A Study On Machine Learning Algorithms For Fall Detection And Movement Classification

Y. Choi, A. S. Ralhan, and S. Ko
Department of Electrical and Computer Engineering
College of Engineering
University of Saskatchewan

***Abstract*—Falls among the elderly is an important health issue. Fall detection and movement tracking are therefore instrumental in addressing this issue. This paper responds to the challenge of classifying different movements as a part of a system designed to fulfill the need for a wearable device to collect data for fall and near-fall analysis.**

Four different fall trajectories (forward, backward, left and right), three normal activities (standing, walking and lying down) and near-fall situations are identified and detected. Different machine learning algorithms are compared and the best one is used for real time classification. The comparison is made using Waikato Environment for Knowledge Analysis (WEKA), one of the most popular machine learning software. The system also has the ability to adapt to the different gait characteristics of each individual. A feature selection algorithm is also introduced to reduce the number of features required for the classification problem.

I. Introduction

A. Fall Detection

Falls among the elderly is a major health concern accounting for approximately half of all injury-related admissions in hospitals in the over 65 age group [6]. Falls are responsible not only for disabling fractures and other physical injuries, but for also causing psychological trauma which can reduce the independence and confidence of older patients [22]. Detection of falls is a particular problem. Although the concept of a fall is ingrained in the common sense but it is difficult to describe the precise nature and cause of a fall in many cases.

Complex internal neural and muscular mechanisms control the body's postural stability [14]. Older adults who have experienced a fall or frequent near-falls are typically assessed through subjective evaluation or questionnaires to determine the exact nature of fall and the circumstances under which the falls occurred [2]. These documentation methods have advantages and disadvantages in terms of accuracy, costs and time commitments. However, the extent and accuracy of recall of the fall activity is always a consistent limiting factor. Therefore, robust and reliable methods for detecting and analyzing the circumstances around a fall are required. This has prompted a great deal of research in fall detection and movement classification. To detect falls, several different approaches have been used. Sixsmith [23] used an array of infrared detectors for fall monitoring. Other methods such as using video cameras, door alerts, pressure mats have been in place for some time now and are discussed by Miskelly [15]. Noury [19] used infrared position sensors and magnetic switches to monitor the behavior. Use of accelerometers and gyroscopes has gained widespread popularity in detecting ambulatory motion and machine learning algorithms are the most intuitive way of detecting and classifying different types of falls [1], [17]. Earliest works involving use of accelerometers for fall detection were published by Lord and Colvin [12] in 1991. William [24] used a belt device and detected the shock of impacting the ground and is able to determine if the patient is lying down by using a mercury tilt switch. A sensor attached to the armpit detects the change in velocity and when it exceeds a threshold, the sequence from upright position to lying posture and the absence of movement after the fall [18]. Most of these devices have the primary objective of distinguishing normal movement from a fall event and often have a high rate of false alarms, which is one of the reasons for keeping these devices from gaining more use in daily life [17]. The data from the waist mount accelerometer is passed through a Gaussian filter to remove noise and then a 3D body motion model is used to map the data to motion types [13]. Hwang [7] used a combination of tilt meters, gyroscopes and accelerometers to detect falls with limited success.

Current literature supports the need for more accurate fall and near fall detection, particularly with the increase in older adults as the baby boomers age. In this work, machine learning techniques are employed, so that not only can the falls be detected, but they can be further classified into subcategories depending on the trajectory. The normal movements are also further classified into subcategories. The availability of information about different movement types, the direction of fall and pre-fall position helps in pre-fall and post-fall analysis. Accurate classification of the normal movement coupled with the direction of fall, gives the complete sequence of events that led to a fall.

B. Machine Learning

Machine learning is the domain of science that explores the ability of machines to understand data. It involves developing algorithms that would enable computers to learn complex patterns and make intelligent decisions based on these algorithms. Learning itself covers a broad range of processes and is thus hard to define. With respect to machines, it can be said that the machine learns whenever it changes its structure, program or data in such manner that its future performance improves

978-1-4244-9224-4/11/$26.00 ©2011 IEEE

[16]. Machine learning can broadly be categorized into two fields: Unsupervised Learning and Supervised Learning. In unsupervised learning, the machine tries to identify groups of similar data from a larger dataset. In other words, it tries to form clusters of data based on some criteria such as cost functions. The machine has no prior knowledge of the classes the data belongs to, it only tries to identify natural clusters or groups of data. Supervised learning on the other hand learns from the test set which contains classified data and predicts the classes of unseen data. The present work is concerned with supervised learning. The test data classified into different movement types is inputted to the machine and from this labeled data the machine will learn the pattern and accordingly predict the movement types upon receiving new unseen data.

