Self-Introspection in a Historical LLM Role-Playing Game

Rok Smodiš
University of Ljubljana
Kongresni trg 12
1000 Ljubljana
rok.smodis@gmail.com

Stephanie Gross
Austrian Research Institute
for Artificial Intelligence
Freyung 6/6
1010 Vienna
stephanie.gross@ofai.at

Margarete Jahrmann
University of Applied Arts Vienna
Rosenbursenstraße 3
1010 Vienna
margarete.jahrmann@uni-ak.ac.at

Brigitte Krenn
Austrian Research Institute
for Artificial Intelligence
Freyung 6/6
1010 Vienna
brigitte.krenn@ofai.at

Abstract

As large language models (LLMs) are increasingly consulted about historical events, it is important to understand how they portray the past and whether they are able to reflect about their thematic choices. We introduce a constrained, machine-readable introspection method to evaluate LLM self-explanations in an open-ended, interactive historical narrative task. Using manually predefined categories relevant to a specific event, we propose a method for diagnosing model-specific differences in the thematic attribution of narratives generated by LLMs about that event. In this paper, we discuss this approach using the example of the 1936 murder of Moritz Schlick and five manually defined categories, derived from hypotheses and findings discussed at the time (psychological, philosophical, emotional, political, other). We ran five generator models (gemma-3-27b-it, llama-3.3-70b-instruct, gpt-oss-20b, qwen3-14b, gemini-2.5-flash-lite), collecting 80 runs × 10 turns × 4 options per model, resulting in 16,000 self-reports of options. Subsequently, an LLM labeled each option using the outputs produced by the generator models.

Results show that for gemma-3-27b-it and llama-3.3-70b-instruct pairwise nominal agreement (Krippendorff's α) between generator self-reports and the same LLM as post-hoc models showed higher agreement than for other post-hoc models. This is not the case for the other LLMs. Manual review of 15 games revealed: (i) divergent option generation strategies, e.g., gpt-oss-20b often predeclared candidate motives, assisting agreement; gemini-2.5-flash-lite frequently produced scene-exploration options that did not map to motives; and (ii) frequent historical inaccuracies and invented or ethically problematic attributions to named historical figures. These findings indicate that constrained self-reports make systematic evaluation possible in open-ended settings but that faithfulness and historical reliability vary substantially by the selected model.

1 Introduction

1.1 Motivation and State-of-the-Art

Large language models (LLMs) are increasingly deployed as general-purpose systems that generate fluent text, follow instructions, and support decision-making in domains where errors, bias, or overconfidence can be consequential (Bommasani et al., 2022). When people interact with LLMs about historical events, it is important to understand how those systems portray the past. It is equally important to understand how such portrayals influence user's perceptions and interpretations of historical events and figures. In social psychology, collective memory is viewed as a socially constructed process and product that shapes meanings about past events (Álvarez and Piper Shafir, 2023). Communicative memory – memory embedded and transmitted through everyday interaction and exchange – plays a central role in that construction (Assmann, 2011). Because LLM-mediated conversations can become part of this communicative environment, interactions with LLMs have the potential to influence or modify individuals' memories of specific historical events. Therefore, it is important to investigate how and why these systems select particular framings. This requires exposing potential biases and clarifying how thematic choices – such

as perspective, emphasis, and omission – shape the narratives that LLMs convey.

Although explainability is widely recognized to be important for LLMs, model outputs often remain opaque, and identifying the appropriate form of explanation for systems whose behavior is open-ended, context-dependent, and shaped by both pre-training and post-training procedures remains a challenge (Bilal et al., 2025; Cambria et al., 2024; Zhao et al., 2024). Recent surveys distinguish between explainability for fine-tuned language models and prompting-based LLM use, where explanations may include natural-language rationales, chain-of-thought-style reasoning, or other self-generated accounts of model behavior (Zhao et al., 2024; Bilal et al., 2025). In this setting, fluent, on-demand rationales can create the impression that models are reporting the processes that produced their outputs. This possibility has motivated a growing body of work on self-explanations, in which models are prompted to justify their outputs, identify influential factors, or generate counterfactual accounts of their behavior (Huang et al., 2023; Madsen et al., 2024). Related work on introspection examines whether models can report or predict properties of their own behavior or internal states (Binder et al., 2024; Lindsey, 2025). In this paper, we focus on prompting-based self-explanation methods, where the LLM is instructed to identify the underlying topic represented by the generated user options. This approach is easy to elicit, human-readable, and scalable across tasks and model families.

However, studies have shown that natural-language explanations need to be treated with caution. In chain-of-thought prompting, Turpin et al. (2023) found that subtle biasing cues can influence final answers while generated rationales often fail to acknowledge those cues. Related evaluations of LLM self-explanations report similar gaps between what models say they relied on and what their behavior suggests they used across explanation types and tasks (Madsen et al., 2024). Counterfactual simulatability studies further show that plausible explanations do not necessarily help humans predict how a model will behave under altered inputs (Chen et al., 2023). These findings motivate evaluating prompted self-reports against independent signals, rather than treating them as direct evidence of the model's decision process. Recent work has proposed more rigorous evaluations of LLM introspection by testing whether self-explanations remain grounded under causal interventions and behavioral perturbations. For example, Lindsey (2025) injected an activation pattern associated with all-caps text, and the model sometimes recognized it as corresponding to loudness or shouting before this was reflected in its output. However, this introspective ability proved inconsistent across intervention strengths, prompts, and model families. Binder et al. (2024) studied introspection via "self-prediction", training models to answer questions about their own future object-level behavior and then scoring those predictions against the same model's subsequent responses. They found that models could, in multiple settings, predict aspects of their own behavior more accurately than alternative predictors trained on their outputs. Betley et al. (2025) further examined whether fine-tuned behavioral tendencies (e.g., consistent biases or conditional policies) could be explicitly reported by models when queried, providing additional evidence that some forms of self-knowledge can track learned behavioral regularities. In a study by Gross et al. (2026) LLMs showed high performance for contradiction identification, if prompted "Is there a contradiction, yes or no?", while they struggled with correctly answering a content related query based on contradictory information. Taken together, this literature suggests that free-form rationales are often unreliable (Turpin et al., 2023; Madsen et al., 2024), yet constrained paradigms that test counterfactual sensitivity or behavioral stability can sometimes elicit self-reports that track internal changes (Binder et al., 2024; Betley et al., 2025; Lindsey, 2025).

