Comparison of Nearest Neighbour and Neural Network Based Classifications of Patient's Activity

Matti Pouke
Department of Information Processing Science
University of Oulu
P. O. Box 3000
FIN-90014 University of Oulu, FINLAND
Email: Matti.Pouke@oulu.fi

Risto T. Honkanen
Department of Information Processing Science
University of Oulu, Kajaani Unit
P. O. Box 51
FIN-87101 Kajaani, FINLAND
Email: Risto.Honkanen@oulu.fi

***Abstract*—This paper presents a comparison of 1-nearest neighbour (1-NN) and neural network based classification of patient activity. The data for classification was acquired from two 6 degree-of-freedom accelerometers deployed at the wrists of a patient. Instead of calculating statistical values, we studied the use of data samples acquired from 200ms time window. The best results were achieved with the 1-nearest neighbour algorithm. The overall accuracy of the 1-NN method was nearly 100%. The learning method for neural network used was the backpropagation with momentum. According to our experiments, the results of classification were more accurate with 1-NN in comparison with the result of neural network (93.4%).**

***Index Terms*—ubiquitous computing; classification; neural network; 1-nearest neighbour; accelerometer sensors**

I. Introduction

Ubiquitous computing is a model of human-computer interaction in which information processing has been integrated into everyday objects and activities. A part of research in ubiquitous computing is developing methods to detect and classify user activities based on often noisy sensor data. An example of human activity classification is the identification of the daily activities of elderly persons living on their own.

The activity identification can be used for safety related functions such as fall detection or help the patient in daily tasks offering assistive services such as memory aids. A cheap and lightweight solution to collect activity data is the use of accelerometer sensors [1], [6], [7], [9]. Accelerometers usually return a real value estimate of acceleration and/or gyrostatic value of a sensor along *x*-, *y*-, and *z*-axises.

In this work we describe an ubiquitous computing system built into a hospice. The system was used for activity data gathering from two elderly patients. The activity data of one patient is identified with two classification methods and the results from the both methods are compared.

Our system consists of two 6 degree of freedom accelerometer sensors, a number of proximity sensors, a personal data assistant (PDA), and a server. The data is collected by accelerometer and proximity sensors and sent to the PDA using Bluetooth. The server receives the data from the PDA by a wireless network connection and writes it into a hard disk. The data is then postprocessed by e.g. a classification algorithm.

In a number of previous studies, classification of activities is based on calculated features of accelerometers, e.g. acceleration mean and standard deviation values, correlation coefficients, and/or mean crossing values for each axis acceleration [1], [6], [7], [9]. We decided, however, to use samples of data acquired from a 200 ms time window instead of the use of statistically calculated values. According to our experiments it seems to be a reasonable method to collect the learning data.

We experimented with two classification methods and a backpropagation neural network based classification in our study. The classification methods we used were J48 and the 1-nearest neighbour algorithm. The best results were achieved with the 1-nearest neighbour algorithm with the overall accuracy of nearly 100%. We then studied the use of multilayer perceptron using backpropagation as learning method. The neural network based classification achieved the overall accuracy of 93.4%. We evaluated the 1-NN classifier with 10-fold cross validation. On the other hand, for the neural network the data set was divided in three sets used for training, validation, and testing. This was done because of classification tools we used. The unbalanced design may result an unfair comparison of classification models.

Section II shortly discusses the related work. The design of experimental setup and preliminary work are presented in Section III. Section IV introduces the data acquisition process. In Section V we introduce architectures of recognition models we experimented. Section VI presents the results of our studies. Section VII gives conclusions and discusses the future work.

II. Related Work

In their paper, Ravi et al. present an activity recognition system based on accelerometer sensor data deployed at the pelvic region of the testees [9]. They used decision trees (C4.5), decision tables, naive Bayesian classifier, and nearest neighbour algorithms as classifiers offered by the Weka Machine Learning Toolkit [3]. According to their studies, the best performance were achieved by the decision tree classifier with the overall accuracy of 84%.

