Skeleton-Based Action Quality Assessment via Partially Connected LSTM with Triplet Losses

Xinyu Wang, Jianwei Li(✉), and Haiqing Hu

School of Sports Engineering, Beijing Sports University, Beijing, China
jianwei@bsu.edu.cn

Abstract. Human action quality assessment (AQA) recently has attracted increasing attentions in computer vision for its practical applications, such as skill training, physical rehabilitation and scoring sports events. In this paper, we propose a partially connected LSTM with triplet losses to evaluate different skill levels. Compared to human action recognition (HAR), we explain and discuss two characteristics and countermeasures of AQA. To ignore the negative influence of complex joint movements in actions, the skeleton is not regarded as a single graph. The fully connected layer in the LSTM model is replaced by the partially connected layer, using a diagonal matrix which activates the corresponding weights, to explore hierarchical relations in the skeleton graph. Furthermore, to improve the generalization ability of models, we introduce additional functions of triplet loss to the loss function, which make samples with similar skill levels close to each other. We carry out experiments to test our model and compare it with seven LSTM architectures and three GNN architectures on the UMONS-TAICHI dataset and walking gait dataset. Experimental results demonstrate that our model achieves outstanding performance.

Keywords: LSTM · Action quality assessment · Triplet loss · Skeleton sequence

1 Introduction

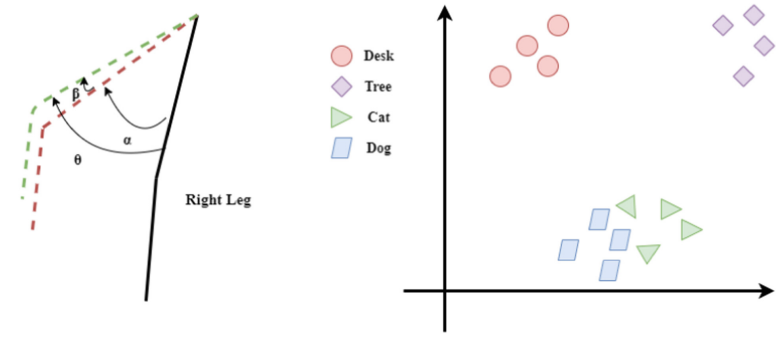
Automatic action quality assessment has attracted research interest in recent years because of its practical applications, such as skill training [1–3], physical rehabilitation [4, 5] and scoring sports events [6–8]. RGB videos [6, 7, 9, 10] and joint coordinates [4, 11] are widely used for this task. Unlike RGB videos, models based on skeleton data not only reduce the number of parameters but also focus on the human body itself, not environmental noise. Recent advances have provided reliable methods based on skeleton data in HAR. However, there are still many works to complete in AQA.

Compared to HAR, we discover two characteristics of action quality assessment: fine granularity, which makes it a challenging problem, and continuity. The process of

Supported by the Open Projects Program of National Laboratory of Pattern Recognition under Grant No. 202100009, and the Fundamental Research Funds for Central Universities No. 2021TD006.

S. Yu et al. (Eds.): PRCV 2022, LNCS 13536, pp. 220–232, 2022.
https://doi.org/10.1007/978-3-031-18913-5_17

improvement in action is a continuous process. The former has already been referenced in numerous articles [7, 8, 10], but the latter has not.



(a) Fine Granularity (b) Continuity

Fig. 1. Examples of two characteristics with (a) fine granularity and (b) continuity.

Fine Granularity. AQA is a challenging process due to the intricacy of human motion. Human motion can be defined as the movements of joint points, which are split into small and large amounts of movement. In fact, large amount of movement of joint points are usually regarded as the features of motion, such as the joint of the hand in the action "drinking". But experts may focus on small action changes, represented by small amounts of movement or small changes in large amounts of movement. Therefore, it is necessary to make models focus on small differences in the skeleton data. As shown in Fig. 1(a), the dashed lines in red and green represent the final positions of different subjects' right legs. α and θ are large amounts of movement.

Continuity. To better illustrate this characteristic, we will take a classification problem as an example. Say there is a problem classifying "desk", "tree" and "cat". It is obvious that there is no order of preference in these labels. Feature vectors, transformed from the inputs of the same class, form a cluster in feature space. There are long distances between clusters from different classes. But if samples of "dog" are added to this task, the distance between "cat" and "dog" is closer than other distances because of similarities between animals, as shown in Fig. 1 (b). In AQA, labels representing different skill levels are continuous, which shows hierarchical relations. The feature vectors of samples don't distribute randomly in feature space.

