Vol.2016-MBL-81 No.25
Vol.2016-ITS-67 No.25
2016/12/9

スマートフォン音声データを用いた歯磨き行動評価のためのニューラルネットワーク構造の検討

Joseph Korpela[1,a)]　前川卓也[1,b)]

概要：Previous methods for skill assessment using ubiquitous computing have relied on a two-tiered approach that consists of an initial activity recognition process followed by a skill assessment process that uses the results of activity recognition as input. The intermediate activity recognition process used in those methods increases the burden placed on researchers when designing and training their skill assessment system. In this paper, we propose a method for skill assessment that removes the need for an intermediate activity recognition process. We exploit the ability of deep neural networks to extract high-level features from input data to allow us to run a skill assessment model that takes raw sensor data as input. We evaluate our method on the task of toothbrushing performance evaluation, and show that deep neural networks have the potential to compete with more traditional skill assessment systems.

Preliminary Investigation on Using Deep Learning to Evaluate Toothbrushing Performance with Smartphone Audio

Joseph Korpela[1,a)]　Takuya Maekawa[1,b)]

1. Introduction

One promising area of research that is now attracting attention in the pervasive computing community is the extension of human activity recognition to performance evaluation. Using techniques developed by the pervasive computing community for human activity recognition, it is possible to create systems that can automate performance evaluation for many domains, including sports [1], [2], [3], [4], [5] and health care [6], [7], [8], [9]. Advances in automated performance evaluation provide doctors with new tools to evaluate patients, coaches with new insights into their athletes' performance, and everyday users with low-cost access to expert advice.

The typical flow used when performing skill assessment with pervasive computing is as follows. First, data is collected from a sensor such as a microphone or an accelerometer while the subject is performing the target activities. Next, domain specific features are extracted from the data, with these features specialized to the target activities. An activity recognition model is then trained, so that the activities can be automatically recognized in sensor data, e.g., hidden Markov models (HMMs) may be trained to recognize various classes of activity in audio data. Finally, skill assessment is conducted by examining features in the sensor data when the target activities are recognized, e.g., measuring the variance and duration of audio segments for segments corresponding to the activity classes recognized. In the case of our previous study on toothbrushing performance assessment using smart phone audio [8], for example, Mel-frequency cepstral coefficients (MFCCs) were first extracted from the audio, then toothbrushing classes corresponding to brush-stroke type and mouth location were recognized using HMMs. The output of the HMMs were then used to generate independent variables, such as the duration

1　大阪大学 大学院 情報科学研究科
Osaka University, Suita, Osaka Prefecture 565–0871, Japan
a)　joseph.korpela@ist.osaka-u.ac.jp
b)　maekawa@ist.osaka-u.ac.jp

spent brushing in different areas of the mouth, for support vector machine (SVM) based regression models which predicted performance scores. This process required a great deal of input from domain experts, from determining appropriate classes and labeling audio data for activity recognition to determining appropriate independent variables to use during skill assessment. While such systems have been used successfully in the past, they create skill assessment models that are highly specific to the target activity.

Human activity recognition systems rely on a diverse range of input data (e.g., audio vs. kinematic data), with the appropriate type of data to use dependent on the activities being recognized. For example, while audio data can be used to recognize some daily-life activities, it would be ineffective when recognizing surgical tasks. And likewise, while it is practical to record a large array of kinematic data for surgical tasks, i.e., the linear and rotational movements of multiple tools and their controls used during surgery, daily-life activities are normally restricted to the data that can be collected by a small number of sensors in a smartwatch or smartphone in order to reduce the impact of collection on the end user. Moreover, even when dealing with the same type of sensor data, the characteristics of the target activity's movements have often necessitated analysis techniques that are tailored to that specific activity, e.g., analyzing a patient's motion during rehabilitation sessions using a sinusoidal model of accelerometer data [9] vs. measuring the quality of a rock climber's holds by measuring the signal energy in the accelerometer data [4].

