DeepSignal: A data driven approach to generate non verbal social signals

Prashanth Kurella
University of Minnesota
kurel002@umn.edu

Edwin Nellickal
University of Minnesota
nelli053@umn.edu

Samuel Hamann
University of Minnesota
haman152@umn.edu

Abstract

Humans are social animals and communicate through both verbal and non-verbal social signals, abstract ideas like emotions are well expressed through these social signals. We present a novel approach to solve this "social signal prediction" problem, ultimately allowing machines to communicate with humans at the fullest extent possible and therefore pushing us further towards a social artificial intelligence. We first formulate the social signal prediction problem as a high dimensional vector sequence to sequence mapping task, use Gated Recurrent Unit cells and an attention mechanism that has proven to be effective in Neural Machine Translation tasks to solve this problem. We also compare our results with the current state-of-the-art methods used to solve this problem.

1. Introduction

Analysis of physical states of human bodies is a well-studied area [6][7][10][31] of computer vision, from reconstructing skeletons to tracking individuals within a crowd. As a subset of the research, social signal analysis captures the imagination of more than most, as computer scientists and mathematicians map physical observations onto either internal or social cues. Unfortunately, large-scale analysis of these phenomena has fairly strict requirements for the data in order for models to be robust. Typically the data that is collected is limited in either the range of motion available to the subjects [25] or are not able to richly capture their motion in a crowded environment [5].

Joo et al. therefore created and analysed such a motion data set[18], by creating a scenario where three subjects participated in the Haggling game: two subjects must negotiate with the third, to persuade him/her to buy their respective products. By using a multi view setup[21][19][20] with RGB-D cameras to track several participants and record the positions and some orientations of the subjects within the frame, the original data set also included annotations for which subject(s) were speaking in a given frame. Using the position and the orientation data, we will create several different models to try to predict the position and orientation of the buyer within a sequence of frames based upon the observed positions and orientations of the sellers within the same sequence.

Joo et al. also created models for predicting the various social signals of the participants, but their models were limited to frame-by-frame analysis by their convolutional model such that they were not examining whole sequences. As such, the majority of our work involved extending their analysis with sequence-to-sequence (seq2seq) analysis, following the work of Sutskever et al.[30]. By examining whole sequences, our model should capture more of the natural transitions and motion of the subjects, resulting in a more robust and accurate model of human behavior. In particular, our data representation allows for a better predictor of the relationships between the joints and gestures than previous works.

2. Related Work

The computational analysis of human motion can be traced back to the work of Gross and Shi [11], who started a motion capture database that tracked the subjects' movements while walking on a treadmill back in 2001. Our immediate predecessors, Joo et al., created a dataset of observed joint positions for a triadic social situation as well as their own 1D convolutional model, which we use as the baseline comparison for our own methods [19][21][18].

To analyze the subjects' behavior our dataset, we looked for models that had performed well on motion in previous work. Ghosh et al. used a Dropout Autoencoder Long-Short Term Memory (LSTM) network to create more natural-looking models for human bodies in motion, informing us to implement a dropout layer within our own networks [8]. Yan et al. use Motion Transformation Variational Auto-Encoders (MT-VAE) to learn motion sequences which we have used extensively in our further investigations [36]. For the overall pipeline, we referred to the work of Walker et al.[32], as it was used to produce videos predicting the motion of human subjects in frame using LSTMs, Variational Autoencoders (VAE), and Generative Adversarial Networks (GAN).

Another aspect beyond simple human motion is the requirement for modelling the influence of social signals, which was produced by Gupta et al. in their work with GANs in pedestrian scenes. Since we only had pure position data, we sought to extract additional trainable features therefrom. Similar work was pursued by Mikolov et al. in the natural language processing for processing phrases, an application analogous to our problem here [27]. Likewise, Kulkarni et al. and Reeds et al. had worked on analyzing latent differences in images to produce analogies, or relationships between those images and their semantic meaning [24][28]. Lastly, we examined the works of Wang et al. for action-transformation relation in videos [34] and Zhou and Berg in predicting temporal relationships of subjects in videos [37] to produce more realistic overall motion in our videos.

