

# Implementation of a Flexible Bayesian Classifier for the Assessment of Patient's Activities within a Real-time Personalized Mobile Application

Vanja Miskovic

Faculty of Information Technology  
Slobomir P University, SPU  
Bijeljina, Bosnia and Herzegovina  
vanja.elcic@gmail.com

Djordje Babic

School of Computing  
Union University  
Belgrade, Serbia  
djbabic@raf.edu.rs

**Abstract**—This paper presents an implementation of a mobile application that provides a real-time personalized assessment of patient's activities by using a Flexible Bayesian Classifier. The personalized assessment is derived from data collected from the 3-axial accelerometer sensor and the counting steps sensor, both widespread among nowadays mobile devices. Despite the fact that online mobile solutions with Bayesian Classifier have been rare and insufficiently precise, we have proven that the accuracy of the proposed system within a defined data model is comparable to the accuracy of decision trees and neural networks.

**Keywords**—activity recognition; real-time mobile application; Flexible Bayesian Classifier.

## I. INTRODUCTION

Major goals of an intelligent system for patient monitoring are detection of health deterioration and notifying medical staff if a patient's health condition is life-threatening. A basic prerequisite for making a correct assessment is a reliable, accurate, timely source of raw data which indicate patient's vital signs. Further, the collected raw data is modeled with context-aware approach, in other words, the raw data is placed in an appropriate context. This is necessary because the collected data as raw facts cannot help in decision-making. It is necessary to analyze data features that can contribute to the making of the right decision. The development of the sensor network technologies (Bluetooth [1] and ZigBee [2]) focuses on health care ("HealthCare" profiles) and development of standardized wearable medical sensing devices (Continua Health Alliance [3]). The following parameters can be pointed out as the most important Quality of Context (QoC)-parameters [4]: precision, probability of correctness, trust-worthiness, resolution and up-to-datedness. In addition to the vital signs, a very important precondition for assessing the patient's health condition is also a correct assessment of physical activities which the patient performs at a given time. Therefore, all intelligent systems designed for patient monitoring, as for an example the system in [5], firstly assess the patient physical activity and secondly provide an evaluation of the patient's health. The considerable physical activities of the patient are

resting, sitting, standing and physically demanding activities, for example climbing up the stairs or running.

In order to estimate the activity of the patient, appropriate sensing devices are required. A very common approach to estimate activity is a combination of 3-axial accelerometers and/or 3-axial gyroscope and/or 3-axial magnetic sensors. The patient should always carry/wear the aforementioned sensors because he/she should be monitored by the system continuously in everyday activities. Various placements of the sensing devices on human body have been proposed. The most common placements of sensing devices are: mobile phone in a front pocket of trousers [6], accelerometer sensors on five different positions: elbow, wrist, knee, ankle and hip [7], mobile phone in left pocket, right pocket, belt, wrist and upper arm [8], mobile phone in hand or where patient typically holds it while doing specific activity in [9] etc. Usually the research focus is on offline testing of designed systems, whereas very few researchers implement and test systems in a real time (online) [9, 10]. From the beginning, the basic concept has been that these systems are generally applicable to any patient, while the latest studies show examples of personalized systems [8]. The personalized system gives an estimate for the patient based on his/her collected data. In a personalized system, the data must be collected for every patient in order to train the system and later to make an assessment. This type of systems should be more accurate than the generalized systems which are based on a common large database with collected data, where the same data are used for assessment for any patient.

This paper presents an implementation of the mobile application that provides real-time personalized assessment of physical activity. The presented personalized mobile application collects sensor data from a patient using his/her mobile phone in a free living environment, both for system training and decision making. A target group for the presented application is a group of chronic patients and process of collecting sensor data for certain activities may be limited. In addition, in real life, a certain activity is done less often than other, for example we climb up the stairs less often than we walk, and unfortunately sitting is the most common body

posture. Therefore, the application allows that a different number of data instances are collected per activity. The patient's movement data is gathered using a 3-axial accelerometer sensor and sensor for counting steps, both available in nowadays mobile devices. The mobile phone is designated to be carried in pockets of trousers, skirts or track suits - where it fits in the execution of certain activity. All above system's features are selected in accordance with the understandable desire of patients that the monitoring system has to be easy to carry and use [11].