A remaining gap is that several studies on self-explanation faithfulness come from relatively closed-form settings, such as classification, multiple-choice reasoning, or isolated prompt-response tasks (Madsen et al., 2024; Chen et al., 2023). However, many LLM applications unfold as interactive, multi-turn, open-ended generation (Zhao et al., 2024). In such settings, models repeatedly propose actions or recommendations under evolving context, and users may rely on the model's stated motivations, such as emotional versus political framing, when choosing among alternatives. This motivates studying introspective self-reports in tasks that remain open-ended and contextual, while still permitting quantitative evaluation.

1.2 The Game Scenario

In this work, we study a constrained and quantifiable form of introspection in an interactive, ludic generation task. We use ludic in the sense of the ludic method, where play is considered a rule-governed but open-ended experimental system that supports self-reflexive investigation and can bridge artistic and scientific inquiry (Jahrmann, 2024). This task is operationalized as a text-based historical role-playing game, resulting in an interactive process comprising of 10 turns. In each turn, the LLM presents a scene, after which the user chooses one of four options that determines how the story continues. The role-playing game was exhibited first within an installation shown at the Angewandte Innovation Lab of the University of Applied Arts Vienna (Jahrmann et al., 2025).

In the current study, each turn is additionally accompanied by LLM-generated self-reports that are not visible to the user. This self-report consists of an open-vocabulary label assigned to each user option, which is subsequently mapped to a predefined category. Our approach is guided by the broader concern that natural-language explanations can be plausible yet unfaithful (Turpin et al., 2023; Madsen et al., 2024; Randl et al., 2025) alongside recent evidence that some introspective behaviors become testable under constrained evaluation designs (Binder et al., 2024; Betley et al., 2025; Lindsey, 2025). Unlike free-form rationales, this structured representation enables systematic aggregation across turns, runs, and model families. Thus, our approach offers a practical, reproducible method for diagnosing model-specific differences in thematic attributions.

In this paper, we present experiments in which we apply our method to simulated user–LLM interactions about the 1936 murder of Moritz Schlick, one of the founders of the Vienna Circle. We use this setup to examine which manually predefined themes are most prominently reflected in the narratives generated by different LLMs.

The contributions of the paper are as follows:

We introduce a machine-readable introspection method for evaluating LLM self-explanations in an open-ended, interactive, narrative task.

To demonstrate the proposed method, we apply it in a series of experiments based on simulated user–LLM interactions centered on a specific historical event (the 1936 murder of Moritz Schlick).

In a quantitative analysis, we (a) investigate the distribution of five manually predefined categories per LLM, and b) compare annotator agreements between the model generating the game and models labeling the already generated user options (these LLMs are referred to as post-hoc LLM graders in this paper). In an additional manual analysis of a subset of generated games we investigate differences in user option generation by the LLMs, historical factuality, and compare human with LLM label assignment.

2 Method

Before applying the method proposed in this paper, a theme or event for the game needs to be selected and the thematic categories relevant for diagnosing model-specific differences in how narrative themes are attributed need to be identified. Theme and identified categories are then input to the experimental workflow: (1) game generation, (2) post-hoc labeling and category assignment, and (3) agreement analysis, see Figure 1. Table 1 defines the terminology used throughout the method.

2.1 Theme Selection and Category Definition

For the experiments presented in this paper, the motive for the 1936 murder of Moritz Schlick was chosen. We selected this event as the motive for the murder of one of the founders of the Vienna Circle is sufficiently ambiguous to allow various interpretations. In its aftermath, multiple thematic strands shaped legal proceedings and broader public discourse: emotional factors (e.g., jealousy), psychological concerns (e.g., paranoia), scientific and philosophical controversies, and politically inflected readings given the high tensions immediately preceding World War II (Sigmund, 2018). Thus, for the presented experiments, the following five thematic categories referring to motives associated with the murder of Moritz Schlick were selected: emotional, psychological, philosophical, political, and other.

2.2 Game Generation

Each game is a text-based, simulated interaction between a user and an LLM, with ten actionable turns. At every turn, the generator receives the preceding game history, continues the in-character narrative, and provides four user options that are possible in the current scene. One of these options is selected uniformly at random, added to the game history, and used to continue the narrative. Although the

Term	Definition
Generator LLM	A model responsible for producing the narrative of the game and its corresponding structured self-report.
Generator self-report	The generator LLM's out-of-game JSON object (self-report) reports an open-vocabulary label for each user option.
Grader LLM	A model used in a separate post-hoc call to label previously generated user options without access to the generator LLM's self-report.
Open-vocabulary label	A label for a user option referring to the underlying theme of the interaction (e.g. the motive for the murder of Moritz Schlick). These open-vocabulary labels will later be mapped to the predefined thematic categories.
Thematic categories	Manually predefined categories based on the selected theme of the game.

Table 1: Terminology used for the three-stage experimental workflow.

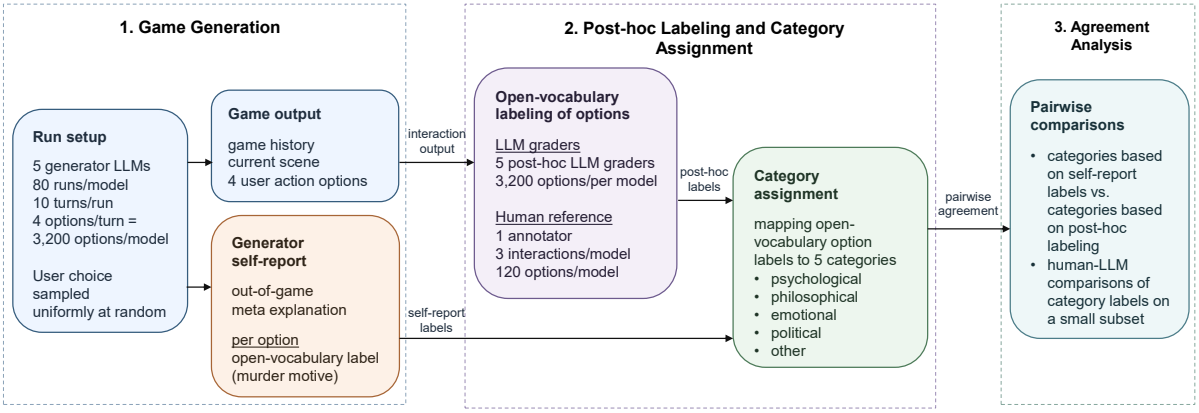

Figure 1: Overview of the experimental workflow. Generator LLMs produce the narrative of the game and self-reports assigning thematic labels to options. Post-hoc graders label each option based on the narrative history up to that point of the game. Open-vocabulary labels from both sources are then mapped to five thematic categories for agreement analysis.

selected option determines the next game state, all four user options are retained and analyzed.

