certified that necessary amendments as pointed out by GEC members and local evaluators of the scholar have also been incorporated in the said thesis.

Signature: __________

Name of Supervisor **Dr. Ayesha Maqbool**

Date: __________

Brig
Head of Dept of CSE
Mil College of Sigs (NUST)

Signature (HOD): __________

Date: 5/9/23

Brig
Dean, MCS (NUST)
(Asif Masood, Phd)

Signature (Dean/Principal) __________

Date: 11/9/23

Declaration

I certify that this research work titled "*Robust Deep Learning Model for Accurate Fall Detection using Smartphone Sensor Data*" is my own work. The work has not been submitted to anybody else for review. The usage of content from other sources has been properly acknowledged and referred to.

Samina

Signature of Student

Samina Arshed

NUST MCS 00000320922

DEDICATION

This thesis is dedicated to *my late father Syed Arshad Saeed and my late husband Muhammad Anees*

And to my mother, daughter, and son whose constant support and exceptional cooperation made it possible and led me to this accomplishment.

ABSTRACT

Falls and associated medical conditions are a serious concern in healthcare, especially among elderly people. The ratio of older people is growing every year, making it even more crucial to have effective fall prevention and detection systems in place. The main objective of this research is to provide a comprehensive solution from dataset creation to application of effective modeling techniques in the context of Fall Detection systems (FDS). A new dataset is presented based on scripted Activities of Daily Life (ADL) performed by elderly people. This representative data combined with young volunteer's simulated falls data is the key differentiation to offer a relevant basis for prediction. The proposed work intends to develop a binary classification framework that can analyze the data and correctly categorize falls and no falls by distinguishing falls from complex fall like activities of daily life (ADL). Recurrent Neural Networks (RNNs) having the ability to handle sequential data and capture temporal dependencies are used in this research for robust fall detection based on smartphone accelerometer data. The accuracy of FDS is most of the time assessed by the performance metrics only, without performing organized testing on varied set of data. Adequate strategies along with thorough testing across several datasets is imperative for a reliable FDS. To add to the sophistication of this research the employed fall detection techniques are assessed using a variety of public datasets. Proposed models are evaluated for the reference new dataset and two other publicly available datasets. With all these considerations the proposed approach and model has shown better performance as compared to previously adopted models.

Keywords: Accelerometer, fall detection, Deep Learning, smart phone data, Random Forest, BiLSTM, feature selection, Machine Learning

Acknowledgements

I am thankful to Allah S.W.T to have instilled the strength and courage in me to work. For He is the one who has guided me throughout this work at every step and for every new thought which came to me. Indeed, I could have done nothing without the will and help of Almighty Allah.

I would like to express special thanks to my supervisor Dr. Ayesha Maqbool and co-supervisor Dr. Ahsan Shahzad for their help and support throughout my thesis. The ‘can do’ attitude and motivation by Dr. Ayesha was invaluable. I would also like to pay special thanks to Dr. Ahsan for tremendous support and guidance, each time I got stuck in something, he came up with the solution. Without their help I wouldn’t have been able to complete my thesis.

I am grateful to GEC members for this thesis, Lt. Col. Khawir Mehmood and Assistant Professor Dr. Tauseef Rana for their meticulous review and feedback that has helped to improve my work and contents of this thesis.

My acknowledgment would be incomplete without thanking the biggest source of my strength, my family. Finally, I would like to express my gratitude to all my friends and the individuals who have encouraged and supported me through this entire period.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

1. INTRODUCTION

In this chapter the introduction to fall detection, techniques and categories of fall detection systems are discussed. Moreover, the significance of fall detection has been highlighted. The research problem, research questions and methodology of how research was conducted are also mentioned in this chapter.

1.1 Falls

Falls are commonly defined as "inadvertently coming to rest on the ground, floor or other lower level, excluding intentional change in position to rest in furniture, wall or other objects" [1]. It is important to have a universally agreed definition of Fall to avoid misinterpretations. To understand the reason and basis, a fall from the perspective of elderly is a sudden jerk or state when they lose balance. However, in medical terms it is a serious incident that can lead to injuries, hospitalization and even to death.[2].

How often a person falls is directly related to age factor and aged people are more susceptible to falling. Approximately 28-35% of people aged 65 and over fall each year and increasing to 32-42% for those over 70 years of age.

Different risk factors also contribute to vulnerability and can be categorized as Behavioral factors that relate to use of multiple medications, excess alcohol intake, lack of exercise and inappropriate footwear. Falls can result from a complex interplay of various biological, physiological, and environmental factors. Some of the biological factors that can contribute to an increased risk of falls include muscle weakness, impaired balance and Gait, dizziness and vertigo, neurological conditions, cardiovascular issues, chronic conditions like diabetes and conditions like dementia that can lead to confusion and poor decision-making, increasing the likelihood of accidents. Environmental factors also play a significant role in contributing to falls, especially in homes and public spaces. These factors can create hazards that increase the risk of accidents. Irrespective of the cause for a fall, it can be critical if the injured person does not get quick assistance.

Falls can also have economic impacts on both individuals and society. These impacts can be seen in various sectors, including healthcare, workplace productivity, and insurance costs. Falls can cause physical injury and mental trauma which can even lead to anxiety and depression in elderly people. According to a factsheet shared by WHO in April 2021, falls are a major public health problem globally. An estimated 684,000 fatal falls occur each year, and over 80% of fall-related fatalities occur in low- and middle-income countries, with regions of the Western Pacific and Southeast Asia accounting for 60% of these deaths [3].

The control mechanism and strategies for improving the overall environment that can help to reduce the elderly fall occurrences and provide emergency relief in case of an event are broadly termed as fall prevention or fall detection [4]. While this is active research domain for past two decades, the focus of this study is on improving the accuracy fall detection.

1.2 Fall Detection Techniques

Fall detection techniques aim to identify when an individual experiences a sudden drop to the ground or a loss of balance. These techniques can be broadly categorized into several types based on the technology and sensors used. Here are some of the common fall detection techniques:

Inertial Measurement Units (IMUs): IMUs consist of accelerometers and gyroscopes and are often integrated into wearable devices like smartwatches or fitness trackers. They can measure changes in acceleration and orientation, allowing them to detect sudden movements associated with falls.

Pressure Sensors: These sensors are often embedded in floors, carpets, or mattresses and can detect changes in pressure caused by a person falling or lying on them.

Computer Vision: Cameras or depth-sensing devices can be used to monitor an individual's movements. By analyzing the video feed or the depth data, fall-like motions can be detected.

Audio Analysis: Microphones can be used to analyze the sounds associated with a fall, such as impact or vocal distress, to trigger an alert.

Machine Learning Algorithms: Various machine learning techniques can be applied to data collected from different sensors to detect fall patterns and distinguish them from normal activities.

Radio Frequency (RF) Signal Analysis: By analyzing radio frequency signals (like Wi-Fi or RFID), changes in the signal pattern caused by a fall can be detected.

Vital Sign Monitoring: In some cases, fall detection systems may combine information from heart rate monitors or other medical devices to detect falls and related health events.

Smart Home Devices Integration: Fall detection can be achieved by integrating data from various smart home devices, such as motion sensors, door/window sensors, and smart cameras.

Location Tracking: GPS or other localization technologies can be used to monitor a person's movement and detect sudden changes or deviations from the expected pattern, which might indicate a fall.

Fall Detection Apps: Some mobile applications utilize smartphone sensors (accelerometers, gyroscopes) to monitor the user's movements and detect falls.

It's important to note that no single technique is foolproof, and a combination of different methods or sensor data fusion may be employed to improve accuracy and reduce false alarms. Additionally, user privacy and data security should be taken into consideration when implementing fall detection systems, especially in sensitive environments like healthcare facilities or private homes.

1.3 Categorization of Fall Detection Techniques

Different strategies for fall detection as briefly discussed above can be broadly categorized into context aware systems or wearable based systems.

Context-aware systems for fall detection take advantage of various contextual information in addition to the standard accelerometer and gyroscope data typically used in fall detection. These additional context-aware features help enhance the accuracy of fall detection and reduce false positives or negatives. By incorporating context-aware elements, fall detection systems can better understand the user's situation, reduce false alarms, and provide more personalized and timely assistance in the event of a fall. As technology continues to advance, we can expect even more sophisticated context-aware systems that further improve fall detection capabilities and enhance overall user safety and well-being.

Wearable-based fall detection systems utilize sensors and wearable devices to monitor an individual's movements and detect potential falls. These systems are designed to be worn on the body, such as on the wrist, waist, chest, or as a pendant. They have many advantages like they provide immediate assistance, continuous monitoring, mobility and independence, user friendly design and long battery life. Despite these advantages, wearable-based fall detection systems may also have some limitations. They can generate false positives or negatives, and the accuracy of fall detection can vary based on device placement and user behaviour. Additionally, wearable devices might not be suitable for individuals who have difficulty wearing or using them consistently.

To overcome the limitations posed by most of the wearable devices another sensor-based detection mechanism is using smartphones sensors. Smartphone-based fall detection systems have gained significant importance due to their potential to improve the safety and well-being of individuals, especially the elderly and those at risk of falling. Sensors are embedded in the phone and with widespread adoption of smartphones it is easy to carry for the elderly. This research is also based on smartphone-based fall detection.

1.4 Motivation

Elderly Falls is posing dramatic challenges in the public health domain and mostly Fall Detection systems are either context aware or wearable based. In the context aware systems sensors are either placed in the environment or rely on vision-based techniques that further raise privacy concerns. For the wearable systems they carry additional cost and elderly may forget to carry them. A very few solutions are based on smartphones, and they also are not optimal enough.

By incorporating the proposed Deep Learning techniques, robust and optimal solution for Fall detection can be devised. To make the systems more effective representative datasets that take elderly ADLs and young adult’s simulated fall data can be taken as input and comprehensive testing across several datasets can be performed.

In the above-mentioned perspective, multidimensional analysis of FDS can be performed by using efficient DL techniques to support the healthcare professionals in elderly care. Therefore, a Deep learning model is chosen to improve FDS using smartphone sensor data.

1.5 Problem Statement

Recent studies narrated that old people are reluctant to perform any physical, intellectual, and tangible functionalities [5]. Accordingly, older people face trouble performing routine work like strolling, running, eating, and getting ready [6]-[8]. The consequences of falls can contribute to health decline and overall mortality in older adults over time. It is known through studies that approx. 35% of aged people experience a fall each year [9]. These facts and figures and the continuous rise in elderly population signify how inevitable it is to have means and measures that can reduce the risks of falls and timely detect their occurrence to generate alarms for immediate assistance. To support the efforts and create a conducive environment, WHO has devised an action plan with defined goals as part of United Nations (UN) program termed as Decade of Healthy Ageing 2021‑2030. This constitutes 10 years of concerted, catalytic, sustained collaboration. Older people themselves will be at the center of this plan, which will bring together governments, civil society, international agencies, professionals, academia, the media, and the private sector to improve the lives of older people, their families, and their communities [10]. To ensure the success of this program and its objectives, it is imperative for the countries worldwide to keep innovating and developing enhanced systems that can fulfill the health care needs of elderly people.

1.6 Aims and Objectives

Following are the objectives of this research.

1. To generate a representative dataset that gives basis for the accurate fall detection.
2. To propose a solution based on accelerometer data collected from smartphone.
3. To design and develop a robust Fall Detection model based on Recurrent Neural Network design.
4. To do comparative analysis of ML and DL techniques in the context of Fall Detection.
5. To analyze and evaluate the performance of proposed model by using different proprietary and public datasets.

Following Fig 1.1 gives a high-level view of this research for detecting falls.

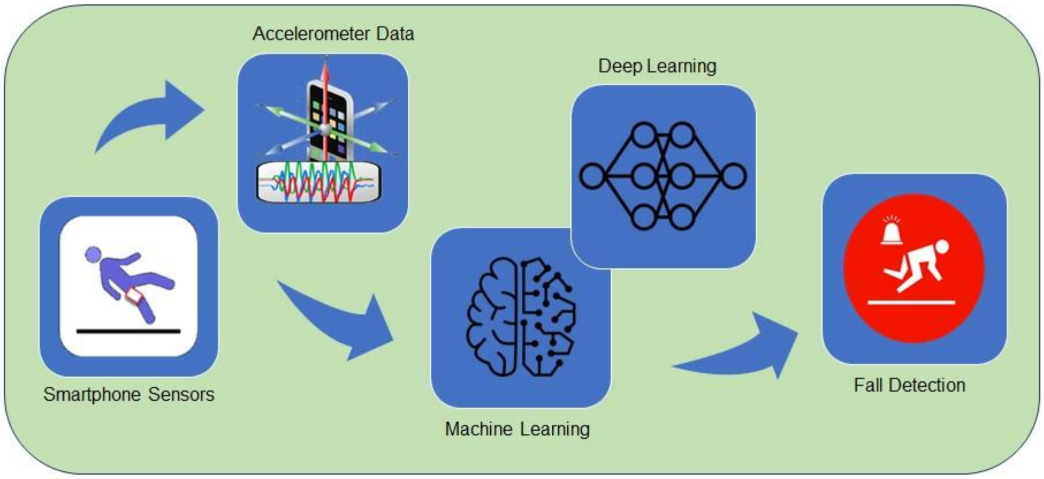

Figure 1-1 High-level view

1.7 Areas of Application

A smartphone-based fall detection system has various areas of application, primarily centred around improving safety and well-being for individuals, especially the elderly and those with certain medical conditions. Some of the key areas of application include:

1. **Elderly Care**: Fall detection systems can be particularly useful for the elderly population, who are more susceptible to falls and related injuries. The system can automatically detect falls and alert caregivers, family members, or medical personnel, ensuring timely assistance.
2. **Medical Monitoring**: Individuals with certain medical conditions or mobility issues can benefit from fall detection systems. These conditions may include epilepsy, Parkinson's disease, multiple sclerosis, etc. The system can monitor for potential falls and enable caregivers or medical professionals to intervene if needed.
3. **Emergency Response**: A fall detection system can be integrated with emergency response services. If a fall is detected, the system can automatically notify emergency services, providing the user's location and other relevant information to ensure prompt assistance.
4. **Workplace Safety**: In industrial or construction settings, where workers might be exposed to hazardous conditions, a fall detection system can be employed to enhance safety. If a

worker falls and is unable to signal for help, the system can alert supervisors or on-site safety personnel.