In the proposed personalized mobile application, the Flexible Bayesian Classifier (FBC) is a chosen algorithm for the assessment and it is implemented according to [12]. Flexible and Naïve Bayesian classifiers (NBC) with minor changes are implemented in the WEKA environment, as well [13], but it is not applicable for Android platforms. There are several sources, such as [10], stating that the Naïve Bayesian classifier shows a very low accuracy. In [10], its accuracy is 48% while the k-nearest neighbor (KNN) algorithm has an average accuracy of 92%. Both algorithms showed that mobile phones' resources are sufficient for them. Naïve Bayesian classifier demanded more processing power (42% compared to KNN's 29%). However, when it comes to memory the situation is opposite and the KNN needed about 21.9 MB while the NBC demanded about 12.6 MB. In [9], a 1-Nearest Neighbor classifier has been implemented in a mobile phone, while a Naïve Bayesian classifier has been tested offline within WEKA environment. Although it seems to be too simple and the current real-time implementations are rare and aren't promising, NBC with the careful selection of the training data showed [14, 15] that in some applications, even in medicine, its effectiveness can be compared with neural networks and decision trees. Therefore, one of the objectives of this paper is to identify the relevant features of the collected data from accelerometer and sensor that counts steps. The following standard activities are selected for testing: sitting, standing, walking, running or jogging, climbing up the stairs and descending the stairs. We show here that, the real-time personalized mobile application using FBC implemented on a mobile phone with average performances provides activity assessment with satisfying accuracy. Convenient periods for prediction as well as the minimum amount of samples in a training data model are determined experimentally. Offline tests are carried out in WEKA environment and the FBC is compared to other algorithms, such as J48 and Multilayer Neural Network. A real-time testing of the proposed mobile application is conducted in a free living environment, and the results are presented in this paper.

## II. RAW DATA COLLECTION AND PRE-PROCESSING

An assessment of activities is made based on the time series obtained from the real-time signals. The data is collected by accelerometer sensor for all three axes: x, y and z. The Android API offers four abstract frequencies for its accelerometer: Fastest, Game, Normal, and UI. However, the various sensors implemented in mobile phones do not support all four frequencies, but they support only one or two frequencies. The fastest frequency supported by the most mobile phones at the market is 50 Hz. This is quite acceptable for the recognition of

physical activity of the patient [8]. The chosen time period for signal analyses is 8s with an overlap of 4s. This practically means that an assessment is actually provided every 4s. The signal features used for the assessment are calculated for all three x, y, z axes, and resulting acceleration  $r$ , which is calculated as:

$$r = \sqrt{x^2 + y^2 + z^2} \quad (1)$$

Android OS runs onSensorChanged() event every time there is a change in the sensor values. Usually, this event does not occur more frequently than 50Hz. Therefore, it is convenient to use a special timer task that is executed on a separate thread. The application runs the thread every 20ms and stores currently detected accelerations on x, y, z axes along with data from sensor which counts steps for later use. In this manner, there are 50 samples for x, y and z axes collected in one second. Within chosen period of 8s there are 400 samples per axis, and all together 1200 samples for all three axes. The obtained signal is preprocessed in order to calculate the energy of the signal. The role of preprocessing is to eliminate noise in the signal which occurs as a result of irregularities in the sensor operation. The noise can be generated also by quick movements of person, such as turning a mobile phone, or walking over obstacles, and so on. First, the signal is smoothed by signal interpolation increasing the number of samples per second to 100. After that, a simple moving average filter of length five is applied. The filter proved to be enough to smooth false peaks, but also to preserve specific forms of signals that allow us to distinguish activities. After preprocessing the signal contains 800 samples within the interval of 8s. Algorithms for data classification extract appropriate features from the raw or smoothed signal, in the same way which human would use to classify activity by observing signals charts. All statistical features are calculated from the raw data. These calculations are acceptable to the capabilities of mobile phones, while calculation of energy of smoothed signal represents moderately demanding task.

Analyzing previous studies [6, 8, 9, 15, 16], we select the following signal features:

1. The signal energies for all three axes: x, y and z, as well as the resultant acceleration  $r$  are calculated by using FFT. FFT is performed over 512 samples twice for each interval with an overlap of 224 samples ( $2 \times 512 - 224 = 800$ ). Before FFT, Hann window function is applied to the signal. The obtained FFT coefficients are normalized with sum of window function coefficients. For the resulting complex FFT coefficients  $x[i]$ , energy is calculated as [15]:

$$E = \frac{\sum_{i=1}^w |x[i]|^2}{w} \quad (2)$$

where  $w$  is a window length.

The signal energy of an acceleration data can discriminate low intensity activities such as lying from moderate intensity activities such as walking and high intensity activities such as jogging [16].

2. Arithmetic means for all three axes  $x$ ,  $y$ ,  $z$  and resultant  $r$  are calculated using raw data.

3. Minimum and maximum values for all three axes  $x$ ,  $y$ ,  $z$  and resultant  $r$  are calculated using raw data.

4. Standard deviations for all three axes  $x$ ,  $y$ ,  $z$  and resultant  $r$  are calculated using raw data:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (s_i - avg\_s)^2}{n-1}}, \quad (3)$$

where  $n$  is a number of samples.