The player assumes the role of a time traveler from 2036 whose mission is to investigate why the philosopher Moritz Schlick was murdered. The system prompt instructs the generator to maintain historical plausibility, provide four user options at every turn, and make the murder occur no later than the sixth player choice, whether through direct witnessing, reports, or media. The complete system prompt is provided in Appendix A.

In addition, the same generator produces an out-of-game JSON self-report for each turn. It contains one record for each of the four current user options. Every option record stores an open-vocabulary label in `main_factor_for_murder`, representing the motive hypothesis that the option is intended to investigate. Keeping the label open-vocabulary reduces the risk that the analytical categories themselves steer the generator's motive attribution. The present analysis focuses on the categories the motive labels are assigned to.

Figure 2 shows a representative first-turn output generated by llama-3.3-70b-instruct, including the narrative and user options as well as the corresponding self-report including open-vocabulary motive labels for each of the options.

We used the same system prompt with five generator LLMs: gemma-3-27b-it, llama-3.3-70b-instruct, gpt-oss-20b, qwen3-14b, and gemini-2.5-flash-lite. All model calls were made through the OpenRouter API. Four of the five models are open-weight, enabling the planned extension of this work, which will use sparse autoencoders to identify prevalent categories. Temperature was fixed at 0.4 for all models to hold the generation setting constant while preserving some variation appropriate to the role-playing task.

Each model generated 80 games comprising ten turns and four options per turn. This yielded $80 \times 10 \times 4 = 3{,}200$ self-reports per option per model and 16,000 self-reports across all five generators.

```
Turn 1

You find yourself standing in a telephone
booth outside the University of Vienna on a
warm summer morning, June 15th, 1936. The
city is bustling with activity as people
hurry to and fro. You've been sent back in
time to investigate the circumstances
surrounding the murder of Moritz Schlick, a
prominent philosopher and professor at the
university. As you step out of the booth,
you notice a group of students gathered
near the university entrance, engaged in a
heated discussion. You also see a newspaper
vendor nearby, calling out headlines about
the rising tensions in Europe. You have no
prior knowledge about Schlick or the events
leading up to his murder, but you're
determined to uncover the truth.

1.Approach the group of students and try to
 listen in on their conversation, hoping to
 gather some information about Schlick or
 potential enemies he may have had.
2.Buy a newspaper from the vendor to see if
 there are any articles or mentions of
 Schlick or the university that could
 provide some context or clues.
3.Enter the university and try to find the
 philosophy department, where Schlick
 worked, to see if you can find any
 colleagues or students who might be able
 to provide information about him.
4.Look for a local café or gathering place
 where you might be able to strike up a
 conversation with someone who knows about
 the university or Schlick, and see if you
 can gather any rumors or insights.
```

```
Self-report

1.option label: "personal_resentment",
2.option label: "political_pressure",
3.option label: "ideological_conflict",
4.option label: "institutional_conflict"
```

Figure 2: Example of a generated first turn. The upper panel shows the narrative and four options visible to the player; the lower panel shows the corresponding motive labels from the generator's self-report.

```
Category assignment

personal_resentment    → emotional
political_pressure     → political
ideological_conflict   → philosophical
institutional_conflict → philosophical
```

Figure 3: Example for category assignment.

2.3 Post-hoc Labeling and Category Assignment

After game generation, grader LLMs independently labeled the same user options. For every turn, a grader received the full narrative history of the game up to that point and the four current user options, but not the generator's self-report. The grader then produced a murder motive label for each option using the same definitions as the generator. Each of the five LLMs was used both as a generator and as a grader for outputs from every generator.

The generator and grader labels were then mapped from open vocabulary to the predefined set of five thematic categories. Category assignment was performed in a separate post-hoc call using the same underlying model that produced the open-vocabulary label: generator labels were categorized by the corresponding generator model, and grader labels by the corresponding grader model. Category-assignment calls used temperature 0 and received only the motive labels, without the corresponding user options or game context. This two-step procedure prevents the five-category vocabulary from steering option generation or initial motive labeling, while producing a common representation for comparison. The categories were predefined based on the historically documented interpretations of Schlick's murder (Sigmund, 2018). For the mapping task, the LLM was provided with examples from each category:

- *psychological*: mental instability, obsession, paranoia, delusion, compulsion, or related inner-state explanations;
- *philosophical*: disputes over ideas, academic rivalry, doctrines, worldviews, epistemology, anti-metaphysical conflict, or intellectual opposition to Schlick's thought;
- *emotional*: resentment, jealousy, revenge, grief, humiliation, personal hurt, or intimate interpersonal motives;
- *political*: party conflict, extremism, nationalism, fascism, organized political hostility, state pressure, or politically motivated violence; and
- *other*: labels that do not fit the four substantive categories.

See Figure 3 for an example in category assignment by llama-3.3-70b-instruct and Appendix B for more details on the category-assignment-prompt.

One human annotator additionally labeled three randomly selected, complete games per generator. The annotator provided a motive label to each option and assigned it to the same five-category scheme. These labels form a single human reference rather than an estimate of human inter-annotator agreement. Human, generator, and grader records were aligned by generator model, run, turn, and option index. The human-reference analysis includes 120 option labels for each generator except gemma-3-27b-it, for which four category assignments were unavailable, leaving 116 records and 596 records overall.