Another implementation of activity recognition system is presented by Pirttikangas et al. in their paper [7]. They at-

PervasiveHealth 2011, May 23-26, Dublin, Republic of Ireland

DOI 10.4108/icst.pervasivehealth.2011.245980

tached the accelerometer sensors on the wrists and on the right tight of the testees. Twenty activities from 20 different users were studied during their experiments. The authors achieved the overall recognition rate of 84% using the decision tree classifier [7].

Work of Ohmura et al. have studied a bluetooth based wearable sensing device for nursing activity recognition [6]. The building component of their system is an accelerometer sensor equipped with a wireless connection between the device and an access point (AP). The whole system consists of a number of wearable sensors, environmental sensors, wireless AP's, backend servers, and a local area networks connecting fixed deployed devices. The recognition accuracy were around 80% using the C4.5 algorithm and exceeded 95% using the 1-nearest neighbour algorithm.

This research compares the results and benefits of a custom-built neural network into classification algorithms, mainly the *k*-nearest neighbour algorithm offered by data mining tools. Comparison of neural networks and *k*-NN algorithm is previously studied in [2] and [5]. In the previous studies, areas of interests were daily flow forecast [2] and quality control in the food industry [5].

This work belongs to an ongoing research not only to develop context aware services for ambient assistive living environments, but to also create a virtual test environment for these services. The purpose of the VE is to prototype the services with actual user data. A previous work in this research described a simple classification system used within the dataset acquired from a hospice. This research implements significantly improved classification methods to the same dataset. [8]

III. Design of Experimental Set-up

The sensor network is the same as presented in [8]. The network consisted of two 6 degree-of-freedom (6-DOF) accelerometers, a master module sensor, several proximity sensors, a PDA, and a laptop. The properties of sensors are presented in Table I. We used WAA-006 type sensors as 6-DOF accelerometer sensors and WAA-001 type sensors as proximity and master module sensors. The manufacturer of the sensors is ATR Promotions, Japan and their predecessors were used in the work of Ohmura et al [6].

The 6-DOF sensors returned the following estimates: a time stamp, acceleration along *x*-, *y*-, and *z*-axises and the angular velocity along *x*-, *y*-, and *z*-axises. The data collection was executed by researchers placing proximity sensors in appropriate locations in the research space and attaching the personal sensors to the testee. In this work, we omitted the data acquired from the proximity sensors. The set of personal sensors consisted of a master module placed into a pocket of the subject and two accelerometers attached to the wrists. The data acquisition process was initiated with the PDA and data was automatically recorded as the subject performed his activities. The PDA and the sensors communicated with a Bluetooth connection whereas the PDA and the laptop running the server software communicated through a WLAN. A schema of sensor network used is presented in Fig. 1.

TABLE I
PROPERTIES OF SENSORS USED IN EXPERIMENTS

Property	WAA-001	WAA-006
CPU	Renaissance Technology H8 7.3728 MHz	Renaissance Technology H8 7.3728 MHz
Size [mm]	38.0 × 39.0 × 10.0	38.0 × 39.0 × 10.0
Weight [g]	17	20
Battery life [h]	~ 4.5	~ 6
Communication	BT 2 Ver. 1.2 Class 2	BT 2 Ver. 2.0+EDR
Sensors	Hitachi ±3G, 200 Hz	Hitachi ±2G/4G, 500 Hz (acc.) InvenSence ±500 deg/s (x, y gyro) Epson Toyocom ±300 deg/s (z gyro)

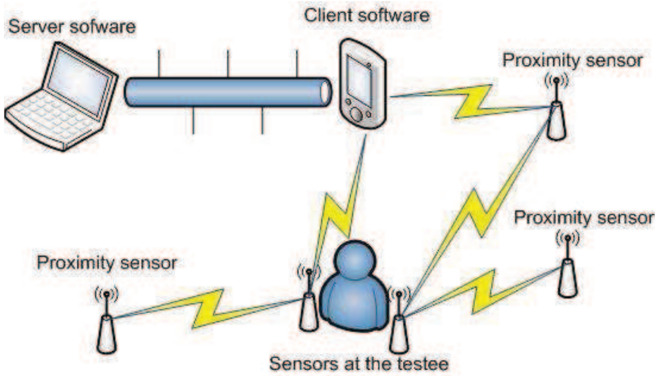

Fig. 1. Schema of sensor network we used .

