CarMem: enhancing long-term memory in LLM voice assistants through category-bounding

Johannes Kirmayr, Lukas Stappen, Phillip Schneider, Florian Matthes, Elisabeth André

Angaben zur Veröffentlichung / Publication details:

Kirmayr, Johannes, Lukas Stappen, Phillip Schneider, Florian Matthes, and Elisabeth André. 2025. “CarMem: enhancing long-term memory in LLM voice assistants through category-bounding.” In *Proceedings of the 31st International Conference on Computational Linguistics (COLING 2025): industry track, January 19–24, 2025, Abu Dhabi, UAE*, edited by Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, Steven Schockaert, Kareem Darwish, and Apoorv Agarwal, 343–57. Stroudsburg, PA: Association for Computational Linguistics (ACL). https://aclanthology.org/2025.coling-industry.29/.

CarMem: Enhancing Long-Term Memory in LLM Voice Assistants through Category-Bounding

Johannes Kirmayr[1,2], **Lukas Stappen**[1], **Phillip Schneider**[3], **Florian Matthes**[3], **Elisabeth André** [2]

[1]BMW Group Research and Technology, Munich, Germany
[2]Chair for Human-Centered Artificial Intelligence, University of Augsburg, Germany
[3]Chair for Software Engineering for Business Information Systems,
Technical University of Munich, Germany

Correspondence: Johannes.Kirmayr@bmwgroup.com

Abstract

In today's assistant landscape, personalisation enhances interactions, fosters long-term relationships, and deepens engagement. However, many systems struggle with retaining user preferences, leading to repetitive user requests and disengagement. Furthermore, the unregulated and opaque extraction of user preferences in industry applications raises significant concerns about privacy and trust, especially in regions with stringent regulations like Europe. In response to these challenges, we propose a long-term memory system for voice assistants, structured around predefined categories. This approach leverages Large Language Models to efficiently extract, store, and retrieve preferences within these categories, ensuring both personalisation and transparency. We also introduce a synthetic multi-turn, multi-session conversation dataset (CarMem), grounded in real industry data, tailored to an in-car voice assistant setting. Benchmarked on the dataset, our system achieves an F1-score of .78 to .95 in preference extraction, depending on category granularity. Our maintenance strategy reduces redundant preferences by 95% and contradictory ones by 92%, while the accuracy of optimal retrieval is at .87. Collectively, the results demonstrate the system's suitability for industrial applications.

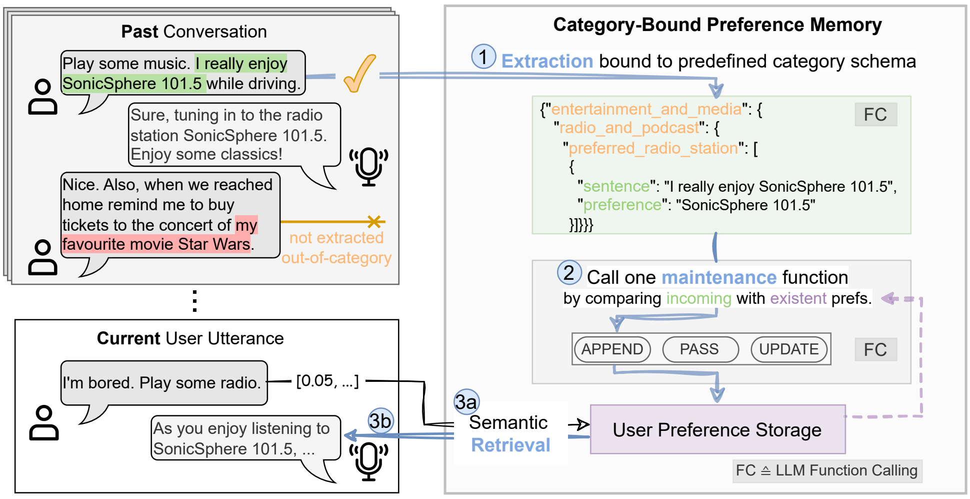
1 Introduction

Memory retention is essential in human interaction for building long-term relationships (Alea and Bluck, 2003; Brewer et al., 2017). Similarly, virtual dialogue systems aim to leverage conversation memories for a more personalised user experience. Large Language Models (LLMs) have become a prominent technology in powering such virtual dialogue systems. Given that LLMs are inherently stateless, all relevant memories need to be presented during each interaction. Presenting all past messages to an LLM degrades performance (Liu et al., 2024) and increases costs. Therefore, an external preference memory system is needed that selectively presents a relevant subset of previously extracted memories for the current conversation turn. However, when engaging with virtual non-human assistants like an in-car personal voice assistant, limitations and concerns arise:

