# Evaluation of Emotion-Conditioned Response Generation for Role-Playing Agents Using Large Language Models: A Case Study with Facial Expression Labels from Visual Novel Data

Shinji Muraji[1,*], Rafal Rzepka[1] and Toshihiko Itoh[1]

[1]Hokkaido University, Kita 14, Nishi 9, Kita-ku, Sapporo, Hokkaido, 060-0814, Japan

**Abstract**

In recent years, research on role-playing agents that leverage the powerful conversational capabilities of large language models (LLMs) has been actively conducted. While many studies have explored how LLMs can generate utterances that reflect a character's individuality, the examination of utterance generation conditioned on the target character's emotions has not been sufficiently conducted. To address this gap, we used scenario data from a visual novel game in which *facial expressions*, one of the key components of emotional information, are explicitly annotated. We trained an LLM to generate responses conditioned on facial expression labels by incorporating these labels into the input during fine-tuning, and evaluated the resulting utterances in terms of their character-likeness. Our experiments revealed three key findings: (1) conditioning on facial expression labels improves the perceived character-likeness of the generated utterances; (2) providing the LLM with facial expression labels that do not correspond to the character's actual expressions can still enhance the perceived character-likeness of the generated utterances; and (3) there remains a gap between how humans and LLMs process emotional information in dialogue generation.

**Keywords**

Emotional Intelligence, Role-Playing Agent, Large Language Model, Emotion Processing, Facial Expression Label, Character-Likeness, Visual Novel, Dialogue Generation

## 1. Introduction

Recent advances in large language models (LLMs) have enabled dialogue systems to produce utterances that convey individuality. To achieve consistent and engaging role-specific behavior, many studies have developed role-playing agents that emulate particular characters, evaluating their ability to reproduce the target character's knowledge, personality, and linguistic style [1, 2].

However, the emotional expressiveness of such agents has largely remained a black box. Emotions play a central role in defining individuality, yet previous works rarely treat emotional information as an explicit conditioning factor in role-playing generation. To address this gap, we investigate whether explicitly conditioning utterance generation on *facial expressions*, a key form of emotional information, enhances the perceived character-likeness of generated responses. An overview of this concept is shown in Figure 1.

Collecting reliable emotion labels for dialogue data is challenging due to subjective variation among annotators, making it difficult to obtain ground-truth emotional information for a target character. As a result, it has not been fully investigated whether conditioning on emotional information can improve the perceived character-likeness of generated utterances. In contrast, visual novel games provide naturally aligned multimodal data in which each utterance is explicitly paired with the character's facial expression in the original work. We treat these facial expressions as ground-truth emotional information and use them to condition LLM-based utterance generation, evaluating whether such conditioning improves the role-playing consistency of responses.

*LaCATODA 2026: The 10th Linguistic and Cognitive Approaches to Dialog Agents Workshop, Jan 26, 2026, Tampines, Singapore*
*Corresponding author.
twr427m@gmail.com (S. Muraji); rzepka@ist.hokudai.ac.jp (R. Rzepka); t-itoh@ist.hokudai.ac.jp (T. Itoh)

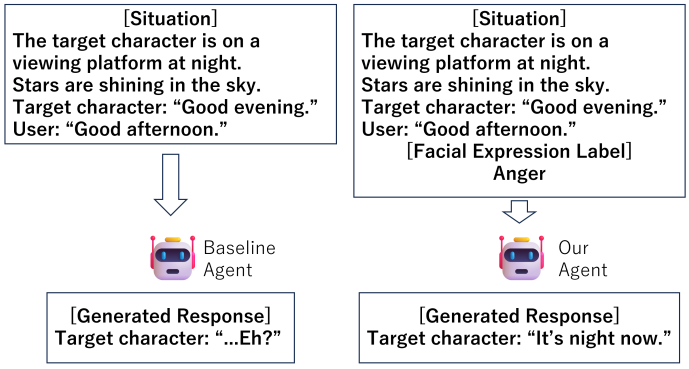


**Figure 1:** Conceptual illustration of response generation conditioned on facial expressions. The left example uses the situation only, while the right example conditions on both the situation and the facial expression label.

Furthermore, to examine the role of label reliability, we compare three conditions: (1) using ground-truth expressions from the game, (2) using LLM-generated expressions predicted from the scene, and (3) using randomly assigned expressions. Through this comparison, we explore how the source and accuracy of emotional information affect the character-likeness of generated utterances.

## 2. Related Work

### 2.1. Role-Playing and Emotion-Conditioned Agents

Prior research on role-playing agents generally follows two main approaches: prompt-based and fine-tuning strategies [1, 2, 3, 4]. Prompt-based methods aim to elicit character-like behavior through instruction design or retrieval, whereas fine-tuning methods adapt model parameters for specific characters. Emotion-related work includes EmoCharacter, a benchmark for assessing the emotional fidelity of role-playing agents, which reports that fine-tuning on real dialogue data and in-context learning can improve emotional fidelity [5]. In parallel, EmotionalRAG introduces emotion-aware retrieval to enhance role-playing without relying on fine-tuning the base LLM [6].