For the actual training and architecture itself, we sought to streamline and optimize our performance efficiently, using the following works to inform our methods. Heck and Salem examined the performance of various minimal gated unit (MGU) models within recurrent neural networks (RNN) [14], and Dey and Salem likewise showed that other variant Gated Recurrent Units (GRU) successfully reduce the number of parameters and amount of expense needed to train RNNs. Luong et al.[26] implemented an attention-based network to improve upon the previous best-performing neural machine translation (NMT) software. Bahdanau et al.[4] propose an alternative to the standard encoder-decoder designs for NMT, by searching for parts of a source sentence relevant to a target word without using a fixed-length vector. Sutskever et al. [30] produced a multilayered LSTM which performed better than a phrase-based SMT system with performance close to the previous best result.

3. Baseline

The most directly comparable baseline performance was set by Joo et al.[18], where they used a used a 1D convolution based Neural Networks following the works of [16]. The linear model fails to capture the dynamics of human motion, therefore generating a 'mean pose' with no natural looking body motion, having a very low Mean Squared Error (MSE) in joint re-construction.

The 1D Auto Encoder was able to generate a more natural looking upper body motion, but failed to be quantitatively better than the linear model. However due to the multi modal nature of human motion, quantitative metrics like average joint error cannot express the quality of generated motion sequences.

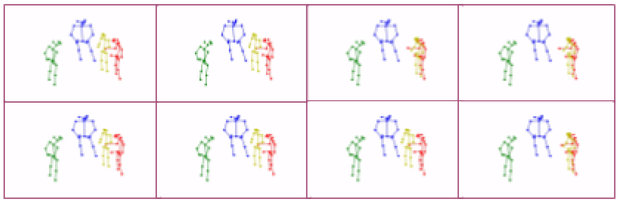


Figure 1: Depicted above are several frames generated by the 1D convolutional network from Joo et al. [18], where the red, green, and blue skeletons are the ground truth positions and the yellow one is the prediction generated by their Body2Body model. The model has very limited movement, but gets the social formations right.

4. Data Description

Our examined data consists of x-y-z positions for 19 key points for each subject, where 14 bones can be reconstructed based upon a prior spatial relations between the joints. This data was extracted from publicly available data sets posted by Joo et al. on their GitHub repository [13]. Using the position data within each frame, we infer the motion of the bodies using the change in positions.

To prevent excessive noise from contaminating our data, we conduct pre-processing by examining the bone lengths at each time step and recording which frames show an excessive amount of variation in any of the lengths. After the problematic frames are found, we then partition the data to prevent potential contamination of our training set. This is done by removing all problematic frames and a certain number of frames that precede each of them (in this case, 100), thus ensuring our data is of high-enough quality for robust training and modelling. We then proceed to create sequences of fixed length (t) after filtering the problematic frames for each of the subjects.

We then proceed to convert the the joint locations from the global co-ordinates into local co-ordinates for the skeleton by using the root of the skeleton as origin. We represented the root as its global translation from the initial starting position. This meant our final representation of skeleton is also a 57 dimensional vector. We then proceed to standardize the representation by subtracting the mean and standard deviation from each skeleton.

We represent the body motion sequences using the following notation: $\boldsymbol{B}, \boldsymbol{L}, \boldsymbol{R}$ refer to the buyer's, the left seller's and the right seller's body motion sequences respectively, the suffix represents a specific frame in the sequence for example, $\boldsymbol{B_1}$ refers to the first frame of the buyer's body motion sequence.

With this format in mind, the original data must first be organized into separate trials, from which we extract several appropriate sequences of data during pre-processing. Each of these contiguous sequences of various lengths is treated

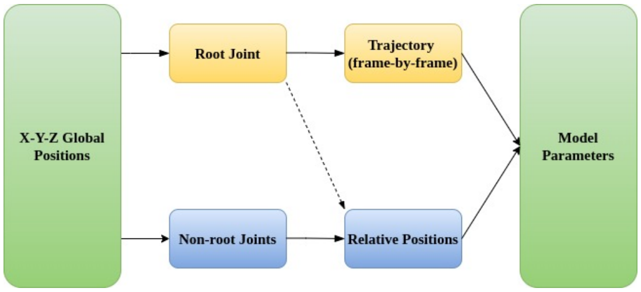


Figure 2: To create a better model, we transform the original X-Y-Z global coordinates for each joint. For the root joint, we use the frame-by-frame trajectory values to capture the overall movement of the subjects, while all of the non-root joints have their global positions transformed to positions relative to the root joint. Then these transformed data points are joined to create the model's parameters.

as a batch, from which we sub-sample to create 64 mini-batches of 80 frames each. Once these are created, we randomly select all of the mini-batches originating from 10% of the original trials as our testing set and then use all of the others for our training set. This way we can avoid having data highly correlated with the training set contaminate the testing set.