5. **Home Automation**: Fall detection can be integrated into smart home systems. If a fall is detected, the system can automatically turn on lights, send notifications to family members or caregivers, and even call for emergency help.
6. **Health and Wellness Tracking**: For fitness enthusiasts or individuals keen on monitoring their overall health, a fall detection system can be an added feature in health and wellness apps. It can help track falls, analyze patterns, and provide insights into potential risk factors.
7. **Assisted Living Facilities**: Fall detection systems can be deployed in assisted living facilities or nursing homes to improve resident safety. Caregivers can be alerted immediately in the event of a fall, ensuring rapid response, and reducing the risk of complications.
8. **Public Health Programs**: Governments or public health organizations can utilize fall detection systems as part of initiatives to monitor and support the elderly population or individuals with specific medical conditions.

It's important to note that while smartphone-based fall detection systems have shown promise in these areas, they may not be 100% accurate, and false alarms can occur. These systems should be viewed as complementary tools to enhance safety and care, rather than as a replacement for medical attention or human monitoring. Additionally, privacy considerations and user consent should be considered when implementing such systems.

1.8 Thesis Outline

This thesis is structured into following chapters:

- **Chapter 2:** Literature review and detailed study on fall detection systems and past research done in this regard are provided in this chapter.
- **Chapter 3:** Proposed Framework, Methodology with different stages and their significance, newly proposed dataset and reference public datasets are covered in this chapter.
- **Chapter 4:** This chapter provides a detailed account of experimental setup and implementation details for model training and validation. Results of training and evaluation are also discussed in detail in this chapter.
- **Chapter 5:** Conclusion and Future work is presented in this chapter.

Chapter 2

2. LITERATURE REVIEW

Artificial Intelligence (AI) based solutions are proven to be a great enabler in this regard. May it be through generative AI, robots or machine learning and deep learning techniques; AI is entirely reshaping the healthcare services landscape [11]. In recent years, many studies are conducted on detecting and avoiding falls. Researchers have mainly explored three types of methods i.e. context aware systems, wearable systems and ambient/fusion systems.

Solutions built on artificial intelligence (AI) have shown to be excellent enablers in this area. Artificial intelligence (AI) is fundamentally changing the landscape of healthcare services, whether through generative AI, robotics, machine learning, or deep learning approaches [12]. Several research has been done in recent years on detecting and preventing falls. From these categories, four approaches namely context aware systems, wearable systems, ambient/fusion systems, and smartphone-based systems have been the major focus of research.

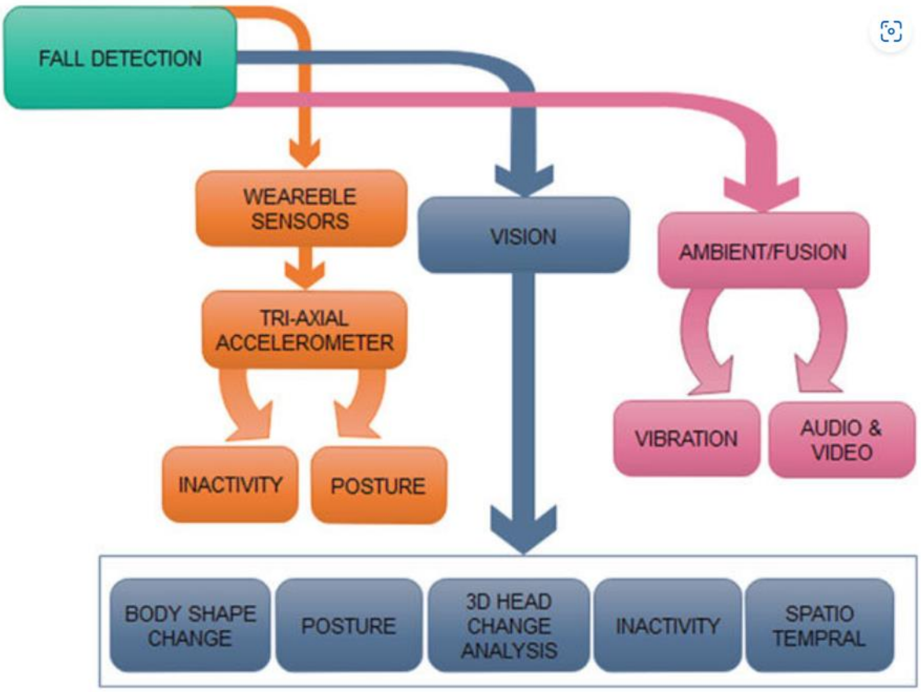

Figure 2-1. Categories of Fall Detection

2.1 Context Sensitive fall detection

Context-sensitive fall detection refers to technology and research that aim to improve the accuracy and reliability of fall detection by considering various contextual factors surrounding the fall event. These factors can include the user's environment, activity, physiological data, and more. The goal is to develop systems that can differentiate between actual falls and benign activities that might mimic a fall-like motion, thus reducing false alarms and improving overall system performance. These solutions can be powered by computer vision, use camera, radars or sensors attached to devices. As they often collect and process sensitive user information, research is conducted to ensure the privacy and security of user data. Techniques for data anonymization, secure context sharing, and user consent mechanisms are also explored.

As represented in Fig 2.2 below the context aware systems have many sub-branches.

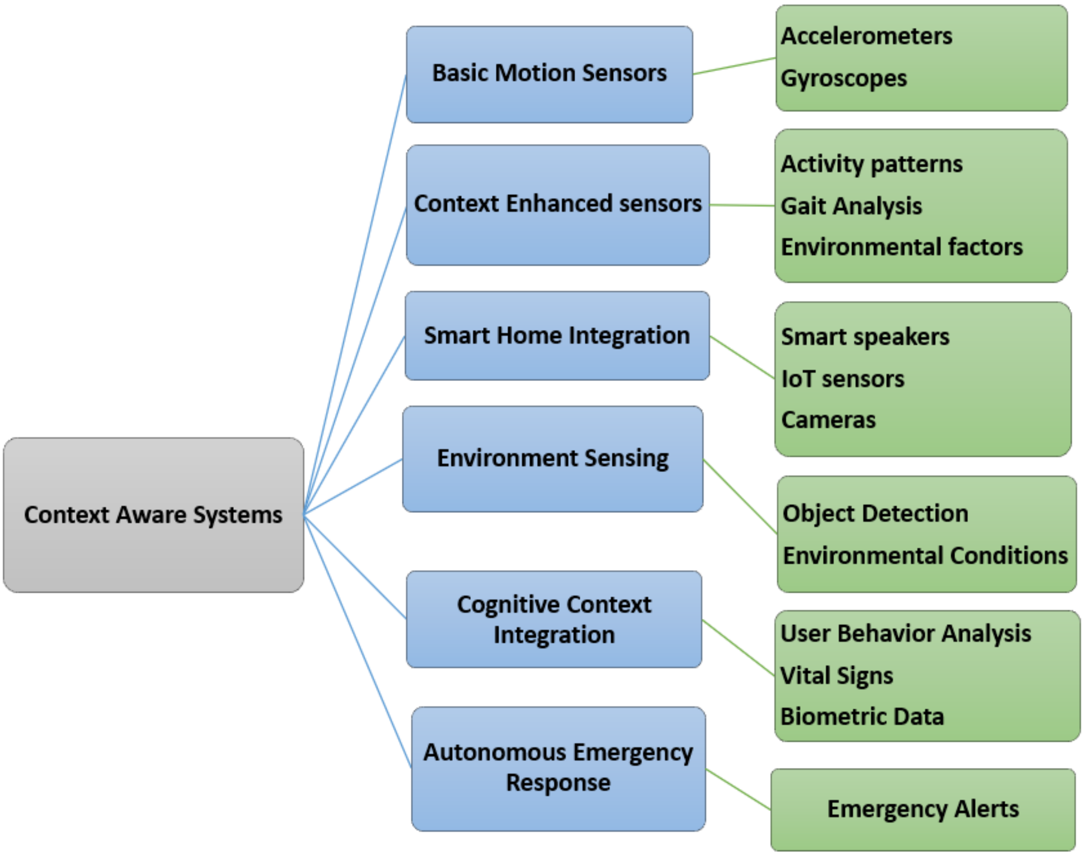

Figure 2-2 Context Aware Systems for Fall Detection

A recent study and detailed review of the fall detection methods is done by Newaz et. al. [13] and a list of strategies based on Information of Things (IoT), biomedical assistive technology and wearables along with the gaps in each research is provided. It also reviews the ethical considerations in each research along with shortcomings. Xiong [14] has used a skeleton-based 3D approach that is trained using neural network (S3D-CNN) and classifies the public and proprietary datasets for fall and no fall. This research claims best accuracy of 99% as compared to existing studies. Debard et al. [15] have used surveillance camera images from simulated environment for elderly fall detection. They used the aspect ratio, fall angle, head state and center speed features to extract fall specific data. They have then applied motion detection techniques along with classification algorithm to specify a fall event. However, it restricts the elderly to indoor environment and thus not feasible enough. Moreover, such systems are prone to difficulties such as limited coverage, expensive installation costs, false alarms, and privacy concerns (particularly in video-based systems).

C. Mosquera-Lopez et. al. [16] has presented research on fall detection system for the patients of multiple sclerosis (MS) disease who are prone to frequent falls due to vulnerability associated with disease and that increases with time. A context-aware system, using inertial and time of flight sensors is developed that used acceleration and movement features. This system has shown a sensitivity of 92.14%, and significantly reduced false alarms.

The above list provides a snapshot of the diverse research in the field of context-sensitive systems. The ongoing development of artificial intelligence, machine learning, and sensor technologies continues to fuel advancements in this field.

2.2 IoT enabled fall detection

IoT (Internet of Things) based fall detection research focuses on utilizing interconnected devices and technologies to detect and respond to falls, particularly in elderly individuals. Wearable devices play a crucial role in this area by providing continuous monitoring and real-time data collection. These systems have embedded smart electronic gadgets within the micro controllers that can be tied to the clothing or worn on the body as inserts or frill [17]. These devices continuously monitor the wearer's movements and provide data that can be analyzed to determine if a fall has occurred. Wearable gadgets for fall detection are usually coupled with the inertial sensors; for example, accelerometer, gyroscope, inertial measurement units (IMU), pressure sensors and barometric altimeter. The main advantage of such systems is that they are cost-effective. IoT technology has become more affordable over time, making these fall detection systems relatively less costly as compared to traditional monitoring solutions or hiring round-the-clock caregivers. Devices can be attached to multiple body locations and offer flexible portability rendering the devices relatively easy to operate. Despite many advantages and penetration IoT fall detection systems may sometimes generate false alarms (false positives) or fail to detect actual falls (false negatives). Factors like sudden movements, incorrect sensor positioning, or rapid changes in posture could trigger false alarms. On the other hand, a fall might not always trigger the sensors, leading to missed alerts. Fig 2.3 shows the main components of an IoT enabled solution.

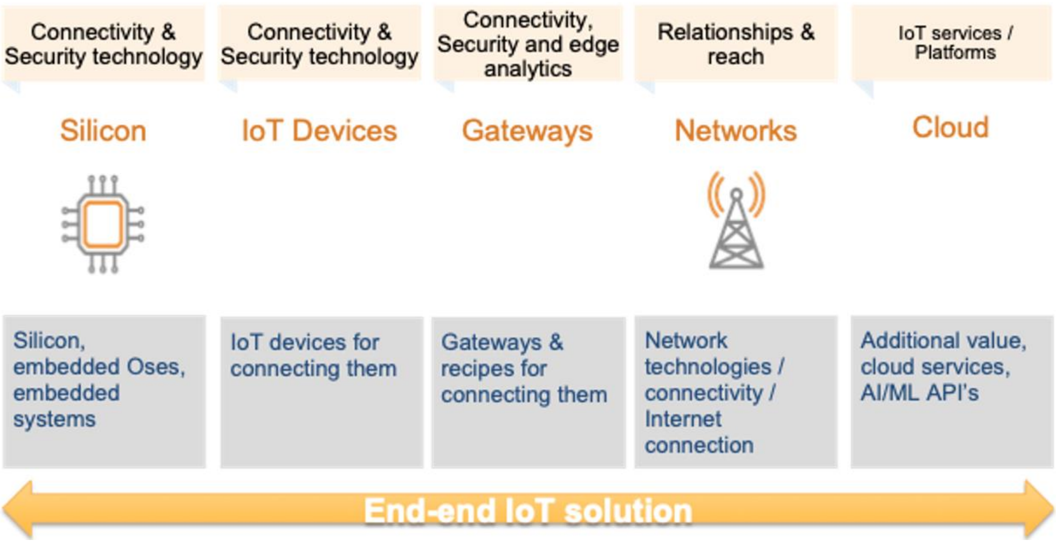

Figure 2-3 IoT-enabled System

Waheed M. et. al. [18] has used wearable sensors to detect elderly falls by employing strategies to reduce the noise in image data and provides solution for missing data values. The suggested noise-tolerant fall detection system focuses on Recurrent Neural Networks (RNNs) and BiLSTM memory stack. Two publicly available datasets SisFall and UP-Fall are used in this study and system shows improved performance in all aspects with accuracy of 97.21% and 97.41%, sensitivity of 96.97% and 99.77%, and specificity of 93.18% and 91.45%, on the input datasets respectively. Kim et. al. [19] in their paper has envisioned a complete healthcare solution based on Augmented Relaity (AR) and IoT-enabled software system for the caregivers. It includes monitoring and alerts service for the elderly and their attendants and families. It uses the techniques of object detection combined with information retrieval from devices placed in the vicinity of elderly on the pattern of a smart city solution. The evaluation shows great performance in all the relevant aspects of processing time, prediction, latency and connection for information and alert transmission with values as less as between 767ms and 1,283ms.

Bugarin et. al [20] has designed a vision enforced solution using camera. This research applied Random Forest machine learning algorithms along with deep learning techniques from MobileNetv2 CNN based MediaPipe Pose with improved results and a message with video sent to the listed contacts in the integrated alarm system. M. Amir et. al. [21] In this threshold-based detection method is proposed that has used ADXL335 accelerometer as an embedded sensor in the wearable device. The transmitter or FDS-Tx is worn and tied to the clothes to record data and is sent back to the cloud system through XBee module. IoT controller does the computation and sends the analysis to cloud connected server. It gave remarkable results with 97% sensitivity, 69% specificity and 83% accuracy.

