Proceedings of the 4th International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/54/20241421

A comparative study of the deep learning based model and the conventional machine learning based models in human activity recognition

Shaoyang Wang

High School Affiliated to Fudan University, Shanghai, 200433, China

oriolehuangsh@hotmail.com

Abstract. Human activity recognition (HAR) has been widely studied as a research field in human behavior analysis due to its huge potential in various application domains such as health care and behavioral science. Recently, deep learning (DL) based methods have also been successfully applied to predict various human activities. This research aims at building different Python-based models to perform HAR using smartphones and calculating and comparing the accuracy of the models to select the optimal one. Four models were built to classify and predict human activities: Deep Convolutional Neural Network (DCNN), Support Vector Machine (SVM), Decision Tree (DT), and Random Forest (RF). The results of the experiments in this paper show that the Deep Convolutional Neural Network achieves an average recognition accuracy rate of 95.49%, exceeding the other three models. The underlying reason may be that Deep Convolutional Neural Network is based on a more advanced algorithm — deep learning technique.

Keywords: Deep Learning, SVM, Decision Tree, RF, HAR

1. Introduction

Human activity recognition, as a research direction in the field of artificial intelligence, has attracted increasing attention with the development of society and the improvement of living standards. So far, various models have been applied to human activity recognition. The dynamic time-warping (DTW) model is the easiest because it does not require training or huge quantities of data, but the recognition accuracy is low when the data has multiple classifications [1]. Apart from DTW and other manual feature extraction methods, large varieties of machine learning techniques have played important roles in inferring activity details, including SVM, the Hidden Markov Model, etc. However, in these traditional machine learning methods, the technology of extracting relevant features cannot adapt well to complex activities. This maladaptation is particularly prominent in the context of multimodal and high-dimensional sensor data influx [2]. They can recognize simple activities well, such as walking. But for more complex activities, such as drinking tea, it is difficult to infer [3]. To overcome this drawback, in recent years, deep learning has been increasingly employed in this field and has demonstrated increased computational powers. In fact, deep-learning-based algorithms are capable of learning relevant features automatically and extracting and classifying features of various human activities efficiently. While multiple models have been proposed to classify and predict human activities, few of them have been

DOI: 10.54254/2755-2721/54/20241421

compared systematically in terms of their accuracy rates. Therefore, the current research further explores the working mechanisms of the Deep Convolutional Neural Network, SVM, DT, and RF models and calculates their respective accuracy rates in recognising six daily activities, including walking, walking upstairs, walking downstairs, sitting, standing, and lying. Research in this domain may lead to the hypothesis that Deep Convolutional Neural Network will produce the highest accuracy rate among the four models because it is based on deep learning algorithm, while the others are all powered by conventional machine learning techniques.

2. Literature review

Nurwulan and Selamaj compared the performance of DTs in human activity recognition. In comparison, they used acceleration and jerk data. The conclusion showed that RFs have better performance than other decision tree models in identifying human activity patterns. The underlying reasons may be that it combines the methods of bootstrap aggregation and randomization in the selection of data node segmentation and therefore improves classification accuracy [4]. Nurwulan and Selamaj also compared RF with other classifiers. The results showed that RF was indeed superior in accuracy to other machine learning technologies, but more time-consuming than other models [5]. Studies have also been performed on SVMs that can solve both linear and non-linearity classification problems by changing the kernel function. In the study on child activity recognition, Nam and Park used the accelerator sensor and the pressure sensor to collect samples and achieved high accuracy in data classification with SVMs [6]. Recent research has directed more attention to deep-learning models, particularly CNNs. Ronao et al. compared one-dimensional CNN with traditional machine learning models such as SVM and DT. This comparison is to classify the activity data recorded by smartphone sensors. In the end, they concluded that the CNN model is definitely more accurate [7]. Zebin et al. compared two-dimensional CNN with traditional machine learning methods. This comparison is to classify the six daily activities recorded by 12 volunteers in terms of accuracy and computational costs. The results showed improvements on both dimensions [8]. Attempts have also been made to improve and supplement the functions of CNNs in multiple dimensions. Sajjad et al. analysed the facial expressions to predict human behaviours using the light-weight CNN, and the experimental evaluations demonstrated the superior performance of CNN for both facial expression recognition and human behaviour understanding [9].