(1) Privacy Concerns: End-users may have concerns about the extraction and storage of private information from their interactions. In Europe, the GDPR (Commision, 2016) enforces data minimization, requiring that data be "adequate, relevant, and limited to what is necessary" for the purposes it is processed under Article 5(1)(c). Additionally, the EU AI Act (Parliament and Council, 2024) mandates a high degree of transparency, reinforcing the need for clear communication about how user data is handled. (2) Technological Constraints: In-car voice assistants are limited in the information they can actually use due to the restricted action space of the vehicle's systems. For example, the preferred radio station can be set as a parameter in the entertainment system, while the favourite movie genre is not applicable. Unbounded information extraction would lead to irrelevant and resource-inefficient storage of memories.

Our work addresses these industry-relevant challenges by proposing a category-bound preference memory system. This system restricts information extraction, with a focus on user preferences, to hierarchically predefined categories. Thereby, companies pre-define categories to prevent capturing non-actionable information, and users have the control to further refine this by opting out of specific categories. An overview of the category-bound preference memory flow is shown in Figure 1. The memory system consists of three main components. (1) Extraction, which captures in-category preferences after conversations while ignoring out-of-category ones. (2) Maintenance based on Bae

Proceedings of the 31st International Conference on Computational Linguistics: Industry Track, pages 343–357
January 19–24, 2025. ©2025 Association for Computational Linguistics


Past Conversation
Category-Bound Preference Memory
1 Extraction bound to predefined category schema
FC
not extracted
out-of-category
2 Call one maintenance function
by comparing incoming with existent prefs.
APPEND
PASS
UPDATE
FC
Current User Utterance
[0.05, ...]
3a
Semantic
Retrieval
3b
User Preference Storage
FC ≙ LLM Function Calling

Play some music. I really enjoy SonicSphere 101.5 while driving.

Sure, tuning in to the radio station SonicSphere 101.5. Enjoy some classics!

Nice. Also, when we reached home remind me to buy tickets to the concert of my favourite movie Star Wars.

```
{"entertainment_and_media": {
 "radio_and_podcast": {
  "preferred_radio_station": [
  {
   "sentence": "I really enjoy SonicSphere 101.5",
   "preference": "SonicSphere 101.5"
  }]}}}
```

I'm bored. Play some radio.

As you enjoy listening to SonicSphere 101.5, ...

Figure 1: High-level memory flow: After a conversation, preferences are extracted (1) based on the predefined category schema (e.g. preferred radio station). Topics outside the category schema, such as favourite movies, are not extracted. (2) Before inserting a new preference, it is compared to existing preferences for consistency, applying the most suitable maintenance operation: append, pass, or update. Within the next conversation (3), the voice assistant retrieves semantically relevant preferences (3a) from the user storage (3b) to provide a personalized response.

et al. (2022), which keeps the preference storage up-to-date by calling a maintenance function before storing a preference. (3) Retrieval, which semantically retrieves relevant preferences for the current user utterance to provide personalized responses.

Furthermore, we introduce a carefully constructed synthetic dataset. This dataset focuses on an in-car voice assistant context with multi-turn interactions. The dataset is designed to evaluate the main components of the external memory system. We benchmark our system on the dataset. In summary, the main contributions of this work are:

1. Category-bound preference memory system based on user-assistant conversations.
2. Closed-world in-car conversational dataset CarMem with benchmark values for main components of our long-term memory system.