Matos-Carvalho et. al [22] has tested a deep leaning model LSTM with different combination of layers for improving the accuracy and performance for classification of fall data gathered from wearable devices and embedded sensors. The study gives a basis and proves the hypothesis that how using multiple layers of LSTM impacts the system output and achieves a 99.13% of accuracy with 4.35% of loss values. It is expected that the results and experiments would help the researchers looking for an ideal deep leaning algorithm for prediction and classification of sequential data.

2.3 Smartphone based fall detection

In comparison to the above techniques, smartphones (SPs) equipped with dedicated detection technology is a far more viable option. Smartphones have become a necessity and almost every adult in most countries carries a mobile (smart) phone. Effective communication, user friendly interfaces and enhanced hardware are important factors that encourage the researchers to build smartphone-based solutions. Smartphones have all the built-in features that make them the most suitable choice for efficient healthcare monitoring including the availability of multiple sensors, connectivity, processing power and easy adoption especially for elderly people.

Fig 2.4 gives a pictorial representation of different layers of smartphone-based fall detection.

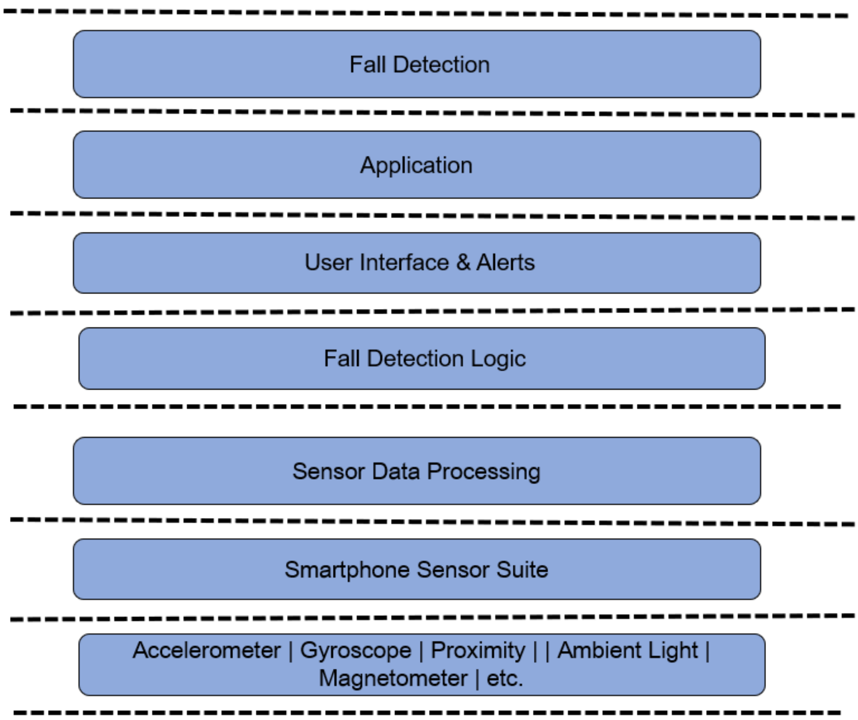

Figure 2-4 Smartphone based fall detection.

Researchers have carried out studies based on smartphones with a combination of approaches. L. Ren and Y. Peng [23] developed a global classification of current fall related research based on four aspects: fall detection and prevention systems based on different sensor devices and analysis algorithms, low-power technologies, and sensor placement in fall-related systems. Stampfler T. et al [24] in their research have screened 267 studies and provided an account of 15 studies that detailed pervasive fall detection and alerting applications based on smartphone accelerometers. S. Usmani et al [25] have visualized the application of Machine Learning techniques for smartphone and wearable device sensors. It gives a detailed account of recent trends and studies covering datasets, ML algorithms and performance metrics. Sensors or wearable based fall detection frameworks and Smartphone based fall detection systems have been widely reviewed by A. Singh et al [26]. Recently, Lee et al. [27] proposed enhanced threshold-based fall detection system in which one more threshold value was incorporated to classify fall events from the normal activities of daily living (ADLs). The system achieved an accuracy of 99.38% whereas 650 test activities including 11 distinct types of daily tasks were carried out.

A. Chelli et.al [28] have tested the performance of four machine learning algorithms i.e., artificial neural network (ANN), K -nearest neighbors (KNN), quadratic support vector machine (QSVM), and ensemble bagged tree (EBT) using smartphone' acceleration and angular velocity data. By applying specific feature extraction techniques, they achieved the best possible performance. Abdullah et. al. [29] used an app called Physics Toolbox Sensor Suite to gather smartphone accelerometer data and applied a neural network-based classification algorithm to distinguish falls from regular life activities.

2.4 Fall Detection Methods

Every technology needs some basic principles, rules, and regulations to work smoothly. Clarifying the different types of falls aids in methodological research. In addition, it contributes to and leads the development of new algorithms. Falls must be considered in all possible circumstances. There are some shared traits as well as crucial differences between these falls. There are three commonly applied methods for detecting falls: a straightforward analytical/threshold-based method, machine learning based methods and deep learning-based methods. A detailed account of research done using these methods specific to smartphone sensors is given below.

2.4.1 Threshold-based Methods

Analytical/threshold-based methods for fall detection are algorithms designed to identify falls based on specific criteria or thresholds. These methods often rely on data from sensors or wearable devices to monitor a person's movements and detect unusual patterns that may indicate a fall. Less computing power is required by these systems. They are also less complex than other modern algorithms. The accuracy of the framework often depends on predetermined limit values.These systems capture information on human movement using the various mobile phone sensors (such as the accelerometer, gyroscope, magnetometer, and so forth). The sensor data is then compared to predefined threshold levels. When a fall is detected, warning agencies are often used to alert the rescue agencies and medical aid authorities.

Wang et al. [30] proposed a study in which two novel inertial parameters i.e., acceleration cubic-product-root magnitude (ACM) and angular velocity cubic product- root magnitude (AVCM) were introduced to work on the selectivity of threshold-based fall recognition techniques. To improve accuracy, Tsinganos et al. [31] devised a threshold-based technique and used the K nearest neighbor (KNN). The system achieved accuracy equivalent to related studies with 97.53% sensitivity and 94.89% specificity. Pipanmaekaporn et al. [32] proposed a technique that focuses on reducing false positives by finding feature patterns in data from an integrated accelerometer sensor. The found patterns are employed as features in the construction of a robust and reliable classifier for fall detection using decision tree learning. The experimental findings using the Mobi Fall dataset, a benchmark for assessing fall detection systems in mobile phones showed that the proposed technique outperforms two accelerometer-based fall detection algorithms for smartphones in terms of performance and false positive rates.

Even though threshold-based algorithms use less processing, provide the highest accuracy, and are simple to implement; threshold-based systems often struggle to accurately distinguish between actual falls and normal movements or activities that might trigger the threshold. The challenge lies in finding the optimal threshold values that can correctly identify all classes.

2.4.2 Machine Learning-based Methods

Machine learning algorithms play a crucial role in fall detection by enabling automated and accurate identification of falls and non-fall activities based on sensor data. They can automatically analyze sensor data, such as accelerometer readings from wearable devices, without the need for manual intervention. This allows continuous monitoring of individuals and immediate detection of potential falls. They can adapt to variations in individual movements and fall patterns. They can learn from new data and continuously improve their performance, making them more reliable over time. Sensor data, especially from wearable devices, can be noisy and prone to artifacts. ML algorithms can be designed to handle noisy data and still make accurate predictions, thus increasing the robustness of the fall detection system.

The significance and application of machine learning algorithms is increasing as more and more research is taking place in this field. Already, plenty of research is done using support vector machines (SVMs), k-nearest neighbors (k-NN), decision trees (DT) and random forest (RF) to categorize falls from regular life activities.

Asier Brull Mesanza ei al. proposed SVM-based fall detection system which used data provided by a Sensor-enabled Tip which can be attached to different Assistive Devices for Walking (ADW) [33]. The model used two modules connected in series. The first one detects all falls, while the second differentiates between user and ADW falls. This latter module is designed to avoid false positives due to ADW accidental falls. Feature evaluation of training data set is implemented to detect the most relevant features to design each Machine Learning-based module. Once the training dataset is processed by the Random Forest (RF) Algorithm, a set of SVMs will be trained to implement algorithm of each module. The proposed approach provided high Fall Detection Ratios of over 90%. Gunale et.al.[34] in their research have used k-nearest neighbor classifier and given a novel method for identifying falls. They have used distinct features related to Orientation angle, ratio of fitted ellipse, Motion Coefficient, and Silhouette threshold. They achieved accuracy above 95% using video sequence data. Palmerini et. al. [35] have performed extensive research on acceleration signals recorded by an inertial sensor on the lower back and obtained data of 143 real-world falls from the FARSEEING repository. Using multiphase machine learning approaches attained sensitivity of 80%, and a very low false alarm rate of 0.56/hour.

Diana Yacchirema ei al. [36] proposed IOT and ensemble machine learning algorithm-based fall detection system for indoor environment. The wearable device is embedded with a 3D axis accelerometer for capturing the movements of elderly people in real time. The acceleration readings are then processed and analyzed using an ensemble random forest (RF) model. This model also alerts emergency services in case a fall is detected. The accuracy of this model was above 94%.

Shahzad et al. [37] have devised a mobile application that captures the falls and sends alerts. This is based on a proprietary dataset called FallDroid based on the simulated fall and ADL data of adults between the age of 28 to 60 years. Derived from FallDroid the android app is also termed FallDroid. In this research the accelerometer data obtained from smartphone sensors is used for training the classifier. Features are extracted using threshold-based mechanism and then machine learning based multiple kernel learning support vector machine (MKL-SVM) algorithm is applied. This study used smartphone data collected from thigh and waist locations. Remarkable performance improvements have been noted especially for waist location data providing accuracy of 97.8%, sensitivity 99.5%, and specificity 95.2% This study has used as a benchmark by many recent researchers for classifying the potential fall and trickly ADL data from real falls.

2.4.3 Deep Learning-based Methods

Deep learning (DL) algorithms use deep neural networks to automatically learn hierarchical representations from raw sensor data. Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) are commonly used for fall detection tasks. Instead of using traditional feature extraction techniques, pattern recognition tasks are carried out in deep learning-based frameworks. Deep learning models are made up of several layers. Each layer separates a chunk of the provided data or simulates certain changes in the data. The last layer consists of artificial neurons.

Deep learning models can automatically learn relevant features from raw data without requiring manual feature engineering. This is beneficial as it reduces the need for domain-specific knowledge and potentially uncovers patterns that might not be apparent through manual feature extraction. Xiaodan et al. [38] employed the DL techniques to devise a novel model using Gated Recurrent Units (GRU) architecture. The results are tested using two open datatsets for fall detection and validated against six ML classifiers and while it outperformed them an accuracy of 99.56% was achieved. Abdullah [39] proposed a solution in which the classification between fall and the non-fall event was done using data collected from a smart phone sensor (tri-axial accelerometer) and neural net pattern recognition app. They achieved a classification accuracy of 90.6%.

Escaño et. al. [40] endeavored to come up with a single solution that can be applicable for two types of classification problems i.e. fall detection and human identification. It is claimed that the single model need not be trained specially for any new subject rather the proposed model is equally good for new data and servs as a universal algorithm for both tasks which is a great achievement. Wang [41] analyze various lightweight and shallow neural networks that require lesser storage and computational resources. The research concludes with a lightweight supervised convolutional neural network achieving 99.9% detection accuracy for resource constraint wearable sensors.

2.5 Datasets used for Fall Detection

For the identification and study of falls, researchers have developed or used a variety of datasets. The most important are tFall, UMAFall, UPFall, MobiFall, and DLR datasets. A brief account of the research based on common datasets with prediction and outcomes is discussed here.

The UP-Fall dataset is a publicly available dataset based on different human activities related to fall detection and was generated by Espinosa et al. [42]. They provided a fall detection system that uses several cameras and a 2D CNN inference strategy to analyze images in certain time frames. Additionally, they provided an optical flow method for obtaining characteristics that accumulates information on the total velocity in two subsequent images. The proposed multi-vision-based technology recognizes human falls with an accuracy of 95.64% when compared to state-of-the-art methods utilizing a straightforward CNN network design. According to Casilari et al. [43], UMAFall is an innovative collection of movement traces collected by meticulous simulation of several predetermined activities of daily life, along with recordings of falls. The UMAFall dataset was collected by placing sensors on the bodies of subject for tracing their movement. It generated data from five locations for which separate sensors were attached to the body. By tracking the movements of 17 experimental subjects, it distinguishes between three different types of falls. The data included magnetometer sensor data, acceleration and gyroscope data and sampling data from smartphone. Several "Fall Detection" datasets have been used for calculating experimental results published in this research.

Martínez-Villaseñor et al. [44] presents the Fall Detection Dataset based on raw data of daily movements collected from 17 young adults. There were 11 types of activities recorded in data in addition to fall events. The objective of collecting and making this dataset available for research was to aid the efforts of detecting falls using recent trends and advancements of machine learning. It also provides a range of experimental video and machine learning options, including pattern recognition. Using acceleration data from cellphones, Micucci et al. [45] offered an amazing work that presents a comprehensive dataset for human activity detection.

The efficiency of the suggested dataset has been demonstrated by the authors through the presentation of thorough statistics, experimental settings, and benchmark results. In conclusion, this research is a significant addition that offers future researchers a helpful tool to assess their models.

While fall detection is an active research topic for past two decades and researchers have used different techniques to improve the accuracy of detection and enhance the performance of applied algorithms, it is felt that there are still gaps to come up with a generalized solution. In these studies, limited datasets are used that undermine the efficacy of the model. The accuracy and relevance of sensor data is an important factor like most of existing research work is based on simulated young adults' data that is not equally applicable for elderly. Difficulties in dataset collection and scarcity in publicly available datasets are major impediments for the development of impactful algorithms for fall detection. These gaps are considered and addressed in this research as outlined in the next chapters.

3. PROPOSED FRAMEWORK

To cover the identified gaps, we propose a robust high-performance smartphone-based fall detection system for elderly. A new dataset is generated from accelerometer signals that is better representative of old age population. A subset related to thigh location data is used in this research. From literature review it is also established that proper evaluation of designed algorithms is not conducted using several datasets that reduces the prediction validity. To overcome these missing aspects, we have used four datasets for performance analysis.

The proposed model's core intelligence is provided by ML and DL techniques, feature selection with ranking, enabling multidimensional analysis of data dynamically, all this aids in improving performance of the classification model. Fig 3.1 below gives a general overview of the proposed framework.