In this paper, based on two characteristics of AQA, we propose a partially connected LSTM with triplet losses. Based on the characteristic of fine granularity, we design a partially connected layer to precisely capture the relations among corresponding joints. All nodes on neighboring layers in the original LSTM model are fully connected. In this way, the joints interact with each other, which may have good or bad effects. For example, "standing with feet shoulder width apart" requires models to focus on the subjects' shoulders and feet, which means that the wrist joint is a negative factor. To avoid it, each part is represented by a graph, which constructs a diagonal matrix, activating the model's

parameters selectively. Based on the characteristic of continuity, extra information is introduced to the loss function. Extras consists of two triplet loss functions [12], which are used to make the clusters of classes in feature space distributed in order. In the proposed model, the positive and negative samples in the triplet loss are restricted by the distance between classes of different skill levels, rather than the same or distinct classes.

In the experiments, we test our model on the UMONS-TAICHI dataset [3] and the walking gait dataset [13]. We make a comparison of seven LSTM architectures and three GNN architectures. Finally, we compare the experimental results of models before and after adding the triplet losses. In summary, we have made the following three main **contributions** to this work.

- We explain and discuss two characteristics and countermeasures of AQA, compared to HAR.
- We propose a partially connected layer and apply this structure to the LSTM for assessing the quality of action from skeleton data.
- We introduce triplet losses to the loss function, based on the character of continuity, which significantly improve models' performance.

2 Related Work

2.1 Action Quality Assessment

We classify the tasks of AQA based on two factors: certainty of action and annotation type. Actions are decomposed into several certain or uncertain motion units. We all know that each diving consists of various action units, such as somersaults and twists. The final score is made up of the difficulty score and the completion score. Recently, because of the available data from the Olympic projects [6, 7, 9], assessment of uncertain units has been extensively studied. Xu et al. [6] splits video into 9 clips, which were put into 9 different C3D networks, and then used two parallel LSTMs to encode the execution and difficulty scores. Parmar et al. [7] uses related auxiliary tasks, such as counting somersaults and twists, to improve the model's performance.

But in some cases, we just want to know how well that moves, namely the completion score, which helps people do some deeper analysis, such as physical rehabilitation, training skills, and detecting abnormalities. Li et al. [14] figures out the differences between diving actions, which is unsuitable for skill assessment. To avoid the influence of differences, it is required that the actions are composed of a series of units based on fixed rules, such as golf swing [1], rehabilitation exercises [4], and karate kicking [2].

Concerning annotation type, the annotation scores are usually replaced by the features of subjects, such as skill levels, physical conditions, due to the great labor cost of the domain experts' professional annotations. This approach, which skips expert grading, converts this task from grading the videos of actions to classifying the subjects. Szczęsna [2] presents a dataset which consists of recordings of 37 karate athletes at different skill levels. JIGSAWS [15], collected from eight surgeons of varying skill levels, has been widely used as a bench dataset in many studies. But there are problems which need to be considered. If there are two subjects, one expert and one novice, doing the same actions, especially for simple actions, the novice is able to perform as well as the expert, which is a misleading guide to the model.

2.2 Graph-Based Methods

Models [10, 16–18] based on graphs of spatial-temporal joint relations have been developed and explored in HAR and AQA. Graph Convolutional Networks (GCNs) are classified as static [16] or dynamic [10, 17, 18] methods by Chen et al. [19] This paper expands it to include more tasks. Song et al. [18] proposes a spatio-temporal attention LSTM to learn discriminative features adaptively. Pan et al. [10] proposes an action assessment network with two learnable relation graphs: the spatial relation graph and the temporal relation graph. Given the complexity of motion and the lack of data in AQA, our model is proposed to find the right patterns via a static method. Like ST-GCN, static methods achieve good performance [16].