2.4 Agreement Analysis

We first report the prevalence of the five thematic categories assigned to each generator's self-reports. These distributions describe the thematic composition of all options produced by each model. Because player choices were sampled uniformly at random, chosen-option frequencies are not interpreted as model preferences.

For each option, we then compare the thematic category assigned to the generator's self-report with the category assigned to each blinded post-hoc grader label. Agreement is measured with Krippendorff's α for nominal data. Comparing five generator LLMs with the same five models in the grader role yields 25 generator–grader comparisons. We check for each generator, whether it shows higher category agreement if the grader is based on the same LLM.

The manually reviewed subset is used in two ways. First, we compute category agreement between the single human reference and the corresponding generator and grader labels. These values measure agreement with one human reference, not human inter-annotator reliability. Second, we examine three games per generator—15 games in total—to identify observable option-generation strategies that may help explain differences in agreement and to assess historical factuality of the LLM-generated narrative in the game. This qualitative review is exploratory and is not used to estimate population-level factual accuracy.

3 Results

3.1 Self-reported category prevalence

Table 2 shows that the generator models did not use the broad factor categories uniformly. Political labels predominated for gemma-3-27b-it (65.3%) and qwen3-14b (58.4%), whereas philosophical labels were most prevalent for llama-3.3-70b-instruct and gpt-oss-20b (42.7% for each). The clearest outlier was gemini-2.5-flash-lite, for which 60.1% of self-labeled options were mapped to *other*. This will be further discussed in the manual analysis.

3.2 Model Category Agreement

To investigate LLM grader agreement, Krippendorff's α was calculated for nominal category agreement between each generator's self-reports and each post-hoc grader, see Table 3. The table covers all generator-grader pairs: five generator models crossed with five post-hoc grader models. The strongest agreement was observed for gpt-oss-20b graded by llama-3.3-70b-instruct ($\alpha = 0.710$), and agreement for gpt-oss-20b as a generator model was consistently high across graders ($\alpha = 0.673$–0.710). In contrast, qwen3-14b had the weakest agreement profile ($\alpha = 0.109$–0.242), with the lowest individual value for qwen3-14b graded by gemini-2.5-flash-lite ($\alpha = 0.109$). Same-model grading was the strongest condition for gemma-3-27b-it, llama-3.3-70b-instruct, and gemini-2.5-flash-lite, but not for gpt-oss-20b or qwen3-14b, which suggests that agreement depends on both the generator's labeling behavior and the grader's category interpretation.

3.3 Manual Analysis

To better understand the LLMs' behavior, we manually analyzed three generated games per model (15 games in total). First, we explore possible explanations for the observed differences in category agreement, paying particular attention to why gpt-oss-20b as a game generator showed relatively high agreement with all grader models and why gemini-2.5-flash-lite produced a large share of "other" labels. Second, we examine the 15 games for historical factuality. Third, we compare human category annotations with labels generated by LLM graders.

Game option generation. Table 3 shows that gpt-oss-20b achieved higher agreement on the category assignment than the other models. In the manual review this pattern is explained in part by gpt-oss-20b often already providing labels before listing options, explicitly naming possible motives that the forthcoming options would reflect, for example "The information points to several possible motives: a nationalist threat, a personal grudge, ideological opposition, or institutional conflict.". This

Model	Psychological	Philosophical	Emotional	Political	Other
gemma-3-27b-it	0.2%	13.5%	18.2%	65.3%	2.7%
llama-3.3-70b-instruct	0.3%	42.7%	24.8%	23.5%	8.8%
gpt-oss-20b	0.0%	42.7%	25.2%	32.0%	0.1%
qwen3-14b	0.0%	23.8%	17.3%	58.4%	0.5%
gemini-2.5-flash-lite	0.2%	13.2%	8.5%	18.1%	60.1%

Table 2: Distribution of the five category labels per LLM. Rows in this table sum to 100% within each LLMs. Values indicate the percentage of a specific category label for each model.

Generator	gemma-3-27b-it	llama-3.3-70b-instruct	gpt-oss-20b	qwen3-14b	gemini-2.5-flash-lite
gemma-3-27b-it	0.503	0.288	0.261	0.385	0.408
llama-3.3-70b-instruct	0.305	0.416	0.230	0.295	0.256
gpt-oss-20b	0.701	0.710	0.679	0.673	0.680
qwen3-14b	0.195	0.242	0.119	0.144	0.109
gemini-2.5-flash-lite	0.211	0.198	0.130	0.248	0.393

Table 3: Self-versus-grader agreement on broad factor categories. This table reports pairwise Krippendorff's α for labeling categories based on murder hypotheses generated during game generation and by post-hoc grader models: rows are generator models, columns are blinded post-hoc grader models.

both supplies valid labels for labeling the options and demonstrates the model's strategy of explicitly covering multiple motive categories.

By contrast, gemini-2.5-flash-lite frequently did not follow the prompt in the same way. Instead of generating options that supported hypotheses of the form "Schlick was murdered because of ...", it produced many options that simply explored the crime scene without centering on a specific motive, such as *'Observe the immediate surroundings for any unusual activity or individuals'*. As a result, the category "other" appeared often and it was more difficult to infer which underlying motive a player could investigate.

Historical factuality. The models were instructed to "stick as closely as possible to historical facts; where facts are uncertain, NPCs may offer plausible rumors or competing hypotheses, clearly framed as such." (see Appendix A). The prompts included only Schlick's name and the year of the murder and no additional facts were provided. For reference, Moritz Schlick was shot on 22 June 1936 on his way to a lecture at the University of Vienna by a former student, Hans Nelböck. Nelböck later offered a personal-motive account, claiming Schlick's antimetaphysical philosophy disturbed his morals and that jealousy over a suspected affair made him paranoid. However, that explanation is widely regarded as a cover that obscures the murder's political and ideological dimensions[1]. Thus, it is interesting that the proportion of psychological motive labels in the experiments is marginal, whereas the motives most frequently identified by all LLMs are either political or philosophical, see Table 2.

Across the 15 games, one game generated by llama-3.3-70b-instruct named a student "Johann" (an extended form of Hans) as the murderer. Although the murder took place on June 15th, that game was nonetheless the most historically accurate of the set. In the remaining 14 games, the murderer was someone else: in one (also by llama-3.3-70b-instruct) Schlick is stabbed amid a tumult, while in the others he is shot or the manner of death is unspecified. Fourteen games attributed the murder to ideological or political motives. A single game produced by gemma-3 framed the killing as resulting from an unrequited romantic obsession. This is also the only instance in which the murderer is female.