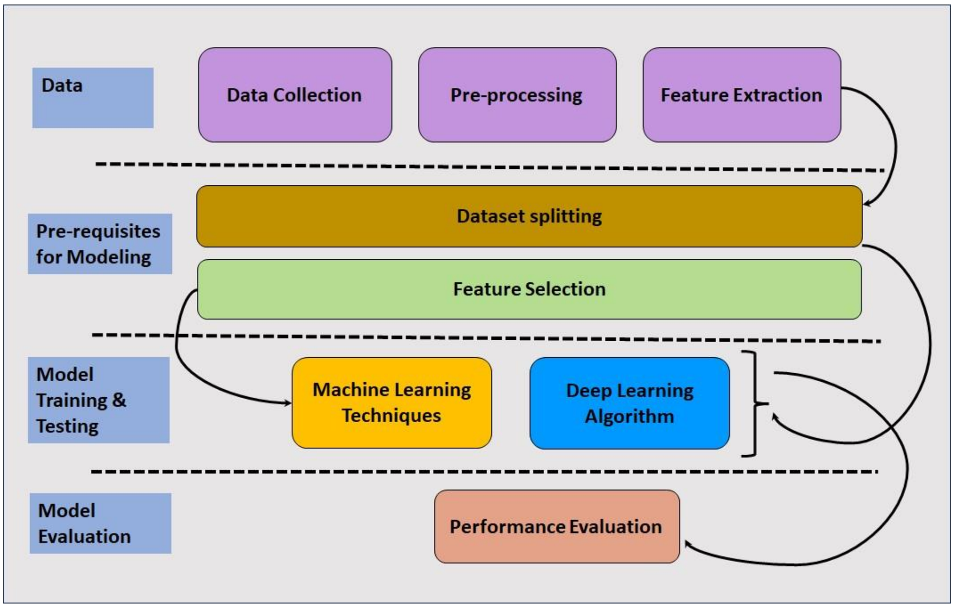

Figure 3-1Overview of Proposed Framework

3.1 Methodology

The proposed methodology employs smartphone accelerometer data that detects potential falls with the built-in accelerometer sensor on a smartphone that measures the acceleration recorded by the device during movement of the subject/person. When a fall occurs, the acceleration pattern changes in a distinctive way and serves as criteria for fall detection algorithm. The main considerations in this research are:

1) Provides a thigh location smartphone-based solution for fall detection systems by using a newly generated Dataset of the elderly activities of daily life (ADLs) along other Smartphone based datasets.

2) A threshold-based algorithm is applied for screening the data and to discard the fall like ADLs in first instance.

3) After Data pre-processing, feature extraction and feature ranking techniques are enforced to improve the accuracy of algorithms.

4) Data is then fed to classification models i.e., Random Forest (ML based algorithm) and to a DL based model BiLSTM.

5) The models learn to distinguish fall patterns from ADLs based on the extracted and selected features.

6) The trained model is then evaluated on a couple of datasets to assess its performance and to come up with an optimal solution.

3.2 Smartphone Accelerometer Data

The design of a reliable, robust, and easy to deploy solution to effectively detect elderly falls must have some basic characteristics like it must be based on representative data, should be tested using cross-dimensional datasets for the purpose of generalization and later when used to facilitate the alerts for medical assistance it must be easy to deploy and easy to use which are derived from the basic design and architecture.

Adequate strategies and comprehensive testing across several datasets are imperative to the success and effectiveness of FDS and the proposed system considers all these crucial factors.

Smartphone continuously collects accelerometer data, which includes three axes: X, Y, and Z. These axes represent the device's movement in three dimensions.

1. X-axis: Represents the acceleration along the horizontal axis, usually left-right movement.
2. Y-axis: Represents the acceleration along the vertical axis, usually up-down movement.

3. Z-axis: Represents the acceleration along the depth axis, usually forward-backward movement.

Fig 3.2 represents how smartphone sensor data is used for fall detection system design.

Figure 3-2 Smartphone Accelerometer sensor

Accelerometers measure acceleration in units of gravity (g) or meters per second squared (m/s^2). When the object is at rest on a flat surface, the acceleration data should show approximately 1g in the Z-axis due to the Earth's gravity. The X and Y axis should show values close to zero since there is no movement. When the object is in motion or subjected to acceleration, the accelerometer data will show fluctuations in all three axes, capturing the changes in movement and orientation over time. These fluctuations can be used to analyze various physical activities, gestures, or movements.

3.3 Data Acquisition

Existing research on fall detection systems usually relies (or are adjusted/- trained) on limited set of human fall data that is basically non-representative of the real fall events [46]. Capturing real-life fall events is a rigorous process which involves a costly infrastructure to capture the fall events. Moreover, the accuracy of a FDS is most of the time assessed by the performance metrics only without performing organized testing on varied set of data. A list of public datasets used for robust testing of the algorithm is also provided hereunder.

3.3.1 FallMEdADL Dataset

To overcome the data limitation, collaboration with a Korean institute has been done for collection of data based on scripted activities performed by the elderly to generate elderly ADL data. We here present the new smartphone accelerometer-based ADL dataset named FallMEdADL.

This elderly ADL data combined with data from young volunteers' simulated falls, serve as a better representative dataset for effective modeling.

Protocols used for Data Collection

For this activity, thirty-five volunteers (26 male and 9 females; age: above 60 years, weight: 59 ± 10.5, and height: 166.7 ± 7.68) were asked to perform the scripted set of fall-like ADLs. We mainly focused on fall-like ADLs that can be easily misinterpreted as falls [37] and an exhaustive list is given in Table 3.1.

Participants used a commercially available smartphone securely placed at thigh position in their pant pocket during the experiments. The smartphone's built-in accelerometer recorded tri-axial acceleration data at a frequency sample of 100 Hz (100 rows of data per sec). To ensure data quality, we carefully calibrated and synchronized the smartphones before each experiment. A consent form was also signed by the participants while the privacy & compliance team of the institute had reviewed and approved the entire process.

Based on the experiments, 178 elderly ADLs were recorded from thigh location that were marked as potential falls. This elderly data combined with young adults' fall data comprising of 175 fall events is used in the classification algorithm. We believe that this dataset will also assist further study of fall prevention and elderly care by acting as a baseline for assessing fall detection algorithms.

This dataset is intended to be published and made available to the researchers who are urged to access it for educational purposes.

Table 3-1 Fall-Like ADLs considered for Data Collection

No.	Description
1.	Walking i) upstairs ii) downstairs iii) few steps iv) quick & slow walking
2.	Lying down i) on floor
3.	Jumping i) Gently ii) quickly
4.	Sitting on i) sofa ii) chair iii) floor iv) ground v) car
5.	Bending down to pick something from floor
6.	Sensor Hit i) a pole ii) by a person
7.	Standing i) bending knees ii) from sofa/chair

3.3.2 FallDroid Dataset

The second important dataset used for cross evaluation is FallDroid dataset proposed by [37] and is a proprietary dataset. This contains complex fall-like ADL events that are tricky for the classification algorithm. This dataset has 287 potential fall ADL and 175 fall events. For this experiment, twenty people volunteered including 17 males and 3 females with an average age of 28.45±2.72, weight of 66.15±10.83 and height of 170.7 ± 7.68 participated in the data collection. Young adults performed simulated falls (forward, backward, lateral left, and right) and nine different types of ADLs (standing up from sitting, sitting down, bending down, lying down, moving up and down stairs, hit instance of the smartphone, jogging, running, and jumping). The smartphone accelerometer sensor data from two LG G2 phones that were used to record data. The participants carried two smart phones, one on the waist location tied to belt and second one placed at thigh location in the pant pocket. An application configured in smartphones collected data at a sampling rate of 64Hz.

3.3.3 Mobiact Dataset

To validate the proposed solution across multiple datasets the third dataset used is a public dataset collected by Biomedical Informatics and released by Vavoulas et al. 2016 [48] and out of available two versions the updated version is used in this study.

A Samsung Galaxy S3 smartphone is used for experiments and was placed in the pant pockets of participants. Using the built-in sensors of gyroscopes, accelerometer, and angular position the data was collected with Sensor-DelayFastest option enabled to attain the maximum feasible sampling rate.

This version listed 12 different ADLs along with 4 different types of falls from a total of 66 subjects with more than 3200 trials were captured. This included the motionless state of the subject and activities like walking, jogging, and standing etc. and a detailed account of these is presented in [48].

From this publicly available dataset we have used the accelerometer sensor data only collected from the pant pocket position.

3.3.4 UniMib SHAR Dataset

This is the fourth dataset used for testing the robustness of proposed algorithms. It is a publicly available dataset collected by Software Architecture Laboratory (sal) [49].

It is based on accelerator samples acquired using a Samsung Galaxy Nexus 19250 smartphone and used for activity recognition and fall detection. The data was captured at a sampling rate of 50 Hz from the built-in triaxial accelerometer The dataset includes 11,771 samples collected from 30 subjects with 24 males and 6 females of ages ranging between 18 to 60 years. The ADLs and fall related activities were performed at least five times by the subjects.

The scripted set of fall-like ADLs and simulated falls were performed while the phone was placed in both left and right pant pockets simultaneously. Different classes and groups of acquired data were made based on ADLs and Falls. The raw accelerometer signals given in the dataset were used in this study. [50]

3.4 Data Preprocessing

3.4.1 Data Filtering

A pre-screening of data is done by applying a simple threshold-based algorithm that filters the fall-like ADLS by applying inactivity and posture tests. Two accelerometer features, namely AAMV and TAV relevant to acceleration and Tilt angle variation are used to apply this filter check. To search for fall-like events/peaks within the recorded dataset a variable size window is detected around impact and when AAMV < .6 with TAV> 10° then it is marked as a fall event.

A fall happens at once and with no repeated instances and same is translated with respect to acceleration that reaches a peak value of 3-g threshold for fall, and for 5 seconds no further activity is observed and acceleration of 1-g corresponding to gravitational acceleration is observed that confirms a fall. A detailed account of this Threshold-based algorithm for data filtering is available in the FallDroid paper [37].

By applying this algorithm to FallMEdADL data the number of fall-like ADLs was filtered to 178 as compared to original 533 fall-like elderly ADLs.

Similarly, passing the FallDroid dataset to algorithm resulted in filtered potential fall related ADLs that were shrinked to 196 as compared to original 287 fall-like events. For Mobiact dataset, the threshold value was 1.9g and consequently 768 falls and 2106 ADLs were detected. For UniMiB SHAR, the threshold value set in the algorithm was 2.4g. This value is different from the FallDroid dataset because by setting this value, all fall events passed the threshold and were considered in the final dataset as per basic requirement of fall detection. As a result, 308 fall events and 202 ADLs were detected.

While this algorithm is simple and computationally efficient, yet it is sensitive to environmental noise and could not correctly detect more complex falls. For this reason, machine learning and deep learning algorithms are used as second step for accurate fall detection.

3.4.2 Down sampling

To use a balanced dataset have applied down-sampling on the UniMib SHAR dataset and used 900 ADLs out of total 2106 and it was carefully ensured that a representation from all types of ADLs is included in the subset of data.

3.5 Feature Extraction

Feature extraction involves transforming the raw accelerometer readings into meaningful representations that capture important information about the movement patterns. Some common features that can be extracted from accelerometer data for fall detection include:

1. **Statistical Features**: Mean, median, standard deviation, variance, skewness, kurtosis, etc., calculated over a window of time.
2. **Frequency Domain Features**: Using techniques like Fast Fourier Transform (FFT) to analyze the frequency components of the signal.
3. **Time-Domain Features**: Root Mean Square (RMS), zero-crossing rate, correlation between different axes, etc.
4. **Machine Learning-Derived Features**: Features obtained from applying machine learning algorithms on the raw data, such as peak detection, slope detection, etc.

The formula and details of these and other significant features are given in table 3.2.

From the accelerometer data features relevant to fall detection are extracted based on the values of Acceleration vector magnitude (AVM). A threshold-based algorithm is used for feature extraction that monitors the acceleration values and when a peak is detected (define as a value of 3g) an impact window is activated. After a wait of 3.5 seconds if no further activity or peaks are noted and subject comes to an acceleration of 1g then it is tagged as a potential fall [37]. The threshold value can be adjusted to account for the recoded traits of data and falls between 1.5g-3g which is very low and helps to curtail false negatives. Fig 3.2 below provides its pictorial representation.

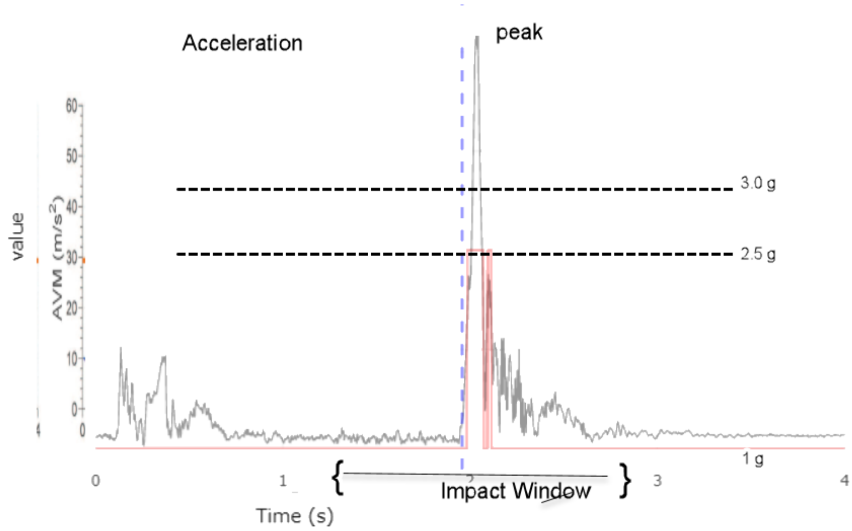

Figure 3-3 Simple Fall Detection using Threshold.

Depending on the context of research and objectives a varying combination of features can be gathered from rich accelerometer data. For this study, data extraction techniques as briefly explained above are used to collect a subset of features relevant to falls and probable falls to be used for fall detection. The focus is on the features that provide valuable insights into the behaviour or movements to be analyzed that are then incorporated in the classification algorithms.

A list with description of the features extracted for this research based on research done by [37] is given in Table 3.2 below.