The methods can be categorized by hierarchical relations in graphs. In most methods [16–18], the human skeleton is treated as a single graph. The complexity of human action manifests in the positive or negative influences between joints. It is hard to explain the complex relations between joints with a single graph. So, part-based models are proposed. Our model constructs multiple graphs based on the human structure and assessment rules. Du et al. [20] proposes a hierarchical recurrent neural network, divided into five subnets. Each part of the skeleton based on physical structure is fed to the corresponding subnet. But the more parts are divided from the skeleton, the more subnets are required, which raises the model parameters. The PB-GCN [21] is designed to learn properties from each part and relations between them by performing a convolution on each partition, and then aggregating them. Si et al. [22] extends the part-based model architecture to graph convolutional LSTM, extracting spatial and temporal features. Instead of independent parameters between different parts, our model shares parameters partially.

3 Methods

We propose a partially connected LSTM with triplet losses for AQA. An overview of the proposed approach is given in Fig. 2. The parts with corresponding activation matrixes are fed into a partially connected LSTM. The Hadamard product is used rather than concatenating the output vectors together to create a high-dimensional vector. The final representation of parts is used in the triplet loss. In the following, we present the details of each technical component.

3.1 Joints Graph and Activation Matrix

There are many methods to construct joint graphs, which are proposed to capture more information about action patterns. For instance, traditional one considers a skeleton graph as $G(V, E)$, where V is the set of k joints and E is the set of m bones. To represent the relations between specific joints and ignore the negative influence of other joints, we divide the full set of k joints into subsets $Vs = V_1, \ldots V_n$. Unlike GCN, the bones of the skeleton are ignored in this work. To some extent, the hierarchical relations of a graph replace the edges among the joints.

A set of vertex matrices $Ve = Ve^0, \ldots Ve^n$, with $Ac^n \in R^{k \times k}$, are diagonal matrices. j_m^n, which is the element on the main diagonal of the vertex matrix, is 0 or 1 according

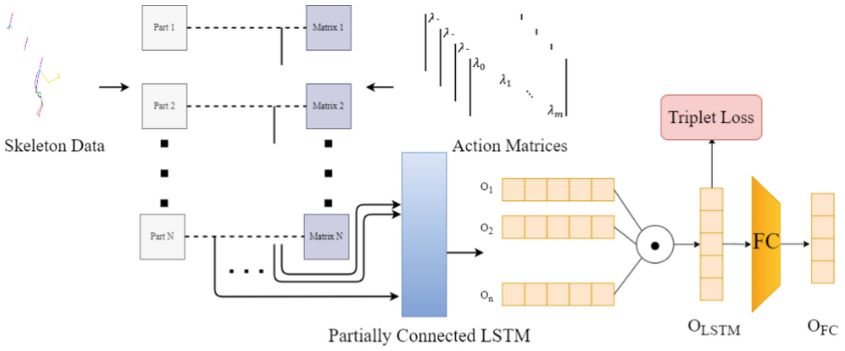


Fig. 2. The architecture of our model.

to whether m-th joint is in the subset V_n or not. Given that the joint is represented by the coordinates (x, y, z), j_m^n is replaced with three corresponding elements.

3.2 Partially Connected Layer

Given the lack of priori information about data, fully and locally connected neural networks are commonly used. Without considering the intrinsic relations between input and output, there are lots of unnecessary connections in the models, which are not conducive to capturing the underlying trend of the data.

Instead of directly multiplying the input vector with the weight matrix, we propose an activation matrix to multiply the weight before the input is put into the model. We can observe that the element in the activation matrix is 1 or 0. According to the basic matrix operation, if the n-th element is 0, the corresponding parameters will be frozen and the nth input node will not participate in the calculation. In this way, the proposed model shares weights partially via activation matrices. Figure 3 shows different processes of calculation in fully and partially connected layers. We can see that the activation matrix activates corresponding parameters in different colors.

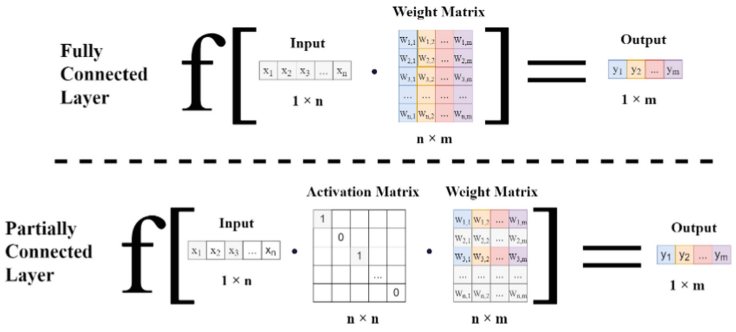


Fig. 3. Fully connected layer and partially connected layer.