5. Skewness is a measure of the asymmetry of the probability distribution. A negative skewness means that distribution is skewed to the left, and in the other words above-average values are more frequent. A positive skewness means that distribution is skewed to the right and below-average values are more frequent.

$$Skew = \frac{m3}{m2^{3/2}}, \quad (4)$$

$$m3 = \frac{\sum_{i=1}^n (s_i - avg\_s)^3}{n}, m2 = \frac{\sum_{i=1}^n (s_i - avg\_s)^2}{n}$$

6. Kurtosis is a measure that indicates whether a probability distribution is flatter or more peaked than the normal distribution. Kurtosis has a negative value for flattened and positive value for pointed probability distribution.

$$Kurt = \frac{m4}{m2^2} - 3, m4 = \frac{\sum_{i=1}^n (s_i - avg\_s)^4}{n} \quad (5)$$

7. In [15], it is stated that by using correlation between axes it is possible to differentiate walking and jogging from climbing the stairs up and down. Therefore, the correlations of  $x$  and  $y$ ,  $x$  and  $z$ ,  $x$  and  $r$ ,  $y$  and  $z$ ,  $y$  and  $r$ ,  $z$  and  $r$  are calculated. The correlation of  $x$  and  $y$  is calculated using a covariance as:

$$corr(x, y) = \frac{cov(x, y)}{\sigma_x \sigma_y}, \quad (6)$$

$$cov(x, y) = \frac{\sum_{i=1}^n (x_i - avg\_x)(y_i - avg\_y)}{n-1}$$

8. In addition to the data collected from the accelerometer, the mobile application collects data from the sensor for counting steps (SENSOR STEP COUNTER). The patient can carry mobile phone in different pockets in clothing, thus the step counter can register more steps or some steps can be skipped. The counter is only reset when the device is rebooted and in the meantime continues to count the detected steps. Therefore, an interesting feature for a single 8s time interval is a number of steps counted during interval. The number of steps is not reliable feature to distinguish the activities, for example the difference between walking and climbing the stairs. However, it is important feature to identify the resting compared to the significant physical efforts.

The [6] gives the signal properties for  $x$ ,  $y$  and  $z$  axes for several activities under controlled conditions. These activities are still standing and sitting, and walking/jogging while the phone with sensor is placed in the pants front pocket. In the case of still sitting a mobile phone with sensor is usually positioned horizontally, thus the Earth's gravitation causes that the accelerometer measures a value of about 9.8 m/s<sup>2</sup> on the axis  $z$ . In this case, there are very small changes of acceleration on the axes  $x$  and  $y$ . Aforementioned values are typical for any activity which is based on sitting without any significant movements, examples of such activities can be working at the computer, turning around, moving the legs, etc. In the same way, for the still standing with a phone in the pants front pocket, the acceleration on  $y$ -axis is close to the gravitational acceleration. This is due to the fact that for standing the mobile phone is in its vertical orientation and its axis  $y$  is orthogonal to the ground.

Leg movements and turning around can disturb these features. The test data collected during everyday activities, which are not strictly controlled; show that they are a reflection of patient's habits and behavior. Figure 1 illustrates the measured signals for all three axes in the case of sitting (above) and standing (below). The signals are shown after pre-processing with resolution of 100 samples per second. Thick lines represent the signal after smoothing explained above, and thin lines which are visible in some areas represent raw interpolated signal. Figure 2 shows signals collected for activities of walking (above) and running (below). A difference between smoothed and raw interpolated signal is more visible here. Figure 2 shows signals when a mobile phone was located in the trousers pocket for walking and in the wider tracksuits front pocket for running. In repetitive activities (i.e. walking and running) regular peaks can be seen at  $y$  and  $z$  axes. In the case of different placement of the phone, a frequency of peaks does not have to follow the same rule [6]. For example in [6], the time between successive peaks on the  $y$ -axis is:  $\frac{1}{2}$  s for walking and  $\frac{1}{4}$ s for running. These peaks clearly indicate that it is the activity with periodic behavior and frequency of these peaks is the highest for running. If signal smoothing decreases significant peak values, minimum and maximum values of raw data will preserve higher peak values on the axis  $y$ .

The signals for activities descending the stairs (above) and climbing the stairs up (below) are illustrated in Figure 3 The descending the stairs signal has a noticeable feature of the series of small peaks on the axis  $y$ . The  $z$ -axis values show a

similar trend with negative acceleration, reflecting the regular movement down each stair [6]. The peaks on the axis  $x$  are small and semi-periodical. The acceleration on the axis  $x$  alternates between positive and negative values. The signal on axis  $x$  has a similar behaviour in both examples. When climbing up the stairs the patient is usually slower and the frequency of peaks on the  $y$  axis should be lower. In the case of climbing up the stairs  $z$  axis has higher positive peaks, and again as with the descending down the stairs it is negative where  $y$  is positive and vice versa.