1) Feature Selection: Usually a data set can have hundreds and thousands of features. This huge dimensionality causes many problems in the process of machine learning. This situation is often referred to as "Curse of Dimensionality" [3]. Feature selection, variable selection or attribute selection is the technique of selecting a subset of relevant features that would result in robust models. In theory, more features should result in a better distinguishing capability by the classifier, however, this is not the case, as redundant features not only slows down the process, but also results in over-fitting [25]. In other words, feature selection is the technique to remove the irrelevant and redundant data for efficient machine learning process. Selecting a subset of relevant features improves the performance in the following ways [4]:

- Alleviating the Curse of Dimensionality
- Generalizing the model
- Reducing the data required for classification, which is very useful for real time applications
- Faster and cost effective predictors

Feature selection has been an active and fruitful field of research and development for decades in statistical pattern recognition, machine learning, data mining and statistics. Yu and Huan [25] described these various applications and note that when there are hundreds and thousands of features present, not all of them add to the information of the target. Both theoretical analysis and empirical evidence show that along with irrelevant features, redundant features also affect the speed and accuracy of learning algorithms and thus should be eliminated as well [9]. Feature selection algorithms can be broadly classified into two types: the wrapper model and the filter model. The wrapper model uses the predictive accuracy of a predetermined learning algorithm to determine the goodness of the selected subsets. These methods are computationally expensive for data with a large number of features [10]. The filter model separates feature selection from classifier learning and selects feature subsets that are independent of any learning algorithm. It uses general characteristics such as distance measures, information gain, correlation coefficients and consistency to determine the feature subset.

The optimum feature selection algorithm introduced in this paper fits under the filter model category. It ranks the features based on distance measurements and then removes redundant features using correlation coefficients. The performance of the algorithm is tested with Naive Bayesian classifier, which is extremely sensitive to the type of features used. If the features are highly correlated, they can get high weightage and reduce the accuracy of classification [21]. Many different approaches have been explored to improve the performance of Naive Bayesian classifier. The four main approaches are feature selection, structure extension, local learning, and data expansion [8].

Most of the algorithms discussed by Hu and Yuan [25] try to determine a feature set which results in an increase in the accuracy of classification. The result is a relaxed approach to the rejection of irrelevant or redundant features. Optimum feature selection algorithm avoids this restraint and uses a more aggressive approach towards rejecting features, which in some cases might result in a slight decrease in classification accuracies.

II. Method

The entire problem of movement classification can be broken down into the following sequential steps.

- Accurate data collection to generate a proper training file, thus providing an accurate representation of various kinds of movement.
- Selection of an appropriate classification algorithm which is able to distinguish these different types of movements accurately.
- Feature selection process to explore the possibility of reducing the number of features required for the classification process.
- Generating the final model using the selected features and the classification algorithm.
- Implementing the model into the real time application, with the ability to continuously update the model to adapt to individual users.

In this section, the process and methodology of each of these steps will be discussed in detail. Data collection methods, the algorithms to be compared and the feature selection algorithm are discussed.

A. Data Collection

The device used for data collection in the FANFARE (Falls And Near Falls Assessment Research and Evaluation) project is a Jennic board of JN5139 series (Figure 1).

The belt worn device is the end device that communicates with the coordinator. The device has a three-axis accelerometer and a two-axis gyroscope. The data consists of 3582 data items distributed in 597 rows and 6 columns. The data was collected for seven types of movements. Out of the seven, three are normal activities: Walking, Standing, and Lying down. The remaining movements include 4 different fall trajectories; Forward Fall, Backward Fall, Left Fall, and Right Fall – of note, patients don't actually fall exactly in these trajectories. The device was worn at chest level for all data collection. Figure 2 shows the interaction between the end device and the coordinator, and the classification steps.

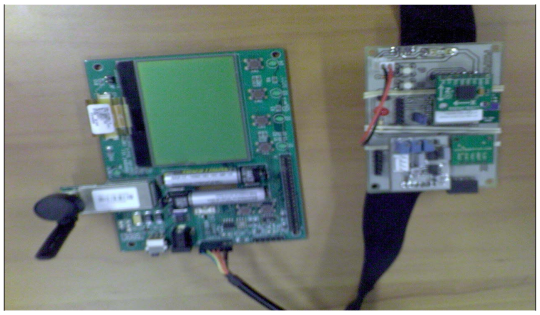

Fig. 1. Fall Detection Device.