3. Methods

3.1. Data sets

The data sets in the present research are retrieved from the website: UC Irving Machine Learning Repository [10]. This website is sponsored by the University of California, Irvine, and can be openly accessed. To downscale the search range, the author only selected the data sets that can do classification and have more than 100 attributes and more than 1000 instances.

After reading and comparing the available data sets, the author finally chose the data set: Human Activity Recognition Using Smartphones. This database was established based on the records of 30 participants engaged in daily life activities. They carry waist smartphones equipped with embedded inertial sensors during activities. The thirty volunteers of the experiment, with smartphones on the wrist, randomly performed six daily activities. The embedded accelerometer and gyroscope captured 3-axial linear acceleration and 3-axial angular velocity at a constant rate of 50Hz. Also, the experiment recorded a feature vector with time and frequency domain variables. This data set has 10299 instances and 70% of the instances were selected as training data and 30% of the instances as test data randomly.

The data preprocessing provides high-quality data for subsequent data analysis and modeling. After dropping duplicate values and missing values, it is also necessary to drop attributes unrelated to the result. Among the 7352 instances, over 6000 instances have values within the range of -0.10 and 0.00. The value of the attribute fluctuates slightly but ends in different results. Therefore, this attribute has little correlation with the result and should be removed to improve the accuracy of the prediction model.

DOI: 10.54254/2755-2721/54/20241421

Consequently, the author dropped all the attributes with instances, which take up a proportion of more than 85%.

3.2. Model architecture

CNN is a multi-layer neural network composed of overlapping convolutional layers for feature extraction and subsampling layers for feature processing. The convolutional layer is comprised of several feature graphs, each with neurons sharing the parameters of the same convolution kernel, obtained from convolutionally checking the input images of the previous layer. Each element in the convolution kernel functions as a weight parameter, coupled with the pixel value of the relevant block of the input image, and then the product is summed with the output pixel obtained via the activation function. It is essentially equivalent to the process of weighting and summing multiple input signals onto a neuron and then activating the output. In fact, CNNs directly employ raw data as input and automatically learn feature manifestations from large quantities of training data.

SVM is a set of guided learning models. It is used for analyzing data and identifying patterns for classification and regression analysis. SVMs take the training data set and predict for each input given. Because SVM is a binary linear non-probabilistic classifier, an SVM training algorithm constructs a model that predicts the category new data are classified into. While the original datasets may contain large quantities of examples, the number of support vectors is small. SVMs can be used for large-scale data.

Decision Tree model is a machine learning technique that can classify dependent variables into varioys groups and recognize major independent variables. For large datasets, this model is efficient in building as it does not require a lengthy training process. It is a supervised classification model. Non terminal nodes display attribute checks, while terminal nodes display judgment results. This is its basic structure.

Random forest is a supervised method. This method will construct a large number of unrelated decision trees within the training time. Each tree in RF can only select a random subset of features. This is different from other decision tree classifiers. This also increases the variation between trees in the model. Due to the low correlation between trees, classification will achieve higher accuracy. But its efficiency is low because a large number of trees can slow down the speed of real-time prediction.

3.3. Building models and modification

The first step in building a CNN model is choosing activation functions that fit the data. If all the layers are fc layers (linear layers), the whole process will become a linear transformation with no need to build multiple layers. To avoid this problem, it needs to add activation functions into the neural network to create a nonlinear connection between two layers so that the data can be fitted better. There are three activation functions that are generally used: the sigmoid function, the tanh function and the ReLU (Rectified Linear Unit) function. The sigmoid function is used for hidden layer neuron output with the value range of (0, 1). It can map a real number to the interval (0, 1). It can be used for binary classification. The graph and the derivative of the sigmoid function are shown in Figures 1 and 2 below.

