Predicting Human Body Anatomical Markers From Markers Generated by Human Pose Generation Algorithm

Bingxian Chen
Stanford University
Bchen72@stanford.edu

Celia Hu
Stanford University
celiahu@stanford.edu

Mentor: Dr. Antoine Falisse
Stanford University
afalisse@stanford.edu

Abstract

Recent advancement in human pose estimation algorithms like the OpenPose have enabled quantification of kinematics and dynamics of human movement from videos. We implemented an attention based bidirectional LSTM model to predict a comprehensive set of human anatomical markers from a limited set of markers generated by pose estimation algorithms. We investigated the effects of attention mechanisms and different hyperparameter choices on the performance of our model.

1. Introduction

This project is part of the pipeline of OpenCap, a software to quantify kinematics and dynamics of human movement from smartphone videos. One step of this process is to use a reduced set of body markers identified from videos, usually through pose estimation models such as OpenPose [7], to predict a comprehensive set of anatomical markers. This comprehensive set is needed for accurate musculoskeletal simulations. Currently, the OpenCap research group uses the LSTM model for this task. Dr. Antoine Falisse who led the OpenCap research is the mentor of our project. The goal and challenge for our project is to explore other model architectures that can be added to improve the accuracy of this model.

The input to our algorithm is a series of body movements represented by 3D coordinates of reduced body markers. The original data is in trc format, which can be read by OpenSim, a software that can visualize musculoskeletal movements. The data is pre-processed by Dr. Falisse and is transformed from trc files to numpy arrays. We then use a LSTM model and output a more complete set of anatomical body markers of the same movement series, still in the form of numpy arrays.

2. Related work

To better understand the problem, we did some literature review regarding human position deep learning and attention models. While LSTM is the commonly used structure for this type of problem, many researchers proposed adding attention features to make the model better suited for the nature of human position. Besides the basic attention model, the spatial-temporal one seems to be a popular adaptation [1][8][9]. Other options include applying GRU with LSTM or using GNN [3][4]. After evaluation, we think the spatial-temporal attention structure could be useful for our project.

One approach from the paper by S. Song et. al. [1] use a local attention model, with spatial attention for joint selection gate and temporal attention for frame-selection gate. The architecture has a spatial attention layer, followed by LSTM layers, and then combined with a temporal attention layer. However, this problem is looking at human action recognition, which is a many-to-one model, different from our many-to-many output.

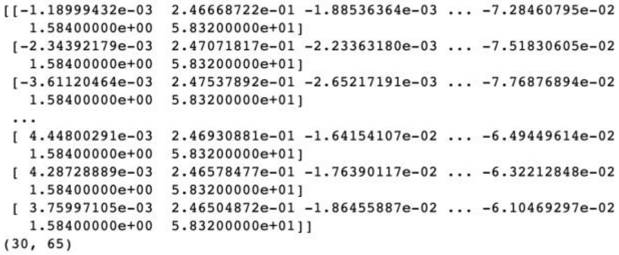
Another approach is a global attention model for spatial information and accumulative learning curve model for temporal information, proposed by Y. Han et. al. [8]. This is also a human action recognition task, which means a many-to-one model. The architecture of this model is a global attention model followed by an accumulative learning curve (ALC) model.

3. Dataset and Features

The dataset of this project is collected and provided by Dr. Falisse. Each example is a series of single-person movements performed in 0.5 second (30 frames at 60 Hz), represented by the 3D coordinates of a set of body key markers. The input is a reduced set of markers, and the output is a more complete set of biomarkers of the same subject over the same time period under the same movement. The mentor suggested using only part of the complete set of markers because they do not care about the outcome of the rest.

The input data provided is each of size (30, 65). Since this project only cares about part of the marker, the input will be reduced to the size of (30*47) before being actually fed into the model. 30 frames represent the length of the motion. 47 represents the feature dimension, including the 3D coordinates of 15 markers, and the height and weight of the subject (3*15+2 = 47). The 3D marker data are expressed with respect to a reference marker and normalized by the subject height. The corresponding output is a 30*105 matrix. While 30 is the same time dimension feature, 105 is the 3D coordinates of the 35 predicted markers (35*3=105).