Based on the presented signal in Figure 3, one can conclude that it is very difficult to distinguish climbing up and descending down the stairs. Sitting and standing are different from all other activities. Moderate walking is differentiated easily from running, as exhaustive activity. For the system for patient monitoring, even these classes of activities: resting, moderate and exhaustive; provide sufficient information for the assessment of vital signs.

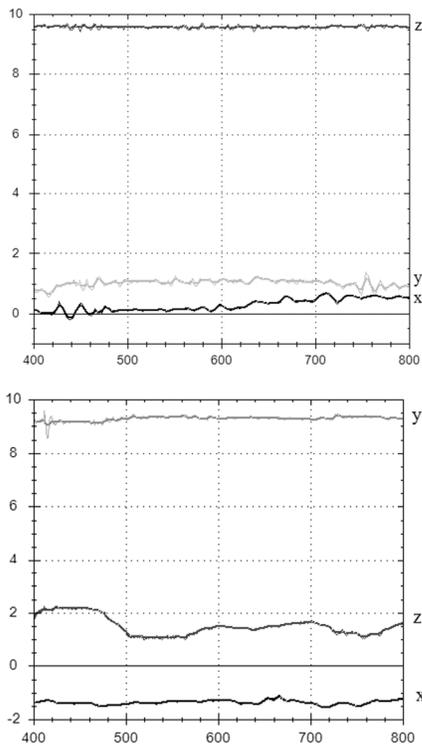


Fig. 1. Examples of  $x$ ,  $y$  and  $z$  signals for sitting (above) and standing (below).

### III. FLEXIBLE AND NAÏVE BAYESIAN CLASSIFIER

Naïve Bayesian classifier (NBC) has a high accuracy if classification model learns from data set that can be represented as a conjunction of discrete and/or continuous attributes [12]. As a result, the classifier gives the most probable class from a limited set of classes  $C$ . The classifier learns from a set of samples. The sample is an  $n$ -tuple of attribute values  $(a_1, a_2, \dots, a_n)$ . Testing is usually performed using a special set of samples or a training data set is divided into  $n$  folds. In such case, the classifier learns iteratively. In

each iteration one fold is used for testing and  $n-1$  folds are used for learning. A task of trained classifier is to determine the class of a new example.

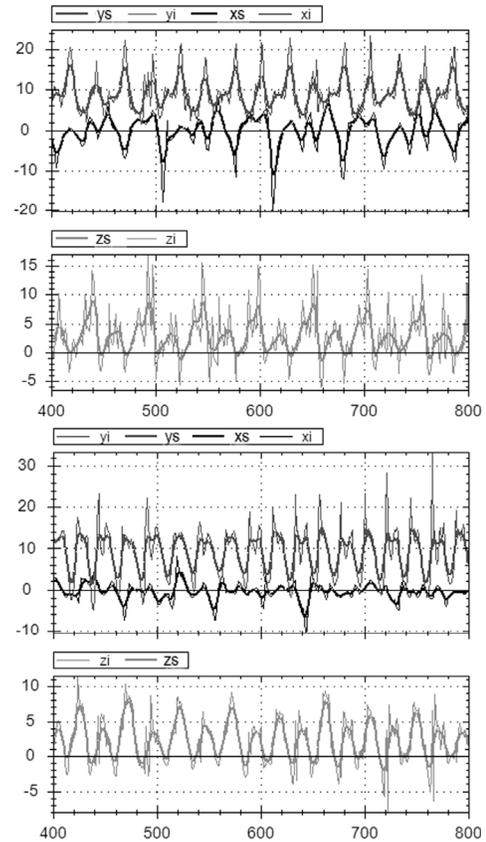


Fig. 2. Examples of  $x$ ,  $y$  and  $z$  signals for walking with phone in a right trousers pocket (above) and running with mobile phone in a wider tracksuits front pocket (below).

Bayesian classifier is a conditional probability model and it can be abstractly described as  $P(C|A_1, A_2, A_3, \dots, A_n)$ . Here,  $C$  is a set of possible classes and  $A_1, \dots, A_n$  is an example of input test data. Bayesian classifier computes the conditional probability of each class for a given tuple of attributes and then predicts the most probable class.