### 2.2. Emotion-Annotated Dialogue Datasets

Emotion recognition in dialogue involves labeling utterances with emotion categories. Representative datasets include MELD [7], where each utterance is annotated with Ekman’s six basic emotions plus Neutral, and EmoryNLP [8], which uses labels drawn from Willcox’s Feeling Wheel. These datasets highlight the high subjectivity and annotation difficulty in emotion-labeling. In contrast, our work uses character facial expressions depicted in visual novel scenes as author-defined emotional signals for conditioning response generation.

## 3. Modeling Role-Playing Agents

We aim to develop an agent that incorporates the target character’s emotional information into its training process. Following previous studies, we construct a role-playing agent using a large language model (LLM) and employ both a role-playing-oriented prompt template and a fine-tuning strategy. The role-playing-oriented prompt template is designed to leverage the LLM’s ability to understand instructions and utilize general factual knowledge about the target character. Meanwhile, the fine-tuning strategy adjusts the model’s internal parameters so that it can reproduce finer aspects of the character’s linguistic style and factual consistency while effectively utilizing emotional information. Each strategy is described in detail below.

### 3.1. Prompt Template

We designed a static prompt template for role-playing to elicit the LLM's character-consistent responses during both training and evaluation. Each prompt includes a brief Wikipedia-based description of the target character and specifies either the *situation* alone or both the *situation* and *facial expression* as conditioning information. A detailed explanation and the full prompt structure are provided in Appendix A.

### 3.2. Fine-Tuning Strategy

We fine-tune a large language model (LLM) to generate responses conditioned on the target character's emotional information. To isolate the effect of emotional conditioning, we prepare two core models: one fine-tuned on the situation only (*baseline agent*) and another fine-tuned on both the situation and the corresponding facial expression (*facial expression-conditioned agent*).

#### 3.2.1. Baseline agent

The baseline agent is trained using (*situation*, *utterance*) pairs, where the *situation* consists of the preceding ten lines of the novel-style scenario text describing dialogue turns, scene setting, and character actions. The model is optimized to maximize the likelihood of reproducing the target character's actual utterances from these contexts.

#### 3.2.2. Facial Expression-Conditioned Agent

The facial expression-conditioned agent extends this setting by incorporating an additional conditioning signal: the facial expression label corresponding to each utterance. It is trained on (*situation*, *facial expression*, *utterance*) triples, learning to generate utterances that reflect both the narrative context and the character's emotional state.

To further examine whether the use of ground-truth facial expressions is necessary, we compare three training variants: (1) using the original ground-truth expressions, (2) using LLM-inferred expressions from the situation text, and (3) using randomly assigned expressions. This comparison allows us to assess whether plausible but non-authentic emotional information can still enhance the perceived character-likeness of generated utterances.

## 4. Construction of a Facial Expression-Annotated Dataset

To train the proposed role-playing agents and examine the effect of emotional information, we construct triplets of (*situation*, *facial expression*, *utterance*) from a visual novel game.

### 4.1. Data Collection

In collaboration with and under the consent of the game's developer, we build a dataset for one target character from a commercial visual novel. Although expanding to multiple characters is desirable, the construction and human evaluation costs are high; therefore, we report results for a single character in this paper. The developer granted permission to use the data and publish the results, but the title and character name had to be withheld. Instead, we report sufficient aggregate statistics to characterize the dataset.

A visual novel presents text in a novel-like format, accompanied by images and voices attached to each script line. A brief overview of this format is given in Appendix B. Each image depicts the current facial expression of a character, which we label and use as conditioning input for the LLM.


**Figure 2:** Example screenshot of a visual novel game. The target character (right) and her friend (left) are shown in a typical dialogue scene. An example of a scene transition in the game is provided in the Appendix B. All illustrations were created by the first author. The original graphical data cannot be shown in this paper due to copyright restrictions.

**Table 1**
Example of the data elements and structure extracted from the visual novel script. Each script line is associated with a facial expression label representing the target character's expression in that scene. Narrative sentences from the protagonist's point of view are also included in the script lines, where occurrences of "I" outside quotation marks refer to the protagonist. In this example, "She" refers to the target character.

| Script line | Facial expression label |
|---|---|
| When I went up to the rooftop, the target character and her friend were there. | Target character's expression: [Anger] |
| Target character: "What are you doing here?" | Target character's expression: [Anger] |
| Protagonist: "I came to apologize." | Target character's expression: [Surprise] |
| She still seemed angry. | Target character's expression: [Suspicion] |
| Target character: "Are you serious?" | Target character's expression: [Suspicion] |

## 4.2. Overall Data Structure

An example of a visual novel game screen is shown in Figure 2. The major elements are highlighted with circles, and their corresponding names are indicated in italics. Note that this example was created for explanatory purposes and is not part of the actual game scenario. The game script consists of narrative text that describes the protagonist's inner thoughts and the scene, as well as character utterances with explicit speakers. Although the narrative parts could be considered noise, rewriting them from the protagonist's perspective into another character's viewpoint would be impractical. Therefore, in this study, we use the original script text as contextual descriptions of the situations in which utterances occur.

Each line in the script corresponds to one or more sentences and is linked to the facial expressions of the characters shown in that scene. We extract the *target character's facial expression* for each line; when the character is absent, the most recent expression is retained. An example is shown in Table 1, illustrating how each script line is paired with the target character's facial expression.

