ORIGINAL PAPER

CrossMark

Human activity classification using long short-term memory network

Anuradhi Malshika Welhenge[1] · Attaphongse Taparugssanagorn[1]

Abstract
Activities of daily living (ADL) can be used to identify a person's daily routine which helps health professionals to provide preventive healthcare. Classification of ADLs is therefore very important. In this study, long short-term memory (LSTM) network, which is an extension of recurrent neural networks, is used. Data collected in MobiAct data set are used to train and test the network. An accuracy of 0.90 is achieved using LSTM network.

Keywords Deep learning · ADL · LSTM · RNN

1 Introduction

Activities of daily living (ADL) classification are used to identify a person's activity level and also used to detect falls since most of the hospitalized elderly people were admitted to hospitals due to falls. By identifying a person's activity level, early diagnosis of some diseases can be made. Several studies were conducted to develop classification algorithms for daily activities using accelerometer data. This can be used in context awareness sensing or fitness and health tracking. Context aware applications can customize their behavior based on the current activity.

A triaxial accelerometer attached to the waist was used by Mathie et al. [1] to classify movements. The algorithm was structured as a hierarchical binary tree. Initially signals from the accelerometer were divided into activities and rest. Activities were further divided into walking, changes in orientation and falls while rest was further sub-divided into standing, sitting, and lying. In the studies conducted using 26 subjects, sensitivity and specificity of 97.7% and 98.7% were achieved. In this algorithm, activities and rest were identified calculating the energy expenditure and comparing that to a threshold. If the value exceeded the threshold, the signal was classified as an activity or else as a rest. Tilt angles of a subject were used to find whether the subject was upright or lying. In the cases where this algorithm could not achieve

good accuracy a rule-based method, which used the tilt angle, duration, energy expenditure, past and future activities, was used to find the probability of standing and sitting. Falls were also classified according to a similar rule-based method.

A kinematic sensor attached to the chest was used in a system developed by Najafi et al. [2] to detect sitting, standing, lying, and walking. This method was tested on community dwelling elderly persons in a gait laboratory and during their normal physical activities as well as on hospitalized elderly persons. The system was able to identify with a high rate of accuracy of the following actions: sit to stand, stand to sit, 62 transfers from bed, 144 posture changes to left, right, back and ventral as well as walking. The device consists of a gyroscope and two accelerometers to measure angular velocity as well as front and vertical accelerations of the trunks, respectively.

Allen et al. [3] investigated two classification methods based on Gaussian mixture model (GMM) and a heuristic rule-based method to identify standing, sitting, lying, and walking using a triaxial accelerometer attached to the waist. Because of the limitation of user data, GMMs can be used to resolve this issue by adapting feature extraction to the GMM to separate gravitational and body components from the acceleration data. These data as well as delta coefficients derived from gravitational and body components as well as energy expenditure were used as features for the GMM. While heuristic rule-based methods had problems in identifying sitting and standing as well as the transition between the two positions, GMM method also had problems in identifying the standing.

✉ Anuradhi Malshika Welhenge
  st118194@ait.asia

  Attaphongse Taparugssanagorn
  attaphongset@ait.asia

[1] Asian Institute of Technology, Khlong Luang, Thailand

In another study, Karantonis et al. [4] implemented a waist mounted device for real-time activity classification using tri-axial accelerometry. This can also identify activity, rest, and postural orientation. In this algorithm, signal magnitude area (SMA) was used to distinguish between activity and rest while the tilt angle was used to classify upright, lying or inverted positions. signal magnitude vector (SVM) which was derived from the accelerometer data and identified possible fall events. If a frequency peak fell between 0.7 and 3 Hz of the fast Fourier transform (FFT) of z axis data, then the upright active was classified as walking. This calculation was done in the computer, as opposed to others, which were done on the wearable device itself.

Various studies have been conducted to design a method to detect falls in real time. If a fallen person can get assistance quickly, then it may help to reduce some of the effects from the fall. In the case the fallen person is unconscious, these devices may help bring quick assistance.

Lindemann et al. [5] developed a fall detector using accelerometers that can be attached to a hearing aid behind the ear. This algorithm can detect ADLs and falls. Sensitivity and specificity of the device were accessed using acceleration patterns and ADLs of one young volunteer and one 83-year-old volunteers. The algorithm used an estimated velocity and measured accelerations to detect the falls. This algorithm showed a false positive only when the hearing aid was hit by the hand. Otherwise it was able to identify falls events and normal activities.