Because of this diversity in both the data collected and the characteristics of the data, most previous research into skill assessment and performance evaluation have used activity recognition techniques that are tailored to a specific target activity [1], [2], [3], [4], [5], [6], [8], [9], [10], which greatly reduces the ability of these techniques to be generalized to other domains. Additionally, most previous techniques place an increased burden on researchers when training the system, as although the end goal is to train a system capable of estimating performance levels, they include an intermediate activity recognition process that requires the additional labeling of individual actions within the data. Furthermore, the tailored features used in previous studies are typically handcrafted and require domain specific knowledge to create. This means that experts in the given domain will need to be greatly involved in their design, which can be burdensome for the experts and costly for the project.

In this paper, we investigate the use of neural networks for performance evaluation. In recent years, neural networks have been shown to be the state of the art recognition technique for a variety of domains, including automatic speech recognition [11], [12], [13], [14], [15], [16] and human activity recognition [17], [18], [19]. Their ability to model complex abstractions of input data allows them to perform recognition tasks with high accuracy without the need for hand-tailored feature extraction processes. For example, in automatic speech recognition studies, neural networks using raw Mel filter banks as input have been shown to achieve state-of-the-art results [14], [15], [16].

We design our neural network for performance evaluation so that it can take raw sensor data as input, and perform evaluation without the need to label individual actions in the data, needing only the skill assessments provided by experts as the ground truths for our evaluation. For example, when evaluating toothbrushing performance, one such network would take raw audio data as input and output a score from 0 to 24 assessing the subject's overall performance. We examine the effectiveness of this technique by implementing a toothbrushing skill assessment system using both deep-feed-forward neural networks and long-short-term-memory (LSTM) neural networks. The contributions of this work are that:

- Our proposed method is the first to use neural networks for evaluating task performance using activity data with only the performance scores as labels, eliminating the need to label individual actions within sessions of data.
- The network architectures used in our method are designed to take entire sessions of activity data as input, allowing us to perform evaluation without labeling individual actions within the session. Because of this, the networks we design are capable of handling input sizes that are much larger than the typical sizes used with deep neural networks.
- We evaluate our method using the task of toothbrushing performance evaluation with audio data.

In the rest of this paper, we first introduce related work, including our previous research into performance evaluation using pervasive computing. We then introduce our proposed method for evaluating toothbrushing performance through the use of deep neural networks.

2. Related Work

2.1 Performance Evaluation

A major area of prior research on performance evaluation using pervasive computing comes from sports performance evaluation. One such example is [1], where wearable sensor data was used to detect sports-related training activities, and the ori-

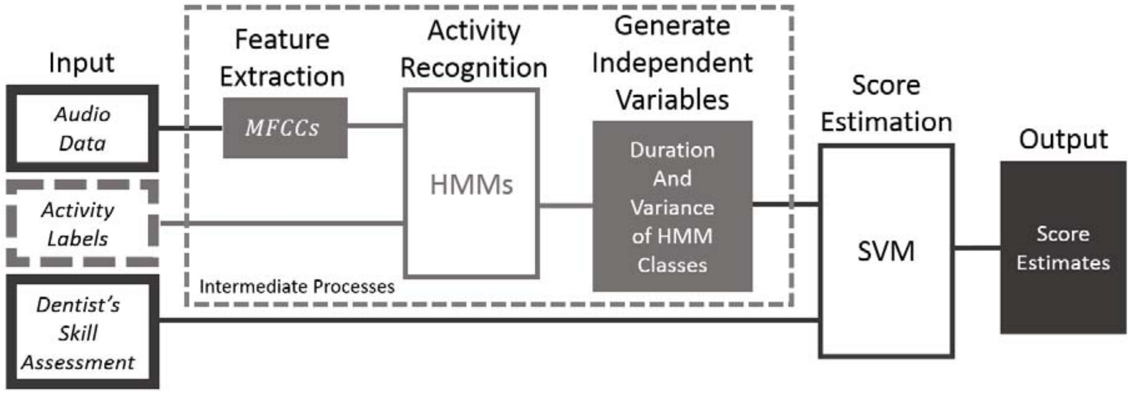

図 1: An overview of the previous process used when evaluating toothbrushing performance using audio data.

