

LLM-driven Multimodal and Multi-Identity Listening Head Generation

Supplementary Material

1. Preliminaries of Large Language Model

Large Language Models (LLMs) take a sequence of text tokens as input and output a distribution over the vocabulary for the next token. The first layer of the LLM is the embedding $E_{word}(\cdot) \in \mathbb{R}^{V_w \times d_w}$, where V_w represents the vocabulary size and d_w is the embedding dimension. $E_{word}(\cdot)$ converts input token indices into embeddings $\{e_1, \dots, e_H\}$, where $e_i \in \mathbb{R}^{d_w}$ and H is the number of input tokens. Additionally, we use a position embedding matrix $P \in \mathbb{R}^{H \times d_w}$ to add positional information to each token's embedding and have $e'_i = e_i + P_i$. Subsequently, let $\{a_1, \dots, a_H\}$ be the input vector sequence to an attention layer in the LLM, represented as matrix $A \in \mathbb{R}^{H \times d_w}$, we use linear projections to generate query, key, and value matrices $Q, K, V \in \mathbb{R}^{H \times d_w}$, respectively. Self-attention is computed between Q and K to obtain attention weights, which are then used to weight V to produce output A' :

$$A' = \text{softmax}(M_T \odot \frac{QK^T}{\sqrt{d_w}}V) \quad (1)$$

where \odot denotes the element-wise product, and $M_T \in \mathbb{R}^{H \times H}$ is a matrix where elements on and below the diagonal are 1, and others are $-\infty$. Finally, A' is fed into LayerNorm and a feedforward network to generate the input to the subsequent layers. The last layer of the LLM is an affine projection layer that predicts the probability of the next token in the sequence.

2. Identity-disentangled VQ-VAE

2.1. Architecture Details

As shown in Fig. 1, we employ a standard CNN-based architecture with 1D convolution (Conv1D), residual block (ResBlock) and ReLU activation for both the encoder and decoder of our identity-disentangled VQ-VAE. Specifically, the residual block in the encoder uses instance normalization to eliminate identity-specific variations, and the residual block in the decoder uses adaptive instance normalization to inject identity information back into the response synthesis process. We use convolution layers with stride 2 and nearest interpolation for temporal downsampling and upsampling operations, respectively. We use $L = 3$ as the number of residual blocks, so the overall downsampling rate $r = 2^3 = 8$.

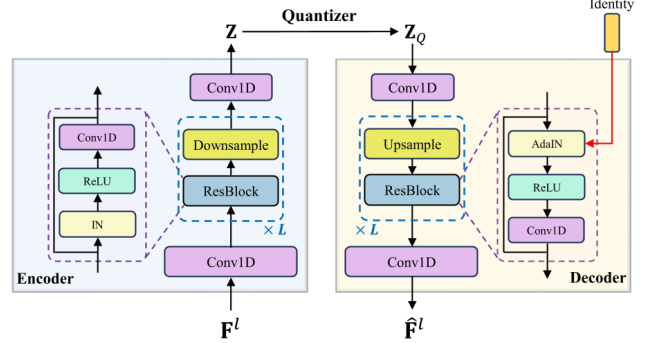


Figure 1. Architecture of the Encoder and Decoder in the proposed Identity-disentangled VQ-VAE.

2.2. Losses Details

We use a set of different losses to train our VQ-VAE:

$$\mathcal{L}_{embed} = \sum_{t=1}^{T/r} \|z_t - sg[Q(z_t)]\|_2, \quad (2)$$

$$\mathcal{L}_{commit} = \sum_{t=1}^{T/r} \|sg[z_t] - Q(z_t)\|_2, \quad (3)$$

$$\mathcal{L}_{rec} = \sum_{t=1}^T \mathcal{L}_1^{smooth}(\hat{f}_t, f_t), \quad (4)$$

$$\mathcal{L}_{veloc} = \sum_{t=1}^{T-1} \mathcal{L}_1^{smooth}(\hat{f}_{t+1} - \hat{f}_t, f_{t+1} - f_t), \quad (5)$$

where sg denotes the stop-gradient operator, and \mathcal{L}^{smooth} denotes the L1 smooth loss function. The final loss function for our VQ-VAE is defined as:

$$\mathcal{L}_{vq} = \mathcal{L}_{embed} + \lambda_{commit}\mathcal{L}_{commit} + \mathcal{L}_{rec} + \lambda_{veloc}\mathcal{L}_{veloc}, \quad (6)$$

where $\lambda_{commit} = 0.02$ and $\lambda_{veloc} = 0.5$ are the weights used to balance individual losses.