Bourke et al. [6] used accelerometers attached to the thigh and trunk to identify supervised simulated falls and ADL. 10 young people were used for the supervised falls and these falls simulate 8 common types of falls of older persons. To detect ADL, 10 elderly persons were used. To classify falls, thresholds were derived from the acceleration data of the simulated falls. When using the upper threshold of the fall in the trunk, all ADL tasks were correctly classified. From that result, they decided that the most suitable place to attach the sensor was the trunk.

Kangas et al. [7] investigated various fall detection algorithms with accelerometers attached to waist, wrist, and head. Using data from ADLs as a reference, three algorithms were used on different phases of a fall. The results showed that good results can be achieved if the accelerometer was worn on head or waist. They concluded that a waist worn device which can identify the impact phase and the lying phase of the fall was ideal for fall detection.

Zheng [8] used a single triaxial accelerometer and a hierarchical scheme as the recognition method. Least squares support vector machines (LS-SVM) and Naive Bayes (NB) algorithm was used as the classifiers. He achieved an accuracy rate of 95.6% for recognition of 10 activities. A Doppler radar system was developed with Hidden Markov models to classify ADLs in an unsupervised manner [9]. An accuracy of 89% was obtained for 6 features.

SmartStep, which is an insole-based ADL classification system, was used with a wrist worn device to compare the accuracy [10]. It achieved a high score for perceived comfort since an insole-based system does not hinder the daily activities. A-Wristocracy, which is a system developed by Vepakomma et al. [11], uses artificial neural networks to classify home activities using a wrist worn device. While it did not use video imaging, feature set was extracted using multimodal sensing suites. It classifies 22 activities with 90% or more accuracy. Ando et al. [12] developed a multisensor approach to identify critical activities of elderly people and persons with neurological pathologies. It combines data from a gyroscope and an accelerometer. This system achieved a specificity of 0.98 while sensitivity is 0.81. K nearest neighbor algorithm was utilized in Physical Activity Classification Algorithm (PAC) which classified 13 ADLs using multiple sensors attached to different body locations [13].

Another approach is gait pattern identification. Several studies used extreme machine learning (ELM) and ANN for the pattern identification. These are also helpful in preventing falls in the elderly [14,15].

Using deep learning methods to classify ADLs is a cutting edge technique. When performing recognition tasks, it is important to identify temporal correlations within the input data. This problem can be addressed using convolutional neural networks (CNNs). However, CNNs have a limitation to capture dependencies within input data. To capture long-range dependencies, deep recurrent neural networks (DRNNs) were proposed by Murad et al. [16]. They presented unidirectional, bidirectional, and cascaded architectures based on long short-term memory (LSTM) DRNNs evaluated their effectiveness on miscellaneous benchmark datasets, and showed superior performance. These deep learning techniques were used in other area, such as to predict protein secondary structure, backbone angles, contact numbers, and solvent accessibility [17]. LSTM RNNs were also used to predict remaining useful life of lithium-ion batteries [18]. In this paper, ADLs are classified using LSTM, which is part of RNN.

The remainder of the paper is organized as follows: Sect. 2 is devoted for the background of this study including deep learning and its application in the field of ADL classification. The methodology is described in Sect. 3. Experiments and the corresponding results are reported and analyzed in Sect. 4. Finally, the conclusion is drawn in Sect. 5.
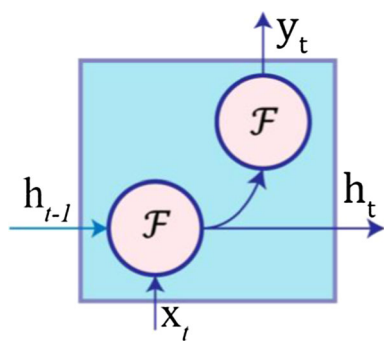
Fig. 1 Architecture of RNN process [16]

2 Background

2.1 Deep learning

Deep learning methods can extract discriminating features from data. Since processing capabilities have also increased complex data analysis can be done in real time. Deep learning algorithms can be supervised, semi-supervised or unsupervised. Artificial neural networks, deep belief networks, deep neural networks, recurrent neural networks (RNN) are some of the algorithms associated with deep learning. These algorithms are used in bioinformatics, speech processing, computer vision, and in many other applications.

