Audio and Text-Driven approach for Conversational Gestures Generation

Korzun V. A.
MIPT
Moscow, Russia
korzun@phystech.edu

Dimov I. N.
MSU
Moscow, Russia
iliyadimov@icloud.com

Zharkov A. A.
MIPT
Moscow, Russia
andrey.zharkov@phystech.edu

Abstract

This paper describes FineMotion's gesture generating system entry for the GENEA Challange 2020. We start by using simple baselines and expand them by using context and combining both audio and textual features. Among the participating systems, our entry attained the highest median score in the human-likeness evaluation and second highest median score in appropriateness.

Keywords: embodied agents, neural networks, gesture generation, social robotics, deep learning, word embeddings

DOI: 10.28995/2075-7182-2021-20-425-432

Генерация разговорных жестов на основе речи и текста

Корзун В. А.
МФТИ
Москва, Россия
korzun@phystech.edu

Димов И. Н.
МГУ
Москва, Россия
iliyadimov@icloud.com

Жарков А. А.
МФТИ
Москва, Россия
andrey.zharkov@phystech.edu

1 Introduction

Gestures are often underrated in human communication. They may contribute a lot to a speech going as far as to change what is being said to the opposite: a simple shrug can make audience question the credibility of the speech. Humans actively use co-speech gestures to convey their emotions or visualize their attitude [5, 9].

The task of generating conversational motions can be used for social robots [16], conversational agents, and even automatic animation of virtual characters. Both rule-based and deep learning approaches have been employed to varying degrees of success. In this work, we propose several models to solve this problem as well as analyze what makes movement seem appropriate and indistinguishable from humans and which features are essential for such a task.

The GENEA Challenge [17] was conducted to explore what kind of models can produce human-like behavior for motion generation. The challenge organizers shared a 3.5-hour long dataset of audio, transcripts, and corresponding motions for body movement as well as several strong baselines. They also conducted a human evaluation of generated motions, consisting of 250 experts.

Our systems were initially built upon baselines [1, 12] provided by organizers. We made several architectural adjustments, but conceptually the core of our systems was not dissimilar from aforementioned models. Our main contributions are adding contextual information and combining both textual and audio information in one model.

Our paper is organized in the following way: section 2 describes related work; section 3 describes data preprocessing, which is shared between all experiments; section 4 describes our models; section 5 contains the discussion of our results; and section 6 contains the conclusion. Our code is publicly available[1] to help other researchers reproduce our results. Our repository also contains a link to trained weights and videos of generated motions.

The dataset used in all experiments is described in [3]. A complete task description along with evaluation of systems proposed for the workshop is described in [17]. Our team was labeled as SD for anonymization purposes.

2 Related work

In [12] authors consider motion generation problem as a mapping of sequence of words to a sequence of human poses. To solve this problem they used sequence to sequence model [15] with soft attention mechanism. The encoder processes input sequence of words which then transmitted to the decoder to generate gesture motions. Word-level features are represented by GloVe [10] embeddings. Gestures are represented by 10 principal components converted from OpenPose [11] features by Principal Component Analysis (PCA). Their sequence to sequence model also has several modifications: decoder hidden state is initialized by hidden state from previous sequence to make series of poses continuous. They also use modified loss

$$\mathcal{L} = \mathcal{L}_{mse} + \alpha \cdot \mathcal{L}_{continuity} + \beta \cdot \mathcal{L}_{variance}$$

where $\mathcal{L}_{mse}$ is a mean squared error, $\mathcal{L}_{continuity}$ is defined as

$$\mathcal{L}_{continuity} = \frac{\sum_{t=2}^{m} \|p_t - p_{t-1}\|}{m-1}$$

where p_t is a pose at time step t. $\mathcal{L}_{variance}$ is defined as negative of the variance o p_t.

In [1] authors consider a slightly different problem: given a sequence of speech features $s = [s_t]_{t=1:T}$ extracted from frames of speech audio at regular intervals t, the task is to generate a corresponding gesture sequence $\hat{g} = [\hat{g}_t]_{t=1:T}$. They use MFCC [2] features to represent audio and features learned by Denosing Auto Encoder to represent gestures. Authors use a recurrent neural network to encode a window of audio features, then this representation of the window used to generate a single frame of gestures. Savitsky-Golay filter [13] is used for smoothing the final predictions.

In the research paper [6] authors propose a GAN approach to gesture generation. They use a 1D UNet for MFCC to motion translation. The discriminator is used to avoid regressing to a mean pose.