From data structured as shown in Table 1, we use the target character's utterances as anchors to extract the preceding ten lines of the script as the *situation*, and the facial expression at the time of the utterance as the *facial expression*. If fewer than ten preceding lines exist, we use all available preceding lines as the context. When a scene transition occurs in the game, we treat it as a new scene and construct separate data for each scene. An example of a resulting triplet generated from the final target utterance in Table 1 is shown in Table 2.

**Table 2**
Example of a constructed triplet used for training. The preceding ten lines of the script are used as the *situation*, the corresponding facial expression label as the *facial expression*, and the target character's utterance as the *utterance*.

| Situation (preceding context) | Facial expression | Utterance |
|---|---|---|
| When I went up to the rooftop, the target character and her friend were there.<br>Target character: "What are you doing here?"<br>Protagonist: "I came to apologize."<br>She still seemed angry. | [Suspicion] | Target character: "Are you serious?" |

**Table 3**
Statistics of the collected dataset.

| Category | Total | Train | Validation | Test |
|---|---|---|---|---|
| Number of target utterances | 3,551 | 2,698 | 433 | 420 |
| Number of scenes | 109 | 87 | 11 | 11 |
| Average utterance length (characters) | 11.86 | 11.77 | 13.50 | 10.72 |
| Average situation length (characters) | 179.27 | 178.23 | 195.55 | 169.15 |

### 4.3. Facial Expression Labels

Each facial expression image of the target character was assigned a short, descriptive textual label so that the LLM could process expressions as linguistic inputs rather than categorical codes. Three annotators (including one of the authors) who had played the visual novel collaboratively created these labels, describing each expression in concise natural-language terms. When a single image conveyed multiple aspects, all agreed-upon descriptors were retained. A detailed description of the labeling procedure and examples of the resulting labels are provided in Appendix C.

Although the game includes more than 45 facial images of the target character, several depict nearly identical expressions with different poses. To simplify the learning task and avoid excessive label granularity, we consulted with the game developer and consolidated them into 14 representative facial expressions. An overview of the final label definitions and their distribution is presented in Appendix D.

The distribution of facial expression labels is notably imbalanced, reflecting the frequent occurrence of serious or emotionally intense scenes in the game. Since our experiments evaluate responses within the same narrative context, we did not apply any rebalancing techniques to the dataset.

We also prepared augmented expression labels generated by an open-source LLM from the preceding context in the script to examine whether inferred emotional information could enhance role consistency. Details of this augmentation process and the prompt used are provided in Appendix E.

### 4.4. Dataset Statistics

This section presents the statistics of the collected dataset. To prepare the data for training, we split it into training, validation, and test sets. Because the target character's utterances in the test set might otherwise appear within the situational context of the training set, we perform the split on a *per-scene basis*. Specifically, the 109 total scenes are divided in an 8:1:1 ratio for the training, validation, and test sets, respectively. The resulting dataset statistics are shown in Table 3.

Since the data were collected in Japanese, sequence lengths are measured in characters, not words. Although the dataset focuses on a single target character, it provides a relatively large-scale resource in which each utterance is paired with detailed emotion-related information derived from facial expressions.

## 5. Experimental Setup

We fine-tune the LLM using the dataset constructed in Section 4 and evaluate the generated utterances in terms of semantic similarity and perceived character-likeness. This section describes the fine-tuning details, including hyperparameters, the models used for comparison, and the evaluation methods.

## 5.1. Fine-Tuning Details

We fine-tune an open-source large language model, `Qwen/Qwen3-32B` [9], using 4-bit quantization and the QLoRA framework [10] under a causal language modeling objective. All computations are performed in `bfloat16` precision to reduce memory usage.

The LoRA rank is set to 256, with a dropout rate of 0.1, applied to the standard attention and MLP projection layers. Training uses the AdamW optimizer with a learning rate of $5 \times 10^{-5}$ and a cosine learning rate schedule. We train for five epochs with an effective batch size of 8, and select the model achieving the lowest validation loss as the final checkpoint. Full training hyperparameters are listed in Appendix F.

## 5.2. Comparison Models

From each triplet (*situation*, *facial expression*, *utterance*) in the test set, Each model generates utterances based on either the situation alone or both the situation and the facial expression. The models compared in this study are as follows:

- **Baseline agents (situation only)**
  - **No fine-tuning:** The original pretrained LLM without fine-tuning, using only prompt instructions to generate responses.
  - **Fine-tuned on situation only:** The model fine-tuned to generate utterances conditioned solely on the situation.
- **Emotion-Conditioned Agents (situation + facial expression)**
  - **Ground-truth expression:** Trained on ground-truth facial expressions extracted from the game.
  - **LLM-augmented expression:** Trained on facial expressions automatically inferred by an LLM from the situation.
  - **Random expression:** Trained on randomly assigned facial expression labels.

The "no fine-tuning" model generates responses solely based on the given prompt instructions, allowing us to examine the effect of fine-tuning by comparing it with the situation-only model. For each emotion-conditioned agent, the same type of facial expression labels used for training are also used at inference time.

To further verify whether the agent truly conditions its responses on the given facial expressions, we also test the ground-truth expression model with mismatched expressions at inference time. Specifically, we evaluate two additional settings in which the model trained with ground-truth facial expressions is given LLM-augmented expressions or random expressions during inference. Thus, in total, we compare seven models.