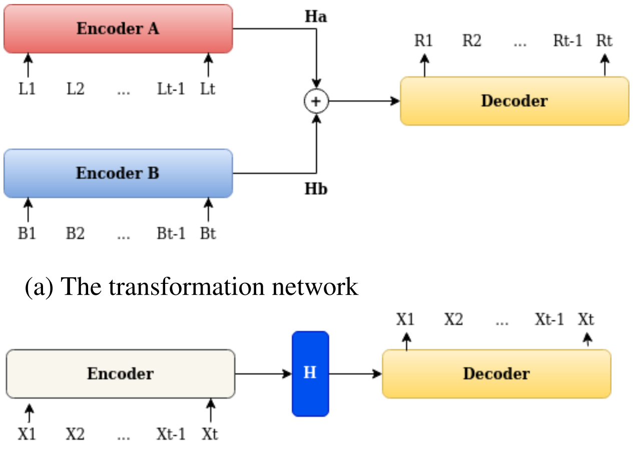
5. Proposed Method

We propose a two step training method, similar to Holden et al[16] and Joo et al[18]. We first train an auto encoder[29] to learn body motion sequences and represent them in a low dimensional latent space. We then train our transformation encoder network with decoder from the Auto Encoder, keeping the decoder frozen.

5.1. Learning Body Motion Representations

We learn to embed the body motions in a small latent space using a sequence auto encoder. The input to this network would be a body motion sequence, referred to here after as $\boldsymbol{X_1} : \boldsymbol{X_t}$. The encoder part of the network is constructed using Layer Normalized GRU cells[3][35] to reduce covariate shift while training, which takes in $\boldsymbol{X_1} : \boldsymbol{X_t}$ as an input and produces a Hidden state output $\boldsymbol{H}$, which we also refer to as the latent motion code. The encoder can have multiple layers of the LayerNorm GRU cells stocked a top each other, we introduced dropout[15] layers between each layer of stacked GRU Encoder to prevent network over fitting.

We found that an encoder with 3 stacked layers with a latent motion code of 512 dimensions with a dropout of 0.2 worked best. The decoder's job was to take in the hidden state and previous time steps output($\boldsymbol{X_{i-1}}$) as inputs at the i^{th} step, and produce the body motion at the i^{th} step as the output. The decoder is constructed similar to encoder, but we incorporate Bahdanau[4] attention by calculating it using the encoder's activations and concatenating the calculated attention vector to the decoder's activation and passing to a batch normalization layer[17] and then finally onto a fully connected layer with same number of hidden units as the output vector (57 in this case). We found that decoder with same number of stacked layers and hidden units as the encoder works best.



(a) The transformation network

(b) The body motion auto encoder

Figure 3: The illustrations of the deep learning architectures used. Here $\boldsymbol{X_1} : \boldsymbol{X_t}$ represents any target body motion sequence that needs to be learnt. $\boldsymbol{B_1} : \boldsymbol{B_t}$ represents buyer body motion sequence, $\boldsymbol{L_1} : \boldsymbol{L_t}$ represents left seller body motion sequence, $\boldsymbol{R_1} : \boldsymbol{R_t}$ represents right seller body motion sequence. Some details have been omitted for ease of illustration

We built and trained this network using the TensorFlow[1] framework, ADAM optimizer[22] with default parameters and a learning rate of 0.0001. We used the Mean Squared Error(MSE), computed over the entire sequence using Eq.1 as the loss function, where $\boldsymbol{X^*}$ represents the sequence generated by the decoder. We trained this network with both the right seller's and the left seller's body motion sequences until convergence with a mini batch size of 64 for faster training. Fig II.(b) diagrammatically represents this architecture. We use Teacher Forcing while training the decoder.

$$L_{MSE} = \sqrt{\sum_{i=1}^{t} (X_i^* - X_i)^2} \tag{1}$$

Method	Mean	Std
Body2Body[18]	**8.72**	2.00
DeepSignal	9.83	3.7

Table 1: Comparison of Average Joint Error between the baseline model and our DeepSignal model.