3.3 Partially Connected LSTM

To avoid the problem of long-term dependency in RNN, Hochreiter [23] proposed Long Short-Term Memory, which is an advanced RNN architecture. Each standard LSTM unit contains four interacting layers: input gatei_t, forget gatef_t, output gate o_t and internal memory cell state c_t, together with a hidden stateh_t.

The activations of the memory cell and three gates are defined as follows

$$\begin{pmatrix} i_t \\ f_t \\ o_t \\ u_t \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ tanh \end{pmatrix} \left(\begin{pmatrix} W_i \\ W_f \\ W_o \\ W_u \end{pmatrix} (x_t) + \begin{pmatrix} U_i \\ U_f \\ U_o \\ U_u \end{pmatrix} (h_{t-1}) + \begin{pmatrix} b_i \\ b_f \\ b_o \\ b_u \end{pmatrix} \right) \tag{1}$$

$$c_t = i_t \circ u_t + f_t \circ c_{t-1} \tag{2}$$

$$h_t = o_t \circ tanh\left(c_t\right) \tag{3}$$

where two fully connected layers $W(x)$, $U(h)$, are the main components of the LSTM unit. The first is the input layer, which takes input at time step t. The second is the hidden layer, which takes a vector storing the values of the hidden units at time t-1 as input.

The dimensions of the input vector are equal to the order of the activation matrix. The activation matrix multiplies the weight directly to share parameters with coupled features. But for the hidden layer, the size of the weight matrix is decided by the number of features in the hidden state. To solve the problem of dimension mismatch, we decomposed the hidden layer:

$$\begin{pmatrix} i_t \\ f_t \\ o_t \\ u_t \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ tanh \end{pmatrix} \left(WVe^n x_t + U_l Ve^n U_r h_{t-1} + \begin{pmatrix} b_i \\ b_f \\ b_o \\ b_u \end{pmatrix} \right) \tag{4}$$

where the weight matrix of the hidden layer is split into two matrices U_l, U_r.

3.4 Triplet Loss

The loss function of cross-entropy is taken as the main component of the function. And extras consist of triplet losses, which have been widely used for ranking [24] and scoring. Given one anchor input X_a, triplet loss [18] is designed to minimize the distance with positive samples X_p and maximize the distance with negative samples X_n at the same time. Our loss function is composed of a set of triplet losses to make better use of hierarchical relationships between samples.

In this work, O_{LSTM}, which is the feature vector of sample X_a, is obtained as:

$$O_{LSTM} = \prod_{k=1}^{n} aO_k \tag{5}$$

where elements corresponding to the same rows and columns of O_k are multiplied together to form O_{LSTM}. While performing each Hadamard product operation, the result is multiplied by a constant a, to avoid output value disappearing.

In the first extra function, a positive sample X_p^1 is taken from a class of the same level, and the distance between the input and negative sample X_n^1 classes is 1. In the second extra function, the negative sample from the first function is changed to a positive sample X_p^2, and as a negative sample X_n^2, a sample is taken from a class of the next two levels. The loss function is defined as:

$$Loss = L_{crossentropy} + L_{triplet}\left(X_a, X_p^1, X_n^1\right) + L_{triplet}\left(X_a, X_n^1, X_n^2\right) \tag{6}$$

4 Experiments

4.1 Evaluation Datasets and Settings

We carry out experiments to test our model in twelve different taijiquan gesture classes on the UMONS-TAICHI dataset and walking gait dataset.

UMONS-TAICHI: It is a dataset of tai chi gestures that includes 13 classes collected from 12 participants at four different skill levels. It is captured by two motion capture systems simultaneously: Qualisys and Microsoft Kinect V2. In this work, we use skeleton data from the Microsoft Kinect V2.

Walking Gait Dataset: It is a dataset of gait that includes normal walking gait and 8 simulated abnormal ones by padding a sole under foot. And we divided different thicknesses into different abnormal levels. Each subject performed 9 walking gaits. Each video in the dataset contains point cloud, skeleton, and frontal silhouette and is acquired in 1200 consecutive frames.

All experiments are carried out with an NVIDIA GeForce GTX 1650 Ti. The neuron size of LSTM cell in the LSTM layer is 128. As shown in Fig. 4, the skeleton is divided into multiple parts based on human structure and assessment rules.