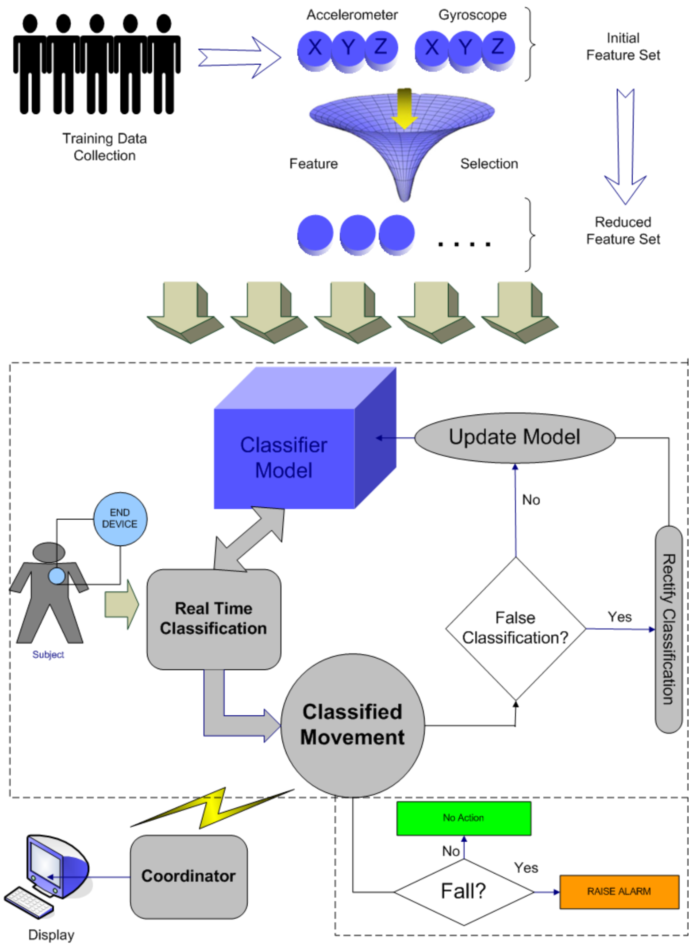

Fig. 2. Block Diagram of Experimental Setup

The training data is collected by imitating the normal movements and fall activities a number of times until we obtained sufficient number of samples for reliable machine learning process. The output of training data collection is a number of samples having a feature set corresponding to the axis of accelerometer and gyroscope. Depending on the type and number of sensors used, the number of features can range anywhere between 5 to 15. This training data set is labeled for different movement types and then is subjected to a feature selection process, which reduces the feature set to lesser number of features based on the relevance and information content of each feature. This final reduced feature set is then used to generate the classification model. A classification model is a set of parameters which is the output of a machine learning algorithm. These parameters can then be used to classify new unseen data. In probabilistic methods for example, the parameters can be means and standard deviations. From these parameters, the likelihood of a movement type can be calculated for every new unseen sample of data. Quality of the training data determines the accuracy of the classification model. Once the classification model is generated, it is inputted into the end device. Now, the device can determine the movements of the subject wearing the device. Every new sample of data is classified as one of the movement types based on the parameters of the classification model. To make the device adaptable to an individual's posture and gait, the data collected from sensors is fed back to the classification model and used to update the features. In this manner, the device adapts to a particular user over a period of time. In the inevitable cases of false alarms, the classification should be rectified and then fed back to the classification model, ensuring that the parameters are updated correctly. All the classification results are wirelessly transmitted to the coordinator, which is connected to a computer displaying the movements in real time. In case the result of the classification is a $Fall$, then an alarm is raised.

B. Daily Activities

This section describes the manner in which data is collected for different activities. The movements are imitated to be as close as possible to the normal routine movements.

1) Lying Down: To collect data for the Lying Down position, the subject is lying down flat on the back. Small controlled movements like slowly turning to left or right side or turning over completely and lying down flat on stomach are also included in this movement class. While shifting positions, no sudden movement is made and the transition in smooth.

2) Standing: For this type, the subject is standing upright at a fixed position. Slow controlled movements like leaning forward, backward, left or right is also considered standing stance. Minor shifting of feet to change the direction the subject is facing is also included in the standing position.

3) Walking: Walking involves normal paced walking or even taking few steps from the standing position. The data collection process takes into consideration any direction changes or even walking backwards. Slight leaning in all the four directions is also allowed while walking, just as in standing position.

4) Fall Forward/Backward/Left/Right: A fall in any direction involves sudden changes in the acceleration values. For a fall forward, the subsect goes from standing position to a lying down position while going in a forward direction in an uncontrolled manner in a fraction of second.

5) Near Fall: A near fall can be described as a state in which the subject is in a precarious position and is on the verge of falling, but recovers and does not actually fall. Every fall is preceded by a near fall situation. So, the samples just before the actual fall, are classified as near fall situations. In these experiments, the near fall situation is not further classified as left/right or forward/backward. Too much leaning in any direction results in near fall risk and is labeled as such. The abnormal leaning needs to be outside what was measured as normal sway for that patient. Also, acceleration of change needs to be included.

C. Classification algorithms

The problem of selecting the best classification algorithm for the movement classification was addressed in [2]. Five different classification algorithms were compared for their classification accuracies and time taken to build the model. The data was collected using the same device discussed earlier in this paper. The experimental results showed that Naive Bayesian Algorithm gives the best performance.

