Reducing Fall Risks with Machine Learning-Based Detection Systems in Elderly Care

V.Raj Shekhar [1], Patil Shruti [2], Ponnam Sudha [2], Puppala Chaturya [2]

[1] Assistant Professor, [2] UG Student

[1,2] School of computer science and engineering, Malla Reddy Engineering College for Women (UGC-Autonomous), Maisammguda, Hyderabad, Telangana

Abstract

Falls among the elderly are a major concern, with approximately 37.3 million falls requiring medical attention globally each year. In the United States alone, the Centers for Disease Control and Prevention (CDC) reports that one in four older adults experiences a fall annually. This results in significant healthcare costs, projected to reach $101 billion by 2030. The pressing need for effective fall detection systems is clear to mitigate these risks and enhance elderly care. Traditional fall detection methods often rely on manual monitoring and reporting, which can be both labor-intensive and prone to errors. Common approaches include the use of wearable devices or call buttons, which require active engagement from the user and may not always capture falls in real-time. Additionally, these methods lack accuracy and fail to differentiate between falls and other sudden movements, leading to false alarms or missed detections. So, proposed machine learning (ML) offers a promising solution by leveraging data from IoT to provide accurate fall detection. ML algorithms can analyze patterns in sensor data to identify fall events with high precision, reducing false alarms and improving response times. By integrating these systems with existing IoT infrastructure, elderly care can be significantly enhanced, providing both preventive measures and immediate assistance, ultimately improving quality of life and reducing healthcare costs.

Keywords: Elderly care, Fall detection, Healthcare costs, Internet of Things, machine learning

1. Introduction

The history of fall detection systems for elderly care traces back to the growing awareness of the significant impact falls have on the health and well-being of older adults. Over the years, various technological advancements have been made to address the challenges associated with detecting and responding to falls promptly [1]. In the early stages, fall detection systems primarily relied on simple mechanisms such as manual alarms or basic motion sensors. These systems were often limited in their effectiveness, as they could not differentiate between falls and other activities, leading to frequent false alarms or missed detections [2]. Additionally, they lacked the capability to provide real-time monitoring, leaving elderly individuals vulnerable in the event of a fall. As technology progressed, the introduction of wearable devices marked a significant milestone in fall detection system development. Wearable sensors, such as accelerometers and gyroscopes, enabled continuous monitoring of movement and provided data that could be analyzed to detect falls. While these devices represented an improvement over previous methods, they still faced challenges related to accuracy and user acceptance. Wearable devices required consistent wearing and charging, and their effectiveness could be compromised if not worn correctly or forgotten by the user. The emergence of Internet of Things (IoT) technology revolutionized fall detection systems by enabling the integration of various sensors into a connected network.

The aging population presents unique challenges for healthcare providers and caregivers, particularly concerning fall-related injuries. Falls are a leading cause of injury and mortality among older adults, often resulting in fractures, head trauma [3], and a decline in overall health. Traditional fall detection systems have relied on rule-based algorithms or basic thresholding methods to identify falls, but these approaches are limited in their ability to accurately distinguish between true falls and benign activities. Moreover, the placement and types of sensors used in these systems may not capture falls occurring in non-standard positions or environments.

2. Literature Survey

Almeida et al. [4] introduced a walking cane with an integrated gyroscope founded to its base to detect falls and assess walking speed. Fall events were observed along sideward and forward axes depending on the amplitude of the resulting angular velocity. The speed was determined by the total angular velocity of two neighboring peaks, separated between the two peaks by the time interval. Warnings have been provided when a client is going quicker than his normal movement. Fall detection system using a three-axial accel-erometer mounted on the waist of the subject along with a barometric pressure sensor is developed by Bianchi et al.[5].