Table 3-2 Features extracted from Accelerometer Signals

Feature No.	Feature Name	Description
1	Avg Absolute acceleration magnitude variation (AAMV)	AAMV $[n]$ = (1/N) $\Sigma i \in w\ in$ |AVM[i + 1]-AVM[i]|
2	Mean	Mean = $\Sigma i \in w\ in$ AVM[i] / N
3	Median	N (odd): Med = AccelerationValues[(N + 1) / 2] N (even): Med = (AccelerationValues[N / 2] + AccelerationValues[(N / 2) + 1]) / 2
4	Range	R = Max(AccelerationValues) - Min(AccelerationValues)
5	Step Count Index (SCI)	SCI = $\sum(\sqrt{(ax^2 + ay^2 + az^2)})$ / g
6-8	Net Axis Acceleration along x,y,z axis SAA	SAA$x/y/z$ = $\Sigma\ i \in win\ \lvert Ax/y/z\ [i]\rvert$
9	Summed Magnitude Area (SMA)	SMA = $\Sigma\ i \in win\ (\lvert Ax\ [i]\rvert + \lvert Ay\ [i]\rvert + \lvert Az\ [i]\rvert)$
10-12	The Angle variation with tilt (TAV along x,y,z)	Avg (TA[i] = arccos ($A_{lp}[i]/AVM_{lp}\ [i]$))
13	Impact Duration Index (IDI)	IDI = $EndTime_i - StartTime_i$
14	Maximum Peak Index (MPI)	MPI = PeakValue / g PeakValue is the magnitude of the highest peak in acceleration. Magnitude = $\sqrt{(ax^2 + ay^2 + az^2)}$ g is the acceleration due to gravity (approximately 9.81 m/s^2)
15	Maximum Value Index (MVI)	MVI = Max(|ax|, |ay|, |az|) / g |ax|, |ay|, |az| are the absolute values of the acceleration components along the x, y, and z axes respectively. g is the acceleration due to gravity (approximately 9.81 m/s^2)
16	Peak Duration Index (PDI)	PDI = PeakTime − StartTime/T T = Total Duration of Window
17	Activity Ratio Index (ARI)	ARI = (Σ|a(t)|) / (N * g) Σ|a(t)| is the sum of the absolute magnitudes of acceleration values over a certain time window. N is the number of samples within the time window. g is the acceleration due to gravity (approximately 9.81 m/s^2).

18	Activity Ratio Index (FFI)	ARI = (Count of High Activity Instances) / (Total Number of Instances)
19	Variance	Var = (1/Total Number of Samples) $\Sigma i \in w\ in\ (Xi - Mean)$
20	Standard Deviation	Std = (√(1/Total Number of Samples)) $\Sigma i \in w\ in\ (Xi - Mean)$
21	Root mean Square (RMS)	RMS = √($x_0^2 + x_1^2 + x_2^2 + \cdots + x_{N-1}^2$)/ Total Number of Samples

3.6 Feature Selection

Feature selection helps to choose a subset of relevant features from the extracted features that are given as input to the machine learning model and to BiLSTM. By selecting the most important and relevant features only, efficiency and performance of the model is improved. It also helps to avoid overfitting.

Different feature selection techniques depending on the type of data, model and specific to implementation framework can be used. For this study 'rankfeatures' function available in MATLAB Statistics and Machine Learning Toolbox is applied with "entropy" and "roc" as the criteria to assess the significance of each feature.[51]

For the BiLSTM implementation we have again selected important features by using Mutual Information (MI) and Eigenvector Centrality (EC) methods. These functions are available in the Feature Selection library (FSlib) [52] in MATLAB Toolbox. Mutual Information is a widely used metric for feature selection in various data analysis tasks, including fall detection. It is used to quantify the amount of information that two random variables share. In the context of feature selection, it helps to determine the relevance of each feature (attribute) with respect to target variable (e.g., fall or no-fall label) in the dataset. Relevant to fall detection, the variables share some mutual or common information measured using this function. Through ranking the features having high mutual information would be ranked higher and thus are best suited for use in the classification. The features with low mutual information are discarded.

Eigenvector Centrality (EC) is a filter-based method in which features are visualized as nodes in affinity graph and then the nodes with central position represents the importance [53]. For the sake of selecting features, they can be ranked first based on the notion of centrality (EC) or can be directly selected. Neighboring nodes are evaluated for their importance and then importance is determined. This method also helps to reduce the number of irrelevant features from the dataset, and this improves the classification accuracy and performance.

Before applying feature selection, data is split into training and testing datasets with the purpose of assessing the performance of proposed models and ensure that they generalize well to new, unseen data. Using the training datasets, performance is further enhanced as explained above by selecting best features before data is fed to the models.

3.7 Classification Algorithms

The objective of this study is to improve the accuracy of fall detection by selecting a representative dataset specifically collected from elderly and to test the robustness of the solution a couple of other data sets are employed as well. The choice of classification model on one hand is based on the data stream and at the same time need to evaluate against available benchmarks already available using the reference data sets. To attain these results two classification models namely Random Forest and BiLSTM are used and new dataset FallMEdADL is used as input to both models with promising results.

3.7.1 Random Forest

Random Forest is a machine learning algorithm that can be used for a variety of tasks, including fall detection. Fall detection involves identifying instances when a person experiences a sudden drop or collapse, which could indicate a potential fall and subsequent need for assistance. It is an ensemble learning method that combines multiple decision trees to make more accurate predictions. Random Forest builds a collection of decision trees during the training phase. Each decision tree is trained on a different subset of the training data, and they operate independently. At each node of every decision tree, a random subset of features is considered for splitting. This randomness helps prevent overfitting and decorrelates the trees. During the prediction phase, each decision tree in the forest independently classifies the input data. The class with the most "votes" from the individual trees becomes the final prediction.

Fig 3.4 shows the typical working of a Random Forest algorithm.

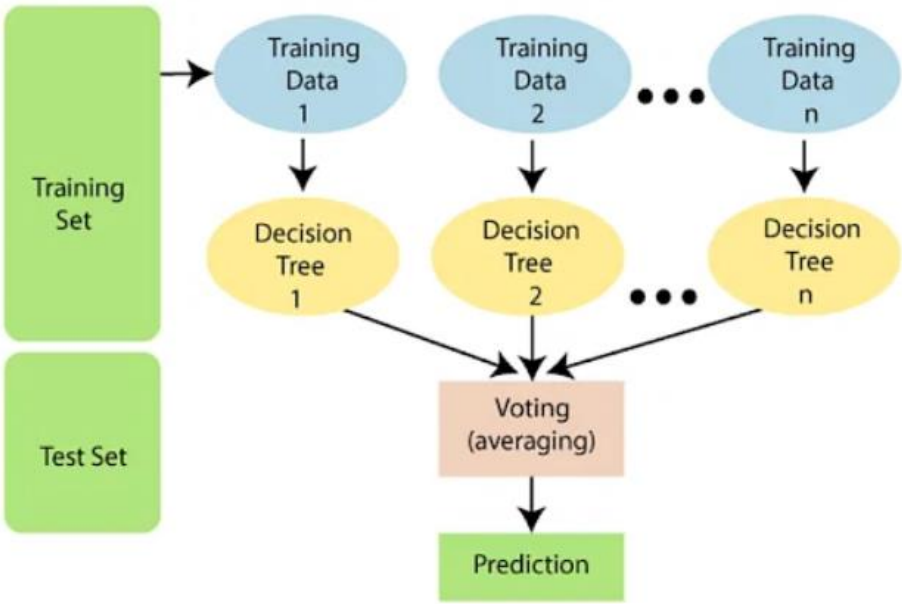

Figure 3-4 Random Forest Algorithm

3.7.2 Bidirectional Long Short-Term Memory (BiLSTM)

Using Bidirectional Long Short-Term Memory (BiLSTM) networks for fall detection is a feasible and effective approach. BiLSTMs are a type of recurrent neural network (RNN) architecture that can capture both past and future context in sequences, making them well-suited for tasks like fall detection where temporal dependencies are important.

BiLSTMs process the sequential data in two directions: from the beginning to the end (forward pass) and from the end to the beginning (backward pass). This bidirectional processing allows the network to capture dependencies not only on the past but also on the future, enhancing its ability to understand the context of the input sequence. Within the BiLSTM architecture, each time step in the sequence is processed by an LSTM cell.

The BiLSTM network is trained using labeled data, where the input sequences are associated with fall or non-fall labels. During training, the network learns to adjust its internal parameters (weights and biases) to minimize the difference between its predictions and the actual labels. Once trained, the BiLSTM model can be used to predict fall events in real-time. Given a new sequence of input data, the model produces an output indicating the likelihood of a fall occurrence. Fig 3.5 shows a simple architecture of BiLSTM classifier.

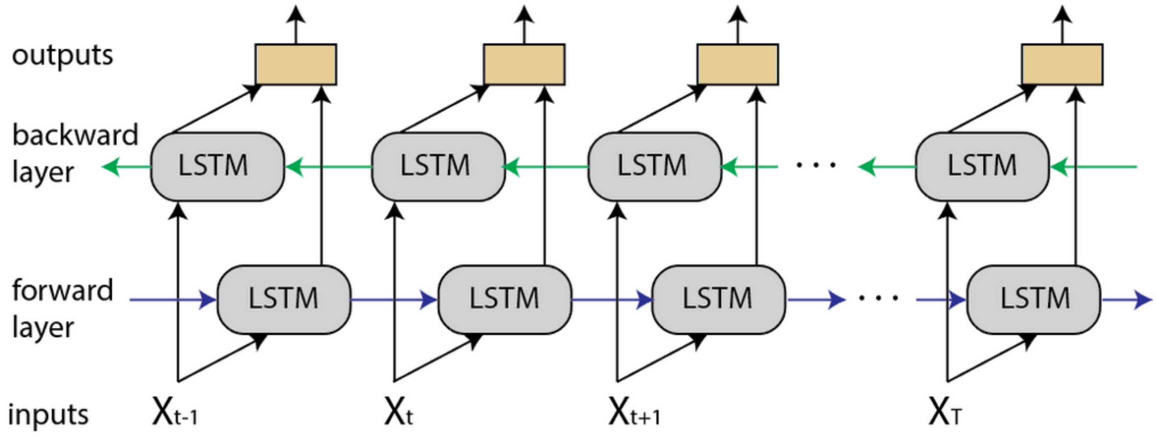

Figure 3-5Architecture of BiLSTM Classifier

The results of classification algorithms are analyzed using performance metrics related to accuracy, sensitivity, and specificity.

Fig 3.6 depicts how different modules and data interact with each other. A detailed account of experimental setup and implementation details is presented in next chapter.

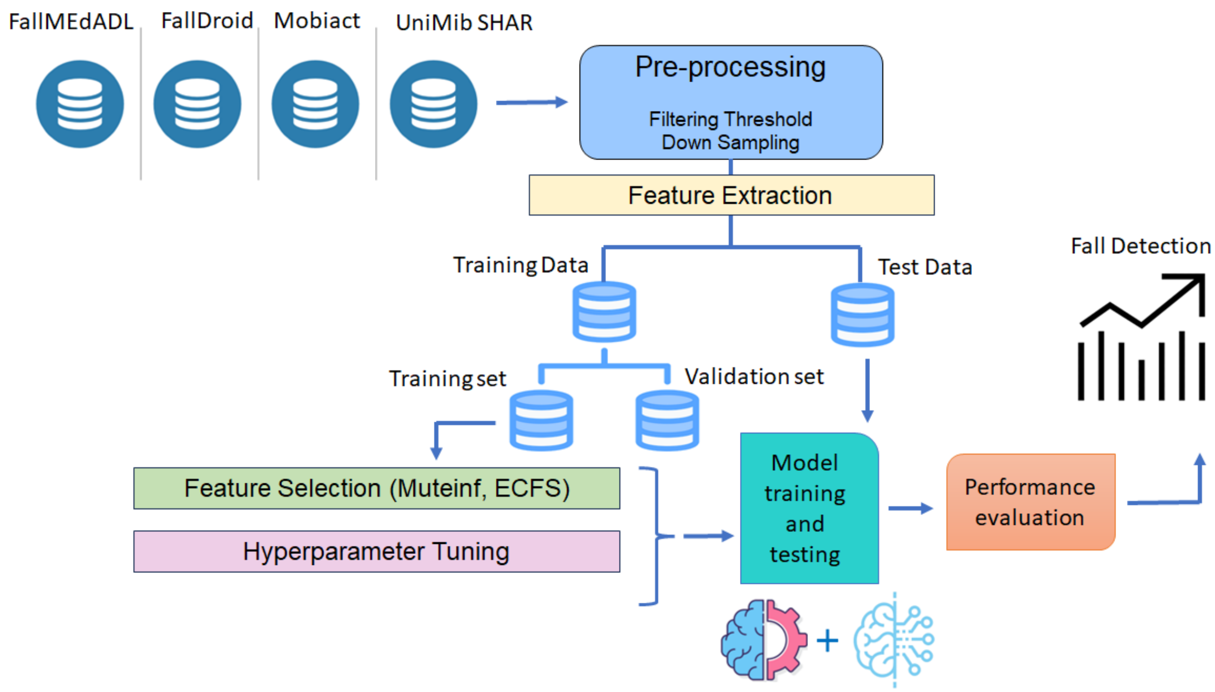

Figure 3-6 Complete Workflow of Proposed solution

Chapter 4

4 EXPERIMENTAL SETUP & RESULTS

In this chapter, we will explain in detail the experimental setup where the learning and concepts of machine learning and deep learning classification are put into implementation. This phase includes defining the parameters, like number of splits, layer configurations, activation functions, and performance metrics selected and applied for evaluation of results. The training process, with convergence criteria along with application of feature selection techniques are also discussed.

4.1 Experimental Setup for FallMEdADL Dataset

In sections 3.3 and 3.4 above, the data preparation methods and functions applied prior to utilizing the data to train the models are described in detail. Section 3.5 describes the features of the dataset that are used to train the ML models. Feature selection techniques are also discussed at length in section 3.6. The next step after feature extraction is to determine the optimum model for both RF and BiLSTM classification models in terms of hyperparameters. Both models are implemented in MATLAB R2023a.