3 Data preprocessing

The challenge organizers provided 23 recordings with an overall length of 3 hours and 40 minutes for training. Each recording consists of an audio file with speech recording, text transcripts, and BVH (bounding volume hierarchy) file with the motion data. The initial motion was captured by 60 frames per second; the generated motions for evaluation were rendered at 20 frames per second. The motion skeleton contained 71 joints, but we used only 15 points corresponding to the upper body without hands and fingers.

We split the dataset for training and validation in the following way: the first recording *Recording_001* was used for validation (12 minutes), while the rest of the recordings were used for training (3 hours 28 minutes total). As the evaluation process is rather long, we used only 1 minute of *Recording_001* for human evaluation, and the remaining part of the sample was used to calculate mean squared error on joints as a sanity check.

[1] https://github.com/FineMotion/GENEA_2020

For all our models we used the same audio and motion data preparation pipeline provided in one of the baselines [1]. For audio representation we used MFCC. We then averaged every five consequent Mel features to align audio features with motions (so that they have 20 FPS each). We represent motion data by 3 dimensional axis-angle rotation vectors for 15 joints. Thus each motion frame has 45 float features. This values are normalized over the mean value on train dataset. All aforementioned transformations of data result in input audio feature matrices to have size (N, 26) and output motion matrices to have size (N, 45), where *N* represents the number of frames in the sample.

We use the term "context window". The context window consists of 61 frames centered around a certain point in time, represented by a frame. We also use a "mean pose" calculated from the training dataset to use it as a starting value in recurrent models.

For paddings we used the MFCCs of silence recording. In text-based models we also used text features in form of GloVe embedding for words in context window.

For all proposed models we smoothed generated motions by applying the Savitzky-Golay filter to them. The length of the filter window and the order of the polynomial are 9 and 3, respectively. We did not use any external data.

4 Proposed models

The task of generating a motion can be summarized in the following way: given a set of audio features $A = (A_1, A_2, ..., A_n)$ and words $W = (W_1, W_2, ..., W_k)$ predict a corresponding set of motions $M = (M_1, M_2, ..., M_n)$.

$$f(W, A) = M \tag{1}$$

During training we minimize MSE loss and use it to assess model convergence. Our final loss functions are modified by additional terms which are described in corresponding model sections.

4.1 Sequence to sequence model

Our first described model is a sequence to sequence model, which is a reimplementation of [12] on sound-based features. The model in the aforementioned paper used words to generate corresponding motions. The competition dataset provides audio, motions and words. Three seconds of speech correspond to 60 poses and usually contain less than 10 words. We have decided to build our system on audio features and use textual information to further improve the quality of the models. Aside from difference in density between the two sets of features, speech obviously conveys more information like emotions, pauses, voice crackling, which are usually lost in text-to-speech systems.

As motions and audio features are mapped on a one-to-one basis, our first model is a simple seq2seq [15] consisting of GRU [7] encoder and decoder over audio and motions. This baseline system is illustrated on Figure 1. The encoder takes several audio features (MFCC) $A_{i-k}...A_i$ from corresponding frames, encodes them into a higher dimensional space represented by $AE_{i-k}..AE_i$ and passes it to the decoder, which predicts the following motions labeled $M_{i+1}...M_m$. Decoder's final layer combines decoder hidden state and encoder-decoder dot-product attention [8] to make a motion prediction. As the decoder requires a pose as the first input, we supply it a previously predicted pose or the "mean pose" if no previous poses are available.

We tried to further improve this model by adding a word encoder, which is illustrated in the dotted box in Figure 1. The simultaneous use of words and audio features has the problem of alignment. The GENEA 2020 challenge dataset had a transcript with words and corresponding time regions. Such markup is hard to annotate. Moreover while it helps to map words to various time windows there is still a problem of combining just a few words and multiple audio features. We use attention mechanism between the two representations for automatic alignment.

Words are embedded using GloVe [10] and are passed to another GRU. The hidden state of word-level encoder is not directly passed to the decoder, but a second encoder-decoder attention vector is calculated, which is supplied to the final layer of the decoder, to make prediction based both on audio, previous poses and words. The words are taken from a 2-second window.