3. Speaker Emotion Details

We first downsample the sequence of speaker facial motions $\mathbf{F}^s = \{f_1^s, f_2^s, \dots, f_T^s\}$ by a rate of r , dividing \mathbf{F}^s into T/r groups. In this way, the groups can be represented as:

$$\mathbf{F}_j^s = (f_{(j-1)r+1}^s, f_{(j-1)r+2}^s, \dots, f_{jr}^s), \quad (7)$$

where $j = 1, 2, \dots, T/r$. For each \mathbf{F}_j^s , we use the emotion recognition module (ER) from EMOCA [1] to predict

the emotion probability distribution for each facial motion within the group, and have:

$$p_k = \text{ER}(f_k^s) \in \mathbb{R}^{N_e}, \quad (8)$$

where $k = (j-1)r+1, (j-1)r+2, \dots, jr$, N_e is the number of emotion states. Then, we average the emotion probability distributions across all motions in the group, yielding:

$$p = \frac{1}{r} \sum_{k=(j-1)r+1}^{jr} p_k \in \mathbb{R}^{N_e} \quad (9)$$

Finally, the emotion state with the highest probability is selected as the emotion token for that group’s facial motions. This process is repeated for all groups, resulting in the final speaker emotion tokens $\mathbf{emo} \in \mathbb{R}^{(T/r) \times 1}$, where \mathbf{emo}_i , $i = \{1, \dots, T/r\}$, is the emotion token for group i .

4. Training Details

(i) For VQ-VAE, we use an AdamW [4] optimizer with $[\beta_1, \beta_2] = [0.9, 0.99]$ and a batch size of 256. We train 200K iterations with a learning rate of $2e^{-4}$ and 100K iterations with a learning rate of $1e^{-5}$. (ii) For the language model, we use an AdamW [4] optimizer with $[\beta_1, \beta_2] = [0.5, 0.99]$ and a batch size of 8. First, we train 60K iterations using \mathcal{L}_{pre} with a learning rate of $5e^{-5}$. Second, we train 150K iterations using \mathcal{L} with a learning rate of $5e^{-5}$, and the learning rate is decayed to $2.5e^{-6}$ for another 50K iterations. Training the VQVAE and the language model on a single NVIDIA A100-40G GPU takes about 22 hours and 38 hours respectively.

5. Inference Efficiency

Our inference speed is 393.1 FPS, with minimal additional computational overhead against LM-listener (405.7 FPS). Specifically, we additionally: i) introduce SpeechTokenizer and EMOCA to extract additional information, extending the input token sequence, and ii) add several AdaIN layers during response token decoding. Preprocessing a 10-minute video with SpeechTokenizer and EMOCA takes 2.05 and 8.66 minutes, respectively. Parallel preprocessing and end-to-end pipelines can further improve the efficiency of our framework.

6. Metrics for Quantitative Results

Following [5], we evaluate our method based on realism ($L2$ and *Frechet Distance (FD)*), diversity (*Variation* and *Diversity*), and synchrony (*Paired FD (P-FD)* and *L2 Affect*), whose details are provided as follows:

- *L2*: Distance to ground truth expression and pose coefficients.

- *Frechet Distance (FD)*: Motion realism measured by distribution distance between generated and ground-truth facial motion. We calculate FD [3] scores in the expression $\mathbb{R}^{T \times d_\psi}$ and pose $\mathbb{R}^{T \times d_\theta}$ space of the full facial motion sequence.
- *Variation*: Variance calculated across the sequence of expression and pose coefficients.
- *Diversity*: Following [6], we randomly sample 30 pairs of listener expression and pose coefficients within a sequence, and compute the average Euclidean distances between the pairs to measure motion diversity in the set.
- *Paired FD (P-FD)*: Quality of listener-speaker dynamics measured by distribution distances on listener-speaker pairs. Specifically, we calculated FD [3] scores on concatenated listener-speaker expression $\mathbb{R}^{T \times (d_\psi + d_\psi)}$ and pose $\mathbb{R}^{T \times (d_\theta + d_\theta)}$.
- *L2 Affect*: Measures the accuracy of the produced listener facial affect across the sequence. We average listener facial affect over a 1-second window and compute the L2 against ground truth in a sliding-window manner.