Different hyperparameters like “No of units”, “Nu of Splits”, and “n estimators” are applied and tested with different values to enhance the predictive capability of the model. The RF model was trained and evaluated by iteratively choosing the number of trees from 1-50. The MaxNumSplits parameter was set to 15 and the Method was set to classification and TreeBagger ensemble for classification is used. The 5-fold cross-validation was used to ensure the fairness of results. A total of 100 iterations were performed on each number of trees and averaged out the results to find out the best tree value for which maximum performance was achieved. Figure 4.1a shows the best performance was achieved for 24 numbers of trees for new dataset and 35 number of trees for FallDroid, Mobiact and UniMib Shar datasets.

A graph is created showing the best accuracies and error rates. A forest is created based on the best estimator value with a given “max depth” and “min sample leaf”.

The graphical view of the first trained classification tree while doing iterations is shown below in Fig 4.1 where value ‘1’ represents a Fall and value ‘-1’ depicts ‘No-Fall’.

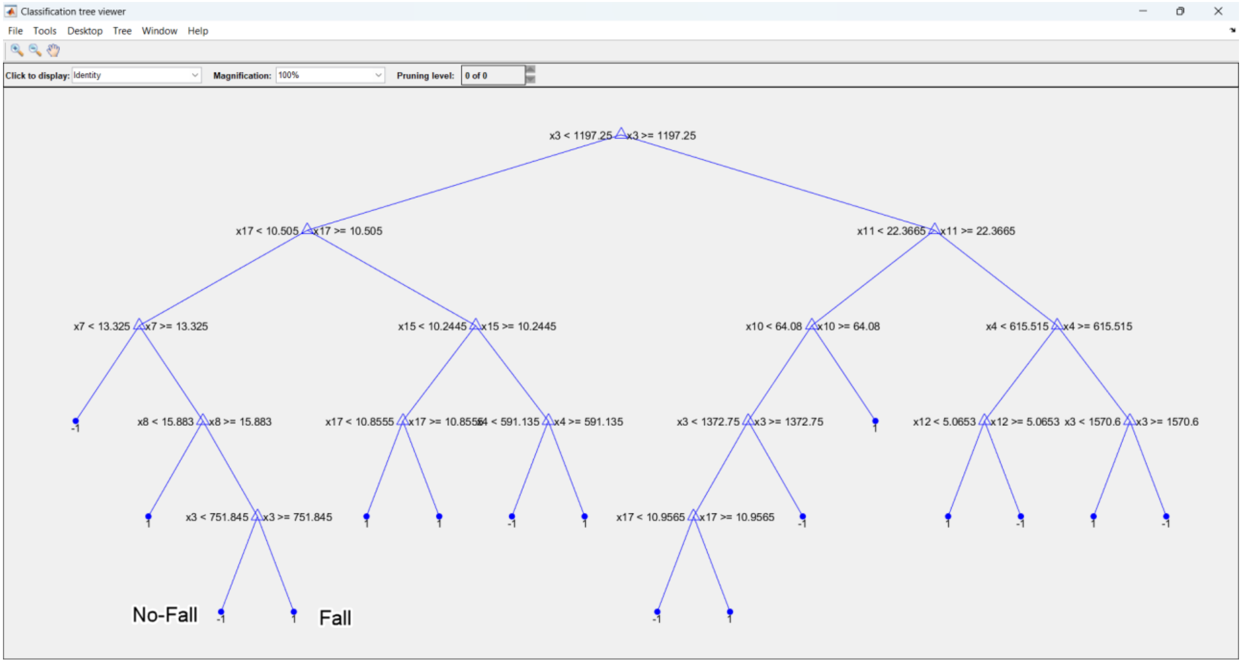

Figure 4-1 Implemented Classification Tree Instance

In the case of the BiLSTM model, the best model was found by iteratively choosing the value of the number of units from 1-90 for a single BiLSTM layer. The complete architecture for the BiLSTM model is shown in Figure 4.2.

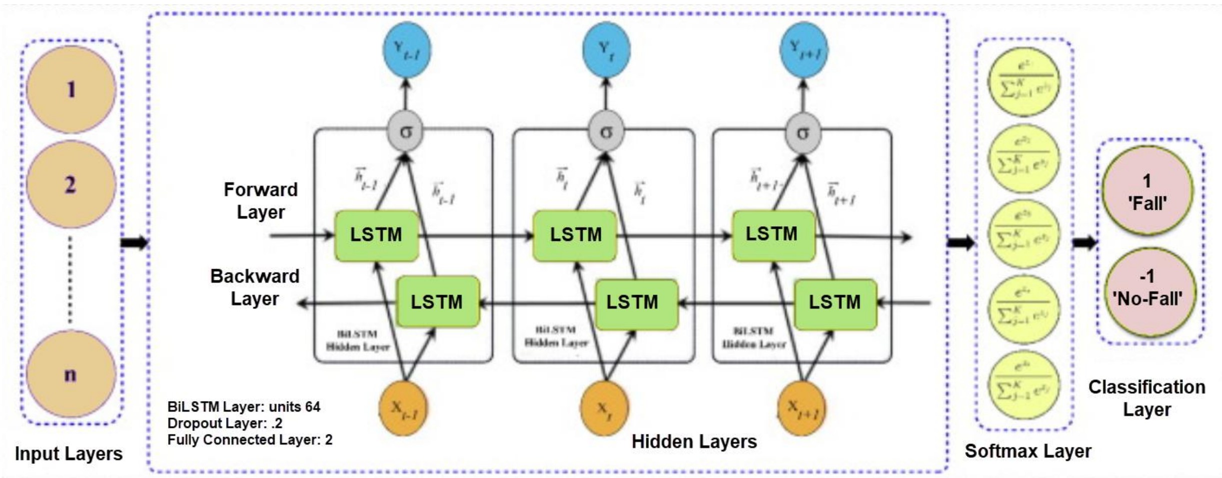

Figure 4-2 Implemented BiLSTM Architecture

A total of 100 iterations were performed on each number of units and averaged out the results to find out the best number of units for which maximum performance was achieved. Tests revealed that 63 units in the BiLSTM layer gave the best output and used as the hyper-parameter.

A typical training instance of BiLSTM Training Progress is shown in Fig 4.3 below.

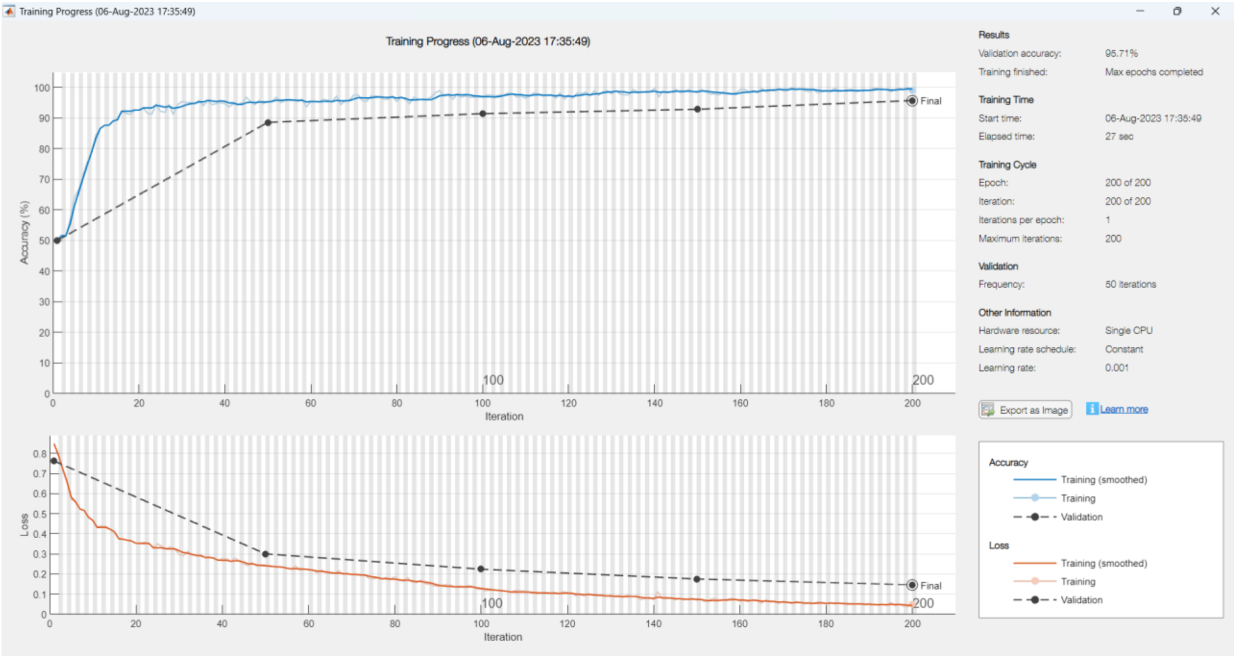

Figure 4-3 Training Progress of an instance of BiLSTM

Table 4.1 shows the hyper parameter values used for both RF and BiLSTM classification models.

Table 4-1 Hyperparameters used for Classification Models

S. No.	Classification Model	Hyper Parameters
1.	Random Forest	Splits = 15 No of folds = 5 Tested with Trees = 1-50 Best results at Trees = 35
2.	BiLSTM	Optimizer = Adam No of epochs=200 Batch size = 4 Learning rate = .001 Dropout = .2 Units Tested = 1-90 Best results with units = 63

The results of RF and BiLSTM for the hyperparameter tuning are shown in Fig 4.4 and Fig 4.5 respectively.

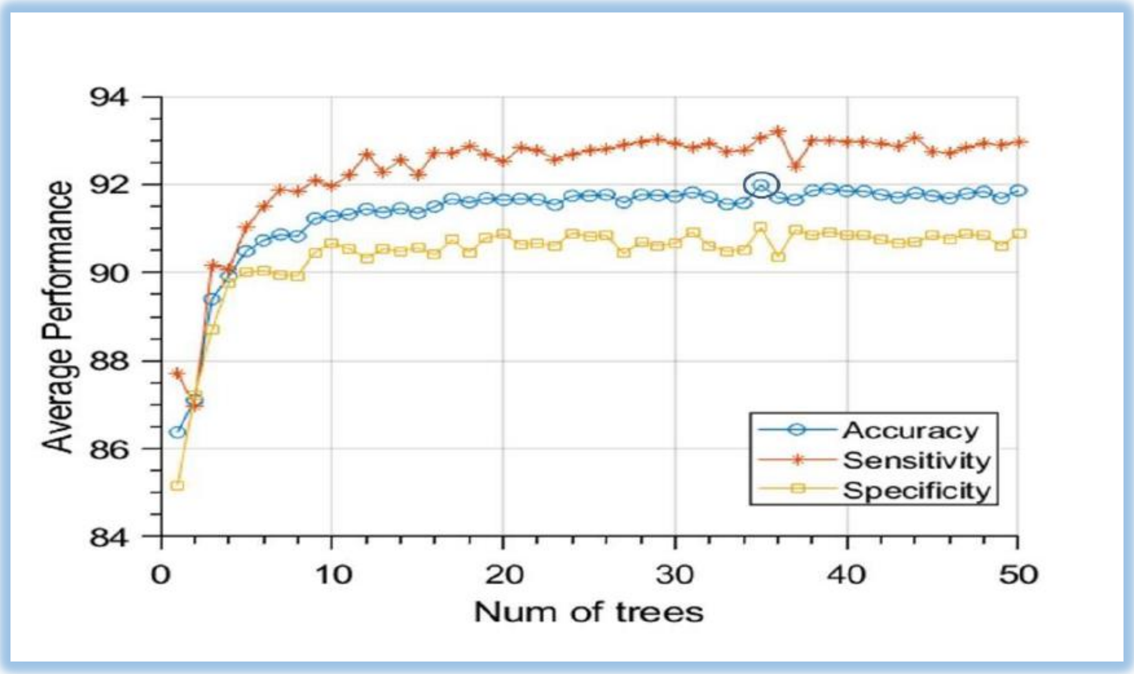

Figure 4-4 Avg. Performance of Random Forest

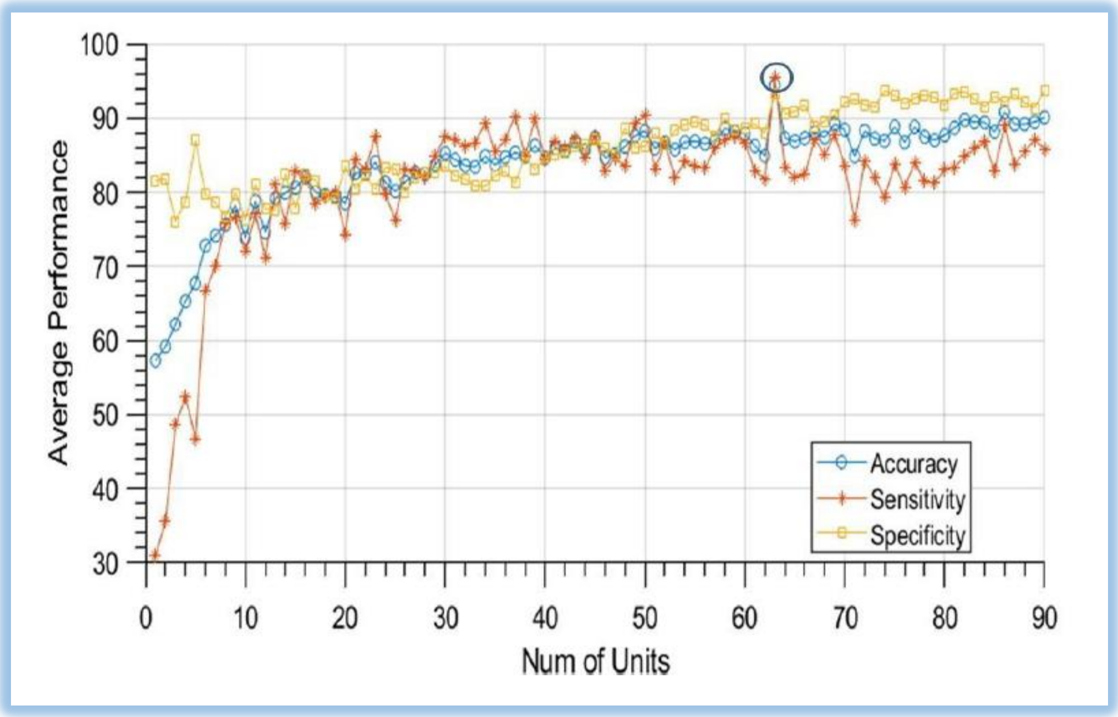

Figure 4-5 Avg. Performance of BiLSTM

After this, the two FS techniques were applied to improve the performance, efficiency, and interpretability of the models and to reduce overfitting.