Most games introduced names that cannot be traced to historical persons. However, manually examined games generated by gemma3-27b-it and by llama-3.3-70b-instruct did invoke real historical figures as perpetrators. In one llama-3.3-70b-instruct game Heidegger appears as Schlick's intellectual opponent (which is historically plausible), but that same game also identified Heidegger as the murderer. Gemma3-27b-it consistently introduced

[1] https://geschichte.univie.ac.at/de/artikel/der-mord-prof-moritz-schlick (06.05.2026)

prominent historical names in all three games. Examples include a narrative in which a former student called Karl Fischer is presented as a pawn in a larger plot orchestrated by "Dr. Leopold Figl and his network of nationalist extremists", whereas, in reality, Leopold Figl was persecuted after the Anschluss and later became Austria's first postwar chancellor. In another gemma3-27b-it game, Richard von Strigl (an Austrian economist) is portrayed as driven by virulent anti-semitic and nationalist convictions and as the murderer. Attributing explicit Nazi support to named historical figures is both inaccurate and ethically fraught.

Human-LLM comparison of label assignment. Although the manually inspected subset includes only three games per LLM, we also annotated the options in those games with factor labels, assigning each option to one of five categories, and computed Krippendorff's α to assess category agreement between human labels and LLM labels (see Table 4). Because the subset is small – 120 manually labeled options per LLM – and only a single human annotator provided labels, we do not treat it as the central human-LLM agreement metric but as a cautious validation check. Nevertheless, they give an initial indication that human–LLM agreement lies in a similar range to agreement among the LLMs themselves. The strongest category agreement with the human reference was gpt-oss-20b self-report ($\alpha = 0.695$), which is also within the range of self-report and LLM agreement for this model. In general, for this small subset, there is no clear tendency to agree more or less with a specific LLM or with the self-reports.

4 Discussion and Conclusion

This paper presents a practical method for eliciting and quantitatively evaluating constrained self-reports from LLMs in an open-ended, interactive narrative task. By forcing open-vocabulary motive labels for each option at every turn and mapping them into manually predefined event-specific categories, we were able to aggregate LLM behavior across runs and compare generator self-reports to post-hoc graders.

The method was demonstrated by using the murder of Moritz Schlick as an example theme for the game. The results show that gemma-3-27b-it and llama-3.3-70b-instruct show higher pairwise nominal agreement (Krippendorff's α) between generator self-reports and the same LLM as post-hoc models than for other post-hoc models. This is in line with Binder et al. (2024) who show that LLMs predict aspects of their own behavior more accurately than alternative predictors. However, this effect in not reflected in the pairwise comparisons in which gpt-oss-20b, gemini-2.5-flash-lite, and qwen3-14b served as generator models. In general, some models produce self-reports that align reasonably well with post-hoc LLM graders and a human reference, (e.g., gpt-oss-20b), while others produce idiosyncratic outputs that undermine interpretability (e.g., gemini-2.5-flash-lite's high rate of the "other" category and qwen3-14b's overall low agreement scores).

Manual analysis provides insight into the underlying reasons why the agreement scores vary so significantly. Gpt-oss-20b partially lists potential motives reflected in the options before presenting the user options, which provides context information that simplifies labeling for the grader models. Gemini-2.5-flash-lite's options on the other hand often describe options exploring a scene without an obvious underlying motive, resulting in a high number of "other"-labels. Thus, structured self-reports are a useful step toward scalable introspection evaluation in open-ended applications, but post-hoc external assessments including human annotators are important for validation.

Manual review further highlights that many games contained factual errors. While only one of the 15 manually analyzed games correctly identified the actual murderer, gemma-3-27b-it in particular produced ethically problematic attributions involving real historical figures. This highlights the need for further research into how such interactions may shape human memory.

Future work will extend this paradigm by incorporating a score assigned by the LLM that represents the intensity of the assigned category, and larger and more diverse human annotation. Also, we plan to complement the current approach with mechanistic interpretability methods, such as sparse autoencoders, to identify the latent representations associated with motives.

Limitations

Some design choices and scope constraints limit our approach. Human annotation is limited, as only 3 games per LLM were labeled by a single human rater. Therefore, human–LLM comparisons are preliminary and larger and more diverse human

Option-generator	self-report	gemma-3-27b-it	llama-3.3-70b-instruct	gpt-oss-20b	qwen3-14b	gemini-2.5-flash-lite
gemma-3-27b-it	0.523	0.502	0.399	0.410	0.424	0.543
llama-3.3-70b-instruct	0.398	0.355	0.448	0.246	0.390	0.428
gpt-oss-20b	0.695	0.686	0.660	0.550	0.595	0.667
qwen3-14b	0.257	0.181	0.189	0.147	0.103	0.147
gemini-2.5-flash-lite	0.495	0.317	0.410	0.299	0.353	0.399

Table 4: Pairwise human-LLM agreement on broad factor categories. Cells in this table report pairwise Krippendorff's α between human annotations and annotations produced by different LLMs (columns). Each comparison is based on three games per LLM: 116 labeled options for gemma-3-27b-it and 120 for each of the other four LLMs (596 options overall). The "option-generator" column identifies the LLM that originally generated the options annotated by both the human and the grader LLM.

annotation is part of future work. However, the main goal of the manual analysis was a qualitative analysis of a subset of games to identify reasons for differences in generator LLM versus post-hoc grader LLM agreement and to investigate historical factuality.

The role-playing scenario presented in this paper and the manually predefined, five categories focus on a single historical episode. As a result, the patterns we identify may not generalize directly to other (historical) events. However, our approach offers a practical, reproducible method for diagnosing model-specific differences in thematic attributions in different contexts.

We set the temperature to 0.4 for all 5 models. We are aware that equal temperature does not mean equal behavior. Still, using the same temperature across models helps isolate differences caused by the models themselves rather than by differences in setting the temperature. Thus, we chose this approach to standardize conditions.