Using Bayes' theorem, the conditional probability can be decomposed as:

$$P(C|A_1, A_2, A_3, \dots, A_n) = \frac{P(C) * P(A_1, A_2, A_3, \dots, A_n|C)}{P(A_1, A_2, A_3, \dots, A_n)} \tag{7}$$

The numerator of the previous equation can be rewritten as:

$$\begin{aligned} P(C) * P(A_1, A_2, A_3, \dots, A_n|C) &= P(C) * P(A_1|C) * P(A_2, A_3, \dots, A_n|C, A_1) \\ &= P(C) * P(A_1|C) * P(A_2|C) * P(A_3, \dots, A_n|C, A_1, A_2) = \dots \\ &= P(C) * P(A_1|C) * P(A_2|C) * P(A_3|C) * \dots * P(A_n|C, A_1, A_2, A_3, \dots, A_{n-1}) \end{aligned} \tag{8}$$

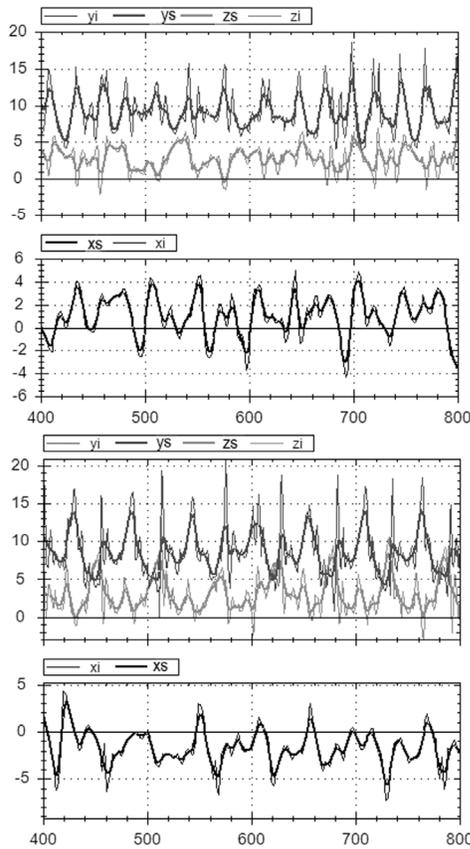


Fig. 3. Examples of signals for activity taking the stairs down (above) and taking the stairs up (below).

The "naive" conditional independence assumes that each attribute  $A_i$  is conditionally independent of any other attribute  $A_j$  for  $j \neq i$ , when the class is  $C$ . In other words:  $P(A_i | C, A_j) = P(A_i | C)$  for  $i \neq j$ . Under independence assumption explained above new expression for the numerator is:

$$P(C) * P(A_1, A_2, A_3, \dots, A_n | C) = P(C) * \prod_{i=1}^n P(A_i | C) \tag{9}$$

Then the Bayes' theorem of (7) is given as:

$$P(C | A_1, A_2, A_3, \dots, A_n) = \frac{P(C) * \prod_{i=1}^n P(A_i | C)}{P(A_1, A_2, A_3, \dots, A_n)} \tag{10}$$

The denominator is often omitted from the calculation of probability, because for each class  $C$  it has the same value. The sum of all probabilities of all possible classes should be equal to the probability of certain event. Thus all calculated probabilities are normalized with sum of all probabilities.

The probability of a certain class (prior) can be easily calculated from the learning data set as a ratio of number of samples in the class and total number of samples. Furthermore, if a domain of an attribute is finite (discrete) set of values, then its probability can be expressed as the number of samples

where the attribute has a certain discrete value  $a$  from a set of discrete values  $A$  and given class is  $c$  from  $C$ ,  $P(A = a | C = c)$ . However, in the most cases attributes have continuous values and the probability for the given class is modeled by some continuous probability distribution function over the range of its values for the given class. The probability of continuous attributes is often modeled using normal or Gaussian distribution, and it is calculated as [12]:

$$P(A = a | C = c) = g(a, \mu_c, \sigma_c), \text{ where} \tag{11}$$

$$g(a, \mu_c, \sigma_c) = \frac{1}{\sigma_c \sqrt{2\pi}} * e^{-\frac{(a-\mu_c)^2}{2\sigma_c^2}}$$

For each continuous attribute and each class it is required to calculate mean and standard deviation. The standard deviation is a measure that is used to quantify the amount of variation or dispersion of data values. A low standard deviation means that elements of given set tend to be very close to the mean, and a high standard deviation indicates that elements of given set have a wider range of values. To clarify the estimation process, consider a small data set in which there are two classes (+ and -), a nominal attribute  $X_1$  which takes values  $a$  and  $b$ , and a continuous attribute  $X_2$  [12]. If there are five training examples: (+;  $a$ ; 1); (+;  $b$ ; 1.2); (+;  $a$ ; 3.0); (-;  $b$ ; 4.4); (-;  $b$ ; 4.5), then the corresponding conditional probabilities are [12]:

$$P(C = +) = 3/5$$

$$P(X_1 = a / C = +) = 2/3$$

$$P(X_1 = b / C = +) = 1/3$$

$$P(X_2 = x / C = +) = g(x; 1.73; 1.21)$$

Here, 1.73 is the average value of  $X_2$  where the class is +, while 1.21 is the standard deviation of  $X_2$  where the class is +.