For applying ‘Mut-Inf’ on the 21 features for FallMEdADL dataset, it was first split into an 80:20 ratio. The training set was sent as input parameter to the Muteinf function implemented in MATLAB FSlib library. This function returns the feature vector based on ranking from the most important to less important. These features were then sent iteratively one by one from top to bottom to find fit for both RF and BiLSTM models. A total of 100 iterations were performed to find out the average performance for all 21 features. For the RF model 15 features gave the best average performance and for the BiLSTM model 9 features were ranked as of high importance and gave the best results.

In the case of ‘ECFS’ function for feature selection, the best average performance for the RF model was achieved with 19 features and for the BiLSTM model with 16 features respectively. A detailed account of feature selection techniques is given under section 3.6.

Figure 4.6 shows the occurrence of features in the top 15 out of all 21 features for the RF model which shows that these features were selected repeatedly in all iterations and are prominent features.

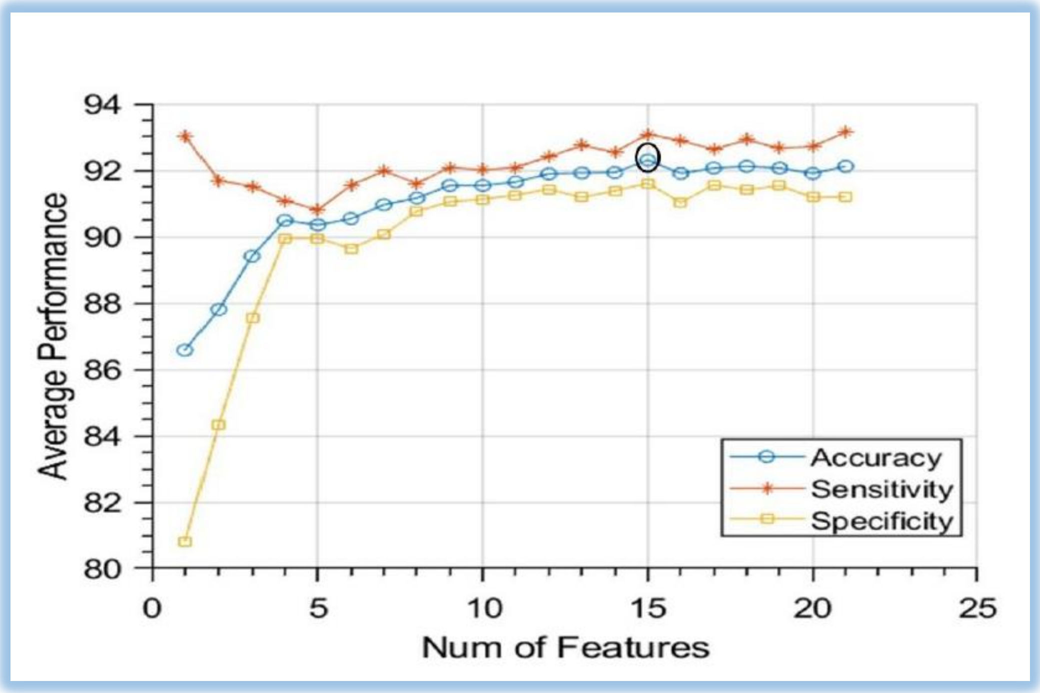

Figure 4-6 Performance of RF using MI FS Technique

Figure 4.7 shows the occurrence of features in the top 9 out of all 21 features for the BiLSTM model.

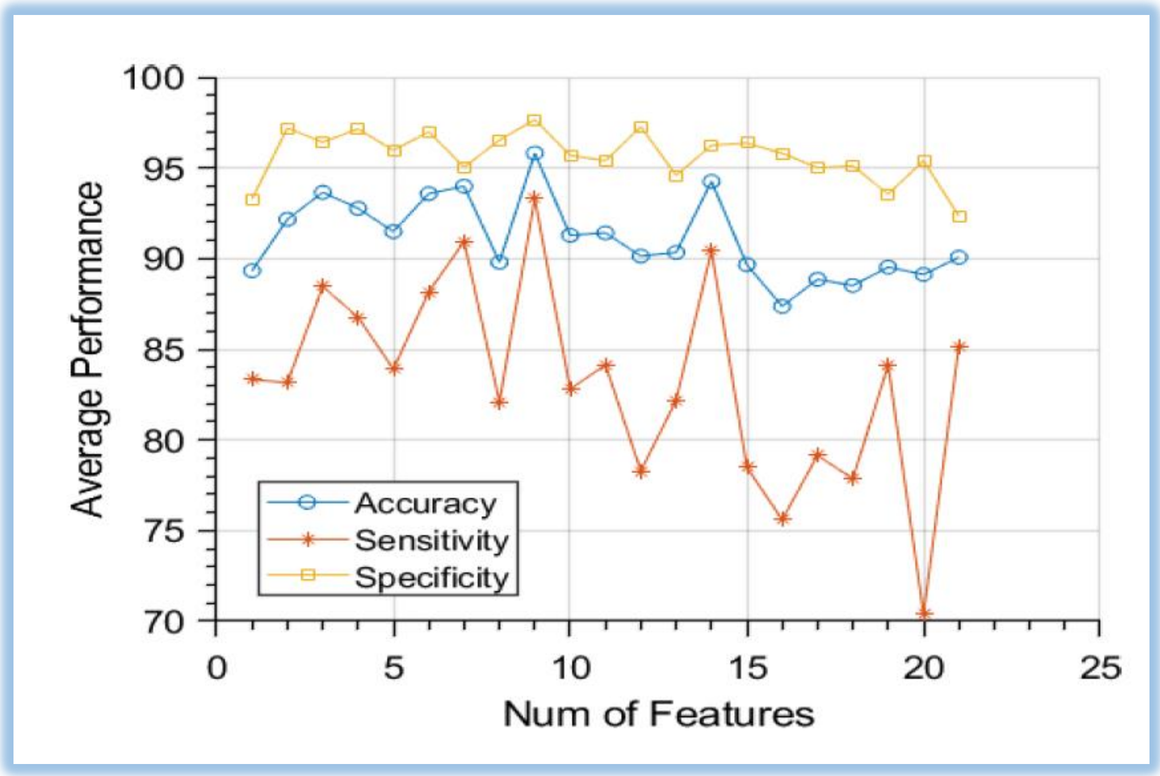

Figure 4-7 Performance of BiLSTM using MI FS Technique

4.2 Overall Testing on FallMEdADL, FallDroid, UniMiB SHAR, and Mobiact Dataset

This critical phase serves as the foundation upon which our entire research effort is built, defining the parameters and methodologies that govern the acquisition of knowledge through computational analysis. Also, in this research we present the cross evaluation and analysis using 4 multiple datasets comprising both public datasets and proprietary datasets along with new elderly data.

Performance evaluation metrics such as accuracy, sensitivity, and specificity are commonly used in the field of ML and DL, particularly in binary classification tasks. These metrics help assess the performance of a classification model by comparing its predictions to the actual ground truth. Accuracy is one of the most basic and intuitive metrics. It measures the overall correctness of the model's predictions. It is calculated as the ratio of the number of correct predictions (both true positives and true negatives) to the total number of predictions (true positives, true negatives, false positives, and false negatives).

Accuracy = (TP + TN) / (TP + TN + FP + FN)

where:

- TP (True Positives): The number of positive instances correctly predicted as positive.
- TN (True Negatives): The number of negative instances correctly predicted as negative.
- FP (False Positives): The number of negative instances incorrectly predicted as positive.
- FN (False Negatives): The number of positive instances incorrectly predicted as negative.

Sensitivity (also known as Recall or True Positive Rate) measures the ability of the model to correctly identify positive instances (the ones that belong to the positive class). It is the ratio of true positive predictions to the total number of actual positive instances.

Sensitivity = TP / (TP + FN)

Specificity: Specificity measures the ability of the model to correctly identify negative instances (the ones that belong to the negative class). It is the ratio of true negative predictions to the total number of actual negative instances.

Specificity = TN / (TN + FP)

These metrics provide valuable insights into the strengths and weaknesses of a classification model and can help guide the model optimization process based on the specific requirements of the application. To ensure the effectiveness of these metrices we had done the data preprocessing and datasets were balanced. The classification models envisaged and discussed in section 4.1 applied on the FallMEdADL dataset are evaluated on the reference datasets of FallDroid, UniMiB SHAR and Mobiact datasets to ensure robust testing of algorithms employed. All these datasets after the feature extraction step were first evaluated on the RF model by using both FS techniques. The RF model proposed on the FallDroid dataset remain fixed by setting the number of trees=35 and MaxNumSplits=15. RF model was trained and tested on the 15 features dataset using a total of 100 iterations and the average performance was calculated.

For the ECFS technique, the top 19 features were fixed, and the RF model was trained and tested on 19 features dataset and average performance was computed. For the BiLSTM model, the FallDroid dataset performed best for the single layer with the number of units=63. So, the model parameters along with other training options listed in Table 4.1 remain the same. Moreover, for MI FS and FC FS techniques, the top 9 features and 16 features were fixed respectively and the BiLSTM model was trained and tested on the 9 and 16 features dataset. A total of 30 iterations were performed and the average performance was computed.

In the case of the UniMiB SHAR dataset, for the MI FS technique, the best ranked features were the top 15 and top 9 for the RF and BiLSTM models respectively. For ECFS, the occurrence of features in the top 19 and top 16 for RF and BiLSTM models respectively.

After applying the feature selection, RF and BiLSTM models were trained and tested for the four datasets to get generalized results. Their performance is evaluated based on measures of accuracy, sensitivity, and specificity. Table 4.2 shows the results of all the average performance measures for the four reference datasets using RF algorithm with feature selection.

Table 4-2 Avg performance of RF Classifier using all datasets

Dataset	Accuracy	Sensitivity	Specificity
FallMEdADL	97.36%	97.12%	94.66%
FallDroid	92.30%	93.09%	91.60%
Mobiact	95.05%	97.10%	93.30%
UniMib SHAR	97.30%	99.38%	94.13%

Table 4.3 shows the results of all the average performance measures for the four reference datasets using BiLSTM algorithm with feature selection.

Table 4-3 Avg performance of BiLSTM Classifier using all datasets

Dataset	Accuracy	Sensitivity	Specificity
FallMEdADL	98.23%	96.94%	99.09%
FallDroid	95.81%	93.31%	97.66%
Mobiact	98.42%	98.50%	97.84%
UniMib SHAR	98.30%	99.48%	96.40%

Table 4.4 compares the performance of newly proposed dataset with BiLSTM classification with the three other datasets. Shahzad et al. [37] applied different ML algorithms on the FallDroid dataset and reported higher accuracy of 91.70% for the MKL-SVM algorithm which is less than the performance achieved by BiLSTM model. This study has thus improved the prediction results as compared to other techniques applied earlier for the FallDroid dataset and provides new direction to the researchers. Ivascu et al. [54] reported results on the UniMiB SHAR dataset for the RF model with an accuracy of 96.21% which was slightly lower than our reported results (See Table 4.3). Moreover, the probability of predicting falls as compared to ADLs (Sensitivity) is higher for BiLSTM and RF model. Alawneh et al. [55] proposed results on UniMiB SHAR (AF-2) provided a dataset for the BiLSTM model and achieved an accuracy of 99.25% which is slightly comparable with our achieved an accuracy of 98.24% ±1.53. However, there is no significant difference in the accuracies achieved by both models. Ajerla et al. [56] employed the LSTM model along with the reliefF filter-based FS technique on the Mobiact dataset and achieved an accuracy of 99% which is again less than the newly dataset-based accuracy.

Table 4-4 Avg performance against multiple datasets

Research Reference	**Dataset**	**Classifier**	**Accuracy**	**Sensitivity**	**Specificity**
Current	FallMEdADL	BiLSTM	98.23%	96.94%	99.09%
	FallDroid	BiLSTM	95.81%	93.31%	97.66%
	Mobiact	BiLSTM	98.42%	98.50%	97.84%
	UniMib SHAR	BiLSTM	98.30%	99.48%	96.40%
[37]	FallDroid	MKL-SVM	91.70%	95.83%	88.01%
[39]	UniMib SHAR	BiLSTM	99%	N/A	N/A

However, the performance of the methodologies discussed in the previous research are not matchable both with respect to performance but also by considering the complexity and relevance of dataset. The datasets used in the past accounted for simple daily life activities and did not have tricky ADLs that were like falls and could become a potential source of more false negatives. In addition to that new research is based on elderly representative data of daily activities.

5 CONCLUSION and FUTURE WORK

5.1 Conclusion

The research presents implementation and comparison of deep learning and machine learning techniques. It also presents a new elderly ADL dataset that when used as input has given remarkable results. In addition to this new dataset two publicly available datasets and a proprietary dataset that contained complex ADLs resembling falls are used for evaluation of classifiers. The ability of a model or algorithm to generalize its learnings from the dataset to new, unseen data depends on the dataset's diversity and relevance to real-world scenarios. A biased or incomplete dataset can limit the applicability of research findings. We have tested and proved that new elderly data that is better representative for elderly fall detection results in higher prediction accuracies and less false alarms or false negatives. This new dataset will be published as well with the intent that researchers can use it for better analysis and prediction. Generally, also when comparing the acceleration of elderly movements, they are naturally different from young adult and all research currently relies on the young adult’s data with average age between 28 to 60. While it is understandable to not get fall data from elderly avoiding any mishaps or injuries, yet the ADL data as used in this research can be collected and compared with probable fall data. Application of BiLSTM and RF on the new dataset along with feature selection techniques on the new dataset FallMEdADL has attained the maximum average accuracy of 99.80%. Also, these architecture and techniques are tested against 3 other reference datasets and results are compared. This robust testing across multiple datasets has given a sound basis for generalization of the algorithms.

5.2 Future Work

To serve the purpose of elderly care and fall detection, the research can be enhanced by developing an android app for the smartphones that incorporate the proposed solution. Then a comparison of online and offline analysis using the newly proposed dataset is also suggested. In addition to making this an end-to-end working solution from fall detection to emergency alarm and response mechanism; the new dataset FALLMEdADL can also be enhanced. Currently, data is collected from single location and work is already in progress to gather elderly ADL data from other locations of Belly, Hand, and Bag. Future study can be based on developing and analyzing the classification algorithms that can effectively consider the additional information and come up with an even more generalized and trustworthy solution.