5.2. Learning to Generate Target's Motion from Conversation Partners' Motion

After training the auto encoder to convergence, we then discard the encoder part of the auto encoder and replace it with the transformation encoder. The transformation encoder consists of two sequence encoders, identical to the encoder used in the auto encoder. They each take in an input sequence and produce a hidden embedding($\boldsymbol{H_a}$ or $\boldsymbol{H_b}$) which are then added together and propagated to the decoder. The decoder takes this combined state and produces the target sequence. We use the same MSE reconstruction loss in Eq.1 to train the network, however this time around freezing the decoder's weights. We found that the same configuration as the auto encoder works best.

6. Results

We have also experimented with an adversarial loss that used another Recurrent Neural Network (RNN) as the discriminator, but it failed to train to completion. We hypothesize that this was due to the vanishing gradient problem, which is still very persistent in RNNs. We have also conducted experiments with a single encoder, which took the concatenated motion states of input sequences as input at each time step. This configuration produced a ”mean pose” without movement, which was much closer to the ground truth sequence. Due to the aforementioned issues, we have opted not to include these configurations in our analysis.

6.1. Quantitative

Table 1 shows the comparison between the baseline model by Joo et al [18] and our DeepSignal model. It shows that our model performs qualitatively worse than the baseline in terms of average joint error. However, given the multi modal nature of the problem, this metric doesn't fully encapsulate all of the complexity of the scenario, as the objective of our model should be to generate one of many acceptable nonverbal responses to the given social stimuli. For example, appropriate reactions to a hand wave could include a reciprocated hand wave or a change in trajectory to approach the original hand waver, or both simultaneously.

6.2. Qualitative

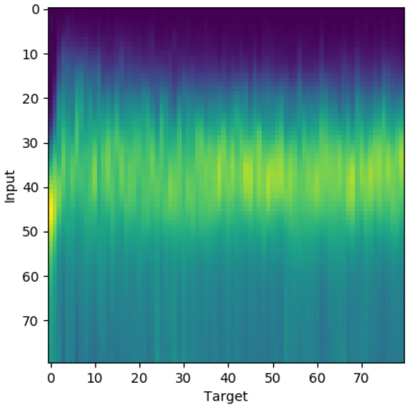


Figure 4: Depicted above is the average Bahdanau attention[4] computed while decoding the testing sequences plotted as a heat map. It appears that the early parts of sequence seem to have much smaller impact during the decoding of target sequence compared to the middle parts of the sequence

In Figure 4, we show a direct comparison between our DeepSignal model's predictions and those of the Body2Body. Upon inspection of the predicted skeletons, we found that, while the Body2Bodys' predicted skeletons had greater overlap with the ground truth than DeepSignals', this was likely due more to the social formation and overall positioning of the skeleton rather than modelling the gestures or behaviors of specific limbs. Combining the qualitative results here with the quantitative ones from above, it appears likely that the baseline Body2Body model was overfit upon the average position of the ground truth skeletons and learned an average relation between the joints, without learning the appropriate trajectories or changes in position for each joint.

7. Conclusion and Analysis

DeepSignal is able to generate fluid and vivid human like motion, however still suffers from some severe flaws. It suffers from a ”root drift” problem where the subjects' motion tends to look like they're on a friction less surface like ice and that they're drifting away while walking.We hypothesize that this is due to lack of explicit modelling of footsteps in our data and the ability to represent unnatural human motion using our representation of the motion modes. The multi modal nature of the problem and lack of a definite quantitative metric to compare the models make this problem extremely challenging. The multi modal nature of the generated sequences also paves the way for non deterministic generative models like Generative Adverserial Networks(GANs)[9] [12] [2] or Variational Auto Encoders (VAEs)[23] [33] which have been successfully applied to sequential generative tasks in other domains.