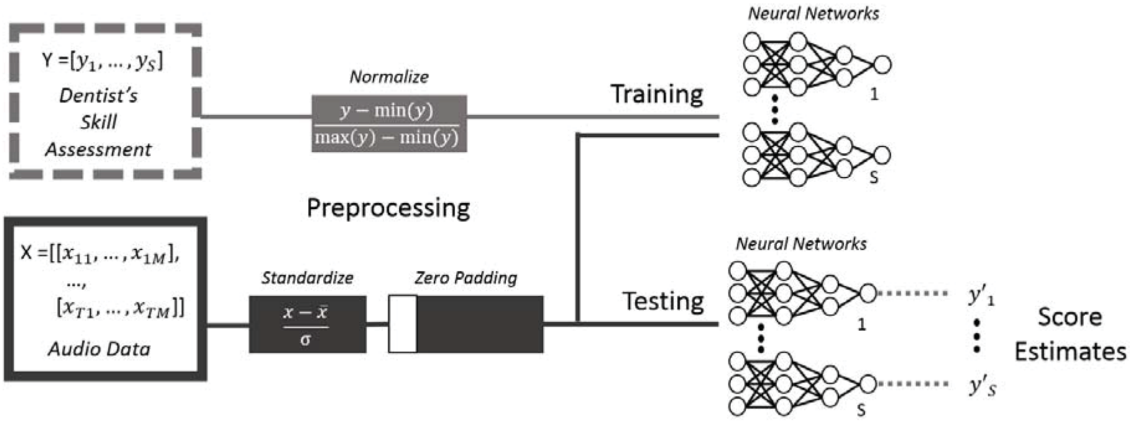

図 2: An overview of the training and testing phases used when using neural networks to evaluate toothbrushing performance using audio data.

entations of body parts during the activities then used to identify potential factors associated with injuries in athletes. In [2], acceleration sensors were used to collect kinematic data during swimming activity as a basis for providing immediate automated feedback to swimmers on their swimming technique. Accelerometer data was also used as the basis for skill assessment in [4], where data was collected during rock climbing activity using wrist-worn sensors to provide feedback to climbers that estimated their skill based on a set of criteria that is used to evaluate competitive climbing. In [5], an automated skill assessment system was designed to provide feedback to amateur horseback riders based on data collected from inertial measurement units attached to horses' legs.

In addition to the use of automated performance evaluation for athletes, techniques have also been developed for use in exercise and physical rehabilitation. In [3], a smartphone was used to provide automated feedback to users on their performance during exercise on a balance board. In [9], personalized support for physical rehabilitation was achieved by creating a system that learned a sinusoidal motion model for correct movement during sessions supervised by a physical therapist, and provided exercise feedback to the user in later sessions by comparing their movement to that motion model.

Another important area of research on automated performance evaluation is on its medical applications. In [6], deep learning was used to estimate the severity of symptoms in patients with Parkinson's Disease in naturalistic settings, based on accelerometer data collected from wrist-mounted sensors. In contrast to our current work, the above approach conducted performance evaluation by creating a tailored set of features specific to the target activity as the basis for action recognition and performance evaluation.

The most relevant previous work on skill assessment using pervasive computing is [7]. They propose a method for skill assessment using accelerometer data that is able to automatically extract features from a symbolic representation of the data using stochastic rule induction. Their technique does not rely on tailored features and thus provides a more generalized approach to skill assessment than previous work. In contrast, our system removes the need for any feature extraction by conducting skill assessment using raw data as input to the neural networks, allowing the neural networks to automatically extract features from the data.