- **Emotion-Conditioned Agents (situation + facial expression)**
  - **Ground-truth expression (inference with LLM-augmented expression)** Trained with ground-truth expressions and evaluated using LLM-augmented expressions at inference time.
  - **Ground-truth expression (inference with random expression)** Trained with ground-truth expressions and evaluated using randomly assigned expressions at inference time.

Table 4 summarizes the training and testing conditions for each model.

## 5.3. Evaluation Method

We evaluate the utterances generated by each agent in terms of two aspects: (1) semantic similarity to the corresponding gold utterance, and (2) perceived character-likeness of the generated utterance with respect to the given situation. Each evaluation method is described in detail below.

**Table 4**
Summary of the training and test data used for each model. "Sit." = situation, "GT" = ground-truth expression, "LLM" = LLM-augmented expression, "Rand." = random expression.

| Model | Training Data | | | | Test Data | | | |
|---|---|---|---|---|---|---|---|---|
| | Sit. | GT | LLM | Rand. | Sit. | GT | LLM | Rand. |
| No fine-tuning | | | | | ✓ | | | |
| Situation only | ✓ | | | | ✓ | | | |
| Ground-truth expression | ✓ | ✓ | | | ✓ | ✓ | | |
| LLM-augmented expression | ✓ | | ✓ | | ✓ | | ✓ | |
| Random expression | ✓ | | | ✓ | ✓ | | | ✓ |
| Ground-truth expression (inference with LLM-augmented expression) | ✓ | ✓ | | | ✓ | | ✓ | |
| Ground-truth expression (inference with random expression) | ✓ | ✓ | | | ✓ | | | ✓ |

#### 5.3.1. Semantic Similarity

Following prior studies [5] , we automatically evaluate the generated utterances by measuring the cosine similarity between their sentence embeddings and those of the corresponding gold utterances. To focus purely on the utterance content, quotation marks and surrounding tokens were removed before embedding. For sentence embedding, we used *plamo-embedding-1b* [11], a model known for its strong Japanese semantic representation capability.

#### 5.3.2. Character-Likeness Evaluation

For a more rigorous evaluation of character-likeness, we conducted a human annotation study. Annotators were presented with pairs of a situation and a generated utterance and asked to rate how consistent the utterance was with the target character across four aspects: overall impression, personality, linguistic style, and knowledge consistency. Each aspect was scored on a 7-point Likert scale. These evaluation criteria were designed with reference to prior studies on role-playing agents [1, 12], focusing on aspects potentially related to emotional expression. A 7-point scale was chosen to capture finer distinctions than a 5-point scale.

Because some gold utterances in the dataset are short and generic (e.g., "Yeah"), it is unrealistic for all gold utterances to consistently receive the maximum score. Therefore, we also included the gold utterances in the annotation process to establish a human upper bound. As a result, each annotation session consisted of eight utterances per situation, including seven generated utterances and one gold utterance, each rated across four criteria.

Since this annotation is costly, we randomly sampled 100 situations from the 420 test cases for evaluation. For each situation, annotators rated the utterances (presented in random order) based on the four aspects above. Each utterance was rated by three different annotators, and the mean of their scores was taken as the utterance score. The model-level score was computed as the average across the 100 situations. When two models generated identical utterances, the utterance was evaluated only once, and the same score was assigned to both.

Annotators were required to be sufficiently familiar with the target character to understand their emotional expressions; therefore, only those who had played through and completed the original visual novel were eligible. A total of nine volunteers (including one author, who evaluated in a blinded setting) participated in the annotation.

To assess the reliability of the human judgments, we measured inter-annotator agreement using Krippendorff's $\alpha$, obtaining a value of $\alpha = 0.44$. Although the task is inherently subjective, this level of agreement indicates a reasonable level of consistency among annotators.

**Table 5**
Semantic similarity (cosine similarity) between generated and reference utterances

| Model | Cosine Similarity (%) |
|---|---|
| No fine-tuning | 63.50 |
| Situation only | 69.91 |
| Ground-truth expression | **71.05** |
| LLM-augmented expression | 70.22 |
| Random expression | 70.52 |
| Ground-truth expression (inference with LLM-aug.) | 70.14 |
| Ground-truth expression (inference with random) | 68.76 |

**Table 6**
Results of human evaluation

| Model | Overall | Personality | Linguistic Style | Knowledge |
|---|---|---|---|---|
| No fine-tuning | 1.83 | 1.95 | 2.04 | 1.90 |
| Situation only | 4.23 | 4.24 | 4.57 | 4.05 |
| Ground-truth expression | 4.24 | 4.25 | 4.54 | 4.04 |
| LLM-augmented expression | **4.42** | **4.45** | **4.63** | **4.18** |
| Random expression | 4.18 | 4.19 | 4.46 | 3.97 |
| Ground-truth (inference: LLM-augmented) | 4.10 | 4.19 | 4.38 | 3.89 |
| Ground-truth (inference: random) | 4.12 | 4.09 | 4.38 | 3.88 |
| Gold | 5.85 | 5.82 | 5.83 | 5.60 |

# 6. Results

In this section, we present the evaluation results of semantic similarity and perceived character-likeness, followed by additional analyses and discussion.

## 6.1. Semantic Similarity Results

The results of the automatic evaluation based on semantic similarity are shown in Table 5. The model trained and inferred with ground-truth facial expressions achieved the highest similarity to the gold utterances, consistent with previous studies. Interestingly, from the perspective of semantic similarity, using random facial expressions during training yielded slightly higher scores than training without any emotional conditioning.