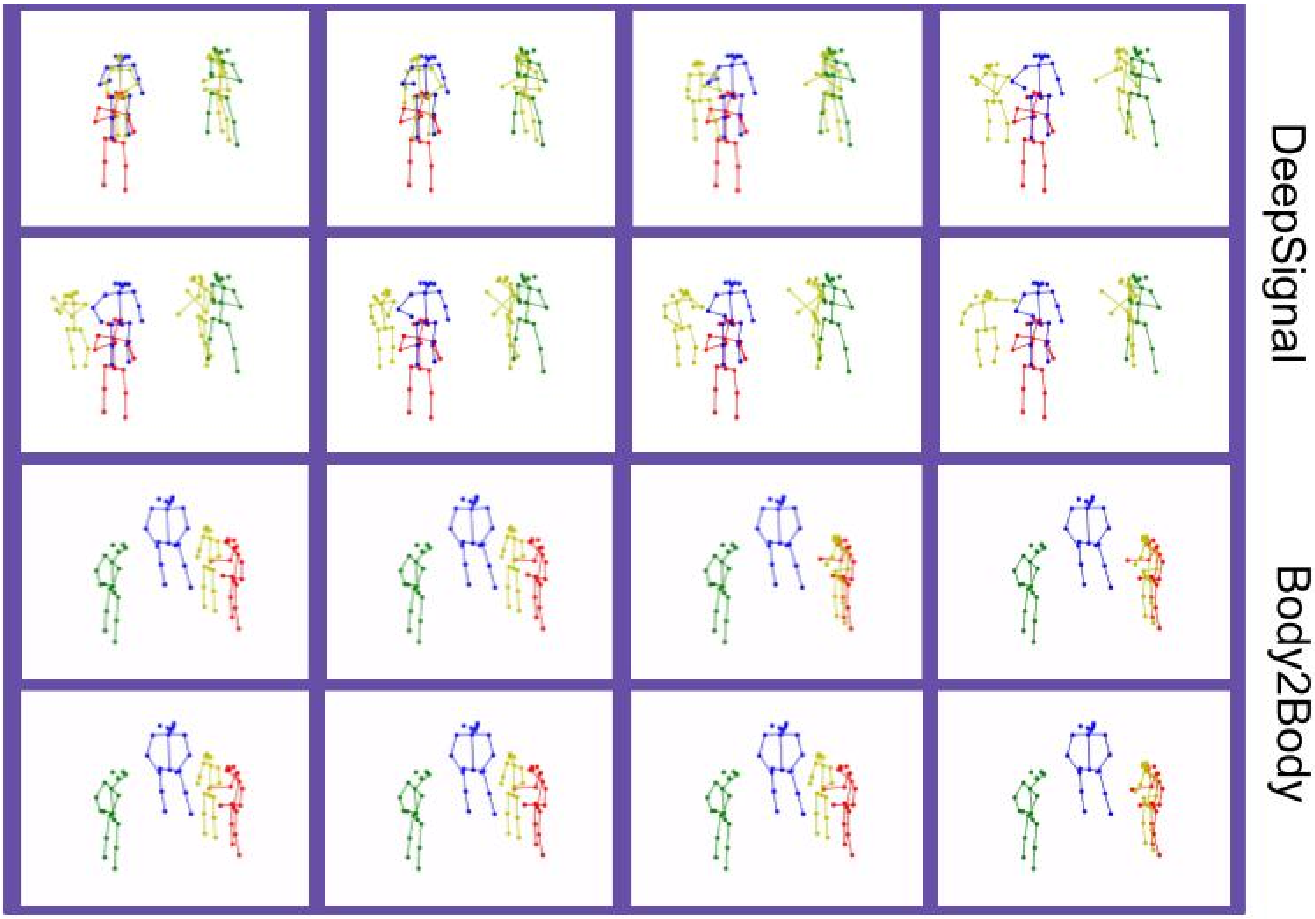


Figure 5: Predicted skeletons alongside their ground truth skeletons. The DeepSignal pictures show predicted skeletons for two subjects, compared to the single predicted skeleton in Body2Body, but the movements are clearly more dynamic and realistic movement in the frames generated by each of the DeepSignal models in comparison to that of the Body2Body model: The body itself moves more dynamically within the environment, and the limbs show more expression.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org. 3

[2] D. Acharya, Z. Huang, D. P. Paudel, and L. V. Gool. Towards high resolution video generation with progressive growing of sliced wasserstein gans, 2018. 4

[3] J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization, 2016. 3

[4] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate, 2014. 2, 3, 4

[5] L. Bazzani, M. Cristani, D. Tosato, M. Farenzena, G. Paggetti, G. Menegaz, and V. Murino. Social interactions by visual focus of attention in a three-dimensional environment. *Expert Systems*, 30(2):115–127, 2013. 1

[6] C. Bregler. Learning and recognizing human dynamics in video sequences. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, CVPR '97, page 568, USA, 1997. IEEE Computer Society. 1

[7] A. Efros, A. Berg, G. Mori, and J. Malik. Recognizing action at a distance. volume 2, pages 726–733, 01 2003. 1

[8] P. Ghosh, J. Song, E. Aksan, and O. Hilliges. Learning human motion models for long-term predictions, 2017. 1