2.2 Long short-term memory

Long short-term memory (LSTM) is an extension of RNNs. RNNs differ from the traditional neural networks in which their inputs and outputs are assumed to be independent of each other. But in RNNs it has a memory which can remember previous computations. There are multiple nodes in a hidden layer in RNNs as shown in Fig. 1. Each node calculates current hidden state h_t and output y_t using previous hidden state h_{t-1}, and input x_t as

$$h_t = F\left(W_h h_{t-1} + U_h x_t + b_h\right), \tag{1}$$

$$y_t = F\left(W_y h_t + b_y\right), \tag{2}$$

where weight for the hidden to hidden recurrent connection, input to hidden connection, and hidden to output connection are denoted by W_h, U_h and W_y, respectively. b_h and b_y are bias terms for hidden and output states. F is an activation function which is a nonlinear function and is chosen from hyperbolic tangent, sigmoid or rectified linear unit [16].

LSTMs typically have a similar architecture, but they use a different function to calculate hidden state and memory in LSTMs are called cells. These cells take previous hidden state and current input and decide what to keep. In addition, these cells can capture long-term dependencies.
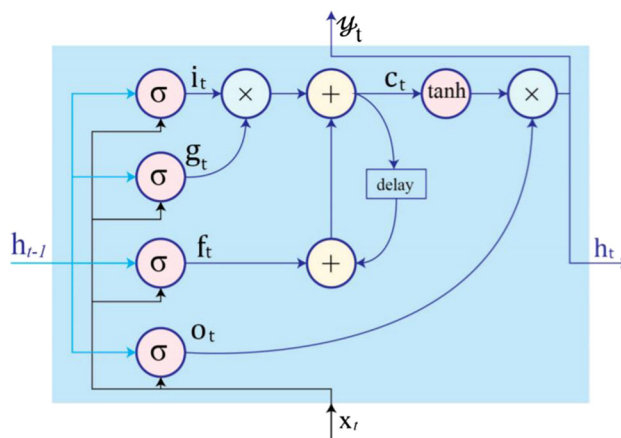

Fig. 2 Architecture of LSTM process [16]

In Fig. 2, the architecture of LSTM process is shown. f_t is the forget gate, i_t is the input gate, o_t is the output gate, and g_t is the input modulation gate. c_t and h_t are internal state and hidden state, respectively. b, U, and W are learning parameters of cell gates. Subscripts of these learning parameters denote the cell gate. b_f, U_f, W_f are learning parameters of forget gate, b_i, U_i, W_i are learning parameters of input gate, b_o, U_o, W_o are learning parameters of output gate, and b_g, U_g, W_g are learning parameters of input modulation gate. They are, in detail, expressed as

$$f_t = \sigma\left(b_f + U_f x_t + W_f x_{t-1}\right), \tag{3}$$

$$i_t = \sigma\left(b_i + U_i x_t + W_i h_{t-1}\right), \tag{4}$$

$$o_t = \sigma\left(b_o + U_o x_t + W_o h_{t-1}\right), \tag{5}$$

$$g_t = \sigma\left(b_g + U_g x_t + W_g h_{t-1}\right), \tag{6}$$

$$c_t = f_t c_{t-1} + g_t i_t, \tag{7}$$

$$h_t = \tanh(c_t) o_t, \tag{8}$$

where σ is the sigmoid function and $\tanh(\cdot)$ is the hyperbolic tangent function.

3 Method

In this study, acceleration data from 10 subjects are used in a LSTM network to classify 5 different activities, i.e., standing, walking, jogging, jumping, and climbing stairs. Initially a third-order median filter is used on the acceleration signals to remove abnormal noise spikes. Acceleration signals consist of gravitational component and body component. To separate gravitational component from body component, low-pass filtering is done. Low-pass filtered signal is the gravitational component [gravitational acceleration (GA)] and by

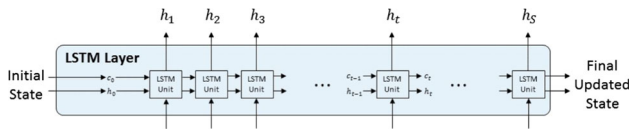**Fig. 3** Block diagram of the classification process [19]

Fig. 4 Block diagram of the LSTM layer [19]

subtracting that signal from raw acceleration (RA) signal we get the body component [body acceleration (BA)] as

$$BA = RA - GA. \tag{9}$$

These body components are used to calculate the signal magnitude area (SMA) as

$$SMA = \frac{1}{T} \left(\int_0^T x(t) \, dt + \int_0^T y(t) \, dt + \int_0^T z(t) \, dt \right), \tag{10}$$

where T is the length of the acceleration signal.