Furthermore, for the model trained with ground-truth expressions, inference with LLM-augmented expressions resulted in higher similarity than inference with random expressions. This indicates that when emotion-conditioned training is employed, conditioning the model on facial expressions closer to the original data can enhance semantic similarity. However, both inference settings performed worse than the model trained and inferred with random expressions, suggesting that when high-quality facial expression information is unavailable at test time, training with actual expressions offers limited advantage in terms of semantic similarity.

## 6.2. Results of Character-Likeness Evaluation

Table 6 shows the results of the human evaluation of character-likeness. We discuss the findings across several aspects below.

### 6.2.1. Effect of Fine-Tuning

The difference between the no fine-tuning model and the situation-only model demonstrates that our fine-tuning setup and data volume were effective. Because we did not apply dynamic retrieval

or knowledge augmentation, the relatively low scores of the no fine-tuning model likely reflect its reliance on prompt instructions alone. The situation-only model achieved around four points on the seven-point Likert scale, corresponding to "neutral" across all criteria, indicating that the adopted fine-tuning strategy raised the perceived quality to an acceptable level.

#### 6.2.2. Effect of Training with Ground-Truth Expressions

When comparing the situation-only model and the ground-truth expression model, little difference was observed. This suggests that conditioning on ground-truth facial expressions did not necessarily make the generated utterances more characteristic of the target character in this experimental setting. However, because the latter model explicitly incorporates emotional information rather than treating it as an implicit factor, it has the potential to enable finer control over the agent's expressive behavior. Further investigation of this controllability is left for future work.

#### 6.2.3. Which Expression Data Should Be Used?

Among the models trained and tested with different types of expression data, the one trained and evaluated with LLM-augmented expressions achieved the highest scores across all criteria, a somewhat unexpected outcome. In principle, the triplets of (situation, expression, utterance) from the original game are internally consistent from a human perspective, and we hypothesized that a model trained on such data would reproduce the character's utterances most faithfully (as supported by its superior semantic similarity). However, the LLM-augmented expressions were inferred solely from the situation and were not guaranteed to align with the actual utterances. Given their low accuracy relative to the true labels, these pseudo-expressions likely introduce noise. Nevertheless, the model trained with them achieved the best human-rated character-likeness. To analyze this unexpected result, we conducted an additional human evaluation to assess whether the generated utterances were consistent with the conditioned facial expressions, as described in the next subsection.

When comparing the ground-truth expression and random expression models, the former achieved slightly higher scores. This indicates that the model did not treat facial expressions merely as random noise but learned some meaningful associations between expressions and utterances. However, since the difference from the situation-only model was marginal, this association was likely weak, potentially due to label imbalance and limited data, which remain an open challenge.

#### 6.2.4. Effect of Changing Test-Time Expressions

Next, we compare cases where the model trained on ground-truth expressions was tested with different types of expressions. When ground-truth expressions were replaced by LLM-augmented or random ones during inference, the scores dropped to the lowest levels among all fine-tuned models. This result suggests that the model had learned dependencies between actual expressions and utterances, and its generation quality degraded when provided with inconsistent or mismatched emotional cues at inference time.

#### 6.2.5. Scores Assigned to Gold Utterances

Finally, we discuss the scores assigned to the gold (original) utterances. The average score for gold utterances was approximately 5.85, indicating that annotators could reliably distinguish authentic lines from generated ones. However, some gold utterances received much lower scores, the lowest being 3.33, showing that even original utterances are not always perceived as "in-character." This highlights that the gold data do not always align with human perception of character-likeness. Therefore, evaluation metrics for role-playing agents should be carefully designed according to the intended purpose of the system.

**Table 7**
Human evaluation of facial-expression consistency

| Model | Character-likeness of Expression |
|---|---|
| Ground-truth Expression | 5.20 |
| LLM-augmented Expression | 4.54 |

### 6.3. Additional Analysis and Discussion

As described in the previous section, the model trained with LLM-augmented facial expressions outperformed the model trained with ground-truth expressions in terms of perceived character-likeness. However, it remains unclear whether the expressions used for conditioning were actually consistent with the generated utterances when viewed by humans. In other words, a triplet of (situation, expression, utterance) might appear inconsistent or unnatural to human observers.

To investigate this, we conducted an additional experiment in which three annotators familiar with the target character (including one of the authors) evaluated whether the conditioned facial expressions matched the generated utterances. Each annotator was first shown a (situation, utterance) pair and then asked: "If the target character were to utter this line in the given situation, would the following facial expression seem appropriate for them?"

Annotators rated the appropriateness of each expression on a seven-point Likert scale. If they could not imagine the target character making the utterance in that situation, they were instructed to select 4 (neutral). Unlike the previous evaluation, which rated character-likeness for *(situation, utterance)* pairs, the present evaluation asks annotators to judge the appropriateness of a *facial expression image* given the same *(situation, utterance)* pair. All evaluations were performed blindly, and annotators did not know which model generated each utterance. The same 100 situations and utterances as before were used, along with the original facial expression images from which each model's expressions were derived during generation.