D. Feature Selection

This section describes the optimum feature selection algorithm. The algorithm is based on the concept of class discrimination ability of features. This means how well a feature can distinguish between different classes [3]. The nominal values of the data sets were replaced by numbers. In the first step, the entire data set is normalized to make the values in the range 0 to 1. This is important as the ranking step involves Euclidean distances and standard deviations. To ensure consistency, all the data values should lie in the same range. The rank of a particular feature is calculated using the Fisher Distance Ranking System [3]:

$$\text{Rank} = \frac{\sum \text{distances between means of each class}}{\sum \text{standard deviation within each class}} \qquad (1)$$

For a particular feature, if the distance between the means values for different classes is large and the deviations within the same class are low, then that feature can better distinguish between the classes. The higher the value of Rank, the better the discrimination ability of that feature, or more valuable the feature. Following are the major steps of the algorithm:

- Normalize the data sets.
- Rank the features and select top features based on parameter θ.
- Identify and separate features with a correlation coefficient more than γ.
- Of the separated feature, keep the feature with a maximum correlation with the target and remove the remaining.
- Use the final feature set for classification.
- Compare the accuracies of the original and the reduced data set.

Once the features have been ranked, the remaining steps of the algorithm are governed by two parameters θ and γ. The prior decides what fraction of the ranked features will be selected and the latter is the threshold of correlation coefficient between any two features, beyond which the two features can be considered to possess redundant information.

The top $\frac{1}{\theta}$ features are used to form the reduced feature set. In case there are missing values in the dataset, they are not used in any of the calculations and are ignored. Then the correlation matrix of the reduced data set is analyzed. The features with correlation coefficient greater than γ are separated. Out of these features, only the feature having maximum correlation with the target in the original unreduced data set is kept and the rest are discarded. If the number of features in a dataset is N, then the following relation is used for the parameters θ and γ.

$$\begin{cases} N < 15 & : \quad (\theta, \gamma) = (2, 0.7) \\ 15 \leq N < 30 & : \quad (\theta, \gamma) = (3, 0.6) \\ N > 30 & : \quad (\theta, \gamma) = (7, 0.7) \end{cases}$$

The algorithm was tested on 18 University of California, Irvine (UCI) data sets and was able to reduce the number of features on an average by 84.5 % with an average increase in classification accuracy of 0.7 % [20]. Two other feature selection algorithms were compared with this algorithm. A Cfs Subset Evaluation algorithm, which evaluates the worth of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy between them [5], reduced the number of features on an average by 70.8 % with an average increase in classification accuracy of 1.8 %. A Consistency Subset Evaluation algorithm, which evaluates the worth of a subset of attributes by the level of consistency in the class values when the training instances are projected onto the subset of attributes [11], reduced the number of features on an average by 63.4 % with an average increase in classification accuracy of 2.2 %.

III. Results

A. Single Node analysis

The following results are for the data collected from a single node worn at the chest. Fig 3 shows a snapshot of the collected data. The first column is the time stamp, followed by acceleration values in X,Y and Z axis respectively. The last two columns are the Gyroscope values for the X and Y axis respectively.

Section II-C shows that the Naive Bayesian classifier is most suitable for the movement classification. In this section, the classifier is used to classify new data which also includes the Near Fall situation. The total number of classes in the data set is 8 and the number of instances are 670. The classification accuracy is 99.4 %, with 3 samples belonging to Near Fall class being misclassified. Table I shows the confusion matrix for the classification result. As can be seen, there are only 4 misclassified instances. One Forward Fall is misclassified as Near Fall. There are two instances of false alarm and one instance of undetected fall.

Now, the performance of the optimum feature selection algorithm is investigated on the data. The data from single node consists of 5 features; Acceleration X axis, Acceleration Y axis, Acceleration Z axis, Gyroscope X axis, and Gyroscope Y axis.

	A	B	C	D	E	F
1	4:14:11 PN	0.044	-0.056	1.014	-0.366	0
2	4:14:11 PN	0.044	-0.056	1.014	-1.832	0
3	4:14:11 PN	0.044	-0.056	1.014	-0.366	-0.733
4	4:14:11 PN	0.044	-0.056	1.014	-0.733	-0.733
5	4:14:11 PN	0.044	-0.056	1.017	0	0
6	4:14:11 PN	0.044	-0.053	1.017	0	0
7	4:14:12 PN	0.044	-0.053	1.014	-0.733	0
8	4:14:12 PN	0.044	-0.056	1.014	0	-0.366
9	4:14:12 PN	0.044	-0.056	1.014	-0.366	0
10	4:14:12 PN	0.044	-0.056	1.014	-0.733	0
11	4:14:12 PN	0.044	-0.056	1.014	0	-0.733
12	4:14:12 PN	0.044	-0.056	1.014	-1.099	0
13	4:14:12 PN	0.044	-0.056	1.014	-0.733	-1.099
14	4:14:12 PN	0.044	-0.056	1.017	-1.099	0
15	4:14:12 PN	0.044	-0.053	1.017	-0.733	0

Fig. 3. Sample Data File Collected From Accelerometer and Gyroscope

TABLE I
CONFUSION MATRIX FOR NAIVE BAYESIAN CLASSIFIER (INCLUDING NEAR FALL)