In [12] the advantage of using the kernel function over Gaussian normal distribution is highlighted. A kernel is a non-negative real-valued integrable function  $K$  satisfying the following two requirements [17]:

$$\int_{-\infty}^{+\infty} K(u) du = 1;$$

$$K(u) = K(-u) \text{ for all values of } u \tag{11}$$

The kernel estimation with Gaussian kernels (one can use other kernel functions as well) look much the same, except that the estimated probability is averaged over a large set of kernels [12]:

$$P(X = x | C = c) = \frac{1}{n_c h} \sum_j K\left(\frac{x - \mu_j}{h}\right),$$

where  $K = g(x, 0, 1)$  and  $h = \frac{1}{\sqrt{n_c}}$  (12)

Kernel functions are calculated for each  $x_j$  from  $XI$  where the class  $c$  is from  $C$  and  $\mu_j = x_j$ . Also  $n_c$  is a number of samples for the given class  $c$ . A parameter  $h$  is called width parameter and it shrinks to zero as the number of instances goes to infinity. Thus, as Flexible Bayesian classifier (FBC) observes more training samples, its probability estimates become increasingly local [12]. FBC is far more demanding when it comes to storage and computational complexity compared to the NBC. However, FBC proved that for 200 samples its accuracy is as high as accuracy of the NBC. While for the distribution of data that does not meet the requirement of normality it is much more accurate than NBC.

A WEKA [13] is an environment for testing algorithms of artificial intelligence and it is very often used in the scientific researches for offline testing. WEKA supports both types of Bayesian classifier. A width parameter  $h$  of the kernel function is calculated as an average of the sum of differences of successive attribute values. Attributes from the collected data set are sorted. Differences of successive values are summed, if values are not identical. At the end this sum is divided with a number of different attribute values. This parameter is called precision in the WEKA environment and it is used to compute the kernel function and for rounding up attribute values. In order to avoid redundancy of the attribute values after rounding up only different values are allocated and stored with the number of repetitions of these values (weight). In this way, the amount of data for storing is significantly reduced when the attribute has values which change rarely. This leads also to the reduction of the number of evaluations of kernel function. For example, the step counter sensor has a value zero for activity of sitting.

#### IV. OFFLINE TESTING OF MODELED DATA IN THE WEKA ENVIRONMENT

The data is prepared for testing in the WEKA testing environment in the following manner. Eight features explained in Section 2 are extracted from the raw class labelled data, and after that the data is normalized by scaling it between 0 and 1. The selected method of testing is a cross-validation with ten folds. It is a fact that there are certain sorts of activities which are done more frequently, for example sitting, standing and walking, compared to activities which occur rarely, for example use of the stairs and running. Taking this fact into account, a training data set is defined with more samples for more common activities.

The simulation results show that the Flexible Bayesian Classifier with kernel estimator correctly classifies 92.68% of all the samples. True positive rates, which measure the proportion of positives that are correctly identified as such, are: 1 for sitting, 1 for standing, 0.859 for walking, 0.958 for running, 0.761 for climbing the stairs, 0.866 for descending downstairs. Furthermore, the Naïve Bayesian Classifier with Gaussian estimator correctly classifies 88.18% of all the samples. Here true positive rates are: 0.99 for sitting, 0.988 for standing, 0.786 for walking, 0.952 for running, 0.658 for climbing the stairs, 0.724 for descending downstairs. These differences in accuracy can be seen in confusion matrix in Table I and Table II.

We have tested the accuracy of the Flexible Bayesian Classifier for data set without features 5 and 6 (skewness and kurtosis). In this case FBC correctly classifies 90.21% of samples, and true positive rates by class are as follows: 1 for sitting, 1 for standing, 0.837 for walking, 0.958 for running, 0.662 for climbing the stairs, 0.756 for descending the stairs. In comparison with previous FBC results, the use of these two features slightly increases the percentage of correctly classified samples for activities of walking and using stairs. The accuracy of FBC for data set without feature 7 (correlation) is 91.64%, and true positive rates by class are as follows: 1 for sitting, 1 for standing, 0.823 for walking, 0.955 for running, 0.757 for climbing the stairs, 0.848 for descending the stairs. In comparison to the first results of FBC, it can be seen that the feature 7 affects all activities except sitting and standing. The accuracy of FBC for data set without feature 8 (number of steps) is 92.03%, and the true positive rates by class are as follows: 1 for sitting, 1 for standing, 0.853 for walking, 0.955 for running, 0.734 for climbing the stairs, and 0.843 for descending the stairs. In comparison to the first results of FBC it can be seen that this features 8 and 7 in the same way affect the estimates.