The results are shown in Table 7. The model trained with ground-truth expressions produced utterances that were judged by humans as more consistent with the given expressions than those generated by the LLM-augmented expression model. This indicates that the latter model learned to associate utterances with expressions that humans did not perceive as coherent.

Nevertheless, the character-likeness scores for the LLM-augmented model were higher overall. This suggests that, for fine-tuning an LLM to generate utterances perceived as more in-character, perfect visual or emotional consistency between expressions and utterances may not be necessary. Even expression labels that appear inconsistent to humans can serve as useful conditioning signals, guiding the model toward more character-like generation.

## 7. Conclusion and Future Work

In this study, we developed a role-playing agent using a static prompt designed for role-playing and a fine-tuning strategy. We examined whether conditioning utterance generation on facial expressions, an essential component of emotional information, would lead to responses that appear more consistent with the target character.

Experimental results showed that when a pretrained LLM inferred facial expressions from context and used those inferred labels for conditional learning, the generated utterances were perceived as more character-like. In contrast, the model trained with ground-truth facial expressions did not achieve higher character-likeness scores than the situation-only model, although it produced utterances that were more consistent with the conditioned expressions than those of the LLM-augmented expression model. These findings suggest a gap between the expression–utterance correspondence perceived by humans and the correspondence that facilitates character-like generation in LLMs.

To enable detailed analysis, we limited our dataset to a single target character, allowing us to collect high-quality expression–utterance pairs and conduct fine-grained evaluations. As future work, we plan

to extend our experiments to multiple characters to investigate inter-character differences in emotional modeling. Furthermore, because our current approach did not employ retrieval augmentation, the model's expressive diversity may have been constrained by its internal parameters. We therefore intend to explore retrieval-based augmentation of character information to better leverage emotional cues such as facial expressions and other affective signals during generation.

## Declaration on Generative AI

During the preparation of this work, the author(s) used ChatGPT-5 in order to: Grammar and spelling check. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

## References


[1] X. Wang, Y. Xiao, J.-t. Huang, S. Yuan, R. Xu, H. Guo, Q. Tu, Y. Fei, Z. Leng, W. Wang, J. Chen, C. Li, Y. Xiao, InCharacter: Evaluating personality fidelity in role-playing agents through psychological interviews, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 1840–1873. URL: https://aclanthology.org/2024.acl-long.102/. doi:10.18653/v1/2024.acl-long.102.

[2] Y. Shao, L. Li, J. Dai, X. Qiu, Character-LLM: A trainable agent for role-playing, in: H. Bouamor, J. Pino, K. Bali (Eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, 2023, pp. 13153–13187. URL: https://aclanthology.org/2023.emnlp-main.814/.

[3] C. Li, Z. Leng, C. Yan, J. Shen, H. Wang, W. MI, Y. Fei, X. Feng, S. Yan, H. Wang, L. Zhan, Y. Jia, P. Wu, H. Sun, Chatharuhi: Reviving anime character in reality via large language model, 2023. URL: https://arxiv.org/abs/2308.09597. arXiv:2308.09597.

[4] N. Wang, Z. Peng, H. Que, J. Liu, W. Zhou, Y. Wu, H. Guo, R. Gan, Z. Ni, J. Yang, M. Zhang, Z. Zhang, W. Ouyang, K. Xu, W. Huang, J. Fu, J. Peng, RoleLLM: Benchmarking, eliciting, and enhancing role-playing abilities of large language models, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Findings of the Association for Computational Linguistics: ACL 2024, Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 14743–14777. URL: https://aclanthology.org/2024.findings-acl.878/. doi:10.18653/v1/2024.findings-acl.878.

[5] Q. Feng, Q. Xie, X. Wang, Q. Li, Y. Zhang, R. Feng, T. Zhang, S. Gao, EmoCharacter: Evaluating the emotional fidelity of role-playing agents in dialogues, in: L. Chiruzzo, A. Ritter, L. Wang (Eds.), Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), Association for Computational Linguistics, Albuquerque, New Mexico, 2025, pp. 6218–6240. URL: https://aclanthology.org/2025.naacl-long.316/. doi:10.18653/v1/2025.naacl-long.316.

[6] L. Huang, H. Lan, Z. Sun, C. Shi, T. Bai, Emotional RAG: Enhancing Role-Playing Agents through Emotional Retrieval , in: 2024 IEEE International Conference on Knowledge Graph (ICKG), IEEE Computer Society, Los Alamitos, CA, USA, 2024, pp. 120–127. URL: https://doi.ieeecomputersociety.org/10.1109/ICKG63256.2024.00023. doi:10.1109/ICKG63256.2024.00023.

[7] S. Poria, D. Hazarika, N. Majumder, G. Naik, E. Cambria, R. Mihalcea, MELD: A multimodal multi-party dataset for emotion recognition in conversations, in: A. Korhonen, D. Traum, L. Màrquez (Eds.), Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 527–536. URL: https://aclanthology.org/P19-1050/. doi:10.18653/v1/P19-1050.

[8] S. M. Zahiri, J. D. Choi, Emotion detection on TV show transcripts with sequence-based convolutional neural networks, CoRR abs/1708.04299 (2017). URL: http://arxiv.org/abs/1708.04299. arXiv:1708.04299.

[9] Qwen Team, Qwen3 technical report, 2025. URL: https://arxiv.org/abs/2505.09388. arXiv:2505.09388.