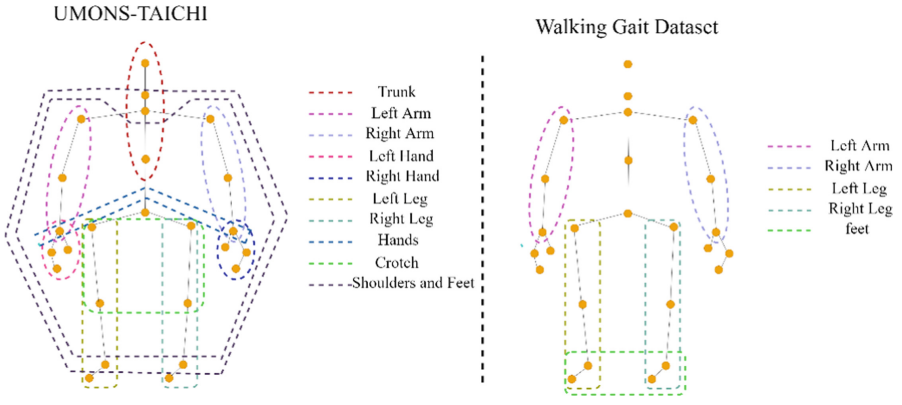


Fig. 4. The parts divided from the human skeleton.

4.2 Data Preprocess

To improve generalization and reduce the risk of over-fitting, models require a large amount of data during the training process. Furthermore, in a small dataset, model performances are excellent and similar, making it difficult to compare and analyze each model. Given the limited size of public dataset for action quality assessment, it is necessary to use data augmentation strategies. According to the different characteristics of the two datasets, we have formulated the following strategies, respectively.

For the UMONS-TAICHI, samples are divided into training and test sets, which are of equal size. Three data augmentation procedures increase the size of the training set to 12 times. First, we select random time steps from the sequence to reduce them to a specific length. Second, we randomly rotate the 3D coordinate in the range of $[-15°, 15°]$, along the x, y axis. The Cartesian coordinates of a vector are mapped to new coordinates by the multiplication of the rotation matrix. Third, apart from the joint of crotch, we randomly add Gaussian noise in data with the $\theta = 0, \sigma = 0.01$.

For the walking gait dataset, each video contains 1200 frames, which is composed of a lot of samples of walking. But the clips of samples are not split from the video. It is totally different from UMONS-TAICHI. If the strategy of video cropping as above is used again, the sample most likely contains a chaotic action sequence. Therefore, the frame sequence but not the frame itself is randomly selected from the video.

Figure 5 shows different data augmentation strategies for the UMONS-TAICHI and the walking gait dataset.