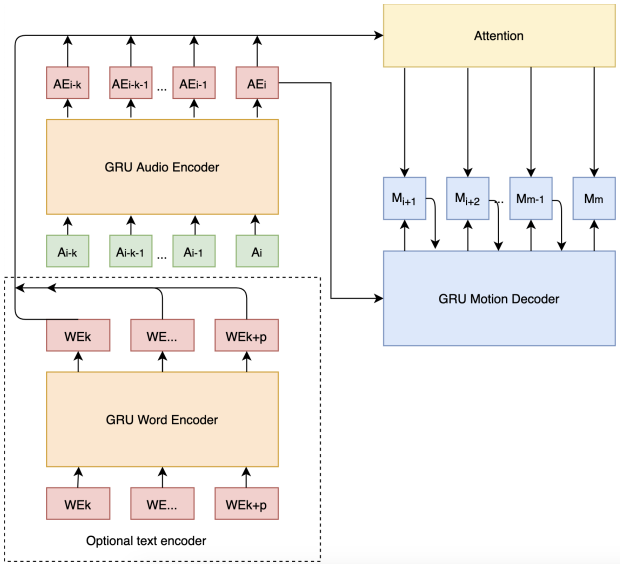

Figure 1: Scheme of baseline seq2seq model on audio features with optional word-level encoder.

We tuned several hyperparameters and training strategies. As the authors in [12] we employed continuity loss and variance loss to make the generated motions more fluid and natural. The addition of variance loss significantly improved co-speech gesture quality. We trained model with learning rate of 0.001 using Adam optimizer; audio encoder was a 2-layered bidirectional GRU with the hidden dimension of 150 units; word encoder was a single-layered GRU, both input and output dimensions were set to 100 units; decoder was a single-layered GRU with hidden dimension of 150 units. The model was trained for 100 epochs with a batch size of 512, where each sample contained 10 previous poses and 20 poses for prediction.

We also explored various combinations of windows sizes for encoder and decoder. We did not find larger windows to be beneficial to the quality of our predictions and we kept the same window sizes as in the original paper: we use 10 previous frames to predict the following 20 frames.

Another strategy we tried to employ is a variation of scheduled sampling [14]. During training our autoregressive decoder models a following function:

$$f(h_{i-1}, \tilde{m}_{i-1}, E) = m_i, \tag{2}$$

where h_{i-1} is decoder's hidden state, $\tilde{m}_{i-1}$ is the true motion on a previous time step and E corresponds to encoder states. During teacher forcing we replace the real motion $\tilde{m}_{i-1}$ with previously generated motion m_{i-1} with a probability of 0.5. The main idea behind it is to help the model to explore the error space and become more robust. In the end we found out that not supplying real poses at all was the best option and the rest of our models are using their own predictions during training, just as it would happen during inference. This may be attributed to variance loss: the model was rewarded for making different poses, which likely resulted in a pretty constant deviation from true poses.

We also do not save hidden state of encoder and decoder between batches during training. Each training sample is processed individually without knowledge of previous time period, but during inference the model always supplies it's state for the next segment. This may be the reason behind choppiness in predicted movement. We used smoothing to eliminate this shortcoming.

We'd like to state that our evaluation of hyperparameters is rather subjective: all the changes were judged by a small group of people on a one-minute sample from the validation recording. It is quite possible that we misjudged some of our experiments because of an unsuitable time sector or a simple human error.

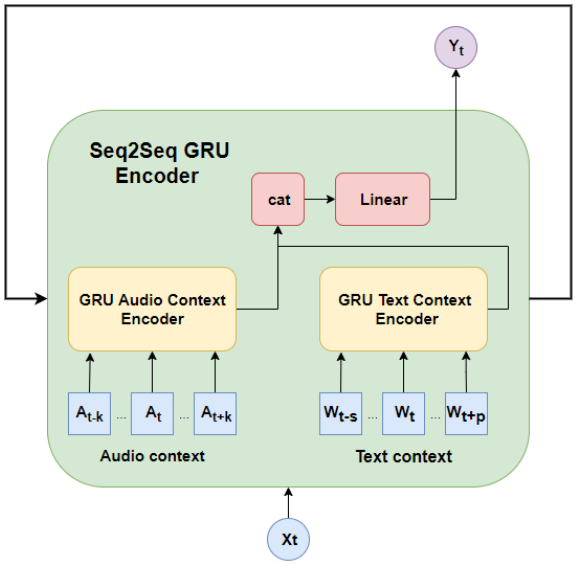

Figure 2: Scheme of contextual encoder.

4.2 Contextual encoder

The second model is inspired by [1]. We have decided to keep sequence to sequence model and enhance it with contextual representations. In our basic sequence to sequence encoder each input corresponds to a single frame.