		Classified event							
		A	B	C	D	E	F	G	H
Actual event	A	85	0	0	0	0	0	0	1
	B	0	82	0	0	0	0	0	0
	C	0	0	90	0	0	0	0	0
	D	0	0	0	87	0	0	0	0
	E	0	0	0	0	80	0	0	0
	F	0	0	0	0	0	87	0	0
	G	0	0	0	0	0	0	85	0
	H	1	0	0	1	0	0	1	70

A:Forward Fall B:Lying C:Standing
D:Right Fall E:Walking F:Backward Fall
G:Left Fall H:Near Fall

The following are the features selected by the optimum feature selection algorithm; Acceleration X axis, Acceleration Y axis, and Acceleration Z axis.

TABLE II
PERFORMANCE COMPARISON OF FEATURE SELECTION ALGORITHMS ON DATA FROM SINGLE NODE

	Features used	Accuracy (%)
NBC with Optimum Feature Selection	3	98.8
NBC with Cfs Subset Eval	5	99.4
NBC with Consistency Subset Eval	4	99.4

From Table II it can be seen that using only these three features, the accuracy of classification is 98.8 %, which is 0.6 % lower than the accuracy achieved using all five features. Table III shows the confusion matrix for the results.

In real time application, the classification is completed in the end device and then transmitted to the coordinator. More features result in more loops in the classification program, resulting in fewer samples processed. With 5 features, the sampling rate is at 10 samples/sec and with 3 features, it increases to 18 samples/sec. This increase in sampling rate helps in monitoring even subtle changes in movement. Therefore, this slight drop in classification accuracy is justified if the number of features are being reduced. This effect is even more considerable when multiple nodes are used.

TABLE III
CONFUSION MATRIX USING FEATURES SELECTED BY OPTIMUM FEATURE SELECTION ALGORITHM

		Classified event							
		A	B	C	D	E	F	G	H
Actual event	A	85	0	0	0	0	0	0	1
	B	0	82	0	0	0	0	0	0
	C	0	0	90	0	0	0	0	0
	D	0	0	0	87	0	0	0	0
	E	0	0	0	0	80	0	0	0
	F	0	0	0	0	0	87	0	0
	G	0	0	0	0	0	0	84	1
	H	1	0	0	5	0	0	0	67

A:Forward Fall B:Lying C:Standing
D:Right Fall E:Walking F:Backward Fall
G:Left Fall H:Near Fall

As can be seen from the confusion matrix, the increase in misclassification is only due to 3 additional features belonging to a near fall being classified as some sort of fall activity. Table II compares the performance of the three algorithms.

As is evident from the discussions in this section, both Cfs Subset Evaluation and Consistency Subset Evaluation algorithms are able to maintain the accuracy, but could not reduce the number of features considerably. Optimum feature selection algorithm on the other hand reduces the number of features considerably while maintaining or increasing acceptable accuracy.

B. Double Node Analysis

In this section, the experimental results using two nodes are described. The motivation behind using multiple nodes is the greater accuracy achieved in classifying different movements and also more number of movement types that can be detected. Below are the details of the setup and the data collection process:

- The second device consists of a three-axis accelerometer and a one-axis gyroscope.
- The first device is still worn around chest and the second device is worn on the thigh just above the knee.
- Both devices are on the left side of the body.
- The methodology for data collection is the same as used with the single node setup.
- An additional movement type can be classified by using the two nodes. It is the $Sitting$ position.
- The data for the sitting position is collected by having the subject sit on a chair and allowing slight shifting of feet and sway of upper body to account for normal sitting postures including upright, leaning back or leaning forward.
- The resultant data file has 9 features and 9 classes as shown in Figure 4.

Table IV shows the confusion matrix for the classification results.