TABLE I. FLEXIBLE BAYESIAN CLASSIFIERS CONFUSION MATRIX

a	b	c	d	e	f	classified as
507	0	0	0	0	0	<i>Sitting = a</i>
0	501	0	0	0	0	<i>Standing = b</i>
0	0	437	1	11	60	<i>Walking = c</i>
0	0	14	338	0	1	<i>Running = d</i>
0	0	33	0	169	20	<i>Upstairs = e</i>
0	0	4	0	25	188	<i>Downstairs = f</i>

TABLE II. NAÏVE BAYESIAN CLASSIFIERS CONFUSION MATRIX

a	b	c	d	e	f	classified as
502	0	0	0	5	0	<i>Sitting = a</i>
0	495	0	0	0	6	<i>Standing = b</i>
0	0	400	3	20	86	<i>Walking = c</i>
0	0	17	336	0	0	<i>Running = d</i>
0	0	49	0	146	27	<i>Upstairs = e</i>
0	0	30	0	30	157	<i>Downstairs = f</i>

TABLE III. J48 DECISION TREE CONFUSION MATRIX

a	b	c	d	e	f	classified as
507	0	0	0	0	0	<i>Sitting = a</i>
1	500	0	0	0	0	<i>Standing = b</i>
0	0	459	2	25	23	<i>Walking = c</i>
0	0	1	351	1	0	<i>Running = d</i>
0	0	30	0	177	15	<i>Upstairs = e</i>
0	0	19	0	24	174	<i>Downstairs = f</i>

TABLE IV. MULTILAYER PERCEPTRON CONFUSION MATRIX

a	b	c	d	e	f	classified as
507	0	0	0	0	0	<i>Sitting = a</i>
0	501	0	0	0	0	<i>Standing = b</i>
0	0	491	0	10	8	<i>Walking = c</i>
0	0	2	349	1	1	<i>Running = d</i>
0	0	14	0	199	9	<i>Upstairs = e</i>
0	0	12	0	12	193	<i>Downstairs = f</i>

J48 decision tree algorithm correctly classifies 93.9% of samples, and true positive rates by class are as follows: 1 for sitting, 0.998 for standing, 0.902 for walking, 0.994 for running, 0.734 for climbing the stairs, and 0.802 for descending the stairs. The Multilayer Perceptron correctly classifies 97% of the samples, and true positive rates by class are as follows: 1 for sitting, 1 for standing, 0.965 for walking, 0.989 for running, 0.896 for climbing the stairs, and 0.889 for descending the stairs.

#### V. REAL TIME TESTING OF THE PROPOSED MOBILE APPLICATION

As stated above, we have designed the mobile application for testing purposes. It allows users to check the activity that will be carried out, and then to initiate the assessment of the activity. The application is based on the described Flexible Bayesian classifier which evaluates the data collected in periods of 8s, with 4s overlap. Therefore, the process of assessment must be done in less than 4s. The Flexible Bayesian classifier used for the activity assessment uses a personalized patient's data model that is implemented within Java desktop application. At the first stage, the data collected from accelerometer and step counter is preprocessed and all described features are obtained, such as average, minimum or maximum value of signal or energy, etc. After that, the data is normalized. Further, all prior probabilities and the total number of samples per class are calculated, including precision for each attribute, values of the kernel functions followed with their weights. All assessed activities and activities checked by user, along with the system time required for the assessment are stored in a log file for later analysis. We used device with Android OS 5.0.2. for testing. The CPU ARMv7 rev 3 (v7I) has four cores with maximum frequency equal to 1190.4 MHz, and minimum frequency equal to 300.0 MHz. The number of samples collected for all activities is roughly equal to 2300. The log files show that with used mobile phone and its CPU, the process of assessing the data with preprocessing can be executed in less than 2s. The required memory resources are approximately 40 MB.

TABLE V. REAL-TIME TESTING USING MOBILE PHONE

a	b	c	d	e	f	classified as
641	0	0	0	0	0	<i>Sitting = a</i>
3	415	0	0	0	1	<i>Standing = b</i>
0	0	715	0	8	11	<i>Walking = c</i>
0	0	4	200	4	5	<i>Running = d</i>
0	0	45	2	82	18	<i>Upstairs = e</i>
0	2	60	0	6	94	<i>Downstairs = f</i>

The real-time analyses explained above shows similar behavior as the WEKA classifier in simulation. The mobile application is tested for all activities and the results are presented in Table V. The activities of sitting and standing are mostly correctly assessed. The activities of walking and running are also highly accurately assessed, their true positive rates are: 0.974 and 0.939, respectively. The activities related to the use of stairs are more often wrongly assessed and falsely interpreted as walking. The testing of the stairs activities in the

real-time situations shows that they are less accurate compared to the WEKA environment. The results of testing show that true positive rates are only: 0.558 climbing the stairs and 0.580 descending the stairs.