IV. Data Acquisition Process

The data was acquired from a hospice for Veterans in Oulu, Finland. The subjects consisted of two elderly patients with ages of 84 and 90 years. In this study, the comparison was studied using only the data of the younger patient. As the data acquisition experiment required pre-planning, the subjects were interviewed before the experiment. During the interviewing process, the daily activities of the subjects were discovered. According to these activities, key locations of actions in their living environments were pointed out. Proximity sensors were placed onto these key locations. As the purpose was to find out whether regular daily activities of the elderly can be recognized, no strict plan or schedule was made for the activities; It was agreed that the participants would perform some of their usual daily pastimes. During the data acquisition the subjects acted alone according to their own pace without much guidance. However it was important for researchers to encourage and entertain the elderly patients to some extent.

V. Architectures of the Recognition Models

We now present an recognition model for our architecture. Section V-A presents the nearest neighbour model we used. In

Section V-B we introduce neural network based classification model.

A. Nearest Neighbour Model

The k-nearest neighbour algorithm is a classification method based on instance-based learning. The training samples of a k-nearest neighbour method are vectors in a multidimensional feature space, each vector having a class label. The dimensionality of the space equals to the amount of features in the dataset. During the training phase, the algorithm simply stores the feature vectors and their corresponding class labels. During the classification phase, the euclidian distances from training samples to test samples are calculated. The k closest training samples determine the classification of a sample by majority vote. Ties can be broken at random. In this study, 1-nn algorithm is used which means that the class of the closest sample alone determines the classification of a testing sample. This study used the Weka Machine Learning Toolkit [3] for the utilization of the 1-NN -algorithm.

B. Neural Network Model

Let $u_j = f\left(\sum_{i \in pred(j)} w_{ij} u_i\right)$ be our basic computational unit i.e. model neuron having i inputs and an output, where f is an activation function, $pred(j)$ is the set predecessors of the node, $w_{i,j}$ is the weight of input i, and u_i is the input value from the predecessor i. The model neuron computes a function f of the weighted sum of its inputs.

A multilayer perceptron is said to be a feedforward artificial neural network that maps a set of input data onto a set of outputs [4]. A multilayer perceptron consists of a number of layers (l) each having a number of computational units or neurons. We call neurons at the layer 0 as input neurons and neurons at the layer $l-1$ as output neurons. Let $u_j^{l'}$ be the output value of a neuron j at a level l'. When it receives its input values i, it calculates its output using the equation

$$u_j^{l'} = \begin{cases} d_j & \textit{if } l' = 0, \\ f\left(\sum_{i \in pred(j)} w_{ij} u_i\right) & \textit{otherwise}, \end{cases} \tag{1}$$

where d_j is the input value for the input neuron u_j^0, f is an activation function of the neuron u_j, and w_{ij} weight coefficients for input values from neurons u_i. Let $r_p, (p = 0, \ldots, m-1)$ be the expected values of the calculation with the input values $d_q, (q = 0, \ldots, n-1)$. Now, we can evaluate error values δ_p of output neurons u_p by equation

$$\delta_p = r_p - u_p. \tag{2}$$

The overall error E of the neural network can be expressed as [4]

$$E = E(W) = \frac{1}{2} \sum_{p=0}^{m-1} \delta_p^2, \tag{3}$$

where m is the number of output neurons.