Vol.2016-MBL-81 No.25
Vol.2016-ITS-67 No.25
2016/12/9

2.2 Audio Recognition using Deep Neural Networks

Audio recognition, and in particular automatic speech recognition, using deep neural networks is a major area of research, with deep neural networks representing the state-of-the-art. This research has included speech recognition with both long-short-term memory (LSTM) networks [14], [15] and deep-feed-forward networks [13]. In this research, we use the network architectures used in automatic speech recognition as the basis for our network architectures. However, since we do not label individual events in sessions of data, our networks are run on much larger input data, e.g., 15,000 time steps in a single instance.

In [20], the authors proposed a method for creating interference-robust neural networks to perform tasks such as ambient scene identification and stress detection using audio data collected on a smartphone. Their method increases interference robustness by combining their labeled training data with a large amount of unlabeled data from various locations that capture a variety of background noises.

2.3 Activity Recognition using Neural Networks

Neural networks have also begun to make an impact in human activity recognition using wearable sensor data. In [17], neural networks were used to perform activity recognition on a Snapdragon 400 SoC (representative of smartwatch hardware), with the networks outperforming other traditional methods of activity recognition in terms of accuracy. Work has also been done on exploring the effectiveness of convolutional neural networks (CNNs), feed-forward deep neural networks (DNNs), and LSTMs when performing activity recognition on wearable sensor data [18], with their results indicating that neural networks can outperform previously published methods. Additionally, [19] reported state-of-the-art results when performing activity recognition on wearable sensor data using a CNN-LSTM hybrid network.

2.4 Toothbrushing Performance Evaluation

Figure 1 shows the flow of our previous method for evaluating toothbrushing performance with smartphone audio data. In our previous study on toothbrushing performance assessment using smart phone audio [8], MFCCs were extracted from the audio and used as input to HMMs to recognize toothbrushing activity classes. These classes corresponded to brush-stroke type, i.e., fine vs. rough stroke, and mouth locations, i.e., inside vs. outside surface of the teeth and front vs. back teeth. For example, one class would correspond to brushing the outside surface of the front teeth with a fine stroke. We then used the output of the HMMs to generate independent variables that corresponded to the duration of each class and the variance of the data for each class. These independent variables were then used to train SVM-based regression models to predict a user's toothbrushing performance, with the ground truths for the regression models assigned by a dentist who evaluated each session of toothbrushing using video data. The dentist assigned 12 scores total, corresponding to the quality of the brushing stroke (*Stroke*), the duration of brushing (*Duration*), and how well the brushing covered each area of the mouth (*Coverage*) for four areas of the mouth: inside and outside surfaces of the front and back teeth.

The dashed gray portions of Figure 1, *Activity Labels* and the Intermediate Processes, correspond to the portions of the standard model for skill assessment via pervasive computing that we attempt to eliminate in this study.

3. Proposed Method

Figure 2 shows an overview of our proposed method. We begin with two vectors of input data: a Sx1 vector for each of the S scores being estimated for the current session, e.g., three scores corresponding to quality of the *Coverage*, *Stroke*, and *Duration* of a user's toothbrushing, and a TxM vector for the T samples of M dimensional data in the session. We first standardize each session of sensor data to have zero mean and unit variance, and normalize the evaluation scores to a range of [0,1]. The sensor data is then zero padded so that input data for all sessions have the same length. We then use the sensor data and scores to train a neural network for each of the scores. During the test phase, we use the networks to predict each of the S evaluation scores.

3.1 Preprocessing

Each dimension of the input data was standardized using the equation:

$$x' = \frac{x - \bar{x}}{\sigma}$$

where x' is the standardized input vector, x is the raw input vector, $\bar{x}$ is the mean of the input vector, and σ is the variance of the input vector.