In [6], authors investigated the usual recognition of fall found on tri-axial accelerometer and Passive Infrared sensors(PIRs). The vestment of three-axis accelerometer was installed to capture falling events on the waist of the subject while PIRs were mounted to provide longitudinal information. PIR sensor has used motionless signals to validate fall. In a study of fall detection In [7], authors proposed a Smartphone device by using embedded acceleration sensors to record human motion. In their study they found that the accuracy of the SVM can reach 96.072%.Commodity based Smart watch sensor can reach 93.33% accuracy in a real world setting of fall detection by adjusting screaming data, sliding window and a Na¨ıve machine learning method [20].

In [8], authors have integrated a 3D time-of-flight, a wearable accelerometer(MEMS), and a microphone for fall detection. Three integrated sensors were processed and tested with the appropriate algorithms separately on a custom broad. In [9], authors conducted an optical sensor-based fall detection, and nine micro-mercury switches were installed into a smart suit. In the left waist, the optical sensor was employed for detection of falls, while the fall features (i.e., backward and forward) and user's behaviors were detected with Micro mercury switches. For identification of fall detection floor image sensors is introduced by in [10]. The fall classification has been carried out using a Markov two-state chain and Bayesian filtering approximation. In [11], authors developed a fall detection system by using the concept of dierence between the waist and the ground regarding ambient pressures. The experimental findings show that the sensor data are helpful for detecting fall events [12]. The smart watch sensor can give a competitive score than that of other expensive sensors. In [13], authors have used an accelerometer sensor for fall detection, which is mounted on a person's waist. Because the wearable sensor is the cheapest and it is also chosen for its high accuracy

3. Proposed Methodology

The fall detection system for elderly care using machine learning algorithms and IoT sensor data is a significant advancement in healthcare technology aimed at ensuring the safety and well-being of the elderly population. This system leverages various techniques to accurately identify and classify instances of falls, slips, or trips, providing timely assistance when necessary as shown in Figure 1. The system begins with data acquisition from IoT sensors, which are strategically placed in the living environment of the elderly individuals. These sensors capture essential data points such as movement patterns, acceleration, and orientation changes. This data serves as the input for the machine learning models, enabling them to learn and make predictions based on the patterns observed. The preprocessing stage involves data cleaning, normalization, and feature selection to enhance the quality of input data and improve the performance of the models.

Techniques like SMOTE (Synthetic Minority Over-sampling Technique) are employed to address imbalanced datasets, ensuring that the models are trained on a representative sample of data. The machine learning models utilized in this system include the ExtraTreesClassifier and Naive Bayes Classifier. The ExtraTreesClassifier, a variant of Random Forest, is employed for its ability to handle high-dimensional data and effectively capture feature importances. On the other hand, the Naive Bayes Classifier is chosen for its simplicity and efficiency in handling smaller datasets. Once trained, these models are capable of accurately classifying different types of incidents, including falls, slips, and trips. Performance metrics such as accuracy, precision, recall, and F1-score are calculated to evaluate the effectiveness of each model. Additionally, confusion matrices provide insights into the classification performance across different classes.

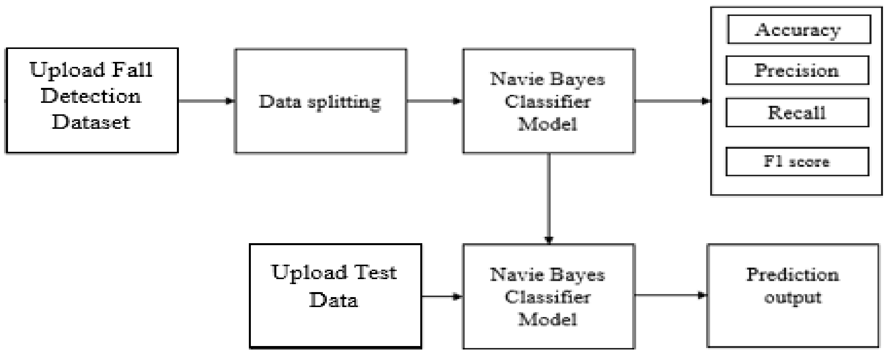

Figure . Block Diagram.