After that zero crossing rates for each axis of acceleration are calculated. These rates along with SMAs are used as features for the input to the LSTM. It is specified with a output size of 400, number of classes as 5 and to output the last of the sequence. Training options used are 300 maximum epochs, mini batch size of 10 and is trained using stochastic gradient descent with momentum. These parameters are chosen using trial and error method. The network is trained on 5 activities of 20 subjects, namely standing, climbing stairs, jogging, jumping and walking. Finally the network is tested on 5 activities from another 20 subjects and the accuracy is calculated.

A sequence input layer inputs sequence into the network. An LSTM layer learns long-term dependencies between time steps of sequence data. As shown in Fig. 3 to predict class labels network has a fully connected layer, softmax layer and classification output layer. In Fig. 4 block diagram of the LSTM layer is shown. h denotes the hidden state and c denotes the cell state of each LSTM unit.

4 Experiments and results

4.1 Datasets

The data used in this study are from MobiAct data set which is publicly available [20]. Data are collected from a smartphone while subjects are performing different types of activities. A

Samsung Galaxy S3 device with the LSM330DLC inertial module was used to capture the motion data. The gyroscope was calibrated prior to the recordings using the device's integrated tool. For the purpose of data capturing, an Android application was developed that records raw data for acceleration, angular velocity and orientation with the enabled parameter SENSOR_DELAY_FASTEST. This provides the highest possible sampling rate.

In an attempt to simulate daily usage of mobile phones, the device was located in a trouser pocket freely chosen by the subject in any random orientation. For the falls, the subjects used the pocket on the opposite side of the falling direction. Each sample is stored along with its time stamp in ns.

For the generation of the MobiAct dataset 57 subjects (42 men and 15 women) were recorded while performing the predefined activities. The subjects' age spanned between 20 and 47 years (average: 26), the height ranged from 160 to 189 cm (average: 175), and the weight varied from 50 to 120 kg (average: 76). 50 subjects completed successfully all ADLs and 54 subjects completed all falls. In total, 10 trials had to be removed from the dataset due to errors in acquisition.

4.2 Result and discussion

An accuracy of 0.90 is achieved using above LSTM settings. The results are shown in Table 1.

The network is trained by initially dividing the data set in to small groups called mini batches. In the table mini batch loss is the difference between predicted value and the true value and mini batch accuracy is the percentage accuracy of the predicted value compared to the true value. Base learning rate is the speed by which the network is trained. Accuracy of training and loss of training, and the confusion matrix are shown in Figs. 5 and 6, respectively. In Fig. 5, light line is the training accuracy which is the classification accuracy on each individual mini batch. Dark line is the smoothed training accuracy, obtained by applying a smoothing algorithm to the training accuracy. It is less noisy than the unsmoothed accuracy, making it easier to spot trends. Similarly in the graph below light line is the loss and dark line is the smoothed loss. The figure marks each training Epoch using a shaded background. An epoch is a full pass through the entire data set. In the confusion matrix, the classification of target class versus output class is shown. In the figure numbers 1, 2, 3, 4 and 5 are standing, climbing stairs, jogging, jumping, and walking, respectively. When classifying there is a small confusion between jogging and jumping which are similar activities. This method was then compared to the Gaussian mixture model (GMM) classifier. For the same dataset, GMM recorded an accuracy of 82% which is lower than the accuracy recorded by LSTM.

Table 1 The results of LSTM showing time elapsed, mini batch loss, mini batch accuracy, and base learning rate