Error of a weight coefficient w_{ij} is proportional to the partial derivative $\partial E / \partial w_{ij}$. Because the error E depends on the values of neurons u_i and the values of neurons depend on weights w_{ij}, partial derivatives can be calculated by using a chain rule [4]

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial u_j} \frac{\partial u_j}{\partial w_{ij}}. \tag{4}$$

We used the sigmoid function $f(x) = 1/(1 + e^{-\lambda x})$ as the activation function, where λ is called to be the gain factor of the sigmoid function. A reason to this was that the sigmoid function is continuous and it's derivative can be expressed as $f'(x) = f(x)(1 - f(x))$ [4]. The minimization of the error requires that the weights of a neuron are changed in the direction of the negative gradient component. Adjustments of the weights w_{ki} on the path from $k^t h$ to the $i^t h$ node can be expressed as [4]

$$\Delta w_{ki}^{l'} = \begin{cases} -\eta(\delta_i) u_i (1 - u_i) x_k & \textit{if} l' = 0 \\ -\eta u_i (1 - u_i) \sum_{j \in succ(i)} (\Delta_j \times w_{ij}) x_k & \textit{otherwise}, \end{cases} \tag{5}$$

where $\eta \in]0, \ldots, 1]$ is the learning constant, x_k is the value of $k^t h$ input of the neuron i, j refers the index of the successors of the neuron i, and $\Delta_j = \delta_i u_i (i - u_i)$.

VI. Experimental Results

We experimented our studies with a multilayer perceptron having two hidden layers. Section VI-A presents the preparation of the data. We sketch our results with the 1-Nearest Neighbour model in Section VI-B and with neural network in Section VI-C.

A. Preparation of Data

Initially, we had a series of data measured with 10 ms time interval. The data consisted of x-, y-, and z-acceleration values measured by accelerometers and x-, y-, and z-rotation acceleration values measured by gyroscopes deployed at the right and left hand of the patient. The overall number of data per a time step were 12 real values and a time stamp. We then classified activities of the patient in 15 classes, e.g. in "*Plays piano*", "*Walks*", and "*Plays puzzle*".

The overall number of measured instances after the initial classification was around 170'000 instances. In order to prepare the data for learning purposes of a neural network, we picked up every three instances belonging to the same activity class and having 100 ms time interval with each other and combined the data as a new learning instance. The procedure was applied over all the raw material. At the same time we omitted activity class "*Random*" (which mostly consisted of idle sensor data captured before the video recording). The remaining number of instances were around 122'000 instances. From this material we randomly picked up instances into a learning set, a testing set, and a validation set of a neural neural network, 40'000 instances in each. Activities we experimented consisted of the patient's usual daily activities:

TABLE II

CONFUSION MATRIX OF 1-NN ANALYSIS.

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	←Classified as
12474	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	a = ”*Plays a piano*”
0	30692	0	0	0	0	1	0	2	0	1	0	0	22	0	0	b = ”*Sits*”
0	0	2860	1	0	0	0	0	0	0	0	0	0	0	1	0	c = ”*Gets up*”
0	0	0	8401	0	0	0	0	0	0	0	0	0	0	0	0	d = ”*Walks*”
0	0	0	0	250	0	0	0	0	0	0	0	0	0	0	0	e = ”*Opens a door*”
0	1	0	0	0	2581	0	3	0	0	0	0	0	0	0	0	f = ”*Sits down*”
0	0	0	0	0	0	10065	0	0	0	0	0	0	0	0	0	g = ”*Shoots blowdart*”
0	4	0	0	0	0	0	30162	0	0	0	2	1	0	0	0	h = ”*Plays a puzzle game*”
0	0	0	0	0	0	0	0	6521	0	0	0	0	0	0	0	i = ”*Gives a puzzle*”
0	2	1	0	0	0	0	1	0	3966	0	0	0	0	0	0	j = ”*Points*”
0	1	0	1	0	0	0	1	0	0	6107	0	0	0	0	0	k = ”*Takes a puzzle*”
0	0	0	0	0	0	0	0	0	0	0	2820	0	0	0	0	l = ”*Drops a puzzle*”
0	0	0	0	0	0	0	0	0	0	0	0	1486	0	0	0	m = ”*Talks on a chair*”
0	31	0	0	0	0	0	0	3	0	0	0	0	4623	0	0	n = ”*Sits on a rockchair*”
0	0	0	0	0	0	0	0	0	0	0	0	0	0	400	0	o = ”*Spins a mover*”
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	180	p = ”*Puts away a puzzle*”

TABLE III

CONFUSION MATRIX OF NEURAL NETWORK ANALYSIS.