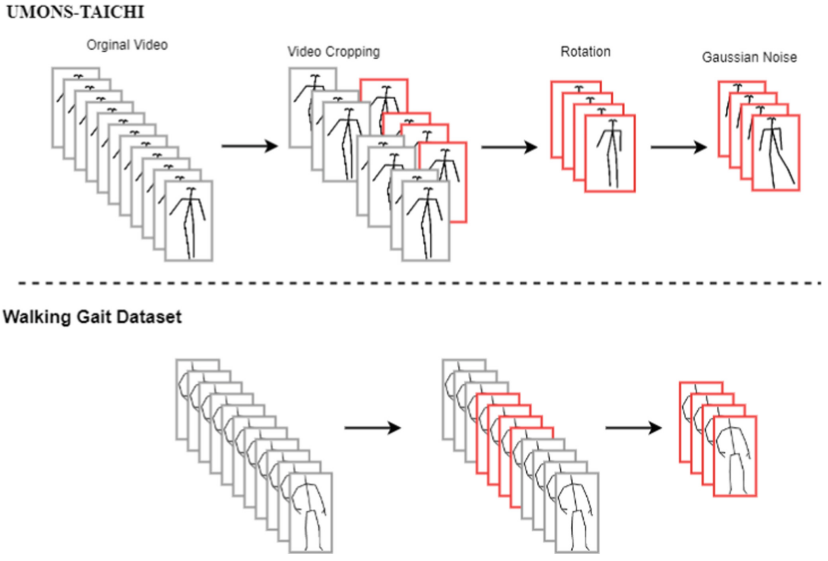


Fig. 5. Two data augmentation strategies

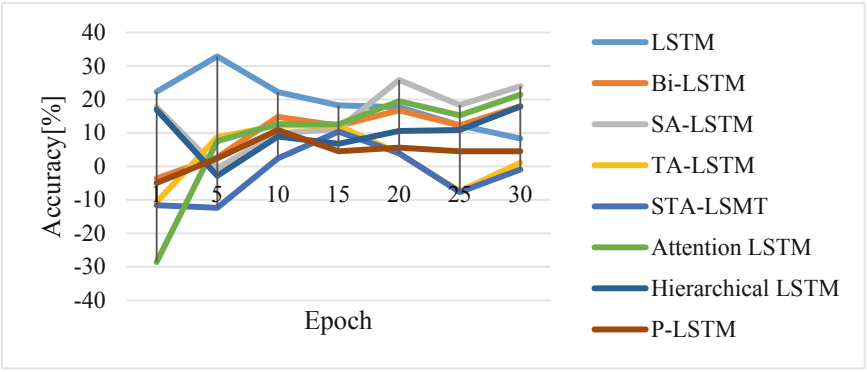


Fig. 6. Training accuracy minus test accuracy.

Table 1. Experimental results of partially connected LSTM on the UMONS-TAICHI (%).

Action		1	2	3	4	5	6	7	8	9	10	11	12
Accuracy	Tr	100	100	100	100	100	100	100	100	100	100	100	100
	Te	62.5	56.3	91.9	93.4	87.2	91.9	94.0	95.1	93.7	80.3	78.9	72.7
Spearman	Tr	100	100	100	100	100	100	100	100	100	100	100	100
	Te	62.9	22.5	94.8	94.0	82.2	92.3	92.3	96.1	95.7	85.2	85.8	66.4

4.3 Experimental Results and Analysis

The experimental results of partially connected LSTM tested in 12 different taijiquan gesture classes. The number of samples of actions 1 and 2 is only 32. We can see that the proposed model is overfitted in Table 1. Partially connected LSTM gets 100% train accuracy and test accuracy is over 90% in actions 3, 4, 6, 7, 8, 9.

In the next set of experiments, we compare our architecture with seven other deep LSTM architectures. LSTM and Bi-LSTM don't pay attention to spatial or temporal relations between actions. Compared to LSTM, SA-LSTM learns spatial patterns and TA-LSTM learns temporal patterns from data. STA-LSTM [18] is a joint spatial-temporal attention network. All of them automatically produces their attention map while training. It is mentioned in Sect. 2 that the right pattern is difficult to learn. Instead of learning attention weights, the attention mechanism calculates them from the hidden state of the decoder. Both hierarchical LSTM and partially LSTM are designed by specific graph structure. Table 2 shows a comparison of the highest accuracy of each model in 30 epochs for action 8. Table 3 shows a comparison of the highest accuracy and spearman correlation of each model in 50 epochs for the walking gait dataset. Figure 6 shows the difference between training accuracy and test accuracy. We discover that models with sub-network learning attention weights automatically perform poorly. Expect our model, the differences between training and test accuracy are all over 10%. Our model has improved the test accuracy to 95.1%, respectively. It turned out that partially LSTM

reduces over-fitting by avoiding entering too many features at the same time. On the walking gait dataset, our model obtains accuracy of 70.0%.

Table 2. Experimental results of LSTM architecture on the UMONS-TAICHI (%).

Method	Accuracy	
	Train	Test
LSTM	94.1	89.9
Bi-LSTM	100	89.9
SA-LSTM	88.0	64.0
TA-LSTM	89.7	80.9
STA-LSMT	88.3	82.0
Attention LSTM	99.1	86.5
Hierarchical LSTM	100	93.3
Ours	**100**	**95.5**

Table 3. Experimental results of LSTM architecture on the walking gait dataset (%).

Method	Accuracy		Pearson correlation		Spearman correlation	
	Train	Test	Train	Test	Train	Test
LSTM	50.4	48.8	63.9	68.9	70.5	73.2
Bi-LSTM	60.5	57.1	73.8	80.3	75.5	80.4
SA-LSTM	29.2	30.2	47.4	57.8	51.1	60.8
TA-LSTM	52.2	45.2	68.9	65.8	71.0	67.8
STA-LSMT	25.6	23.4	49.8	41.1	58.4	54.9
Attention LSTM	51.4	47.6	69.9	70.9	71.2	74.1
Hierarchical LSTM	49.0	43.7	65.7	65.7	65.4	68.3
Ours	**70.0**	**68.3**	**89.9**	**89.5**	**90.5**	**90.7**

In the third experiment, we train other advanced methods listed in Table 4 on the UMONS-TAICHI dataset. All graph-based models achieve 100% train accuracy. This shows that the spatial or temporal pattern is beneficial for models to assess the quality of action. However, models exhibit varying degrees of overfitting.