We decided to represent each frame as a 3-second window around it, which resulted in 61 frames. We used two additional GRU encoders to encode the audio and textual context window as displayed on Figure 2. The audio encoder consists of 3 linear layers with batch normalization and RELu activation. Those layers are used to project audio features for the one-layer one-directional GRU. All audio encoder layers have hidden size 150. The textual encoder is bidirectional one-layer GRU over GloVe embeddings and hidden size of encoder is similar to the embeddings size which is 100. The outputs of both context encoders are concatenated and projected to be passed as inputs to the seq2seq encoder with hidden dimension of 150 units. The rest of the model is a simple sequence to sequence architecture with attention, which was described earlier.

We train this model with Adam optimizer with the learning rate of 0.001 and the batch size of 50. The final model was trained for the 100 epochs, however the target loss stabilized after 80th epoch. Furthermore, motions generated after 80th and 100th epochs were virtually identical.

4.3 Adversarial training

Even a single speaker has a significant variation of his movements even in extremely similar situations, same phrases and contexts. However, so far we described only models which tried to recreate the same movements as the ground truth, even if it was not the only correct behaviour, but one of the many possible motions. To try to overcome this problem we used adversarial training (as done, i.e. in [6, 4]).

The generator model produces motions from audio, while discriminator model tries to classify real and generated motions. The generator loss is

$$L_G = L_{base}(G) + \lambda L_{adv}(G, D), \tag{3}$$

where L_{base} contains whatever non-adversarial components of generator loss and L_{adv} represents adversarial loss with weight λ. In all of our experiments we used non-saturating GAN loss.

We tried several discriminator models based on blocks of (1D convolution, 1D batch normalization, LeakyRelu(0.2)). After series of that blocks we flatten the outputs of convolutional block and apply two

more linear layers. We varied total number of blocks from 2 to 6 with at least two of them reducing spatial dimension (stride > 1).

Unfortunately, the training with adversarial loss was not stable (especially for relatively high λ values around 10.0). Sometimes we got interesting and diverse results (mostly for small λ values around 0.1), however the quality was still lacking in comparison with our best model so in the final system adversarial training was not used.

5 Results and discussion

The challenge organizers used two human-evaluation metrics for evaluation:

- **Human-likeness** - the generated motion should be realistic for human. The evaluation participants should score the motion file without audio by this criterion.
- **Appropriateness** - the generated motion should match the corresponding audio. So participants score motion with audio.

Summary statistics (sample median and sample mean) were provided in [17] and are listed in table 1. The challenge organizers provided results for all participating systems (with label SX), baselines (BA[1] and BT[12]), natural (N) and mismatched (M) motion capture. Our system is labeled as SD.

	Human-likeness		Appropriateness	
ID	Median	Mean	Median	Mean
N	$72 \in [70, 75]$	67.6 ± 1.8	$81 \in [79, 83]$	73.8 ± 1.8
M	"	"	$56 \in [53, 59]$	53.3 ± 2.0
BA	$46 \in [44, 49]$	46.2 ± 1.7	$40 \in [38, 41]$	40.4 ± 1.8
BT	$55 \in [53, 58]$	54.6 ± 1.8	$38 \in [35, 40]$	38.5 ± 1.9
SA	$38 \in [35, 41]$	40.1 ± 1.9	$35 \in [31, 37]$	36.4 ± 1.9
SB	$52 \in [50, 55]$	52.8 ± 1.9	$43 \in [40, 45]$	43.3 ± 2.0
SC	$57 \in [55, 60]$	55.8 ± 1.9	$50 \in [48, 52]$	50.6 ± 1.9
SD	$60 \in [57, 61]$	58.8 ± 1.7	$49 \in [46, 50]$	48.1 ± 1.9
SE	$49 \in [47, 51]$	49.6 ± 1.8	$47 \in [44, 49]$	45.9 ± 1.8

Table 1: Summary statistics of user-study ratings

Our systems were built upon baselines, which allows us to estimate the importance of proposed modifications. Our final submission has the highest median score among the participating systems and baselines. It also has the second highest score by appropriateness. Although our system shows strong improvement over baselines, it is still far behind human generated motions.

Challenge organizers used a special set of mismatched audios and human motions during the human evaluation. This approach was not surpassed by any system. That means that our synthetic generated motions are significantly less appropriate than random human movement.

To select the best model we compared them on validation data using human evaluation among the members of our team. The seq2seq model with contextual encoder was unanimously chosen as the best model, however seq2seq with attention over text and audio was a close second.

We found out that our team was looking for specific sorts of movements during the motion evaluation: we generally were looking for correspondence between motions and verbal pauses. We were more inclined to vivid movements, even if they were choppy, and last but not least - we were always looking for fast and sharp movements coinciding with loud and aggressive speech patterns.