#### VI. CONCLUSION

Real-time testing using the mobile applications shows that it is possible to use the Flexible Bayesian classifier to predict accurately the patient's activity if features are carefully selected. The offline test results show that for the selected data model, the accuracy of the Flexible Bayesian classifier can be compared to the accuracy of the J48 decision tree and the Multilayer Perceptron. Based on obtained results, it can be concluded that the Flexible Bayesian classifier can very precisely distinguish resting, moderate activities and exhaustive activities such as running. The patient carries the phone with sensor at arbitrary position making the proposed system non-intrusive. The presented data model can be extended with new sensor devices, for example with gyroscope, which would increase the accuracy even further. In this way, the presented FBC implementation shows that it is one of the best choices in the field of mobile healthcare services.

#### REFERENCES

- [1] Bluetooth Medical Devices WG, "HDP Implementation Guidance White Paper", 2009, [goo.gl/oyZhwS](http://goo.gl/oyZhwS)
- [2] ZigBee Alliance, "ZigBee Wireless Sensor Applications for Health, Wellness and Fitness", 2009, [goo.gl/9vuvnY](http://goo.gl/9vuvnY)
- [3] Continua Health Alliance, "Overview Presentation: The Next Generation of Healthcare: Personal Connected Health & Wellness", 2010, <http://www.continuaalliance.org>
- [4] T. Buchholz, A. Kupper, M. Schiffrers, "Quality of Context: What It Is And Why We Need It", 10th Workshop of the Open View University Association OVUA'03, Geneva, Switzerland, July, 2003
- [5] T. Garcia-Valverde, A. Muñoz, F. Arcas, A. Bueno-Crespo, A. Caballero, "Heart Health Risk Assessment System: A Nonintrusive Proposal Using Ontologies and Expert Rules", BioMed Research International, Article ID 959645, Vol. 2014,
- [6] J. R. Kwapisz, G. M. Weiss, S. A. Moore, "Activity Recognition using Cell Phone Accelerometers", ACM New York, Newsletter, Vol. 12, No. 2, pp. 74-82, 2010
- [7] L. Bao, S. S. Intille, "Activity Recognition from User-Annotated Acceleration Data", Lecture Notes in Computer Science, Vol. 3001, pp. 1-17, 2004
- [8] M. Shoaib, S. Bosch, O. Durmaz Incel, H. Scholten, P. J. M. Havinga, "Fusion of Smartphone Motion Sensors for Physical Activity Recognition", Sensors, Vol. 14, pp. 10146-10176, 2014
- [9] S. Das, B. Perez, F. W. Olin, A. Perrig, "Detecting User Activities using the Accelerometer on Android Smartphones", Journal of Multimedia and Information System Vol. 2, No 2, pp. 233-240, 2015
- [10] M. Kose, O.D. Incel, C. Ersoy, "Online human activity recognition on smart phones", In Proceedings of the Workshop on Mobile Sensing: From Smartphones and Wearables to Big Data, Beijing, China, pp. 11-15, April 2012
- [11] M. Ziefle, C. Rocker, "Acceptance of pervasive healthcare systems: A comparison of different implementation concepts", Pervasive Computing Technologies for Healthcare (PervasiveHealth), IEEE 4th International Conference, Munich, pp. 1-6, March 2010
- [12] G. H. John, P. Langley, "Estimating Continuous Distributions in Bayesian Classifiers", 11th Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, pp. 338-345, 1995

- [13] WEKA, Weka 3: Data Mining Software in Java, <http://www.cs.waikato.ac.nz/ml/weka/>
- [14] T. Mitchell, *Machine Learning*, McGraw Hill, New York, 1997
- [15] I. Cleland, B. Kikhia, C. Nugent, A. Boytsov, J. Hallberg, K. Synnes, S. McClean, D. Finlay, "Optimal Placement of Accelerometers for the Detection of Everyday Activities", *Sensors*, Vol. 13, pp. 9183-9200, 2013
- [16] G. J. Welk, J. A. Differding, R. W. Thompson, S. N. Blair, J. Dziura, P. Hart, "The utility of the Digi-walker step counter to assess daily physical activity patterns", *Medicine & Science in Sports & Exercise*, Vol. 32, pp. S481-S488, 2000
- [17] Wikipedia, "Kernel (statistics)", [https://en.wikipedia.org/wiki/Kernel\\_\(statistics\)](https://en.wikipedia.org/wiki/Kernel_(statistics))