The scores used to train the networks were normalized to the range [0, 1] using the equation:

$$y' = \frac{y - min(y)}{max(y) - min(y)}$$

where y' is the normalized score, y is the raw score, $min(y)$ is the minimum score possible, and $max(y)$ is the maximum score possible.

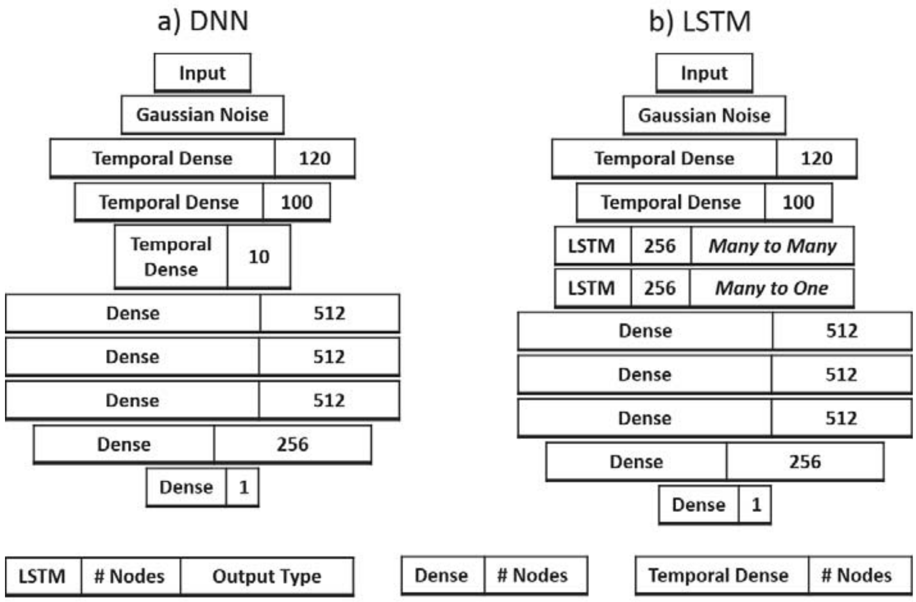

図 3: The two neural network architecture used for skill assessment in this study: a deep-feed-forward network (DNN) and a recurrent network (LSTM).

3.2 Network Architectures

In this study, we investigate performance evaluation with neural networks using two different architectures, which we will refer to as: *DNN* and *LSTM*. We use the following types of layers in these architectures:

Dense A simple feed-forward layer that does not exploit any temporal structure in the data.

Gaussian Noise Adds additive zero-centered Gaussian noise with sigma 0.1 to the input data during the training phase.

LSTM An LSTM recurrent layer, where each node processes the input vector by stepping through each time step in the data in order. If the layer is a *Many to Many* layer, then each node in the layer outputs a value for each time step, else if the layer is a *Many to One* layer, then each node returns only a single value corresponding to its output at the final time step.

Temporal Dense Treats each time step t in the input vector as a separate input to a Dense layer, i.e., splits the TxM input vector into T vectors and runs a Dense layer using each of those T vectors as input. The output of this layer is a vector of size TxN, where N is the number of nodes in the Dense layer.

All LSTM layers used the hyperbolic tangent function as their activation function, all internal DNN layers used the rectified linear unit function as their activation function, and the DNN output layers used the sigmoid function as their activation function.

3.2.1 DNN

Figure 3(a) shows the architecture for our DNN model. It takes a session's TxM vector as input and first uses three Temporal Dense layers to reduce the dimensionality of the input down to 10, reducing the GPU memory necessary for later layers. Note that this dimensionality reduction is necessary when working with large instances of data, since the data is too large to process in the GPU memory at full dimensionality. We then use five Dense layers on the Tx10 output of the Temporal Dense layers to estimate a single performance evaluation score for that session.