Our humble human evaluation has come to a conclusion, that the approach with context encoder helps to make generated motions smoother, because is uses more information, especially for the last frames in a sequence, while basic seq2seq heavily relies on smoothing.

6 Conclusion

In this paper we proposed several modifications for existing approaches in co-speech gesture generation. In our approach we combined text and audio features and thus were able to outperform text- and

audio-only baselines. Our models were rated highest for Human-likeness metric and second highest for appropriateness, however compared with the real data (human gestures) there is a striking gap in our system's performance and real motions, meaning that there is still a lot to be improved.

Our team also did not explore various sound preprocessing techniques, which could result in a more high-dimensional vector input representation, which would allow models to extract a more rich set of features.

We believe that future research should focus on multimodal representations. We also believe that the quality of generated motions will increase with the expansion of the dataset, which will enable the researchers to train more sophisticated models, like GANs or transformers.

7 Acknowledgements

The reported study was funded by RFBR according to the research project № 20-31-90051

References

[1] Analyzing input and output representations for speech-driven gesture generation / Taras Kucherenko, Dai Hasegawa, Gustav Eje Henter et al. // Proceedings of the ACM International Conference on Intelligent Virtual Agents. — IVA '19. — 2019. — P. 97–104.

[2] Davis Steven, Mermelstein Paul. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences // IEEE transactions on acoustics, speech, and signal processing. — 1980. — Vol. 28, no. 4. — P. 357–366.

[3] Ferstl Ylva, McDonnell Rachel. Investigating the use of recurrent motion modelling for speech gesture generation // IVA '18 Proceedings of the 18th International Conference on Intelligent Virtual Agents. — 2018. — Nov. — Access mode: `https://trinityspeechgesture.scss.tcd.ie`.

[4] Generative adversarial nets / Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza et al. // Advances in neural information processing systems. — 2014. — P. 2672–2680.

[5] Knapp Mark L, Hall Judith A, Horgan Terrence G. Nonverbal communication in human interaction. — Cengage Learning, 2013.

[6] Learning individual styles of conversational gesture / Shiry Ginosar, Amir Bar, Gefen Kohavi et al. // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. — 2019. — P. 3497–3506.

[7] Learning phrase representations using RNN encoder-decoder for statistical machine translation / Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre et al. // arXiv preprint arXiv:1406.1078. — 2014.

[8] Luong Minh-Thang, Pham Hieu, Manning Christopher D. Effective approaches to attention-based neural machine translation // arXiv preprint arXiv:1508.04025. — 2015.

[9] Matsumoto David, Frank Mark G, Hwang Hyi Sung. Nonverbal communication: Science and applications. — Sage Publications, 2012.

[10] Pennington Jeffrey, Socher Richard, Manning Christopher D. Glove: Global vectors for word representation // Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). — 2014. — P. 1532–1543.

[11] Realtime multi-person 2d pose estimation using part affinity fields / Zhe Cao, Tomas Simon, Shih-En Wei, Yaser Sheikh // Proceedings of the IEEE conference on computer vision and pattern recognition. — 2017. — P. 7291–7299.

[12] Robots learn social skills: End-to-end learning of co-speech gesture generation for humanoid robots / Young-woo Yoon, Woo-Ri Ko, Minsu Jang et al. // Proceedings of the IEEE International Conference on Robotics and Automation. — ICRA '19. — 2019. — P. 4303–4309.

[13] Savitzky Abraham, Golay Marcel JE. Smoothing and differentiation of data by simplified least squares procedures. // Analytical chemistry. — 1964. — Vol. 36, no. 8. — P. 1627–1639.

[14] Scheduled sampling for sequence prediction with recurrent neural networks / Samy Bengio, Oriol Vinyals, Navdeep Jaitly, Noam Shazeer // Advances in Neural Information Processing Systems. — 2015. — P. 1171–1179.

[15] Sutskever Ilya, Vinyals Oriol, Le Quoc V. Sequence to sequence learning with neural networks // Advances in neural information processing systems. — 2014. — P. 3104–3112.

[16] To err is human (-like): Effects of robot gesture on perceived anthropomorphism and likability / Maha Salem, Friederike Eyssel, Katharina Rohlfing et al. // International Journal of Social Robotics. — 2013. — Vol. 5, no. 3. — P. 313–323.

[17] A large, crowdsourced evaluation of gesture generation systems on common data: The GENEA Challenge 2020 / Taras Kucherenko, Patrik Jonell, Youngwoo Yoon et al. // 26th International Conference on Intelligent User Interfaces. — 2021. — P. 11–21.