[9] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks, 2014. 4

[10] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. *Transactions on Pattern Anal-*

ysis and Machine Intelligence, 29(12):2247–2253, December 2007. 1

[11] R. Gross and J. Shi. The cmu motion of body (mobo) database. Technical Report CMU-RI-TR-01-18, Carnegie Mellon University, Pittsburgh, PA, June 2001. 1

[12] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks, 2018. 4

[13] M. C. Y. S. Hanybul Joo, Tomas Simon. socialSignalPred Dataset, Feb. 2020. 2

[14] J. Heck and F. M. Salem. Simplified minimal gated unit variations for recurrent neural networks, 2017. 2

[15] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors, 2012. 3

[16] D. Holden, J. Saito, and T. Komura. A deep learning framework for character motion synthesis and editing. *ACM Trans. Graph.*, 35(4), July 2016. 2, 3

[17] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift, 2015. 3

[18] H. Joo, T. Simon, M. Cikara, and Y. Sheikh. Towards social artificial intelligence: Nonverbal social signal prediction in a triadic interaction, 2019. 1, 2, 3, 4

[19] H. Joo, T. Simon, X. Li, H. Liu, L. Tan, L. Gui, S. Banerjee, T. Godisart, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, and Y. Sheikh. Panoptic studio: A massively multiview system for social interaction capture, 2016. 1

[20] H. Joo, T. Simon, X. Li, H. Liu, L. Tan, L. Gui, S. Banerjee, T. Godisart, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, and Y. Sheikh. Panoptic studio: A massively multiview system for social interaction capture. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 41(01):190–204, jan 2019. 1

[21] H. Joo, T. Simon, and Y. Sheikh. Total capture: A 3d deformation model for tracking faces, hands, and bodies, 2018. 1

[22] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2014. 3

[23] D. P. Kingma and M. Welling. Auto-encoding variational bayes, 2013. 4

[24] T. D. Kulkarni, W. F. Whitney, P. Kohli, and J. Tenenbaum. Deep convolutional inverse graphics network. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 2539–2547. Curran Associates, Inc., 2015. 2

[25] B. Lepri, R. Subramanian, K. Kalimeri, J. Staiano, F. Pianesi, and N. Sebe. Connecting meeting behavior with extraversion—a systematic study. *IEEE Trans. Affect. Comput.*, 3(4):443–455, Jan. 2012. 1

[26] M.-T. Luong, H. Pham, and C. D. Manning. Effective approaches to attention-based neural machine translation, 2015. 2

[27] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc., 2013. 2

[28] S. E. Reed, Y. Zhang, Y. Zhang, and H. Lee. Deep visual analogy-making. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 1252–1260. Curran Associates, Inc., 2015. 2

[29] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. 1986. 3

[30] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks, 2014. 1, 2

[31] R. Villegas, J. Yang, Y. Zou, S. Sohn, X. Lin, and H. Lee. Learning to generate long-term future via hierarchical prediction, 2017. 1

[32] J. Walker, K. Marino, A. Gupta, and M. Hebert. The pose knows: Video forecasting by generating pose futures, 2017. 1

[33] W. Wang, Z. Gan, H. Xu, R. Zhang, G. Wang, D. Shen, C. Chen, and L. Carin. Topic-guided variational autoencoders for text generation, 2019. 4

[34] X. Wang, A. Farhadi, and A. Gupta. Actions transformations, 2015. 2

[35] J. Xu, X. Sun, Z. Zhang, G. Zhao, and J. Lin. Understanding and improving layer normalization, 2019. 3

[36] X. Yan, A. Rastogi, R. Villegas, K. Sunkavalli, E. Shechtman, S. Hadap, E. Yumer, and H. Lee. Mt-vae: Learning motion transformations to generate multimodal human dynamics, 2018. 1

[37] Y. Zhou and T. L. Berg. Learning temporal transformations from time-lapse videos, 2016. 2

Contributions

All members wrote the proposal, slides, and report, and designed the experiments together.

Samuel implemented the pre-processing and data cleaning.

Edwin worked on trying out competing 12 models on AWS, also contributed to the standardization script.

Prashanth Worked on creating the training and testing pipeline, data visualization and implementing the model architectures.

Acknowledgements

We thank **Dr. Hyun Soo Park** for his help and mentor ship throughout the project. We also thank **Dr. Hanbyul Joo** for providing us access to the Social Signal Prediction Data set.