3.2.2 LSTM

The architecture for the LSTM network is shown in Figure 3(b). The LSTM layer starts with two Temporal Dense layers that reduce the dimensionality of the data down to 100. The first LSTM layer then takes the Tx100 vector of data as input, and steps through it in sequential order, with a value returned by each node at each time step. The second LSTM layer processes the Tx256 output vector from the first LSTM layer in sequential order, but each node only returns a single value upon reaching the final time step. Each of the values in the second layer's output now represents some feature captured from the entire session. We then use four Dense layers to estimate a performance evaluation score based on the 256x1 output vector from the second LSTM layer.

4. Evaluation

4.1 Test Environment

We evaluated our method using a desktop PC with an In-

表 1: Mean absolute error (MAE) for each method when evaluating the scores at three different granularities.

	Total	CSD	IOxFBxCSD
AvgScore	5.50	2.03	0.79
HMM-SVM	3.31	1.49	0.58
DNN	4.09	1.80	0.69
LSTM	7.97	2.28	0.80

表 2: Error ratio for each method when evaluating the scores at three different granularities.

	Total	CSD	IOxFBxCSD
AvgScore	0.229	0.254	0.393
HMM-SVM	0.138	0.186	0.291
DNN	0.170	0.225	0.344
LSTM	0.332	0.285	0.399

tel i7 6700K CPU with 32 GB ram and a GTX 1080 GPU (8 GB ram). The networks were implemented in Python using the Keras [21] deep learning library with Theano [22] used as the back end. All neural networks were trained using 30 epochs of training.

4.2 Dataset

We evaluated our method using audio data with performance evaluation scores for toothbrushing performance [8]. The data includes 94 sessions of audio data taken from 14 users. The audio was collected using a smartphone placed next to the sink when the users were brushing their teeth.

The scores used for performance evaluation were assigned to each session by a dentist who specializes in dental care instruction. We predicted these scores at three different granularities:

IOxFBxCSD refers to the use of 12 scores, each in the range [0, 2], to evaluate the user's *Coverage*, *Stroke*, and *Duration* for each of four areas of the mouth: the inside surface of the front teeth, the outside surface of the front teeth, the inside surface of the back teeth, and the outside surface of the back teeth, with *Coverage* evaluating how well the user's brushing covered the area, *Stroke* evaluating the quality of the brush stroke used in the area, and *Duration* evaluating the duration of the user's brushing in the area.

CSD corresponds to the use of three scores: *Coverage*, *Stroke*, and *Duration*, which each assigned a score in the range [0, 8]. These scores are each the sum of their corresponding scores for each area of mouth.

Total refers to the use of a single score in the range of [0, 24], used as a coarse-grained assessment of toothbrushing performance in a session of audio. This score is the sum of the 12 IOxFBxCSD scores for the given session.

The Mel filter banks used for the audio data were processed using 40 filter banks, along with the filter banks' second and third order delta components, for a total of 120 dimensions, chosen due to their reported high performance in automatic speech recognition [11], [12]. The filter banks were processed using 2048 samples per window with a step size of 1024 samples. Note that it was not possible to use the actual waveform data for the audio due to its extreme size.

4.3 Evaluation Methodology

We evaluated our method using user independent models, with the dataset split into four groups of users and tests run using leave-one-group-out validation. When evaluating the accuracy of the predictions, we use the mean absolute error (MAE) along with the error ratio. The error ratio is computed as the MAE divided by the maximum score possible for the given score type, e.g., 24 for Total scores. All results are evaluated using the mean absolute error (MAE).

4.4 Methods

When evaluating the effectiveness of our score estimation models, we will use the following methods:

AvgScore Naive method from [8] that simply assigns each score as the mean of all scores in the training data.

HMM-SVM The method proposed in [8] for a user independent score estimator that uses an HMM-based activity recognition model to generate features for an SVM-based regression model.

DNN The deep-feed-forward neural network shown in Figure 3(a).

LSTM The LSTM recurrent neural network shown in Figure 3(b).