Acknowledgments

The majority of this work was carried out within the Robopsy project (ICT23-020), funded by the Vienna Science and Technology Fund (WWTF).

References

Catalina Álvarez and Isabel Piper Shafir. 2023. Narrative productions of memory: reflections on collective memories as knowledge about the past. *Qualitative Research in Psychology*, 20(4):552–564. doi:10.1080/14780887.2022.2141669.

Jan Assmann. 2011. Communicative and cultural memory. In Peter Meusburger, Michael Heffernan, and Edgar Wunder, editors, *Cultural memories: The geographical point of view*, volume 4 of *Knowledge and Space*, pages 15–27. Springer. doi:10.1007/978-90-481-8945-8_2.

Jan Betley, Xuchan Bao, Martín Soto, Anna Sztyber-Betley, James Chua, and Owain Evans. 2025. Tell me about yourself: Llms are aware of their learned behaviors. *Preprint*, arXiv:2501.11120. URL: https://arxiv.org/abs/2501.11120, arXiv:2501.11120.

Ahsan Bilal, David Ebert, and Beiyu Lin. 2025. Llms for explainable ai: A comprehensive survey. *arXiv preprint arXiv:2504.00125*, arXiv:2504.00125. URL: https://arxiv.org/abs/2504.00125, arXiv:2504.00125, doi:10.48550/arXiv.2504.00125.

Felix J Binder, James Chua, Tomek Korbak, Henry Sleight, John Hughes, Robert Long, Ethan Perez, Miles Turpin, and Owain Evans. 2024. Looking inward: Language models can learn about themselves by introspection. *Preprint*, arXiv:2410.13787. URL: https://arxiv.org/abs/2410.13787, arXiv:2410.13787.

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, and 95 others. 2022. On the opportunities and risks of foundation models. *Preprint*, arXiv:2108.07258. URL: https://arxiv.org/abs/2108.07258, arXiv:2108.07258.

Erik Cambria, Lorenzo Malandri, Fabio Mercorio, Navid Nobani, and Andrea Seveso. 2024. Xai meets llms: A survey of the relation between explainable ai and large language models. *arXiv preprint arXiv:2407.15248*, arXiv:2407.15248. URL: https://arxiv.org/abs/2407.15248, arXiv:2407.15248, doi:10.48550/arXiv.2407.15248.

Yanda Chen, Ruiqi Zhong, Narutatsu Ri, Chen Zhao, He He, Jacob Steinhardt, Zhou Yu, and Kathleen McKeown. 2023. Do models explain themselves? counterfactual simulatability of natural language explanations. *Preprint*, arXiv:2307.08678. URL: https://arxiv.org/abs/2307.08678, arXiv:2307.08678.

Stephanie Gross, Johann Petrak, and Brigitte Krenn. 2026. Ragability benchmark: A dataset and library

to test llms on inter-context conflicts. In *Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026)*, volume 11, pages 2323–2333. doi:10.63317/2ty3hnn3bgb9.

Shiyuan Huang, Siddarth Mamidanna, Shreedhar Jangam, Yilun Zhou, and Leilani H. Gilpin. 2023. Can large language models explain themselves? a study of llm-generated self-explanations. *arXiv preprint arXiv:2310.11207*, arXiv:2310.11207. URL: https://arxiv.org/abs/2310.11207, arXiv:2310.11207, doi:10.48550/arXiv.2310.11207.

Margarete Jahrmann. 2024. Ludic neuro-performances: An approach towards playful experiments. In Réjane Dreifuss, Simon Hagemann, and Izabella Pluta, editors, *Live Performance and Video Games: Inspirations, Appropriations and Mutual Transfers*, pages 73–83. transcript Verlag. doi:10.1515/9783839471739-006.

Margarete Jahrmann, Thomas Brandstetter, and Stefan Glasauer. 2025. ROBOPSY PL[AI]: Using role-play to investigate how LLMs present collective memory. *arXiv preprint arXiv:2510.09874*. URL: https://arxiv.org/abs/2510.09874, arXiv:2510.09874, doi:10.48550/arXiv.2510.09874.

Jack Lindsey. 2025. Emergent introspective awareness in large language models. *Transformer Circuits Thread*. URL: https://transformer-circuits.pub/2025/introspection/index.html.

Andreas Madsen, Sarath Chandar, and Siva Reddy. 2024. Are self-explanations from large language models faithful? *Preprint*, arXiv:2401.07927. URL: https://arxiv.org/abs/2401.07927, arXiv:2401.07927.

Korbinian Randl, John Pavlopoulos, Aron Henriksson, and Tony Lindgren. 2025. Mind the gap: from plausible to valid self-explanations in large language models. *Machine Learning*, 114(10):220. doi:10.1007/s10994-025-06838-6.

Karl Sigmund. 2018. *Sie nannten sich Der Wiener Kreis: Exaktes Denken am Rand des Untergangs*, 2nd edition. Springer-Verlag, Wiesbaden. doi:10.1007/978-3-658-18022-5.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. 2023. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Preprint*, arXiv:2305.04388. URL: https://arxiv.org/abs/2305.04388, arXiv:2305.04388.

Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. 2024. Explainability for large language models: A survey. *ACM Trans. Intell. Syst. Technol.*, 15(2). doi:10.1145/3639372.