Epoch	Iteration	Time elapsed (ns)	Mini batch loss (m/s)	Mini batch accuracy (m/s) (%)	Base learning rate
1	1	0.02	1.6087	100.00	0.0100
10	50	1.39	1.6113	0.00	0.0100
20	100	2.83	1.1434	70.00	0.0100
30	150	4.30	0.7378	20.00	0.0100
40	200	5.74	0.7232	40.00	0.0100
50	250	7.15	0.7085	50.00	0.0100
60	300	8.62	0.6965	50.00	0.0100
70	350	10.12	0.6867	60.00	0.0100
80	400	11.60	0.6786	60.00	0.0100
90	450	13.08	0.6709	60.00	0.0100
100	500	14.54	0.6629	60.00	0.0100
110	550	16.00	0.6532	60.00	0.0100
120	600	17.45	0.6409	60.00	0.0100
130	650	18.93	0.6269	70.00	0.0100
140	700	20.40	0.5922	70.00	0.0100
150	750	21.87	0.5870	60.00	0.0100
160	800	23.34	0.5312	70.00	0.0100
170	850	24.83	0.5719	70.00	0.0100
180	900	26.30	0.5849	60.00	0.0100
190	950	27.77	0.5771	60.00	0.0100
200	1000	29.26	0.8577	40.00	0.0100
210	1050	30.74	0.6213	60.00	0.0100
220	1100	32.21	0.5168	60.00	0.0100
230	1150	33.69	0.3995	80.00	0.0100
240	1200	35.16	0.7250	70.00	0.0100
250	1250	36.64	0.4395	80.00	0.0100
260	1300	38.14	1.2448	40.00	0.0100
270	1350	39.64	1.0723	40.00	0.0100
280	1400	41.12	0.8563	50.00	0.0100
290	1450	42.60	0.2744	90.00	0.0100
300	1500	44.09	0.8109	70.00	0.0100
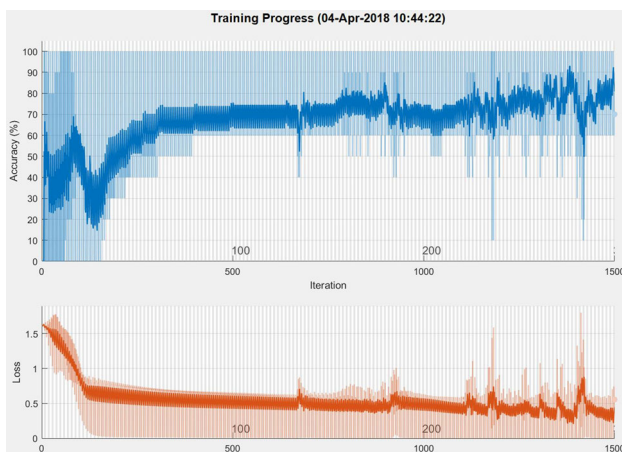
Fig. 5 Accuracy of LSTM process

5 Conclusion

In this study, we classified five activities namely standing, climbing stairs, jogging, jumping, and walking, for 20 subjects using trained LSTM network, which was trained using another 20 subjects. This classifier can be used to identify the context of the user and the identified context can be used in context-based applications. We used SMAs and zero crossing rates were used as input features to the LSTM network making this simpler. In addition, using only 4 features is sort of a benefit in terms of computation time. This study achieves 0.90% accuracy although a very few features are used as inputs. A-Wristocracy, also achieved a 90% accuracy while the Doppler radar system developed with Hidden Markov models to classify ADLs in an unsupervised manner
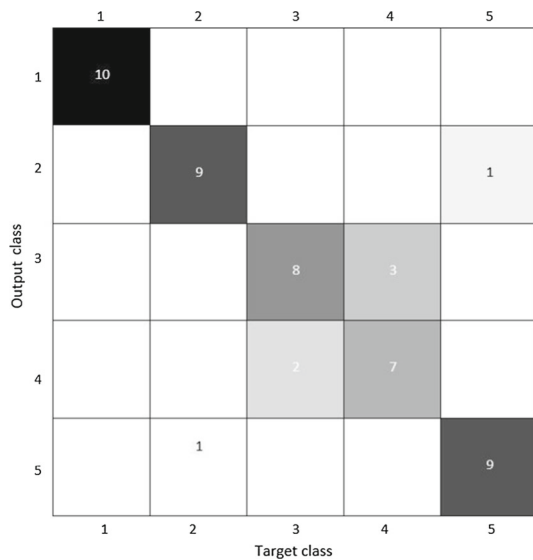
Fig. 6 Confusion matrix of the dataset

achieved an accuracy of 89% for 6 features. Chen et al. also used LSTM to classify activities and achieved an accuracy of 92%; however, they used the entire acceleration signal as the input [21]. In this case, we used features extracted from the acceleration signal since it will reduce the computation time. When classifying there is some confusion between jogging and jumping. This is mainly due to the fact that they are similar activities. As a conclusion, our proposed method outperforms some other methods proposed in the mentioned literature.