[10] T. Dettmers, A. Pagnoni, A. Holtzman, L. Zettlemoyer, QLORA: efficient finetuning of quantized LLMs, in: Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23, Curran Associates Inc., Red Hook, NY, USA, 2023.

[11] Preferred Networks, Inc, Plamo-embedding-1b, 2025. URL: https://huggingface.co/pfnet/plamo-embedding-1b.

[12] Q. Tu, S. Fan, Z. Tian, T. Shen, S. Shang, X. Gao, R. Yan, CharacterEval: A Chinese benchmark for role-playing conversational agent evaluation, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 11836–11850. URL: https://aclanthology.org/2024.acl-long.638/. doi:10.18653/v1/2024.acl-long.638.

```
Please play the role of ${Character} from
${WorkTitle}.
Respond as ${Character}, using the tone,
manners, and vocabulary that ${Character}
would typically use.
You must possess all knowledge that
${Character} is known to have.

### Description of ${Character} from Wikipedia
${WikipediaDescription}

The following is the conversation so far. Please
generate ${Character}'s next line.

### Previous Conversation
${Situation}

### Your Next Line
```

```
Please play the role of ${Character} from
${WorkTitle}.
Respond as ${Character}, using the tone,
manners, and vocabulary that ${Character}
would typically use.
You must possess all knowledge that
${Character} is known to have.

### Description of ${Character} from Wikipedia
${WikipediaDescription}

The following is the conversation so far.

### Previous Conversation
${Situation}

Your current facial expression is
${FacialExpression}.
Please generate ${Character}'s next line in a
way that matches this facial expression.

### Your Next Line
```

**Figure 3:** Prompt templates used during training. The left example conditions on the situation only, while the right example conditions on both the situation and the facial expression. The symbols marked with "$" indicate placeholders for specific information and data of the target character. The text content in the templates is translated from the original Japanese version.

## A. Prompt Template

Previous studies have commonly employed prompt-based strategies to construct role-playing agents, for example by tailoring the instructions given to the LLM or by augmenting the input with character-specific information retrieved from external sources [3, 4]. These approaches aim to leverage the LLM's internal knowledge and improve role-playing behavior through prompt design and retrieval.

While improvements in prompting or dynamic knowledge augmentation are not unrelated to emotional information, optimizing them together with emotion-conditioned training would make the problem setting overly complex. Therefore, we designed a static prompt template to elicit the LLM's capabilities consistently during both training and evaluation.

As static character information, we included concise descriptions sourced from Wikipedia. Inspired by Shao et al. [2], who automatically construct character profiles from Wikipedia for fine-tuning, we did not auto-generate training targets; instead, we used the Wikipedia descriptions only as fixed context within our static prompt template.

## B. Details of Visual Novel Gameplay and Data Structure

For readers unfamiliar with visual novel games, Figure 4 illustrates an example of scene transitions. As described in Section 1, a visual novel is a type of game in which users progress through novel-style text by clicking or pressing keys, while listening to the spoken dialogue and viewing character images. Figure 4 corresponds to the example shown in Table 1. Although the audio is not used in our experiments, many visual novels include voice acting synchronized with character utterances. Thus, the data can be regarded as containing discretized facial expressions linked with the corresponding utterances and voice audio.

## C. Facial Expression Labeling Procedure

Specifically, three annotators (including one of the authors) who had experienced the target visual novel within the past year collaboratively created the labels. The labeling procedure was as follows. First, each annotator independently described each facial expression image in one word or short phrase. Next, all annotators reviewed one another's descriptions and voted for the expression they found most

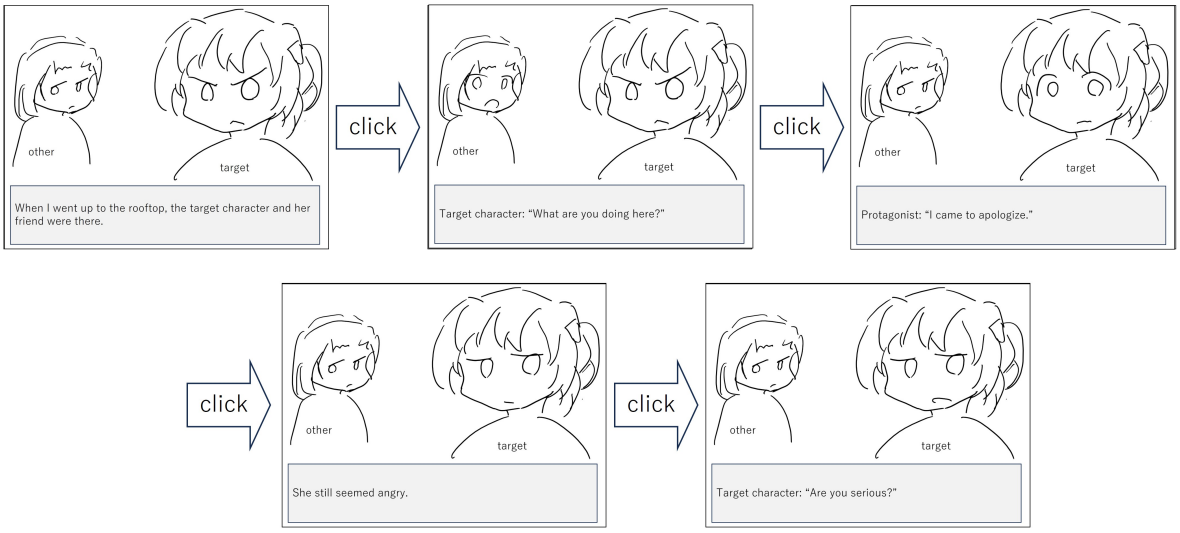


**Figure 4:** Example of scene transitions in a visual novel game. The player mainly reads novel-style text by clicking to advance through the story.