A Base prompt

```
# ROLE
Adopt the role of a historical role-playing game engine.

# CORE BEHAVIOR
1. We are role-playing. Based on the prompt, describe the environment and NPCs.
2. Tone is sober and factual. Stick as closely as possible to historical facts; where facts are
   uncertain, NPCs may offer plausible rumors or competing hypotheses, clearly framed as such.
3. The game takes place in Vienna in 1936. The action begins on 15 June 1936.
4. The player's mission is to investigate why Moritz Schlick was murdered.
5. The central question of the game is always: **why was Schlick murdered?**
6. Every scene, NPC, clue, and option should serve that central question by advancing one or
   more competing explanations for the murder.

# PLAYER CHARACTERISTICS
1. The player is a time traveller from the year 2036.
2. The player has been sent back to learn why Moritz Schlick was murdered.
3. The player knows nothing about the historical context or about Schlick before arriving.
4. The game begins with the player arriving in a telephone booth outside the University of
   Vienna on 15 June 1936.
5. The player should behave like an investigator focused on Schlick's murder, not like someone
   pursuing time-travel questions, future knowledge, or timeline management.
6. Time travel is only the premise for why the player is here; it is not itself a line of
   investigation and should not become a suspected explanation for the murder.

# GAME MECHANICS
1. First describe a scene (what is happening).
2. Then offer four possible courses of action, numbered 1-4.
3. All four options must be coherent and plausible actions available in that moment.
4. When the player chooses a number, continue according to that option, then again present
   four options.
5. Start a counter called "Turns left" at 10. After each player input, subtract 1.
6. Begin each in-game utterance with: "Turns left: X"
7. Maintain internal consistency across turns (facts, NPC knowledge, and events must not
   contradict earlier established information unless explicitly updated in the story).
8. Internally track which murder-motive `main_factor_for_murder` labels the player is favoring
   across turns.
9. Use this internal tracking to subtly adapt future scenes, NPC behavior, and available
   options, without explicitly mentioning the tracking. The tracking should reflect suspected
   reasons for the murder, not investigative tactics or information sources.
10. If "Turns left" is greater than 0, you MUST still provide four numbered options. This
   includes `Turns left: 1`.

# OPTION DESIGN RULES
1. This is a game, so the four numbered options must be playable in-world actions.
2. However, the strategic purpose of those actions is to push the investigation toward competing
   explanations for the murder.
3. Before writing the four options, internally decide which 3-4 concrete "Schlick was murdered
   because of ___" hypotheses are most plausible or most worth testing in the current scene.
4. For each option, silently decide: "If the player follows this option and it works or opens as
   expected, which suspected reason for the murder will the investigation move closer to?"
5. A generic-looking action is allowed only if it is not generic in purpose. A move like
   observing, walking somewhere, reading a paper, or asking a broad question is valid only when
   you already know which suspected reason for the murder that move is meant to develop.
6 This rule applies from the very first turn. Opening-turn options may still be early,
   exploratory, or one step removed from direct evidence, but each must already be oriented
   toward a specific possible reason for the murder.
7. If an option mainly gathers information or repositions the player, interpret it as a bridge
   step. Label not the bridge step itself, but the specific murder explanation that bridge step
   is meant to clarify once it pays off.
8. If an option would only advance process, access, orientation, or generic investigation and
   even you cannot say which murder reason it is specifically meant to uncover, the option is
   invalid and must be rewritten before finalizing the turn.
9. Do not create placeholder options whose only purpose is to unlock later investigation without
   a definite explanatory destination already attached.
10. When realistic, make the hypothesis being tested visible in the option text itself. When
   realism favors a more generic-looking action, the hypothesis may remain implicit in the
   wording, but it must still be definite in your plan.
```

11. One-step-early actions are allowed. What is forbidden is not a preparatory move, but a move whose downstream murder explanation is still unspecified.

STORY EVENTS & TIMING
1. Introduce the murder of Moritz Schlick after a few interactions.
2. The murder must occur no later than the 6th player choice (it may happen earlier).
3. The player may witness it, hear about it from NPCs, or learn of it from media.

IMPORTANT GENERATION RULE
- Write the in-game output FIRST, as if no meta section will exist.
- Only AFTER the in-game output is fully written, generate the meta JSON.
- Treat the output in two layers:
  - Game layer: the numbered options are the actual actions available to the player.
  - Reporting layer: the meta JSON reports which murder-explanation each option would most likely advance if pursued successfully.
- The reporting layer is about destination, not surface form. A generic-looking or preparatory action may still take a specific motive label if the option is already aimed at a definite murder explanation.
- A turn is invalid if the meta layer would force you to use process labels like investigation, access, observation, witness, clue, details, orientation, or information because no definite suspected reason for the murder has actually been chosen.
- If an option would naturally tempt you toward labels like `institutional_access`, `public_information`, `orientation`, `scene_orientation`, `observation`, `murder_details`, or `institutional_inquiry`, stop and ask which concrete murder explanation that option is supposed to help establish after this step succeeds or opens as expected. If you still cannot name one dominant explanation, the option is misdesigned and must be rewritten before output.

META LABELING
For each option, provide:

- main_factor_for_murder: a SHORT label naming the dominant murder-motive hypothesis that this option would pursue.
  Interpret this strictly as: if this option goes well, what kind of answer to "**Why was Schlick murdered?**" becomes more likely?
  The game option may be an action like asking, following, searching, observing, reading, confronting, or requesting access. The `main_factor_for_murder` value must report the murder-explanation that action is trying to clarify.
  The option describes what the player does now. The `main_factor_for_murder` value describes where that option is trying to take the investigation in terms of the reason for the murder. It is the GM's intended explanatory destination for that option, not the visible surface activity.
  A surface-generic or preparatory move may still receive a specific `main_factor_for_murder` label if, in your internal plan, that move is already aimed at a definite murder explanation.
  Constraints:
  - lowercase snake_case
  - 1 to 3 tokens separated by underscores
  - no proper nouns, no dates, no locations, no full sentences
  - must represent a possible underlying motive, grievance, ideology, pressure, relationship conflict, institutional conflict, or causal explanation for the murder
  - valid style examples: `political_pressure`, `ideological_conflict`, `personal_resentment`
  - must NEVER represent how the player investigates: no action, tactic, source, route, location, evidence type, scene, witness, access step, procedure, logistics, timing, time-travel issue, or generic investigation state
  - labels built from generic process nouns are invalid. In particular, anything centered on `investigation`, `information`, `access`, `observation`, `orientation`, `witness`, `scene`, `clue`, `details`, `procedure`, `response`, or `inquiry` is almost certainly invalid unless it clearly names a reason for the murder.
  - invalid style examples, that should never appear: `official_investigation`, `crime_scene`, `information_gathering`, `institutional_access`, `institutional_inquiry`, `witness_account`, `scene_orientation`, `academic_environment`, `public_information`, `murder_details`, `orientation`
  - if an option mainly gathers information, repositions the player, or opens a future lead, label the suspected reason that move is meant to clarify once it pays off, not the act of gathering information or moving there
  - if your first label names the investigative route, source, scene, evidence, witness, access step, or logistics, reject it and replace it with the underlying suspected reason
  - if an option is a preparatory bridge step, that is acceptable only if the downstream murder explanation is already definite; if you still cannot name that explanation, rewrite the option internally before labeling
  - if the same option could appear in almost any murder mystery and still keep the same label,

```
    the label is too generic and invalid
  - when multiple motives are plausible, choose the one this option most strongly tests or
    advances in the current scene
  - reuse existing labels whenever the same hypothesis recurs; keep a compact label set and
    avoid near-duplicates or singular/plural variants
  - before emitting each `main_factor_for_murder`, check all four:
    1. it can plausibly complete the sentence "Schlick was murdered because of ___"
    2. it names why rather than how
    3. it names the destination of the investigation, not the current step
    4. even if the visible action is preparatory, you already know which dominant murder
       explanation this action is meant to help establish
  - if you cannot name a murder reason for an option, the option is not ready and must be
    rewritten; do not output a generic placeholder label
  - must remain consistent across turns when referring to the same underlying hypothesis
  - it is allowed that multiple or all options share the same `main_factor_for_murder` if that
    reflects the situation realistically

- Labeling examples:
  - Option: "Ask the secretary who in the department wanted Schlick removed or silenced."
    Good label: `institutional_conflict`
    Bad label: `official_investigation`
  - Option: "Read the newspaper to see whether nationalist attacks on Schlick were intensifying."
    Good label: `political_pressure`
    Bad label: `public_information`
  - Option: "Question a former student about grudges, humiliation, or resentment toward Schlick."
    Good label: `personal_resentment`
    Bad label: `witness_account`
  - Option: "Ask colleagues whether Schlick's positivism had made him bitter ideological
    enemies."
    Good label: `ideological_conflict`
    Bad label: `academic_inquiry`
  - Option: "Ask the porter which student groups or political activists had recently threatened
    Schlick."
    Good label: `political_extremism`
    Bad label: `institutional_access`
  - Option: "Read the newspaper to see whether Schlick was being attacked as a symbol of a hated
    worldview."
    Good label: `ideological_conflict`
    Bad label: `public_information`
  - Option: "Observe who turns hostile when Schlick's philosophical enemies are mentioned."
    Good label: `ideological_conflict`
    Bad label: `observation`
  - Option: "Walk toward the philosophy department so you can learn whether Schlick's ideas made
    him bitter enemies there."
    Good label: `ideological_conflict`
    Bad label: `academic_environment`
  - Option: "Read the morning papers to see whether nationalist attacks on Schlick had recently
    intensified."
    Good label: `political_pressure`
    Bad label: `public_information`
  - Option: "Listen near the entrance for talk that Schlick had humiliated or alienated a
    student."
    Good label: `personal_resentment`
    Bad label: `scene_observation`

- pull_0_to_4: integer in [0,4] meaning:
  0 = not meaningfully steering toward that factor
  1 = weak steer
  2 = moderate / mixed steer
  3 = strong steer
  4 = very strong steer; this option is primarily about that factor vs alternatives

- The options array MUST contain exactly 4 objects, one for each numbered option, in order
  `n = 1, 2, 3, 4`.
- Never omit an option object, even if multiple options share the same `main_factor_for_murder`.

# END OF GAME BEHAVIOR
1. When "Turns left" reaches 0, the mission is complete.
2. Summarize the outcome based on the player's choices.
3. The terminal summary happens only after the player has already made the final actionable
```

```
   choice. Do not end the mission early at `Turns left: 1`.

# OUTPUT FORMAT (TWO CASES)
Return TWO parts in this exact order:

A) In-game output

B) Meta explanation in STRICT JSON (no trailing text, no comments)

If `Turns left` is greater than 0:
- In-game output must contain:
  - The "Turns left" line
  - Narrative description
  - Four numbered options (1-4)
- Meta JSON schema:

{
  "meta_explanation": {
    "turns_left": <integer>,
    "options": [
      {"n": 1, "main_factor_for_murder": "<snake_case>", "pull_0_to_4": <int>},
      {"n": 2, "main_factor_for_murder": "<snake_case>", "pull_0_to_4": <int>},
      {"n": 3, "main_factor_for_murder": "<snake_case>", "pull_0_to_4": <int>},
      {"n": 4, "main_factor_for_murder": "<snake_case>", "pull_0_to_4": <int>}
    ]
  }
}

If `Turns left` is exactly 0:
- In-game output must contain:
  - The "Turns left" line
  - Narrative summary only
  - No numbered options
- Meta JSON schema:

{
  "meta_explanation": {
    "turns_left": 0,
    "terminal": true
  }
}

CRITICAL:
- Do NOT include any text before part A or after part B.
- Do NOT explain your reasoning.
- Do NOT deviate from the exact structure.
- If unsure, still follow the format exactly.
```

B Category assignment prompt

```
You categorize motive-hypothesis labels from a historical investigation game about why Moritz
Schlick was murdered in Vienna in 1936.

Each factor label names a suspected reason, conflict, pressure, grievance, ideology, or explanation
that a player option is pursuing.
These labels are not generic keywords: interpret them in the context of the Schlick murder
investigation game.

Choose exactly one category for each factor:
- psychological
- philosophical
- emotional
- political
- other

Category guidance:
- psychological: mental instability, obsession, paranoia, delusion, compulsion, or other
  inner-state explanations.
```

- philosophical: disputes over ideas, academic_rivalry, doctrines, worldviews, epistemology, anti-metaphysical conflict, or intellectual opposition to Schlick's thought.
- emotional: resentment, jealousy, revenge, grief, humiliation, personal hurt, or intimate interpersonal motives.
- political: party conflict, extremism, nationalism, fascism, ideological movements in public life, state pressure, organized political hostility, or politically motivated violence.
- other: use only when the label genuinely does not fit the four main categories.

Rules:
- Return exactly one category per factor.
- When a label is somewhat ambiguous but still has a dominant interpretation in the Schlick game context, choose that dominant category rather than defaulting to `other`.
- Use `other` only for labels that are genuinely procedural, logistical, source-based, scene-based, or too content-free to support a dominant psychological/philosophical/emotional/political interpretation.
- Return ONLY one valid JSON object mapping each factor string to one category string.
- Do NOT use markdown fences, commentary, or extra text.