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	←Classified as	Accuracy
4118	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	a = ”*Plays a piano*”	100.0%
1	9578	0	3	2	5	5	113	31	8	51	28	1	62	0	1	b = ”*Sits*”	96.9%
0	8	918	24	0	1	0	2	0	0	5	0	0	0	3	0	c = ”*Gets up*”	95.5%
0	5	21	2564	0	0	26	7	0	0	1	0	0	1	7	8	d = ”*Walks*”	97.1%
0	0	1	14	57	0	0	0	0	0	0	0	0	0	8	0	e = ”*Opens a door*”	71.3%
33	30	10	27	0	617	22	72	6	2	21	2	0	4	2	0	f = ”*Sits down*”	72.8%
0	20	0	0	0	2	3081	8	2	0	5	0	0	0	2	0	g = ”*Shoots blowdart*”	98.8%
0	164	5	10	0	26	8	9391	65	77	79	37	3	26	6	0	h = ”*Plays a puzzle game*”	94.9%
0	72	0	0	0	22	36	107	1714	16	40	16	0	57	0	16	i = ”*Gives a puzzle*”	81.8%
0	17	0	0	0	1	5	56	13	1148	30	7	0	6	0	3	j = ”*Points*”	89.3%
14	42	0	0	0	24	2	142	16	4	1756	14	0	5	0	0	k = ”*Takes a puzzle*”	87.0%
0	15	72	34	0	36	0	84	10	1	28	644	0	6	6	0	l = ”*Drops a puzzle*”	68.8%
0	0	0	0	0	0	0	0	0	0	0	0	483	0	0	0	m = ”*Talks on a chair*”	100.0%
0	236	6	5	0	0	0	37	3	0	41	4	4	1129	2	0	n = ”*Sits on a rockchair*”	77.0%
0	1	1	8	0	0	0	1	0	0	0	0	0	0	99	0	o = ”*Spins a mover*”	90.0%
0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	46	p = ”*Puts away a puzzle*”	95.8%

a = ”*Plays a piano*”
b = ”*Sits*”
c = ”*Gets up*”
d = ”*Walks*”
e = ”*Opens a door*”
f = ”*Sits down*”
g = ”*Shoots blowdart*”
h = ”*Plays a puzzle game*”
i = ”*Gives a puzzle*”
j = ”*Points*”
k = ”*Takes a puzzle*”
l = ”*Drops a puzzle*”
m = ”*Talks on a chair*”
n = ”*Sits on a rockchair*”
o = ”*Spins a mover*”
p = ”*Puts away a puzzle*”

B. Results Using Nearest Neighbour Model

We experimented 1-nearest neighbour and J48 methods offered by the Weka data mining toolkit [3]. 10-fold cross validation was used with both methods. The 1-nearest neighbour algorithm provided the highest results correctly classifying with accuracy of nearly 100%. The classification rate of each action can be seen from Table II. The confusion matrix figured at Table II shows the classification rate of each activity in addition to how many times each activity was incorrectly classified as other activity.

The classification accuracy was almost 100% in most of actions. The high accuracy rate is probably due to suitably selected sample space and the limited number of test subjects. The slow and bare movements of the elderly might also contribute to easy classification. The actions ”*Sits*” and ”*Sits on a rockchair*” were most often confused among each other. This might be due to a very small difference between the forms of the activities. Even the smallest rocking motions were annotated which might result into an almost identical sensor output with sitting motion (especially if the annotation was start and end times were slightly inaccurate).

C. Results Using Neural Network

We used learning rate $\eta = 0.25$ in our experiments. A $\langle 36, 100, 100, 16\rangle$ multilayer perceptron having an input layer with 36 neurons, two hidden layers each having 100 neurons, and an output layer with 16 neurons were experimented. The overall number of weight coefficients of the network were $n_w = 15200$. The number of learning iterations using the learning set were 225. To avoid over-learning, we tested the network after each 15 learning iterations using the validation set. If the previously calculated total error of the network were bigger that the new one, the new one was saved and learning were continued. On the other hand, if the previously calculated total error were less that the new one, the weight coefficients after the previous (best) calculations were restored and the learning phase were restarted. At the end of the execution the network were evaluated using the evaluation set. The confusion matrix figured at Table III shows the classification rate of each activity in addition to how many times each activity was incorrectly classified as other activity.