**Table 8**
Types and counts of facial expression labels (translated from Japanese)

| Facial Expression Label | Original | LLM-augmented | Random |
|---|---|---|---|
| [Slightly happy, faint smile] | 353 | 482 | 266 |
| [Natural smile, cheerful face] | 57 | 165 | 238 |
| [Serious, expressionless, unsociable] | 672 | 1258 | 265 |
| [Displeased, frown] | 225 | 510 | 262 |
| [Uneasy, confused] | 52 | 426 | 242 |
| [Exasperated] | 61 | 142 | 231 |
| [Interested, attentive] | 397 | 22 | 260 |
| [Surprised] | 26 | 34 | 262 |
| [Angry] | 364 | 16 | 271 |
| [Sad] | 920 | 182 | 254 |
| [Unconvinced] | 307 | 119 | 254 |
| [Astonished] | 2 | 1 | 244 |
| [Crying hard] | 24 | 1 | 233 |
| [Holding back tears] | 91 | 193 | 269 |

appropriate. Finally, all expressions that received at least one vote were adopted as valid labels. This procedure excluded cases where annotators could not provide an adequate description. When a single facial expression was judged to convey multiple aspects, all recognized descriptors were retained in the label, which can therefore consist of multiple terms (e.g., [Uneasy, confused]).

## D. Facial Expression Labels and Distribution

This appendix provides the full list of the 14 facial expression labels, along with the number of samples for the original, LLM-augmented, and randomly assigned datasets (Table 8). The text labels are translated from Japanese.

```
You are a game creator.
You must have complete knowledge of ${target_character_name}.

### Description of ${target_character_name} from Wikipedia
${wiki_description}

Based on the following game scenario, you must decide the next facial expression that
${target_character_name} will make.You can choose from the following 14 facial
expression labels:

[Slightly happy, faint smile]
[Natural smile, cheerful face]
[Serious, expressionless, unsociable]
[Displeased, frown]
[Uneasy, confused]
[Exasperated]
[Interested, attentive]
[Surprised]
[Angry]
[Sad]
[Unconvinced]
[Astonished]
[Crying hard]
[Holding back tears]

Carefully read the game scenario and select the next facial expression for
${target_character_name}. Your answer must consist only of the chosen label, for
example: “[Serious, expressionless, unsociable]”.

Game Scenario:
${situation}
```

**Figure 5:** Prompt template used for data augmentation. The symbols marked with “$” indicate placeholders for specific information and data of the target character. The text content is translated from the original Japanese version.

**Table 9**
Accuracy, Precision, Recall, and F1-score of the Additional Facial Expression Labels

| Data Type | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|
| LLM-augmented | 16.64% | 14.81% | 13.81% | 10.05% |
| Random | 7.21% | 7.07% | 7.01% | 5.53% |

## E. Prompt for Data Augmentation

For the LLM-based data augmentation, we used an open-source model that could run on our available GPU resources (Qwen/Qwen3-32B-AWQ [9]) to predict the target character’s facial expression from the surrounding situation text. If the model failed to output one of the 14 predefined facial expression labels, generation was repeated until a valid label was produced.

We also evaluated how closely the LLM-augmented and randomly assigned facial expression labels matched the ground-truth labels (Table 9). Because each category is weighted equally, macro-F1 scores are reported, which tend to be lower in absolute value. Although improving expression-labeling accuracy is not the main objective of this work, the results show that LLM-based augmentation performs better than random labeling but still has room for improvement.

## F. Training Hyperparameters

This appendix lists the full hyperparameter configuration omitted from the main text. All fine-tuning was conducted using the 4-bit quantized version of `Qwen/Qwen3-32B` [9] under a causal language modeling objective via the QLoRA framework [10], on a workstation equipped with two NVIDIA GeForce RTX 3090 GPUs (24 GB VRAM each), running Ubuntu 22.04 with CUDA 12.6 and PyTorch 2.8.0.

**Table 10**
Full fine-tuning hyperparameter settings

| Item | Setting |
|---|---|
| Quantization | 4-bit (bitsandbytes, NF4) |
| Precision | `bfloat16` |
| Optimizer | AdamW8bit ($\beta = (0.9, 0.98)$, weight decay = 0.01) |
| Learning rate | $5 \times 10^{-5}$ |
| Warmup ratio | 0.05 |
| Scheduler | Cosine |
| Batch size | 1 |
| Gradient accumulation steps | 8 (effective batch size = 8) |
| Epochs | 5 |
| Loss objective | Causal language modeling loss on target utterance only |
| Validation criterion | Lowest validation loss |
| LoRA rank ($r$) | 256 |
| LoRA $\alpha$ | 256 |
| LoRA dropout | 0.1 |
| LoRA target modules | `q_proj`, `k_proj`, `v_proj`, `o_proj`, `up_proj`, `down_proj` |