Finally, we evaluate the effect of additional triplet losses by comparing the models' performance. As shown in Table 5, this approach improves the performance of models especially for LSTM architectures.

Table 4. Experimental results of GNN on the UMONS-TAICHI (%).

Method	Train accuracy	Test accuracy
ST-GCN [16]	100	88.8
2S-AGCN [17]	100	91.0
DGNN [25]	100	77.5
Ours	**100**	**95.5**

Table 5. Experimental results of models with triplet loss on the UMONS-TAICHI (%).

Method	No triplet loss		Triplet loss	
	Train	Test	Train	Test
LSTM	94.1	89.9	97.5	93.3
Hierarchical LSTM	100	93.3	100	96.6
Attention LSTM	99.1	86.5	100	97.8
STA-LSTM	88.3	82.0	99.1	88.8
ST-GCN	100	88.8	100	88.8
Ours	**100**	**95.5**	**100**	**97.7**

4.4 Complexity Analysis

This subsection presents the complexity analysis of the runtime and parameters of our model, compared to LSTM and hierarchical LSTM. We recorded the runtimes of three models separately on the train set of the walking gait dataset. All LSTM models are recreated by us. There is no significant difference in time spent between our model and hierarchical LSTM, as shown in Fig. 7.

The complexity of model is related to the number of parameters in the network. Assume that i is the size of the input vector, h is the size of the hidden layer and o is the size of the output vector. Each standard LSTM cell contains 4 dense layers, which has a set of 2 matrices: U and W. U has dimensions $i \times h$ and W has dimensions $h \times h$. Including bias vectors, the number of parameters for LSTM cell, becomes $4 \times \left(ih + h^2 + h\right)$. All models are comprised of a single hidden layer and output layer. The number of parameters for LSTM, which is constructed with a hidden layer of a single LSTM cell, becomes $4 \times \left(ih + h^2 + h\right) + oh + o$. The hidden layer in hierarchical LSTM is composed of multiple LSTM cells. So, the number of parameters for hierarchical LSTM is $\sum_{k=1}^{n} 4 \times \left(i^k h + h^2 + h\right) + oh + h$, decided by the hierarchical relations. So, without increasing the number of parameters, our model makes use of hierarchical relations.

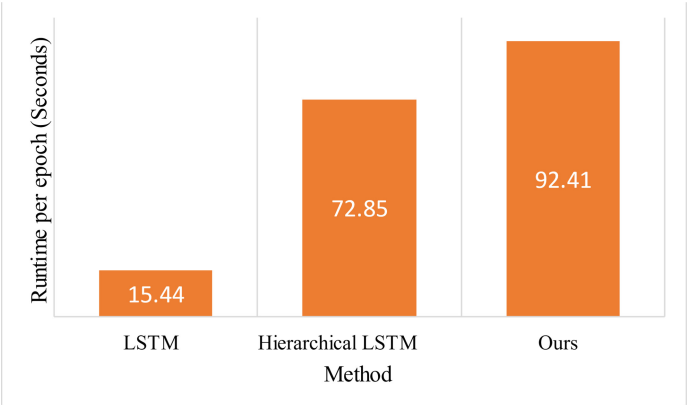


Fig. 7. Runtime per epoch of LSTM models

5 Conclusion

In this paper, we propose a partially connected LSTM with triplet losses for action quality assessment. The Fully connected layer in the LSTM model is replaced by the proposed partially connected layer to explore the hierarchical relations of skeleton graph. Activation matrix is proposed to multiply the weight, which make nodes partially connected. Such an approach can reduce the impact of insignificant features. We introduce two triplet losses to the loss function, which are used to make feature vectors distributed in order. On the UMONS-TAICHI dataset and walking gait dataset, the proposed partially connected LSTM achieves outstanding performance. In future work, we will plan to use the multi-labels fusion method to explore the hierarchical relations in the skeleton to improve the accuracy. We will also focus on employing more advanced models for action quality assessment.