VII. Conclusion and Future Work

In this work, we have compared two classification methods for elderly patient's activity recognition. We presented an experimental sensor network based data acquisition model to capture elderly patients' motion data. We then introduced our classification models which were utilized for the captured data. The basic models were the 1-nearest neighbour model and a neural network model. We achieved an overall accuracy of nearly 100% with the 1-nearest neighbour model and 93.4% with the neural network we used in our experiments.

The main drawback of this study is the small amount of test data; the data of only one patient was analyzed. In future work, a multiple amount of similarly captured memory disorder patient data is to be analyzed. With multiple patient data we can test whether our model can be generalized among many elderly patiens. Larger amount of data also enables experimentation among different datasets containing similar activities recorded at different times. This can be used to test whether one testing set can be used among multiple patiens. Our test results also excite the interest to compare classification accuracy between patiens of different ages.

With raw annotated sensor data, it is rather quick to perform different measurements with Weka classifiers, such as the 1-NN. If features and their calculation methods are known, preparation of a dataset and its classification can be performed within tens of minutes. The actual classification of a new dataset lasts several seconds. A drawback of the neural network based classification is the large time consumption during a learning phase of a neural network. In our experiments, the time consumption of a $\langle 100, 100, 16 \rangle$ was more than two hours. Concerning the neural network method, a subject of our future work is parallelization of backpropagation algorithm using graphics processing unit having hundreds of processing cores.

In this work we have compared 1-NN and neural network based classifiers. Extending the comparison to newer methods such as ensemble learners and support vector machines is another direction of our future work.

Acknowledgements

This research is funded by the Academy of Finland VESC and P-SESC projects. The authors would also like to thank the staff and participants from ODL Veljeskoti Oulu.

References

[1] Ling Bao and Stephen S. Intille. Activity recognition from user-annotated acceleration data. In *Proceedings of the Second International Conference on Pervasive Computing,, PERVASIVE'04*, pages 1–17, 2004.

[2] A. Eskandarinia, H. Nazarpour, M. Teimouri, and M. Z. Ahmadi. Comparison of neural and k-nearest neighbor methods in daily flow forecast. *Journal of Applied Sciences*, 10(11):1006–1010, 2010.

[3] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The weka data mining software: An update. *SIGKDD Explorations*, 11(1), 2009.

[4] George F. Luger. *Artificial Intelligence*. Pearson Education, Inc., 2009.

[5] M. O'Farrell, E. Lewis, C. Flanagan, W. Lyons, and N. Jackman. Comparison of k-nn and neural network methods in the classification of spectral data from an optical fibre-based sensor system used for quality control in the food industry. *Sensors and Actuators B: Chemical*, 111-112:354 – 362, 2005.

[6] Ren Ohmura, Futoshi Naya, Haruo Noma, and Kiyoshi Kogure. B-pack : A bluetooth-based wearable sensing device for nursing activity recognition. *Proceedings of the* 1^{st} *International Symposium on Wireless Pervasive Computing, Jan. 2006*, 2006.

[7] Susanna Pirttikangas, Kaori Fujinami, and Tatsuo Nakajima. Feature selection and activity recognition from wearable sensors. In *Proceedings of International Symposium on Ubiquitous Computing Systems (UCS2006)*, 2006.

[8] Matti Pouke, Seamus Hickey, Tomohiro Kuroda, and Haruo Noma. Activity recognition of the elderly. In *Proceedings of the 4th ACM international Workshop on Context-Awareness For Self-Managing Systems*, pages 46–52, 2010.

[9] Nishkam Ravi, Nikhil D, Preetham Mysore, and Michael L. Littman. Activity recognition from accelerometer data. In *Proceedings of the Seventeenth Conference on Innovative Applications of Artificial Intelligence(IAAI)*, pages 1541–1546. AAAI Press, 2005.