7. Metrics for VQ-VAE Evaluation

We evaluate VQ-VAE performance using three metrics: *Reconstruction (Rec.)*, *Commitment (Commit.)*, and *Perplexity (PPL)*:

- *Reconstruction (Rec.)*: Distance to the ground truth expression and pose coefficients.
- *Commitment (Commit.)*: Mean Squared Error (MSE) between the output feature of the encoder and the output token of the quantizer.
- *Perplexity (PPL.)*: The entropy of the token distribution from the quantizer, where a higher value indicates a uniform distribution across all tokens, and a lower value indicates a concentration on specific tokens.

8. User Study Details

We conducted an online anonymous survey (questionnaire) with university student volunteers. Our survey first presents the survey topic and evaluation metrics (contextual consistency and synchrony), and shows one video at a time. Users could watch the videos for an unlimited duration and rank them based on their preferences. To further illustrate user preferences in the user study, we assign scores ranging from 3 to 1, corresponding to the ranking results of the volunteers for the three responses: **GT**, **LM-listener** [5], and **Ours**, from highest to lowest. As shown in Tab. 1, **Ours** significantly outperforms the **LM-listener** [5] and slightly surpasses **GT**. Using multimodal speaker cues, our method occasionally generated more diverse responses, surpassing the ground truth video, which often remained calm on such occasions.

Method	Video1	Video2	Video3	Video4	Video5	Video6	Video7	Video8	Video9	Video10	Average
GT	2.36	1.73	1.97	2.00	2.12	2.33	1.94	2.03	2.48	2.55	2.15
LM-listener	1.24	1.36	1.24	1.33	1.15	1.18	1.39	1.42	1.61	1.33	1.33
Ours	2.39	2.91	2.19	2.67	2.73	2.48	2.67	2.55	1.91	2.12	2.46

Table 1. Additional User Study Results. Scale: 1-3; the higher, the better.

Method	MultiModal	Identity	L2 ↓	FD ↓	Variation	Diversity	P-FD ↓	L2 Affect(10 ²) ↓
GT					0.1148	2.6053		
Naive	×	×	0.2026	9.6016	0.0313	1.5817	9.9899	11.6484
MM only	✓	×	0.1794	8.2897	0.0297	1.5629	8.6663	10.6285
ID only	×	✓	0.0882	3.5803	0.0123	1.0175	3.8507	5.7948
Full (Ours)	✓	✓	0.0860	3.3939	0.0130	1.0426	3.6768	5.2537

Table 2. Ablation study of the different components (Multimodal-LM and Identity-disentangled VQ-VAE) on the RealTalk dataset [2].

9. Effectiveness of Different Components Results on RealTalk

As shown in Tab. 2, the performance trends are consistent with the results on L2L-trevor [5], confirming that the performance gains can generalize across different datasets.

10. Limitations and Future Work

Our method, while effective, has the following limitations to be addressed in future work:

- *Global Context.* In addition to using only the speaker’s video as input, we can incorporate global context information to assist the model in better understanding the speaker’s intent. This includes contextual factors such as the setting of the conversation and the relationship between the speaker and the listener.
- *Identity Priors.* We can utilize additional identity information as priors to refine the representation of the listener’s identity, such as age, gender, and personality, enabling a more comprehensive modeling of the listener’s response style.
- *Unified Speaker-Listener Generation.* We aim to unify Talking Head Generation and Listening Head Generation within a single LLM model. By using prompts to facilitate identity switching between the speaker and the listener, the model can exhibit different functionalities (outputting conversational content or responses) as the conversation progresses, thereby creating highly interactive digital humans.

References

- [1] Radek Daněček, Michael J Black, and Timo Bolkart. Emoca: Emotion driven monocular face capture and animation. In

Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 20311–20322, 2022. 1

- [2] Scott Geng, Revant Teotia, Purva Tendulkar, Sachit Menon, and Carl Vondrick. Affective faces for goal-driven dyadic communication. *arXiv preprint arXiv:2301.10939*, 2023. 3
- [3] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in Neural Information Processing Systems*, 30, 2017. 2
- [4] I Loshchilov. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 2
- [5] Evonne Ng, Sanjay Subramanian, Dan Klein, Angjoo Kanazawa, Trevor Darrell, and Shiry Ginosar. Can language models learn to listen? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10083–10093, 2023. 2, 3
- [6] Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Yong Zhang, Hongwei Zhao, Hongtao Lu, Xi Shen, and Ying Shan. Generating human motion from textual descriptions with discrete representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14730–14740, 2023. 2