Protocol 22: HRNN Training with MSR Action3D Dataset

15.06.2019

Hierarchical Recurrent Neural Network (HRNN)

1. Introduction

The hierarchical recurrent neural network [1], (Figure 1), was used to train the skeleton data for lameness detection. The network is based on the idea that the actions are dependent on the movements of individual body parts and their combinations. Since the training did not work at all, one reason may be some bugs in network or training codes (cf. protocol21 Dicussion). To debug the codes, the dataset used in the original paper of HRNN, i.e. MSR Action3D Dataset [2], was used for network training for action recognition.

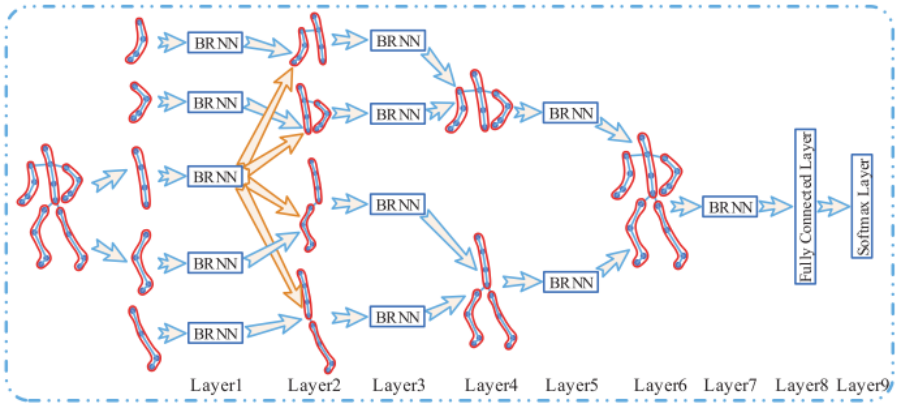

Figure 1: Hierarchical recurrent neural network. The human skeleton is divided into five parts, which are fed into five bidirectional recurrent neural networks (BRNNs) [1].

2. Experiment

- **Dataset:** MSR Action3D Dataset has 20 actions performed by 10 subjects in an unconstrained way for two or three times, 567 samples with 22077 frames. All sequences are around 40 frames and were captured in 15 frames per second, and each frame in a sequence contains 20 3D skeleton joints, as shown in Figure 2. The dataset can be downloaded from [3].

  The 20 classes are:

  > high arm wave (1), horizontal arm wave (2), hammer (3), hand catch (4), forward punch (5), highthrow (6), draw x (7), draw tick (8), draw circle (9), hand clap (10), two handwave (11), side-boxing (12), bend (13), forward kick (14), side kick (15), jogging (16), tennis swing (17), tennis serve (18), golf swing (19), pickup & throw (20).

  The dataset is divided into three action sets AS1, AS2 and AS3, each contains eight actions (classes). These three sets are further split into training and test sets based on the subject ID (odd number for training).

- **Network architecture:** The HRNN has 9 layers (Figure 1). The coordinates of the five body parts are respectively fed into the five RNN in the first layer. As the number of layers increases, the representations extracted from the subnets are hierarchically fused based on the correlation between the parts. All the recurrent layers are bidirectionally RNNs (BRNNs), and only the last recurrent layer (before fully connected layer) consists of LSTM neurons. The network outputs the predicted class from the softmax layer.

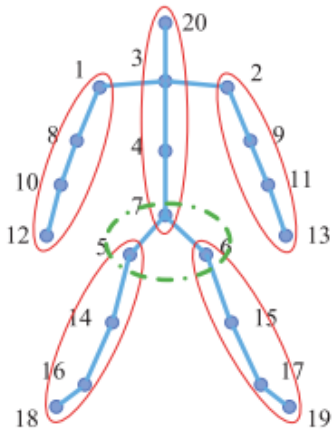

Figure 2: The 20 skeletal joints in MSR Action3D Dataset [1]. The joints are divided into five parts according to the paper.

- **Strategies:**
  - The value of learning rate was fixed in the beginning (0.001) and reduced for every ten epochs.
  - Sequence length: The length was fixed as 40 frames for each video.
  - Batch normalization was added after recurrent layers.
  - Weight noise was added during the training process.
  - Batch size: 16.
- **Result:** The overall accuracy of the three action sets are 52.83%, 70.18%, and 72.41 %, respectively. The confusion matrices for of the three sets are displayed in Figure 3, and the f1 scores for each class are listed in Table 1.
- **Observations:**
  - Training from scratch worked better than using pretrained model of other action set.
  - The initial prediction always have the same label for all the samples. If regularization is applied from the start of training, the network cannot learn properly.
  - Some actions such as forward punch (Class 5) are hard to predict because of their similarity with other actions: horizontal arm wave (Class2), hammer (Class 3). The action draw x could not be predicted because of relatively small data amount.