Figure 1 is an example of how the input array of (30, 65) looks like. Figure 2 is the visualization of trc file in OpenSim. The blue dots represent an example of the input movement, and the green dots represent the corresponding output with a more complete set of markers.



```
[[-1.18999432e-03  2.46668722e-01 -1.88536364e-03 ... -7.28460795e-02
   1.58400000e+00  5.83200000e+01]
 [-2.34392179e-03  2.47071817e-01 -2.23363180e-03 ... -7.51830605e-02
   1.58400000e+00  5.83200000e+01]
 [-3.61120464e-03  2.47537892e-01 -2.65217191e-03 ... -7.76876894e-02
   1.58400000e+00  5.83200000e+01]
 ...
 [ 4.44800291e-03  2.46930881e-01 -1.64154107e-02 ... -6.49449614e-02
   1.58400000e+00  5.83200000e+01]
 [ 4.28728889e-03  2.46578477e-01 -1.76390117e-02 ... -6.32212848e-02
   1.58400000e+00  5.83200000e+01]
 [ 3.75997105e-03  2.46504872e-01 -1.86455887e-02 ... -6.10469297e-02
   1.58400000e+00  5.83200000e+01]]
(30, 65)
```

Figure 1: Input data of numpy array with shape (30, 65).

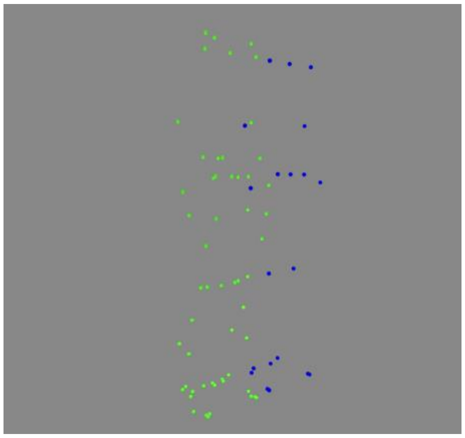


Figure 2: OpenSim visualization of trc file.

The entire dataset contains 121,554 examples, which is split into training, evaluation, and testing sets with distribution of 80%, 10%, 10%. The splitting method makes sure that all of a participant's data resides in only one set.

4. Methods

4.1 Long Short-Term Memory Network (LSTM)

The first model that we examined was the long short-term memory network (LSTM) [5], which is a particular type of recurrent neural network, or a RNN. In general, a RNN is a network that uses sequential data or time series data. It is unique from other neural networks due to its ability to take information from prior inputs to influence its current input and output. The output of RNN depends on the prior elements in the sequence. As shown in Figure 3, an LSTM network uses the input for the current time-step and the output from the previous time-step to produce a new output which is fed to the next time step [2]. In the LSTM architecture, there are three types of gates (input gate, output gate, forget gate) for each memory cell. These three gates regulate the flow of information into and out of each cell.

The characteristics of LSTM make it a great candidate for our task where we input time series data and output time series data, and output at each time-step should depend on prior elements in the sequence. We implemented the LSTM model described by Uhlrich et al. [6] as our baseline model.

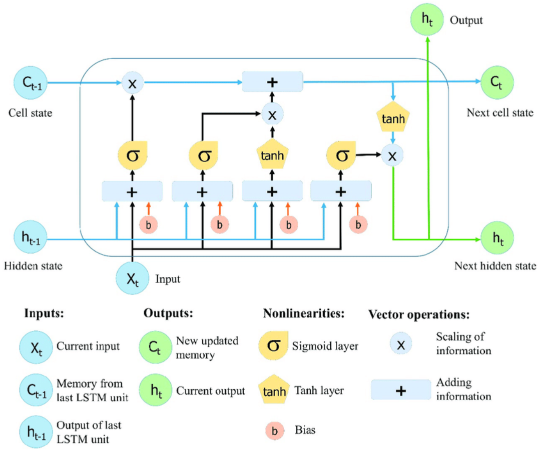


Figure 3: Long Short-Term Memory Neural Network [2]

4.2 Bidirectional LSTM

Bidirectional LSTM is a variant of LSTM that not only takes into account information from prior time-steps but also information from future time-steps when producing output at the current time-step. This is achieved by running a forward LSTM and a backward LSTM model and concatenating their outputs at each time-step. Since our task involves predicting a set of markers' positions in a series of time points, we believe that at any time point t, both input positions before time t (past) and after time t (future) would be helpful to predict marker position at time t. As a result, we hypothesize that using bidirectional LSTM might achieve better accuracy than unidirectional LSTM.

4.3 Attention Mechanism

The attention mechanism is another popular modern deep learning architecture that is used for action classification and pose estimation [1][3][4][8][9]. Although RNN architectures already take into account information provided by prior elements in the sequence, the information could get lost when the input sequence gets longer. The attention mechanism is a good way to propagate contextual information through the entire network with a trainable attention weight that tells the model which members of the input sequence the model should be paying more attention to when producing the output at the current time-step.