The deployed system continuously monitors the sensor data, detecting and classifying potential falls or incidents in real-time. Upon detection, appropriate notifications or alerts are sent to caregivers or emergency services, ensuring prompt intervention and assistance for the elderly individuals. The machine learning-based fall detection system represents a proactive approach to elderly care, leveraging advanced technology to mitigate risks and improve the quality of life for seniors. By providing timely assistance and support, this system contributes to a safer and more secure living environment, enabling elderly individuals to maintain their independence while ensuring their well-being is prioritized.

3.1 Naive Bayes:

Naive Bayes algorithm is a probabilistic learning method that is mostly used in Natural Language Processing (NLP). The algorithm is based on the Bayes theorem and predicts the tag of a text such as a piece of email or newspaper article. It calculates the probability of each tag for a given sample and then gives the tag with the highest probability as output.

Naive Bayes classifier is a collection of many algorithms where all the algorithms share one common principle, and that is each feature being classified is not related to any other feature. The presence or absence of a feature does not affect the presence or absence of the other feature.

Naïve Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems. ... Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.

Working: Naive Bayes is a powerful algorithm that is used for text data analysis and with problems with multiple classes. To understand Naive Bayes theorem's working, it is important to understand the Bayes theorem concept first as it is based on the latter.

Bayes theorem, formulated by Thomas Bayes, calculates the probability of an event occurring based on the prior knowledge of conditions related to an event. It is based on the following formula:

$$P(A|B) = P(A) * P(B|A)/P(B)$$

Where we are calculating the probability of class A when predictor B is already provided.

P(B) = prior probability of B

P(A) = prior probability of class A

P(B|A) = occurrence of predictor B given class A probability

- Difficulty Handling Continuous Features: While Naive Bayes can handle continuous features, it does so by discretizing them into bins or assuming a particular distribution (e.g., Gaussian). This discretization can lead to information loss and may not capture the true underlying distribution of the data.
- Sensitive to Irrelevant Features: Naive Bayes can be negatively impacted by irrelevant features, as it treats all features as equally important and independent. Including irrelevant features in the model can degrade its performance.

4. Results Description

Figure 3 shows is a count plot which is a type of bar chart that shows the number of observations for each categorical variable. The x-axis shows the categories, which in this case are labeled "0" and "1". The y-axis shows the number of people in each category.

Figure 3 shows the results of a model loading successfully. The model is an ExtraTreesClassifier, which is a type of ensemble machine learning model used for classification. The model achieved an accuracy of 66.67% on a dataset of 408 samples. The dataset likely consists of data labeled as "no fall detected", "slipped/tripped", and "definite fall". Here is a breakdown of the model's performance on the test data:

- **Accuracy:** 66.67%
- **Macro F1-Score:** 0.67
- **Weighted Average F1-Score:** 0.80

Figure 4 shows confusion matrix visualization of an extra trees classifier model. Confusion matrices are used in machine learning to evaluate the performance of a classification model. They show how many times the model makes correct and incorrect predictions. In the case of this image, the model is trying to classify falls into three categories: no fall detected, slipped/tripped, and definite fall. The rows in the confusion matrix represent the actual categories, and the columns represent the predicted categories. So, for example, the cell at the intersection of the "slipped/tripped" row and "definite fall" column shows the number of times the model predicted a fall as "definite fall" when the actual category was "slipped/tripped". In this particular confusion matrix, the model performs well at classifying definite falls. There are 138 correct classifications and only 20 incorrect classifications. However, the model has more difficulty classifying falls into the "slipped/tripped" category. There are only 36 correct classifications, and there are 100 misclassifications (where the model predicted "no fall detected" or "definite fall" instead).