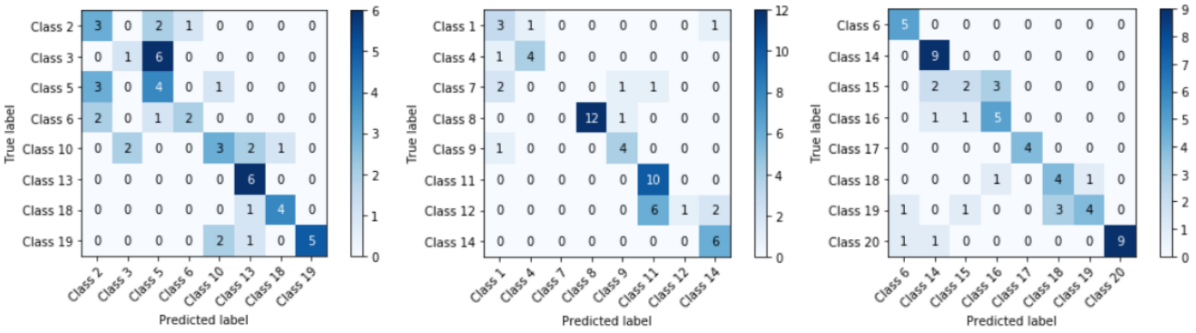

Figure 3: Confusion matrices of the three action sets: AS1 (left), AS2 (middle), and AS3 (right).

	AS1 f1 score		AS2 f1 score		AS3 f1 score
Class 2	0.43	Class 2	0.60	Class 6	0.83
Class 3	0.20	Class 3	0.80	Class 14	0.82
Class 5	0.38	Class 5	0.00	Class 15	0.36
Class 6	0.50	Class 6	0.92	Class 16	0.62
Class 10	0.43	Class 10	0.80	Class 17	1.00
Class 13	0.75	Class 13	1.00	Class 18	0.61
Class 18	0.80	Class 18	0.11	Class 19	0.57
Class 19	0.79	Class 19	1.00	Class 20	0.90

Table 1: f1 scores of the three action sets.

3. Discussion

Since the paper [1] does not give any details of the hyperparameters or tuning strategies, the training process may be quite different, which can be one of the reasons why the result of accurate prediction could not be reproduced. Still, the experiment shows that the hierarchical RNN is able to classify multiple actions using limited amount of skeleton data. For lameness detection, there are fewer classes, but the variation of data is smaller, so the prediction can be more difficult. The subtleties of skeletal joints can be degraded by poor data quality. Besides, some joints may vary from subject to subject, but tell nothing about the level of lameness. As for the training process, there are some issues:

- Training stagnation

  Even though the network has only nine layers, the number of parameters is quite large. The vanishing gradient problem caused the network to stop learning, i.e. the weights could not be updated. Batch normalization was added between after the recurrent layers to reduce the problem. In addition, the learning rate should not be too large.

- Same prediction in the beginning

  As mentioned in the previous protocol, one issue was the network always predicted the same label. The issue appears in action recognition as well: the predicted labels are the same during the first few epochs, but may jump between different labels. The most likely reason is data similarity rather than imbalanced data. Imbalanced data still can be an issue for lameness detection, though. The skeletal data appear similar for different actions, not to mention the data of cow walking. Accordingly, the learning rate cannot be too large or small, otherwise the learning curve cannot converge.

- Overfitting

  The issue of overfitting is hard to avoid even with some regularization strategies.

- Hyperparameters

  Learning rate appeared to be the most vital hyperparameter, but there may be others that were neglected during the training process.

Reference

[1] Y. Du, W. Wang, and L. Wang, Hierarchical recurrent neural network for skeleton based action recognition, in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 11101118, 2015.

[2] W. Li, Z. Zhang, and Z. Liu, Action recognition based on a bag of 3d points, in 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops, pp. 914, IEEE, 2010.

[3] W. Li, MSR Action Recognition Datasets, https://www.uow.edu.au/~wanqing/#Datasets.