4.5 Results

4.5.1 Overall Performance

Table 1 shows the MAE for each method on the three score types. The AvgScore results give a baseline for performance, with results any worse than this row indicating an inability to create a working model. Based on this, it is clean that LSTM was unable to build a working regression model for any type of score, with its results worse than AvgScore in each case. Looking at Table 2, we can see that the error ratios for LSTM were over 10 percent worse than AvgScore.

Comparing the results of DNN to AvgScore, we see that the DNN model was able to train a working regression model, with

DNN outperforming AvgScore for each score type. Comparing DNN with HMM-SVM, the top performing method from our previous study, we see that the neural networks were not able to perform as well. In each case, the error ratio of DNN was a few percent higher than that of HMM-SVM. Based on these results, we believe that the DNN-based evaluation model could be useful for performance evaluation tasks where one is unable to properly train the intermediate activity recognition processes shown in Figure 1.

4.5.2 Processing Times

When conducting performance evaluation, it is often important to be able to give feedback to the user in a timely manner. Because of this, we also examined the processing time needed to conduct evaluation on a single session of data. We found that the average time needed to predict a Total score for a session was approximately 67 msec. Since our architecture calls for a separate network to be trained per score, this time scales linearly with the number of scores. So even in the case of IOxF-BxCSD, DNN is able to predict all 12 scores in under 1 second. Note that this time is based on processing on a GPU, so when the data is collected via smartphone, additional time will be required for data transfer. In the case of LSTM, the average time per session was approximately 6 seconds.

4.6 Discussion

While our method allows us to conduct performance evaluation using raw data, there are restrictions on the formatting of the input data. In the case of audio data, single-channel raw waveform data for audio sampled at 44.1 kHz for sessions lasting for up to 5 minutes results in 13,230,000 x 1 input vectors, which are too large to be processed by neural networks on a single GPU with limited memory. Therefore, it was necessary to choose an alternate representation of the data as input, the Mel filter-bank representation of the data, allowing us to reduce the input vectors' to a size of 15,000 x 120. For other datasets, such as accelerometer data, it should be possible to use raw sensor data as input as the sampling rate is typically much lower than that of audio data. However, even if the raw data is not too large, it may be useful to attempt to use an alternate representation of the data, e.g., discrete Fourier transform. Additionally, it may even be possible to apply this method to video data if the data is first processed into smaller feature vectors on a per frame basis, e.g., by preprocessing frames using a pretrained CNN model such as GoogLeNet [23].

Our results indicate that while a deep-feed-forward network is capable of learning a working regression model, its performance is inferior to that of a more traditional method that uses handcrafted features. However, we feel that one reason for this is the small size of our dataset, i.e., 94 sessions. In projects with larger datasets, the additional data may push the performance closer to that of more traditional methods.

5. Conclusion

This paper proposes a method for conducting performance evaluation through the use of deep neural networks. Our method simplifies the evaluation task by removing the need to conduct intermediate feature extraction and activity recognition tasks. By doing so, we are able to reduce the burden on researchers and domain experts when developing evaluation models. We evaluated our method using the task of toothbrushing performance evaluation using smartphone audio. Our results indicate that a deep-feed-forward network is able to generate a working regression model, but the performance is not yet as good as more traditional methods. In our future work, we hope to apply our method to other datasets and plan to explore additional neural network architectures.

6. Acknowledgment

This work was supported by JSPS KAKENHI Grant Number JP16J01917.

参考文献

[1] A. Ahmadi, E. Mitchell, C. Richter, F. Destelle, M. Gowing, N. E. O' Connor, and K. Moran, "Toward automatic activity classification and movement assessment during a sports training session," *IEEE Internet of Things Journal*, vol. 2, no. 1, pp. 23–32, 2015.

[2] M. Bächlin, K. Förster, and G. Tröster, "Swimmaster: a wearable assistant for swimmer," in *11th International Conference on Ubiquitous Computing*. ACM, 2009, pp. 215–224.