Sigmoid function: $\sigma(x) = \frac{1}{1+e^{-x}}$

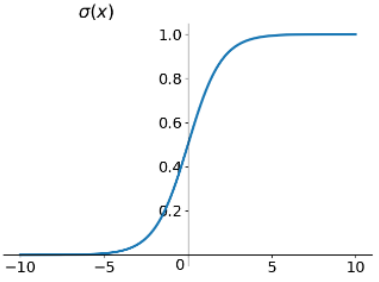

Figure 1. Sigmoid function

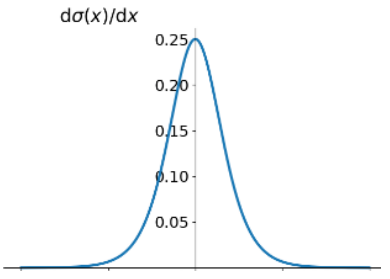

Figure 2. Derivative of sigmoid function

DOI: 10.54254/2755-2721/54/20241421

The function has a problem called gradient vanishing. The optimization method of a neural network is the back propagation of derivatives. This model first calculates the loss corresponding to the output layer. Then it continuously transmits the losses to the upper layer network in the form of derivatives and modify the parameters to reduce losses. The sigmoid function often causes the derivative to become close to zero. This situation makes it impossible to update the parameters, and therefore the neural network cannot be optimized. Also, the function produces an output from 0 to 1, and the output represents the probability of belonging to a category. So, it can only be used in a binary classification project. Thus, the sigmoid function should be excluded.

The tanh function is used for hidden layer neuron output with a value range of (-1, 1). It can map a real number to the interval (-1, 1). The graph and the derivative of the tanh function are shown in Figures 3 and 4 below.

Tanh function: $\tanh(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$

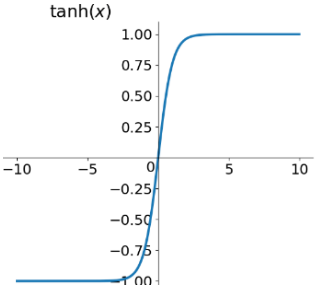

Figure 3. Tanh function

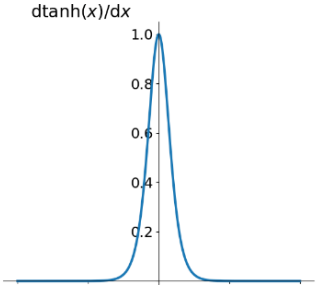

Figure 4. Derivative of tanh function

The tanh function is also used in binary classification projects. Regardless of whether the input is large or small, the output is almost the same. Therefore, its gradient is small, which is not conducive to weight updating. Therefore, the function is also not appropriate for the multi-classification project.

Positive and negative numbers are both changed to zero by the piecewise linear ReLU function. Unilateral inhibition results in sparse activation of neurons in the neural network. The graph and the derivative of the ReLU function are shown in Figure 5 and Figure 6 below.

ReLU function: $ReLU(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases}$

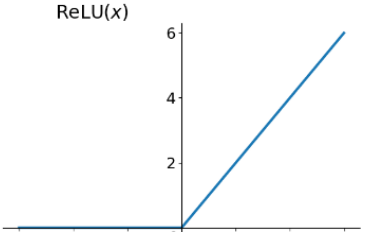

Figure 5. ReLU function

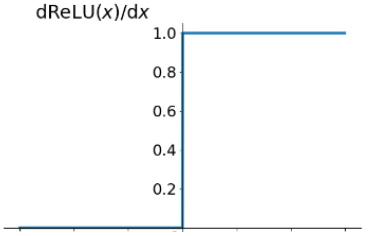

Figure 6. Derivative of ReLU function

The ReLU function does not have the gradient vanishing problem, and the convergence speed is much faster than the Sigmoid function and Tanh function. Most importantly, the function can be used in multi-classification projects. In this case, the author definitely chose this activation function.

In Python, the author built the network using a ReLU function after each linear transformation. There were 561 neurons initially, representing 561 attributes, and 6 neurons eventually, representing 6 classifications. To achieve the best fitting of the data, the author can increase or decrease the number of layers, or change the net.

After that, the author used the Cross Entropy Loss to define the loss and the optimizer: torch.optim. Adam to optimize the fitting. The learning rate was set to 0.01, and it was found that the loss functions were mapped. Figure 7 shows that the loss was gradually decreasing.