References

1. Mathie, M.J., Coster, A.C.F., Lovell, N.H., Celler, B.G., Lord, S.R., Tiedemann, A.: A pilot study of long-term monitoring of human movements in the home using accelerometry. J. Telemed. Telecare **10**, 1–8 (2004)
2. Najafi, B., Aminian, K., Paraschiv-Ionescu, A., Loew, F., Bla, C.J., Robert, P.: Ambulatory system for human motion analysis using a kinematic sensor: monitoring of daily physical activity in the elderly. IEEE Trans. Biomed. Eng. **50**, 711–723 (2003)
3. Allen, F.R., Ambikairajah, E., Lovell, N.H., Celler, B.G.: Classification of a known sequence of motions and postures from accelerometry data using adapted Gaussian mixture models. Physiol. Meas. **27**, 935–951 (2006)
4. Karantonis, D.M., Narayanan, M.R., Mathie, M., Lovell, N.H., Celler, B.G.: Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring. IEEE Trans. Inf. Technol. Biomed. **10**, 156–167 (2006)
5. Lindemann, U., Hock, A., Stuber, M., Keck, W., Becker, C.: Evaluation of a fall detector based on accelerometers: a pilot study. Med. Biol. Eng. Comput. **43**, 548–551 (2005)
6. Bourke, A.K., Brien, J.V.O., Lyons, G.M.: Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm. Gait Posture **26**, 194–199 (2007)
7. Kangas, M., Konttila, A., Lindgren, P., Winblad, I., Jämsä, T.: Comparison of low-complexity fall detection algorithms for body attached accelerometers. Gait Posture **28**, 285–291 (2008)
8. Zheng, Y.: Human activity recognition based on the hierarchical feature selection and classification framework. J. Electr. Comput. Eng. **2015**, 9 (2015)
9. Li, W., Xu, Y., Tan, B., Piechocki, R.J.: Passive wireless sensing for unsupervised human activity recognition in healthcare. In: 2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC), pp. 1528–1533 (2017)
10. Hegde, N., Bries, M., Swibas, T., Melanson, E., Sazonov, E.: Automatic recognition of activities of daily living utilizing insole based and wrist worn wearable sensors. IEEE J. Biomed. Health Inform. **99**, 1 (2018)
11. Vepakomma, P., De, D., Das, S.K., Bhansali, S.: A-Wristocracy: deep learning on wrist-worn sensing for recognition of user complex activities. In: 2015 IEEE 12th International Conference on Wearable and Implantable Body Sensor Networks (BSN), pp. 1–6 (2015)
12. And, B., Baglio, S., Lombardo, C.O., Marletta, V.: A multisensor data-fusion approach for ADL and fall classification. IEEE Trans. Instrum. Meas. **65**, 1960–1967 (2016)
13. Awais, M., Palmerini, L., Chiari, L.: Physical activity classification using body-worn inertial sensors in a multi-sensor setup. In: 2016 IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI), pp. 1–4 (2016)
14. Semwal, V.B., Gaud, N., Nandi, G.C.: Human gait state prediction using cellular automata and classification using ELM. MISP, vol. 748, pp. 135–145 (2017)
15. Semwal, V.B., Raj, M., Nandi, G.C.: Biometric gait identification based on a multilayer perceptron. Robot. Auton. Syst. **65**, 65–75 (2015)
16. Murad, A., Pyun, J.Y.: Deep recurrent neural networks for human activity recognition. Sensors **17**(11), 2556 (2017). https://doi.org/10.3390/s17112556
17. Heffernan, R., Yang, Y., Paliwal, K., Zhou, Y.: Capturing non-local interactions by long short term memory bidirectional recurrent neural networks for improving prediction of protein secondary structure, backbone angles, contact numbers, and solvent accessibility. Struct. Bioinform. **33**, 2842–2849 (2017)
18. Zhang, Y., Xiong, R., He, H., Pecht, M.: Long short-term memory recurrent neural network for remaining useful life prediction of lithium-ion batteries. IEEE Trans. Veh. Technol. **99**, 1 (2018)
19. Long short-term memory networks. https://www.mathworks.com/help/nnet/ug/long-short-term-memory-networks.html (2018). Accessed 6 Sep 2018
20. Vavoulas, G., Chatzaki, C., Malliotakis, T., Pediaditis, M., Tsiknakis, M.: The MobiAct dataset: recognition of activities of daily living using smartphones. In: International Conference on Information and Communication Technologies for Ageing Well and e-Health, pp. 143–151 (2016)
21. Chen, Y., Zhong, K., Zhang, J., Sun, Q., Zhao, X.: LSTM networks for mobile human activity recognition. In: International Conference on Artificial Intelligence: Technologies and Applications (ICAITA) (2016)