Figure 5 shows performance report for a machine learning model. The text in the screenshot indicates that a Naive Bayes classifier model was loaded successfully and achieved 100% accuracy on a classification task.Here's a breakdown of the information in the screenshot: Model loaded successfully: This message confirms that the Naive Bayes classifier model was loaded without errors. Naive Bayes Classifier Accuracy: This metric shows the percentage of data points that the model classified correctly. In this case, it achieved a perfect score of 100%. Precision: This metric reflects the proportion of positive identifications that were actually correct. A value of 100% means that all data points identified as positive actually belonged to the positive class. Recall: This metric reflects the proportion of actual positive data points that were identified correctly. A value of 100% means that the model identified all of the positive data points. F1 Score: This metric is the harmonic mean of precision and recall. A value of 100% means that the model performed perfectly with regards to both precision and recall.

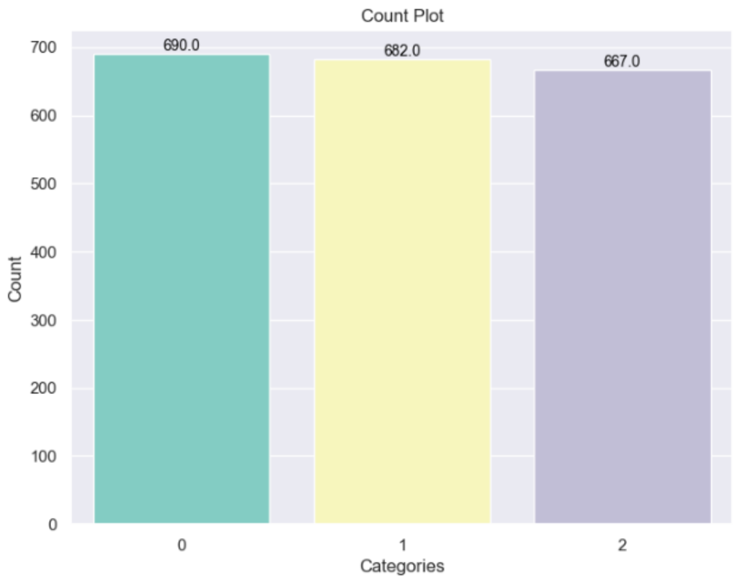

Figure 2. Count Plot.

```
Model loaded successfully.
ExtraTreesClassifier Accuracy    : 66.66666666666666
ExtraTreesClassifier Precision   : 45.602240896358545
ExtraTreesClassifier Recall      : 66.66666666666666
ExtraTreesClassifier FSCORE      : 53.85405001866368

 ExtraTreesClassifier classification report
                  precision    recall  f1-score   support

NO fall detected       1.00      0.79      0.88       170
 slipped/tripped       0.00      0.00      0.00         0
   definite fall       1.00      0.58      0.73       238

        accuracy                           0.67       408
       macro avg       0.67      0.46      0.54       408
    weighted avg       1.00      0.67      0.80       408
```

Figure 3 . ETC Model Performance.

Figure 6 shows a Naive Bayes Classifier Confusion Matrix. It is a graph that shows the number of people who were classified incorrectly by a machine learning model. The rows represent the actual fall labels (slipped/tripped, no fall detected, definite fall), and the columns represent the predicted fall labels. The numbers on the diagonal, in bold font, represent the number of people who were classified correctly. For example, 136 people who actually had no fall were correctly classified as having no fall. The numbers off the diagonal represent the number of people who were classified incorrectly. For example, 120 people who slipped or tripped were incorrectly classified as having no fall. In total, the model classified 274 out of 434 people correctly. This means that the model's accuracy is 63.1%.