DOI: 10.54254/2755-2721/54/20241421

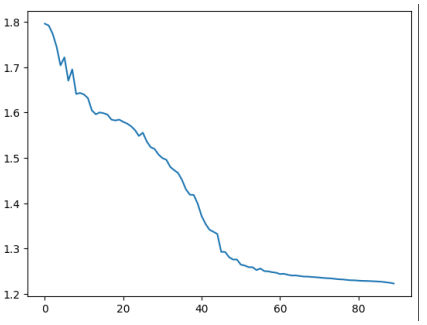

Figure 7. Loss function with the number of iterations(x-axis: number of iterations, y-axis: value of loss function)

The author used this model to fit the test set, compared the predicted results with the true values, and achieved an accuracy of 88.43%.

The programming of the SVM is illustrated in Algorithm 1, while the programming of Decision Tree model and RF model is illustrated in Algorithm 2.

Algorithm 1: Programming of SVM
Input: X,Y **Output**: model from sklearn import svm model=svm.SVC() model.fit(X,Y) **Return** model
Algorithm 2: Programming of Decision Tree model and RF model
Input: Min_samples_split=2, Random_state=0, N_estimators=10 Min_samples=2 **Output**: DT_model, RF_model from sklearn.ensemble import RandomForestClassifier from sklearn.ensemble import ExtraTreeClassifier from sklearn.tree import DecisionTreeClassifier DT_model= DecisionTreeClassifier(max_depth=None, min_samples_split= Min_samples_split, random_state= Random_state) RF_model= RandomForestClassifier(n_estimators= N_estimators, max_depth=None, min_samples= Min_samples, random_state= Random_state) **Return** DT_model, RF_model

In this way, the author obtained the accuracy rate of the other three models. SVM, DF, and RF accuracy rates are 95.04%, 85.95%, and 91.07%, respectively.

4. Results and discussion

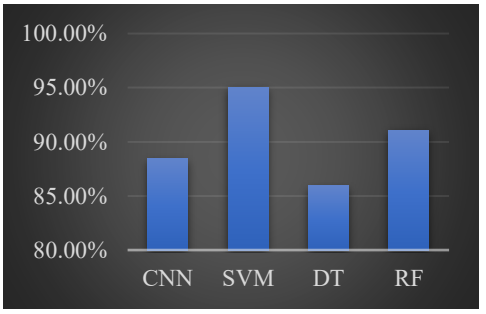

Figure 8. Comparison of the four models

DOI: 10.54254/2755-2721/54/20241421

As is shown in Figure 8, the SVM has the best accuracy rate among the four models. This experimental result is inconsistent with the expected hypothesis that DCNN should yield the highest accuracy. This result implies that the DCNN model needs to be adjusted.

Algorithm 3: Programming of Neural network

Input: Lr=0.002, Gamma=0.9

Output: optimizer, scheduler

```
from torch.optim.lr_scheduler import ExpotentialLR
optimizer=torch.optim.AdamW(net.parameters(), lr=Lr)
scheduler= ExpotentialLR(optimizer, gamma=Gamma)
```

Return optimizer, scheduler

The programming of Neural network is illustrated in Algorithm 3. After several iteration, the author discovered that the loss of the model increased and thus lowered the accuracy rate. The reason is that the thick network layer made the model overfit. Therefore, the number of convolutional layers was reduced and the number of neurons in each layer was modified.

Moreover, it can be found that in the previous model, the loss function did not drop very fast, and the minimum value of the loss function exceeded 1.2, indicating that the learning rate should be adjusted to reduce the value of the loss function as much as possible. However, if the learning rate only changed, the data would be overfitted. Therefore, codes should be added to dynamically adjust the learning rate, which means that with each iteration, the learning rate was increased by 0.9 from the previous learning rate.