	A	B	C	D	E	F	G	H	I	J	K
1	AccelX1	AccelY1	AccelZ1	GyroX1	GyroY1	AccelX2	AccelY2	AccelZ2	GyroX2	Class	
2	0.009	0.967	-0.091	-0.733	-7.326	-0.04	-1	-0.04	0.04	Standing	
3	0.003	0.958	-0.062	4.029	-14.652	-0.04	-1	-0.03	0.04	Standing	
4	0	0.967	-0.053	6.96	-21.978	-0.04	-0.99	-0.04	0.04	Standing	
5	-0.009	0.964	-0.053	12.821	-24.542	-0.04	-1.01	-0.03	0.06	Standing	
6	-0.018	0.967	-0.047	16.85	-23.443	-0.03	-1	-0.04	0.04	Standing	
7	0.358	0.879	-0.446	74.725	-15.751	-0.08	-0.98	-0.05	-0.28	Walking	
8	0.446	0.932	-0.443	87.912	4.029	-0.02	-1.01	0	-0.33	Walking	
9	0.372	0.915	-0.375	76.19	17.949	0.03	-0.99	-0.12	-0.24	Walking	
10	0.103	0.469	0.871	2.564	-9.89	0.37	-0.21	1.02	-0.47	Lying	
11	0.041	0.507	0.812	5.861	-5.495	0.4	-0.23	1.01	-0.5	Lying	
12	0.041	0.528	0.75	2.564	18.315	0.57	-0.2	0.91	-0.18	Lying	
13	-0.062	0.618	0.759	-1.832	-9.89	0.45	-0.18	0.97	0.08	Lying	
14	-0.029	0.721	-0.695	2.564	-1.465	-0.04	-0.05	1.13	-0.17	Sitting	
15	-0.026	0.724	-0.689	3.297	-1.832	-0.04	-0.05	1.13	-0.17	Sitting	
16	-0.026	0.727	-0.683	2.564	-0.733	-0.04	-0.05	1.13	-0.17	Sitting	
17	-0.023	0.733	-0.677	0.733	-2.198	-0.03	-0.05	1.13	-0.18	Sitting	
18	-0.029	0.736	-0.683	5.861	-1.465	-0.03	-0.05	1.13	-0.18	Sitting	
19	0.449	0.667	1.585	-28.938	-166.3	-0.12	-1.01	0.03	0	Near Fall	
20	-0.403	0.531	-1.555	-130.403	398.535	-0.12	-0.99	0.04	0	Near Fall	
21	-4.822	-0.865	1.644	-7.692	109.158	0.57	-0.27	-0.34	0.03	left	
22	-1.489	-0.607	0.205	-25.275	-158.974	1.68	-0.54	0.01	0.16	left	
23	2.263	-0.361	0.95	169.597	115.385	-2.77	0.32	0.62	0.04	right	
24	1.155	1.621	1.378	26.007	77.289	-1.12	-0.47	0.57	-0.02	right	
25	-0.885	0.411	3.473	-20.147	541.392	0.27	-0.64	0.66	-0.26	backward	
26	-0.258	0.211	1.26	-92.308	531.392	0.84	-1.04	0.77	0.08	backward	
27	-0.352	0.513	-4.277	-34.799	-942.857	-0.94	0.46	-1.21	-0.1	forward	

Fig. 4. Sample Data File From Two Nodes

TABLE IV
CONFUSION MATRIX FOR NAIVE BAYESIAN CLASSIFIER (DOUBLE NODE)

		Classified event								
		A	B	C	D	E	F	G	H	I
Actual event	A	23	0	0	0	0	0	0	0	0
	B	0	51	0	0	0	0	0	0	0
	C	0	0	101	0	0	0	0	0	0
	D	0	0	0	22	0	0	0	0	0
	E	0	0	0	0	100	0	0	0	0
	F	0	0	0	0	0	23	0	0	0
	G	0	0	0	0	0	0	20	0	0
	H	0	0	0	0	0	0	0	17	0
	I	0	0	0	1	0	0	0	0	100

A:Forward Fall B:Lying C:Standing
D:Right Fall E:Walking F:Backward Fall
G:Left Fall H:Near Fall I:Sitting

The accuracy of classification is 99.8 %, with one sample of a sitting position misclassified as right fall. This can be the result of an outlier in the samples of the sitting position. Selection of the most relevant features from this set of 9 features is completed using the feature selection algorithm. Table V shows the comparison of the performance of three different classification algorithms.

TABLE V
PERFORMANCE COMPARISON OF FEATURE SELECTION ALGORITHMS ON DATA FROM DOUBLE NODE

	Features used	Accuracy (%)
NBC with OFS	5	97.8
NBC with Cfs Subset Eval	3	95.4
NBC with Consistency Subset Eval	9	99.8

The optimum feature selection selected the following 5 features: Acceleration X axis, Acceleration Y axis, and Acceleration Z axis from the chest device and Acceleration Y axis and Acceleration Z axis from the thigh device.

From Table V, the accuracy of classification using the reduced feature set by optimum feature selection is 97.8 %. The Cfs Subset Eval algorithm reduced the features from 9 to 3, but with a considerable drop in accuracy. Consistency Subset Eval algorithm is not able to reduce the number of features at all and thus the accuracy values remain the same.

TABLE VI
CONFUSION MATRIX FOR NAIVE BAYES CLASSIFIER ON REDUCED FEATURE SET (OPTIMUM FEATURE SELECTION)

		Classified event								
		A	B	C	D	E	F	G	H	I
Actual event	A	23	0	0	0	0	0	0	0	0
	B	0	51	0	0	0	0	0	0	0
	C	0	0	100	0	0	0	0	1	0
	D	0	0	0	22	0	0	0	0	0
	E	0	0	6	0	92	0	0	2	0
	F	0	0	0	0	0	22	1	0	0
	G	0	0	0	0	0	0	20	0	0
	H	0	0	0	0	0	0	0	17	0
	I	0	0	0	1	0	0	0	0	100

A:Forward Fall B:Lying C:Standing
D:Right Fall E:Walking F:Backward Fall
G:Left Fall H:Near Fall I:Sitting

Table VI shows the confusion matrix for the result. The bulk of the resulting misclassification by optimum feature selection is between the walking and the standing positions. There are still no undetected falls. Since accurate distinction between standing and walking is not a priority issue, the reduction in feature set from 9 features to 5 features is a big advantage.