Our implementation of the attention layer is inspired by the spatial attention proposed by Song et al. [9]. At each time step t, given the full set of K markers $x_t = (x_{t1}, \dots, x_{tk})$ where $x_{tk} \in R_3$. The scores $s_t = (x_{t1}, \dots x_{tk})$ for indicating the importance of the K markers are jointly obtained by: $s_t = \tanh\,(W x_t + b)$ where x_t is the output sequence from a LSTM layer with return_sequence set to True, and W and b be the learnable parameter matrix and bias vector. The activation for the kth marker a_{tk} can be calculated by applying the softmax function. The larger the activation the more important the marker is for predicting the position of the new set of markers. We used the attention mechanism along with the bidirectional LSTM model proposed earlier.

4.4 Evaluation Metrics

The loss function that we try to minimize during training is the mean squared error (MSE) loss. For quantitative evaluation of our model's performance, we compute the root mean squared error (RMSE) and mean per marker error (MPME) on the test set. An important feature that our collaborator cares about is the coherence of the predicted markers' movement. We plan to visualize the movement of the predicted markers in a few examples that our collaborator has identified as unsatisfactory to qualitatively evaluate the performance of our model as compared to the baseline model.

5. Experiments/Results/Discussion

We trained our models for 50 epochs with early stopping monitoring validation MSE loss to prevent model overfitting to the training set. We used a batch_size of 64 in all our experiments because increasing the batch_size resulted in memory error. The quantitative results of our experiments are summarized in table 1. The heatmap from our attention output is shown in figure 4. The plots of the training losses are stored in the figures folder on github. They are not included to enforce the 5-page limit. In addition, we also visualized the movement of the predicted markers in a few examples that our collaborator has identified as unsatisfactory to qualitatively evaluate the performance of our model as compared to the baseline model. The visualization of the marker movement will be included in the final presentation.

The results for the three experiments proposed in the method section aligned with our hypothesis. The bidirectional LSTM model performed better than the baseline LSTM model, and the bidirectional LSTM with attention model performed better than the bidirectional LSTM model. As shown in the heatmap in figure 4, in this example, our model learned to pay more attention to certain markers at each time-step, which helped the model to achieve better results.

In addition to the three experiments mentioned in the method section, we also modified our model architecture to add a second attention layer before the last layer to replicate the spatial-temporal attention model proposed by Song et al. [9]. The result of this model was not very good potentially due to the different natures of the two tasks. Song et al.'s spatial temporal attention model was proposed for action recognition, where the model has a time series of joint positions as input and a single action label as output, which has a many-to-one relationship, while our task has a many-to-many relationship. Further task specific modifications of the temporal attention layer may be needed to be applied to our task.

We also conducted hyperparameter experiments with our bidirectional LSTM with attention model. We experimented with learning rate and number of additional LSTM layers in our model. We found that out of the values that we experimented with, our model achieved better results with 1e-4 learning rate and no additional LSTM layers. We did not experiment with even higher learning rate because with higher learning rate, validation MSE loss becomes unstable and triggers early stopping, which prevents learning at an early epoch number (the training error still decreases). The better result achieved by a smaller number of additional LSTM layers might be attributed to the gradient problems associated with deeper RNN networks. Further analysis on the gradients is needed to draw a conclusion.

	RMSE (mm)	MPME (mm)
Baseline (LSTM), lr 5e-5, nLayers 2	8.7	13.1
Bi-LSTM, lr 5e-5, nLayers 2	7.8	12.0
Bi-LSTM + Attention, lr 1e-5, nLayers 2	7.5	11.2
Bi-LSTM + Attention, lr 3e-5, nLayers 2	7.2	10.6
Bi-LSTM + Attention, lr 5e-5, nLayers 2	6.9	10.4
Bi-LSTM + Attention, lr 7e-5, nLayers 2	6.8	10.2
Bi-LSTM + Attention, lr 1e-4, nLayers 2	**6.4**	**9.7**
Bi-LSTM + 2 Attention layers, lr 5e-5, nLayers 2	8.0	12.1
Bi-LSTM + Attention, lr 5e-5, nLayers 3	7.0	10.5
Bi-LSTM + Attention, lr 5e-5, nLayers 1	7.0	10.7
Bi-LSTM + Attention, lr 5e-5, nLayers 0	6.6	10.1

Table 1: Experimental Results. Bi-LSTM refers to bidirectional LSTM; lr refers to learning rate; nLayers refers to additional LSTM layers; 2 Attention layers refers to adding an additional Attention layer after the main LSTM layer to replicate the temporal attention layer suggested by [9]. The Bi-LSTM + Attention with 1e-4 learning rate and 2 additional LSTM layers model achieves better results as compared to other test models.