[3] M. Kranz, A. MöLler, N. Hammerla, S. Diewald, T. PlöTz, P. Olivier, and L. Roalter, "The mobile fitness coach: Towards individualized skill assessment using personalized mobile devices," *Pervasive and Mobile Computing*, vol. 9, no. 2, pp. 203–215, 2013.

[4] C. Ladha, N. Y. Hammerla, P. Olivier, and T. Plötz, "Climbax: skill assessment for climbing enthusiasts," in *2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 2013, pp. 235–244.

[5] R. Thompson, I. Kyriazakis, A. Holden, P. Olivier, and T. Plötz, "Dancing with horses: automated quality feedback for dressage riders," in *2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 2015, pp. 325–336.

[6] N. Y. Hammerla, J. Fisher, P. Andras, L. Rochester, R. Walker, and T. Plötz, "PD disease state assessment in naturalistic environments using deep learning." in *29th AAAI Conference on Artificial Intelligence*, 2015, pp. 1742–1748.

[7] A. Khan, S. Mellor, E. Berlin, R. Thompson, R. McNaney, P. Olivier, and T. Plötz, "Beyond activity recognition: skill assessment from accelerometer data," in *2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 2015, pp. 1155–1166.

[8] J. Korpela, R. Miyaji, T. Maekawa, K. Nozaki, and H. Tam-

agawa, "Evaluating tooth brushing performance with smartphone sound data," in *2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 2015, pp. 109–120.

[9] G. Spina, G. Huang, A. Vaes, M. Spruit, and O. Amft, "COPDTrainer: a smartphone-based motion rehabilitation training system with real-time acoustic feedback," in *2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 2013, pp. 597–606.

[10] K. Moran, A. Ahmadi, C. Richter, E. Mitchell, J. Kavanagh, and N. O' Connor, "Automatic detection, extraction, and analysis of landing during a training session, using a wearable sensor system," *Procedia Engineering*, vol. 112, pp. 184–189, 2015.

[11] C. Liu, Y. Wang, K. Kumar, and Y. Gong, "Investigations on speaker adaptation of lstm rnn models for speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2016, pp. 5020–5024.

[12] Y. Miao and F. Metze, "On speaker adaptation of long short-term memory recurrent neural networks," in *16th Conference of the International Speech Communication Association*, 2015.

[13] T. Ochiai, S. Matsuda, X. Lu, C. Hori, and S. Katagiri, "Speaker adaptive training using deep neural networks," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2014, pp. 6349–6353.

[14] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2015, pp. 4580–4584.

[15] H. Sak, A. W. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling." in *15th Conference of the International Speech Communication Association*, 2014, pp. 338–342.

[16] F. Weninger, S. Watanabe, Y. Tachioka, and B. Schuller, "Deep recurrent de-noising auto-encoder and blind de-reverberation for reverberated speech recognition," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2014, pp. 4623–4627.

[17] S. Bhattacharya and N. D. Lane, "From smart to deep: Robust activity recognition on smartwatches using deep learning," in *2016 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*. IEEE, 2016, pp. 1–6.

[18] N. Y. Hammerla, S. Halloran, and T. Ploetz, "Deep, convolutional, and recurrent models for human activity recognition using wearables," *arXiv preprint arXiv:1604.08880*, 2016.

[19] F. J. Ordóñez and D. Roggen, "Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, p. 115, 2016.

[20] N. D. Lane, P. Georgiev, and L. Qendro, "Deepear: robust smartphone audio sensing in unconstrained acoustic environments using deep learning," in *2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 2015, pp. 283–294.

[21] F. Chollet, "Keras," https://github.com/fchollet/keras, 2016.

[22] Theano Development Team, "Theano: a Python framework for fast computation of mathematical expressions," *arXiv e-prints*, vol. abs/1605.02688, 2016. [Online]. Available: http://arxiv.org/abs/1605.02688

[23] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.