REFERENCES

1. World Health Organization. (2008). WHO global report on falls prevention in older age. World Health Organization. https://apps.who.int/iris/handle/10665/43811.
2. Zecevic AA et al. (2006). Defining a fall and reasons for falling: Comparisons among the views of seniors, health care providers, and the research literature. The Gerontologist, 46:367- 376.
3. “Falls.” https://www.who.int/news-room/fact-sheets/detail/falls. Accessed: 2022-09-05.
4. “Ageing and health.” https://www.who.int/news-room/factsheets/detail/ageing-and-health.
5. C. Qiu, G. Johansson, F. Zhu, M. Kivipelto, and B. Winblad, “Prevention of cognitive decline in old age—varying effects of interventions in different populations,” Annals of Translational Medicine, vol. 7, no. Suppl 3, 2019.
6. S. J. Barber, K. Hamel, C. Ketcham, K. Lui, and N. Taylor-Ketcham, “The effects of stereotype threat on older adults’ walking performance as a function of task difficulty and resource evaluations.,” Psychology and aging, vol. 35, no. 2, p. 250, 2020.
7. A. Cwirlej-Sozanska, A. Wisniowska-Szurlej, A. Wilmowska-Pietruszynska, and B. Sozanski, “Determinants of adl and iadl disability in older adults in southeastern poland,” BMC geriatrics, vol. 19, no. 1, pp. 1–13, 2019
8. V. Guimaraes, A. Pereira, E. Oliveira, A. Carvalho, and R. Peixoto, ˜ “Design and evaluation of an exergame for motor-cognitive training and fall prevention in older adults,” in Proceedings of the 4th EAI International Conference on Smart Objects and Technologies for Social Good, pp. 202–207, 2018.
9. WHO., WHO Global Report on Falls: Prevention in Older Age. World Health.Organization, 2007.
10. The UN Decade of healthy ageing: strengthening measurement for monitoring health and wellbeing of older people. Age Ageing. 2022 Jul 1;51(7): afac147. doi: 10.1093/ageing/afac147. PMID: 35776669; PMCID: PMC9249069.
11. Loveys, Kate Prina, Matthew Axford, Chloe Dome`nec, O` scar Weng, William Broadbent, Elizabeth Pujari, Sameer Jang, Hyobum Han, Zee Jotheeswaran, AT. (2022). Artificial intelligence for older people receiving long-term care: a systematic review of acceptability and effectiveness studies. The Lancet Healthy Longevity. 3. e286-e297. 10.1016/S2666-7568(22)00034-4.
12. Galvão, Y.M.; Ferreira, J.; Albuquerque, V.A.; Barros, P.; Fernandes, B.J. A multimodal approach 617 using deep learning for fall detection. Expert Syst. Appl. 2021, 168, 114226. [CrossRef]
13. Newaz, N.T.; Hanada, E. The Methods of Fall Detection: ALiterature Review. Sensors 2023, 23, 5212. https://doi.org/10.3390/s23115212.
14. Xiong, X., Min, W., Zheng, W. et al. S3D-CNN: skeleton-based 3D consecutive-low-pooling neural network for fall detection. Appl Intell (2020). https://doi-org.ezproxy.uws.edu.au/10.1007/s10489-020-01751-y.

15. Debard, G. *et al.* (2012). Camera-Based Fall Detection on Real World Data. In: Dellaert, F., Frahm, JM., Pollefeys, M., Leal-Taixé, L., Rosenhahn, B. (eds) Outdoor and Large-Scale Real-World Scene Analysis. Lecture Notes in Computer Science, vol 7474. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-34091-8_16.
16. C. Mosquera-Lopez *et al.*, "Automated Detection of Real-World Falls: Modeled From People With Multiple Sclerosis," in *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 6, pp. 1975-1984, June 2021, doi: 10.1109/JBHI.2020.3041035.
17. Ali, S.F.; Khan, R.; Mahmood, A.; Hassan, M.T.; Jeon, M. Using temporal covariance of motion and geometric features via boosting for human fall detection. Sensors 2018, 18, 1918. [CrossRef] [PubMed]
18. Waheed, M.; Afzal, H.; Mehmood, K. NT-FDS—A Noise Tolerant Fall Detection System Using Deep Learning on Wearable Devices. *Sensors* **2021**, *21*, 2006. https://doi.org/10.3390/s21062006
19. Kim, Joo & Saguna, Saguna & Ahlund, Christer & Mitra, Karan. (2021). Augmented Reality-Assisted Healthcare System for Caregivers in Smart Regions. 10.1109/ISC253183.2021.9562927.
20. C. A. Q. Bugarin, J. M. M. Lopez, S. G. M. Pineda, M. F. C. Sambrano and P. J. M. Loresco, "Machine Vision-Based Fall Detection System using MediaPipe Pose with IoT Monitoring and Alarm," *2022 IEEE 10th Region 10 Humanitarian Technology Conference (R10-HTC)*, Hyderabad, India, 2022, pp. 269-274, doi: 10.1109/R10-HTC54060.2022.9929527.
21. N. I. M. Amir, R. A. Dziyauddin, N. Mohamed, N. S. N. Ismail, N. S. A. Zulkifli and N. M. Din, "Real-time Threshold-Based Fall Detection System Using Wearable IoT," *2022 4th International Conference on Smart Sensors and Application (ICSSA)*, Kuala Lumpur, Malaysia, 2022, pp. 173-178, doi: 10.1109/ICSSA54161.2022.9870974.
22. J. P. Matos-Carvalho, S. D. Correia and S. Tomic, "Sensitivity Analysis of LSTM Networks for Fall Detection Wearable Sensors," *2023 6th Conference on Cloud and Internet of Things (CIoT)*, Lisbon, Portugal, 2023, pp. 112-118, doi: 10.1109/CIoT57267.2023.10084906.
23. J. L. Ren and Y. Peng, ” Research of Fall Detection and Fall Prevention Technologies: A Systematic Review,” in IEEE Access, vol. 7, pp. 77702- 77722, 2019, doi: 10.1109/ACCESS.2019.2922708.
24. Stampfler T, Elgendi M, Fletcher RR and Menon C (2022) Fall detection using accelerometer-based smartphones: Where do we go from here? Front. Public Health 10:996021. doi: 10.3389/fpubh.2022.996021.10.
25. S. Usmani, A. Saboor, M. Haris Khan MA, H. Park. Latest Research Trends in Fall Detection and Prevention Using Machine Learning: A Systematic Review. Sensors. 2021; 21(15):5134. https://doi.org/10.3390/s21155134.
26. A. Singh, S. U. Rehman, S. Yongchareon, and P. H. J. Chong, “Sensor technologies for fall detection systems: A review,” IEEE Sensors Journal, vol. 20, no. 13, pp. 6889‑6919, 2020.
27. J.-S. Lee and H.-H. Tseng, “Development of an enhanced threshold based fall detection system using smartphones with built-in accelerometers,” IEEE Sensors Journal, vol. 19, no. 18, pp. 8293‑8302, 2019.

28. A. Chelli and M. Pätzold, "A Machine Learning Approach for Fall Detection and Daily Living Activity Recognition," in *IEEE Access*, vol. 7, pp. 38670-38687, 2019, doi: 10.1109/ACCESS.2019.2906693.
29. Abdullah, Chowdhury Sayef & Kawser, Masud & Opu, Md & Faruk, Tasnuva & Islam, Md. Kafiul. (2020). IEEE WIECON ECE Presentation on "Human Fall Detection using Built-in Smartphone Accelerometer". 10.13140/RG.2.2.19713.89440.
30. F.-T. Wang, H.-L. Chan, M.-H. Hsu, C.-K. Lin, P.-K. Chao, and Y.-J. Chang, "Threshold-based fall detection using a hybrid of tri-axial accelerometer and gyroscope," Physiological measurement, vol. 39, no. 10, p. 105002, 2018.
31. P. Tsinganos and A. Skodras, "A smartphone-based fall detection system for the elderly," in Proceedings of the 10th International Symposium on Image and Signal Processing and Analysis, pp. 53–58, IEEE, 2017.
32. L. Pipanmaekaporn, P. Wichinawakul, and S. Kamolsantiroj, "Mining acceleration data for smartphone-based fall detection," in 2018 10th International Conference on Knowledge and Smart Technology (KST), pp. 74– 79, IEEE, 2018.
33. Asier Brull Mesanza, Ilaria D'ascanio, Asier Zubizarreta, Luca Palmerini, Lorenzo Chiari and Itziar Cabanes, "Machine Learning Based Fall Detector with a Sensorized Tip", IEEE Access, Volume 9, 2021, doi: 10.1109/ACCESS.2021.3132656.
34. Gunale, Kishnaprasad & Mukherji, Prachi. (2015). Fall detection using k-nearest neighbor classification for patient monitoring. 520-524. 10.1109/INFOP.2015.7489439.
35. Palmerini, L.; Klenk, J.; Becker, C.; Chiari, L. Accelerometer-Based Fall Detection Using Machine Learning: Training and Testing on Real-World Falls. *Sensors* **2020**, *20*, 6479. https://doi.org/10.3390/s20226479.
36. Diana Yacchirema1, Jara Suárez De Puga, Carlos Palau and Manuel Esteve, "Fall Detection System for Elderly People using IOT and Ensemble Machine Learning Algorithm", Springer,2019, https://doi.org/10.1007/S00779-018-01196-8.
37. A. Shahzad and K. Kim, "Falldroid: An automated smart-phone-based fall detection system using multiple kernel learning," IEEE Transactions on Industrial Informatics, vol. 15, no. 1, pp. 35–44, 2018.
38. Xiaodan Wu, Yumeng Zheng, Chao-Hsien Chu, Lingyu Cheng, Jungyoon Kim, Applying deep learning technology for automatic fall detection using mobile sensors, Biomedical Signal Processing and Control, Volume 72, Part B, 2022, 103355, ISSN 1746-8094, https://doi.org/10.1016/j.bspc.2021.103355.
39. C. S. Abdullah, M. Kawser, M. T. I. Opu, T. Faruk, and M. K. Islam, "Human fall detection using built-in smartphone accelerometer," in 2020 IEEE International Women in Engineering (WIE) Conference on Electrical and Computer Engineering (WIECON-ECE), pp. 372–375, IEEE, 2020.
40. Rubén Delgado-Escaño, Francisco M. Castro, Julián R. Cózar, Manuel J. Marín-Jiménez, Nicolás Guil, Eduardo Casilari, A cross-dataset deep learning-based classifier for people fall detection and identification, Computer Methods and Programs in Biomedicine, volume 184, 2020, 105265, ISSN 0169-2607, https://doi.org/10.1016/j.cmpb.2019.105265.
41. Wang, G.; Li, Q.; Wang, L.; Zhang, Y.; Liu, Z. Elderly Fall Detection with an Accelerometer Using Lightweight Neural Networks. Electronics 2019, 8, 1354. [CrossRef]

42. Espinosa, R.; Ponce, H.; Gutiérrez, S.; Martínez-Villaseñor, L.; Brieva, J.; Moya-Albor, E. A vision-based approach for fall detection using multiple cameras and convolutional neural networks: A case study using the UP-Fall detection dataset. Comput. Biol. Med. 2019, 115, 103520. [CrossRef] [PubMed]
43. Casilari, E.; Santoyo-Ramón, J.A.; Cano-García, J.M. Umafall: A multisensor dataset for the research on automatic fall detection. Procedia Comput. Sci. 2017, 110, 32–39. [CrossRef]
44. Martínez-Villaseñor, L.; Ponce, H.; Brieva, J.; Moya-Albor, E.; Núñez-Martínez, J.; Peñafort-Asturiano, C. UP-fall detection dataset: A multimodal approach. Sensors 2019, 19, 1988. [CrossRef] [PubMed]
45. Micucci, D.; Mobilio, M.; Napoletano, P. Unimib shar: A dataset for human activity recognition using acceleration data from smartphones. Appl. Sci. 2017, 7, 1101. [CrossRef]
46. Mastorakis, Georgios. "Human fall detection methodologies: from machine learning using acted data to fall modelling using myoskeletal simulation." 2018.
47. S. Abbate, M. Avvenuti, F. Bonatesta, G. Cola, P. Corsini, and A. Vecchio, "A smartphone-based fall detection system," Pervasive Mobile Comput., vol. 8, no. 6, pp. 883–899, 2012.
48. G. Vavoulas, C. Chatzaki, T. Malliotakis, M. Pediaditis, and M. Tsiknakis, "The mobiact dataset: Recognition of activities of daily living using smartphones," in International Conference on Information and Communication Technologies for Ageing Well and e-Health, vol. 2, pp. 143–151, SciTePress, 2016.
49. UniMiB SHAR: A Dataset for Human Activity Recognition Using Acceleration Data from Smartphones http://www.sal.disco.unimib.it/technologies/unimib-shar/
50. Micucci, D.; Mobilio, M.; Napoletano, P. UniMiB SHAR: A Dataset for Human Activity Recognition Using Acceleration Data from Smartphones. *Appl. Sci.* **2017**, *7*, 1101. https://doi.org/10.3390/app7101101.
51. S. Abbate, M. Avvenuti, F. Bonatesta, G. Cola, P. Corsini, and A. Vecchio, A smartphone-based fall detection system," Pervasive Mobile Comput., vol. 8, no. 6, pp. 883–899, 2012.
52. Giorgio (2023).Feature Selection Library (https://www.mathworks.com/matlabcentral/fileexchange/56937-feature-selection-library), MATLAB Central File Exchange. Retrieved July 4, 2023.
53. G. Roffo and S. Melzi, "Features selection via eigenvector centrality," Proceedings of new frontiers in mining complex patterns (NFMCP 2016) (Oct 2016), 2016.
54. T. Ivascu, K. Cincar, A. Dinis, and V. Negru, "Activities of daily living and falls recognition and classification from the wearable sensors data," in 2017 EHealth and Bioengineering Conference (EHB), pp. 627–630, IEEE, 2017.
55. L. Alawneh, B. Mohsen, M. Al-Zinati, A. Shatnawi, and M. Al-Ayyoub, "A comparison of unidirectional and bidirectional lstm networks for human activity recognition," in 2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), pp. 1–6, IEEE, 2020.
56. D. Ajerla, S. Mahfuz, and F. Zulkernine, "A real-time patient monitoring framework for fall detection," Wireless Communications and Mobile Computing, vol. 2019, 2019.