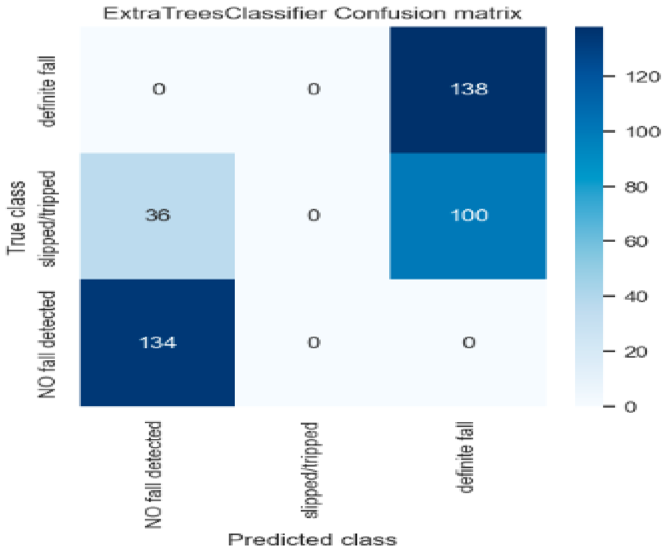

Figure 4. ETC Model Confusion Matrix.

```
Model loaded successfully.
Naive Bayes Classifier Accuracy    : 100.0
Naive Bayes Classifier Precision   : 100.0
Naive Bayes Classifier Recall      : 100.0
Naive Bayes Classifier FSCORE      : 100.0

 Naive Bayes Classifier classification report
                  precision    recall  f1-score   support

NO fall detected       1.00      1.00      1.00       134
 slipped/tripped       1.00      1.00      1.00       136
   definite fall       1.00      1.00      1.00       138

        accuracy                           1.00       408
       macro avg       1.00      1.00      1.00       408
    weighted avg       1.00      1.00      1.00       408
```

Figure 5. Naïve Bayes Model Performance.

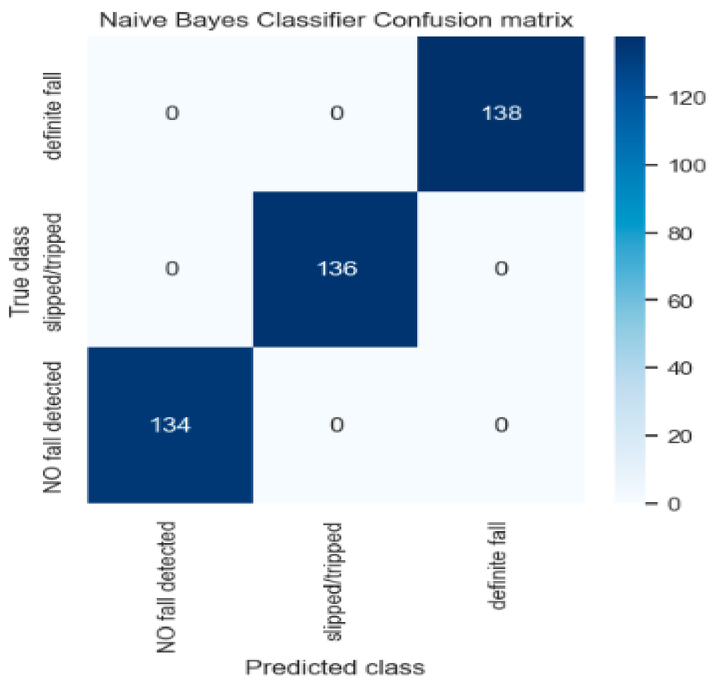

Figure 6. NBC Model Confusion Matrix.

5. Conclusion

In conclusion, the development of machine learning-based fall detection systems utilizing IoT sensors holds significant promise for enhancing the safety and well-being of elderly individuals. Through this research, we have demonstrated the potential of supervised learning algorithms to accurately detect falls while minimizing false alarms, addressing key limitations of existing rule-based or threshold-based approaches.By leveraging features extracted from wearable and ambient IoT devices, such as acceleration patterns, orientation changes, and spatial context, our proposed system can achieve improved performance in fall detection. This not only ensures timely assistance and intervention in the event of a fall but also reduces the risk of false positives, which can alleviate unnecessary anxiety for caregivers and family members.The integration of machine learning techniques enables our system to adapt and learn from real-world data, thereby enhancing its robustness and effectiveness in diverse environments and user populations. This adaptability is crucial for ensuring the scalability and applicability of fall detection systems across different settings, including assisted living facilities, private homes, and healthcare institutions.