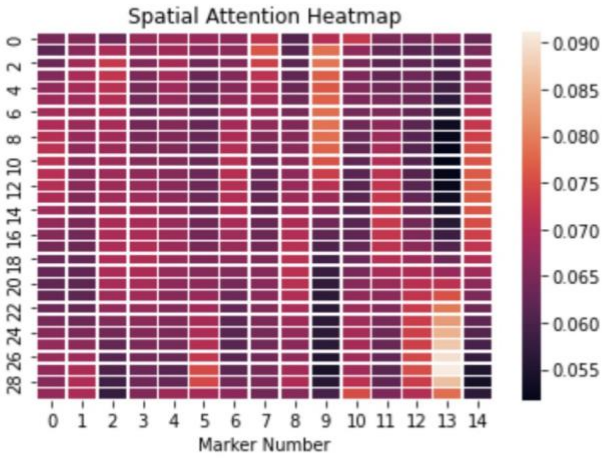


Figure 4: Spatial Attention Heatmap. The x-axis represents the 15 input markers and the y-axis represents the 30 time-steps. The output focuses more on different sets of

input markers at different time-steps. The values in each row sum to 1.

6. Conclusion/Future Work

Our work illustrated the effectiveness of attention-based bidirectional LSTM models in the relatively new field of predicting the position of a comprehensive set of human anatomical markers from a limited set of markers generated by human pose estimation algorithms. The attention-based bidirectional LSTM model performed better than the other proposed models potentially because the attention mechanism allowed the model to focus on more important input features as opposed to paying equal amount of attention to all input features, and the bidirectional LSTM architecture allowed the model to make predictions based on both past and future marker positions.

Due to time constraints, we did not have the opportunity to fully explore the spatial-temporal attention mechanism proposed by Song et al. [9]. If we had more time, we would try to adapt the spatial-temporal attention mechanism to fit our task. Moreover, we would also take more advantage of the attention output. For instance, we could incorporate the attention output and known physical constraints to our loss function. For example, some markers in the input set may be more relevant to predicting anatomical markers that are closer to them. By creating a mapping between input markers and output markers, we could use the attention output to penalize the model for paying attention to relatively irrelevant markers when making predictions about a certain marker's position. We would also benefit from a more comprehensive hyperparameter search if time permits.

7. Contribution

Kevin Chen: coding implementation, result analysis, write-up; Celia Hu: literature review, proposed model structures, write-up.

Special thanks to our Stanford collaborator Dr. Antoine Falisse and our TA Sarthak Consul for providing help and advice through the entire project.

References

[1] S. Song, C. Lan, J. Xing, W. Zeng, and J. Liu, "An End-to-End Spatio-Temporal Attention Model for Human Action Recognition from Skeleton Data." arXiv, Nov. 18, 2016. doi: 10.48550/arXiv.1611.06067.

[2] X.-H. Le, H. V. Ho, G. Lee, and S. Jung, "Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting," Water, vol. 11, no. 7, Art. no. 7, Jul. 2019, doi: 10.3390/w11071387.

[3] S. W. Chu, C. Zhang, Y. Song, and W. Cai, "Channel-Position Self-Attention with Query Refinement Skeleton Graph Neural Network in Human Pose Estimation," in 2022 IEEE International Conference on Image Processing (ICIP), Oct. 2022, pp. 971–975. doi: 10.1109/ICIP46576.2022.9897882.

[4] L. Zhou, W. Zhang, and X. Qian, "Human Action Captioning based on a GRU+LSTM+Attention Model," in 2021 The 9th International Conference on Information Technology: IoT and Smart City, New York, NY, USA, Apr. 2022, pp. 168–173. doi: 10.1145/3512576.3512606.

[5] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[6] S. D. Uhlrich et al., "OpenCap: 3D human movement dynamics from smartphone videos." bioRxiv, p. 2022.07.07.499061, Jul. 10, 2022. doi: 10.1101/2022.07.07.499061.

[7] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields." arXiv, May 30, 2019. doi: 10.48550/arXiv.1812.08008.

[8] Y. Han, S.-L. Chung, A. Ambikapathi, J.-S. Chan, W.-Y. Lin, and S.-F. Su, "Robust Human Action Recognition Using Global Spatial-Temporal Attention for Human Skeleton Data," in 2018 International Joint Conference on *Neural Networks (IJCNN)*, Jul. 2018, pp. 1–8. doi: 10.1109/IJCNN.2018.8489386.

[9] S. Song, C. Lan, J. Xing, W. Zeng, and J. Liu, "Spatio-Temporal Attention-Based LSTM Networks for 3D Action Recognition and Detection," IEEE Transactions on Image Processing, vol. 27, no. 7, pp. 3459–3471, Jul. 2018, doi: 10.1109/TIP.2018.2818328.