The Cfs subset evaluation algorithm reduced the number of features from 9 to 3. However, this is achieved at the cost of reduction in accuracy to 95.4 %. Table VII shows the confusion matrix for these results.

TABLE VII
CONFUSION MATRIX FOR NAIVE BAYESIAN CLASSIFIER ON REDUCED FEATURE SET (CFS SUBSET EVAL)

		Classified event								
		A	B	C	D	E	F	G	H	I
Actual event	A	23	0	0	0	0	0	0	0	0
	B	0	50	0	0	0	0	1	0	0
	C	0	0	100	0	1	0	0	0	0
	D	1	3	0	14	0	0	4	0	0
	E	0	0	0	0	98	0	1	1	0
	F	0	0	0	0	0	21	1	1	0
	G	0	0	0	1	4	0	14	1	0
	H	0	0	0	0	0	0	0	17	0
	I	0	0	0	0	0	0	1	0	100

A:Forward Fall B:Lying C:Standing
D:Right Fall E:Walking F:Backward Fall
G:Left Fall H:Near Fall I:Sitting

As is evident from the confusion matrix for reduced feature subset by Cfs subset evaluation algorithm, the misclassification rate is high. There are in total 21 misclassified samples, of which 9 are undetected fall samples. Though the number of features selected is less than the ones selected by optimum feature selection algorithm, the misclassification rate is too high to be considered acceptable.

C. Real Time Classification

Based on the classifier selected and the chosen features, the final real time implementation is described below. This real time implementation is completed using only a single chest worn device:

- The selected classifier is the Naive Bayesian classifier, so the parameters of the model are the means and standard deviations of each feature within each class and the prior probabilities of each class.
- For each incoming sample of data, the posterior probability of each class is calculated, assuming a Gaussian distribution and using the equation

$$P = \frac{1}{\sqrt{2\pi\sigma^2}} \exp - \frac{(x-\mu)^2}{2\sigma^2} \tag{2}$$

- This probability is calculated for each feature within each class and all are multiplied together and with the prior probability to give the posterior probability of any given class.
- The class with the maximum posterior probability is assigned to the sample and sent wirelessly to the co-ordinator.
- Every sample of data, after classification is used to update the model of the respective class by updating the means and deviations of each feature.

The following are the results of the real time implementation.

- Every fall event is detected correctly.
- 3-4 samples of data for each fall activity.
- Leaning in any direction is detected as a near fall situation.
- Fluctuation between standing and walking positions.

IV. Conclusion

The introduced device prototype developed by the FANFARE team, University of Saskatchewan, can detect and classify different types of falls and different types of normal movements and posture. Near-fall situations are also identified correctly and an early warning capability can be integrated to the device.

A Naive Bayesian classifier, which is a well established and popular algorithm for many machine learning problems, is used for the fall analysis and proved to be efficient in classifying different movements and postures accurately and detecting fall and near-fall situations. Using this classifier for the real time movement classification has resulted in a machine learning model which is easy to update, as it involves only the recalculation of means and standard deviations, and is computationally fast.

There is a fluctuation in classification between walking and standing, which intuitively can be rectified using more number of nodes. The adaptability of the device ensures that the model keeps adapting to an individual's particular gait, posture and lifestyle. The default model caters to the general concept of normal movements, postures and falls. The model can be fine tuned by changing the prior probabilities of each class. Even after this tuning, the device can keep improving the classification process by updating the model continuously. The event of a false alarm poses a problem as of now. The patient must tell the device that it is a normal movement and then the corresponding data should not update the means and deviations of the fall class. The algorithms to update the model are in place, however, the hardware capability to track and send the correct data for updates, in the event of a misclassification is not yet in place.

The optimum feature selection algorithm complements the Naive Bayesian classifier by selecting only the most relevant features necessary for accurate classification of movement. There is a need to use this algorithm as the existing popular feature selection algorithms are unable to reduce the number of features considerably. This algorithm is shown to perform well not only for the movement classification problem, but also for other types of data sets. The test with other UCI data sets showed that the algorithm is fairly generalized and can achieve a significant reduction in the number of features while keeping the classification accuracies at acceptable levels. The comparison with other feature selection algorithms both for movement classification data and UCI datasets, highlighted the difference between them. Where the other algorithms try to improve the accuracy while having a relaxed approach to reducing number of features, the optimum feature selection algorithm is suited to higher reductions in the number of features, while maintaining acceptable levels of accuracy.

Acknowledgements

This research was supported by Natural Sciences and Engineering Research Council of Canada under the strategic project grant # STPGP 35045.