References

1. McNally, W., Vats, K., Pinto, T., et al.: GolfDB: a video database for golf swing sequencing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (2019)
2. Szczęsna, A., Błaszczyszyn, M., Pawlyta, M.: Optical motion capture dataset of selected techniques in beginner and advanced Kyokushin karate athletes. Sci. Data **8**(1), 1–12 (2021)
3. Tits, M., Laraba, S., Caulier, E., et al.: UMONS-TAICHI: a multimodal motion capture dataset of expertise in Taijiquan gestures. Data Brief **19**, 1214–1221 (2018)
4. Liao, Y., Vakanski, A., Xian, M.: A deep learning framework for assessing physical rehabilitation exercises. IEEE Trans. Neural Syst. Rehabil. Eng. **28**(2), 468–477 (2020)
5. Capecci, M., Ceravolo, M.G., Ferracuti, F., et al.: The KIMORE dataset: KInematic assessment of MOvement and clinical scores for remote monitoring of physical REhabilitation. IEEE Trans. Neural Syst. Rehabil. Eng. **27**(7), 1436–1448 (2019)
6. Xu, C., Fu, Y., Zhang, B., et al.: Learning to score figure skating sport videos. IEEE Trans. Circuits Syst. Video Technol. **30**(12), 4578–4590 (2019)

7. Parmar, P., Morris, B.T.: What and how well you performed? A multitask learning approach to action quality assessment. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 304–313 (2019)
8. Parmar, P., Tran Morris, B.: Learning to score olympic events. In: Proceedings of the IEEE Conference on Computer Vision and pattern Recognition Workshops, pp. 20–28 (2017)
9. Parmar, P., Morris, B.: Action quality assessment across multiple actions. In: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1468–1476. IEEE (2019)
10. Pan, J.H., Gao, J., Zheng, W.S.: Action assessment by joint relation graphs. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6331–6340 (2019)
11. Li, H.Y., Lei, Q., Zhang, H.B., et al.: Skeleton based action quality assessment of figure skating videos. In: 2021 11th International Conference on Information Technology in Medicine and Education (ITME), pp. 196–200. IEEE (2021)
12. Schroff, F., Kalenichenko, D., Philbin, J.: FaceNet: a unified embedding for face recognition and clustering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 815–823 (2015)
13. Nguyen, T.N., Huynh, H.H., Meunier, J.: 3D reconstruction with time-of-flight depth camera and multiple mirrors. IEEE Access **6**, 38106–38114 (2018)
14. Li, Z., Huang, Y., Cai, M., et al.: Manipulation-skill assessment from videos with spatial attention network. In: Proceedings of the IEEE/CVF International Conference on Computer 14Vision Workshops (2019)
15. Gao, Y., Vedula, S.S., Reiley, C.E., et al.: JHU-ISI gesture and skill assessment working set (JIGSAWS): a surgical activity dataset for human motion modeling. In: MICCAI Workshop: M2CAI, vol. 3, p. 3 (2014)
16. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In Thirty-Second AAAI Conference on Artificial In-telligence (2018)
17. Shi, L., Zhang, Y., Cheng, J., et al.: Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12026–12035 (2019)
18. Song, S., Lan, C., Xing, J., et al.: An end-to-end spatio-temporal attention model for human action recognition from skeleton data. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31, no. 1 (2017)
19. Chen, Y., Zhang, Z., Yuan, C., et al.: Channel-wise topology refinement graph convolution for skeleton-based action recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 13359–13368 (2021)
20. Du, Y., Wang, W., Wang, L.: Hierarchical recurrent neural network for skeleton based action recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1110–1118 (2015)
21. Thakkar, K., Narayanan, P.J.: Part-based graph convolutional network for action recognition. arXiv preprint arXiv:1809.04983 (2018)
22. Si, C., Chen, W., Wang, W., et al.: An attention enhanced graph convolutional LSTM network for skeleton-based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1227–1236 (2019)
23. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
24. Prétet, L., Richard, G., Peeters, G.: Learning to rank music tracks using triplet loss. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 511–515. IEEE (2020)
25. Shi, L., Zhang, Y., Cheng, J., et al.: Skeleton-based action recognition with directed graph neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7912–7921 (2019)