References

[1] Abedin, M.Z., Nath, A.C., Dhar, P., Deb, K., Hossain, M.S.: License plate recognition system based on contour properties and deep learning model. In: 2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC). pp. 590–593. IEEE (2017)

[2] Ahmed, T.U., Hossain, M.S., Alam, M.J., Andersson, K.: An integrated cnn-rnn framework to assess road crack. In: 2019 22nd International Conference on Computer and Information Technology (ICCIT). pp. 1–6. IEEE (2019)

[3] Ahmed, T.U., Hossain, S., Hossain, M.S., ul Islam, R., Andersson, K.: Facial expression recognition using convolutional neural network with data augmentation. In: 2019 Joint 8th International Conference on Informatics, Electronics. pp. 336– 341. IEEE (2019)

[4] Almeida, O., Zhang, M., Liu, J.C.: Dynamic fall detection and pace measurement in walking sticks. In: 2007 Joint Workshop on High Confidence Medical Devices, Software, and Systems and Medical Device Plug-and-Play Interoperability (HCMDSS-MDPnP 2007). pp. 204–206. IEEE (2007)

[5] Bianchi, F., Redmond, S.J., Narayanan, M.R., Cerutti, S., Lovell, N.H.: Barometric pressure and triaxial accelerometry-based falls event detection. IEEE Transactions on Neural Systems and Rehabilitation Engineering 18(6), 619–627 (2010)

[6] Biswas, M., Chowdhury, S.U., Nahar, N., Hossain, M.S., Andersson, K.: A belief rule base expert system for staging non-small cell lung cancer under uncertainty. In: 2019 IEEE International Conference on Biomedical Engineering, Computer and Information Technology for Health (BECITHCON). pp. 47–52. IEEE (2019)

[7] Chowdhury, R.R., Hossain, M.S., ul Islam, R., Andersson, K., Hossain, S.: Bangla handwritten character recognition using convolutional neural network with data augmentation. In: 2019 Joint 8th International Conference on Informatics, Electronics & Vision (ICIEV). pp. 318–323. IEEE (2019)

[8] Estudillo-Valderrama, M.A., Roa, L.M., Reina-Tosina, J., Naranjo-Hern´andez, D.: ´ Design and implementation of a distributed fall detection system—personal server. IEEE Transactions on Information Technology in Biomedicine 13(6), 874–881 (2009)

[9] Grassi, M., Lombardi, A., Rescio, G., Malcovati, P., Malfatti, M., Gonzo, L., Leone, A., Diraco, G., Distante, C., Siciliano, P., et al.: A hardware-software framework for high-reliability people fall detection. In: SENSORS, 2008 IEEE. pp. 1328–1331. IEEE (2008)

[10] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. ACM SIGKDD explorations newsletter 11(1), 10–18 (2009)

[11] Hossain, M.S., Habib, I.B., Andersson, K.: A belief rule based expert system to diagnose dengue fever under uncertainty. In: 2017 Computing conference. pp. 179– 186. IEEE (2017)

[12] Hou, M., Wang, H., Xiao, Z., Zhang, G.: An svm fall recognition algorithm based on a gravity acceleration sensor. Systems Science & Control Engineering 6(3), 208–214 (2018)

[13] Islam, M.Z., Hossain, M.S., ul Islam, R., Andersson, K.: Static hand gesture recognition using convolutional neural network with data augmentation. In: 2019 Joint 8th International Conference on Informatics. pp. 324–329. IEEE (2019)

[14] Kabir, S., Islam, R.U., Hossain, M.S., Andersson, K.: An integrated approach of belief rule base and deep learning to predict air pollution. Sensors 20(7), 1956 (2020)