References

[1] F. R. Allen, E. Ambikairajah, N. H. Lovell, and B. G. Celler, "An adapted gaussian mixture model approach to accelerometry-based movement classification using time-domain features," in *Proceedings of the 28th IEEE EMBS Annual International Conference*, 2006, pp. 3600–3603.

[2] A. Dinh, Y. Shi, D. Teng, A. Ralhan, L. Chen, V. D. Bello-Haas, J. Basran, S. Ko, and C. McCrowsky, "A Fall and Near-Fall Assessment and Evaluation System," *The Open Biomedical Engineering Journal*, vol. 3, pp. 1–7, 2009.

[3] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. John Wiley, 2001.

[4] I. Guyon and A. Elosseeff, "An Introduction to Variable and Feature Selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[5] M. Hall, "Correlation-based Feature Selection for Discrete and Numeric Class Machine Learning." Morgan Kaufmann, 2000, pp. 359–366.

[6] P. Hawranik, "A Clinical Possibility: Preventing health problems after the age of 65," *Journal of Gerontological Nursing*, vol. 17, pp. 20–25, 1991.

[7] J. Hwang, J. Kang, Y. Jang, and H. Kim, "Development of novel algorithm and real-time monitoring ambulatory system using bluetooth module for fall detection in the elderly," in *Proceedings of the 26th Annual International Conference of the IEEE EMBS*, 2004, pp. 2204–2207.

[8] L. Jiang, D. Wang, Z. Cai, and X. Yan, "Survey of Improving Naive Bayes for Classification," *Advanced Data Mining and Applications*, vol. 4632, pp. 134–145, 2007.

[9] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *International Joint Conference on Artificial Intelligence*, 1995, pp. 1137–1143.

[10] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial Intelligence*, vol. 1, pp. 273–324, 1997.

[11] H. Liu and R. Setiono, “A Probabilistic Approach to Feature Selection - A Filter Solution,” in *13th International Conference on Machine Learning*. Morgan Kaufmann, 1996, pp. 319–327.

[12] C. J. Lord and D. P. Colvin, “Falls in the elderly: Detection and Assessment,” in *Proceedings of the Annual International Conference of the IEEE EMBS*, vol. 13, 1991, pp. 1938–1939.

[13] S. Luo and Q. Hu, “A dynamic motion pattern analysis approach to fall detection,” in *Proceedings of IEEE International Workshop on Biomedical Circuits and Systems*, 2004, pp. 1–5–8a.

[14] J. Massion, “Postural Control System,” *Curr. Opin. Neurobiol.*, vol. 4, pp. 877–887, 1994.

[15] F. G. Miskelly, “Assistive technology in elederly care,” *Age and Ageing, British Geriatrics Society*, vol. 30, pp. 455–458, 2001.

[16] J. Nilsson, *Introduction to Machine Learning*, 1st ed., 2005.

[17] N. Noury, A. Fleury, P. Rumeau, A. Bourke, G. Laighin, V. Rialle, and J. Lundy, “Fall detection - principles and methods,” in *Proceedings of the 26th Annual International Conference of the IEEE EMBS*, 2007, pp. 1663–1666.

[18] N. Noury, A. Tarmizi, D. Savall, P. Boissy, G. Barralon, P. Virone, and P. Rumeau, “A smart sensor for fall detection in daily routine,” in *SICICA*, 2003, pp. 9–11.

[19] N. Noury, H. Thierry, V. Rialle, E. Virone, G. Mercier, G. Morey, A. Moro, and T. Pocheron, “Monitoring behaviour in home using a smart fall sensor and position sensors.” in *Special Topic Conference on Microtechnologies in Medicine and Biology, Annual International IEEE-EMBS*, 2000, pp. 607–610.

[20] A. Ralhan, S. Ko, and Y. Shi, “Optimum feature selection for naive bayesian classifier,” in *International Workshop on Multimedia Signal Processing and Transmission*, 2009, pp. 123–126.

[21] C. A. Ratanamahatana and D. Gunopulos, “Feature Selection FoR Naive Bayesian Classifier Using Decision Trees,” *Applied Artificial Intelligence*, vol. 17, pp. 475–487, 2003.

[22] O. P. Ryannen, S. L. Kivela, R. Honkanen, and P. Laipalla, “Falls and Lying helpless in the elderly,” *Z Gerontol*, vol. 25, pp. 278–282, 1992.

[23] J. N. Sixsmith, “A smart sensor to detect the falls of the elderly,” *IEEE Pervasive Comput.*, vol. 3, pp. 42–47, 2004.

[24] G. William, K. Doughty, and D. A. Bradley, “A smart fall and activity monitor for telecare applications,” in *Proceedings of the 20th International Conference of the IEEE EMBS*, 1998, pp. 1151–1154.

[25] L. Yu and l. Huan, “Efficient Feature Selection via Analysis of Relevance and Redundancy,” *Journal of Machine Learning Research*, vol. 4, pp. 1205–1224, 2004.