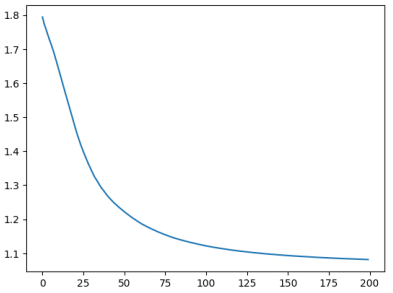

Figure 9. Loss function with the number of iterations after modification (x-axis: number of iterations, y-axis: value of loss function)

Algorithm 4 Programming of calculating accuracy rate of Deep Convolutional Neural Network after modification

Input: test=read('data_test.csv'), drop=delate name

Output: accuracy

```
for i in drop:
    test=test.drop(columns=i)
Y_t=test['Result']
test=test.drop(columns= 'Result')
test=test.values
X_t=test
Y_t=torch.tensor(Y_t)-1
y_t=net(torch.Tensor(X_t))
y_t.argmax(dim=-1)
yy=y_t.argmax(dim=-1)
accuracy=torch.sum(yy==Y_t)/ len(Y_t)
```

Return accuracy = 0.9549

DOI: 10.54254/2755-2721/54/20241421

As shown in Figure 9, the loss function decreased faster and more steadily after modification. After two hundred iterations, the loss function value became less than 1.1, which was obviously better than the previous one and the accuracy rate rose to 95.49%, as is shown in **Algorithm 4**, the highest among the four models. This outcome confirmed the hypothesis that deep learning based convolutional neural network is the more powerful and accurate than traditional machine learning based models in recognizing human activities.

5. Conclusion

This study explores the differences between the four popular models in human activity recognition and has demonstrated that the Deep Convolutional Neural Network outperformed the SVM, DT, and RD with an accuracy rate of 95.49% versus the accuracy rates of 95.04%, 85.95%, and 91.07% produced by the other three models. The results provide evidence that deep learning techniques have a definite advantage over traditional machine learning techniques and will eventually replace them as the most reliable model for human activity recognition. However, in this study, the accuracy rate of Deep Convolutional Neural Network is not far ahead of that of the other models, especially SVM, probably for two reasons. First, the traditional machine learning techniques have been in use for a long time and have therefore been tested, modified, and improved. In a word, these models are technically mature. However, the Deep Convolutional Neural Network has been developed with the rise of deep learning in recent years and is yet to be further optimized before it can develop into a sophisticated technique. Second, Deep Convolutional Neural Network requires repeated parameter adjustments and result testing to approximate the optimal accuracy. This process is prolonged and time-consuming.

References

[1] Chunxiang Zhang et al. (2020). A review of human activity recognition based on cell phone sensors. Computer Science, 47(10), 8.

[2] Nweke, Henry Friday, et al. "Deep Learning Algorithms for Human Activity Recognition Using Mobile and Wearable Sensor Networks: State of the Art and Research Challenges." Expert Systems with Applications 105 (2018): 233-61.

[3] Abbaspour, S., Fotouhi, F., et al. (2020). A comparative analysis of hybrid deep learning models for human activity recognition. Sensors, 20(19), 5707.

[4] Nurwulan, N. R., & Selamaj, G. (2021, March). Human daily activities recognition using decision tree. In Journal of Physics: Conference Series (Vol. 1833, No. 1, p. 012039). IOP Publishing.

[5] Nurwulan, N. R., & Selamaj, G. (2020, October). Random forest for human daily activity recognition. In Journal of Physics: Conference Series (Vol. 1655, No. 1, p. 012087). IOP Publishing.

[6] Nam, Y., & Park, J. W. (2013). Child activity recognition based on cooperative fusion model of a triaxial accelerometer and a barometric pressure sensor. Journal of Biomedical and Health Informatics, 17(2), 420-426.

[7] Ronao, C. A., & Cho, S. B. (2016). Human activity recognition with smartphone sensors using deep learning neural networks. Expert systems with applications, 59, 235-244.

[8] Zebin, T., Scully, P. J., & Ozanyan, K. B. (2016, October). Human activity recognition with inertial sensors using a deep learning approach. In 2016 IEEE sensors (pp. 1-3). IEEE.

[9] Sajjad, M., et al. (2020). Human behavior understanding in big multimedia data using CNN based facial expression recognition. Mobile networks and applications, 25, 1611-1621.

[10] Data Source. archive.ics.uci.edu