Our dataset and code are publicly available. [1]

2 Related Work

Cognitive neuroscience distinguishes between semantic memory (general knowledge) and episodic memory (personal events) (Tulving, 1972). While LLMs effectively cover semantic memory, episodic memory must be handled manually. Personalized dialogue systems aim to leverage episodic memory to enhance user experience by tailoring interactions based on individual preferences. Early approaches used static user profiles (Zhang et al., 2018), while more dynamic methods include memory-augmented networks (Meng and Huang, 2018), memory-augmented LLMs (Wang et al., 2023b), and external memories that continuously update user memories (Xu et al., 2022a,b, 2023). Due to scalability issues with memory-augmented LLMs, we focus on external memory systems that retrieve relevant information as needed. Several works have explored external memories. Park et al. (2023) used an event-based memory in LLM-powered characters for personalized interaction with other characters. MemGPT (Packer et al., 2023) introduces an operating-system-inspired dual-memory structure. Meanwhile, Zhong et al. (2024) enhances its memory mechanism by introducing a human-like forgetting curve.

These advancements, however, have brought new challenges: they deploy unstructured extraction methods, which result in unordered memory pieces in text format, making structured and transparent information an underexplored area. Additionally, with the growing focus on transparency in AI (Adadi and Berrada, 2018), regulations like GDPR (Commision, 2016), and the EU AI Act (Parliament and Council, 2024), there is increasing demand for systems that offer users more control. OpenAI introduced a memory feature in their ChatGPT interface (OpenAI, 2024c), where user control is limited to deleting memories after extraction. Our approach differs by allowing users to control what gets extracted initially through the ability to opt-out from specific category topics.

For maintaining relevant memory, Xu et al.

[1] Dataset and Code is available on https://github.com/johanneskirmayr/CarMem.

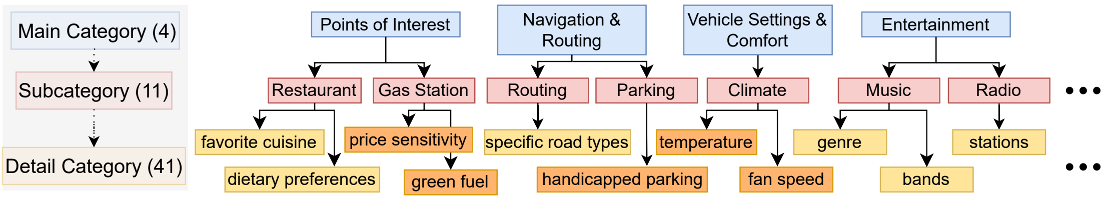


Figure 2: Representative subset of the hierarchically predefined preference categories. There are two types of detail categories: MP (yellow): Multiple preferences within the category are possible, and SP (orange): Single preference within the category is allowed. A full list of categories with attributes is provided in Appendix D.1.

(2022b) use cosine similarity to remove duplicates, and Bae et al. (2022) introduced LLM-driven memory maintenance. We extend this with LLM function calling and structured information representation. Retrieval-augmented generation based on embeddings (Lewis et al., 2020) has been adapted for preference storage and retrieval (Zhong et al., 2024; Wang et al., 2024). In addition, our system leverages category-based storage to enrich embeddings, improving retrieval accuracy.

These advancements are often limited by the datasets available for evaluation. Existing datasets, either focus on user-user conversations (Xu et al., 2022a), are open-domain (Xu et al., 2022b), or consist of only a single conversational session (Zhang et al., 2018). Additionally, datasets such as (Dinan et al., 2020) emphasise the assistant's persona rather than user-specific preferences, making them unfit for evaluating long-term, personalised systems, particularly in the context of in-car voice assistants. To address these gaps, we introduce a synthetically generated dataset. Synthetic datasets have been proven effective in simulating complex, controlled scenarios, especially when real-world data is difficult to obtain (Paulin and Ivasic-Kos, 2023; Gonzales et al., 2023; Wang et al., 2023a).

3 Structured and Category-Bound User-Preference-Memory

Our system manages user preferences through three stages: hierarchical preference extraction, ongoing maintenance, and retrieval for future interactions.

3.1 Preference Extraction

Preferences are extracted from conversations and constrained to predefined hierarchical categories. Relevant categories aligned to the in-car assistant are shown in Figure 2. With this, a user could have a preference for Italian food within the category Points of Interest (Main), Restaurant (Sub), Favourite Cuisine (Detail). Category-bound extraction (1) increases the transparency by showing which preferences are stored and where; (2) allows users to opt out of categories, for instance, due to privacy concerns; and (3) aligns with the limited action space of downstream car functions, avoiding irrelevant preferences. Hierarchical, category-based extraction is achieved via LLM function calling.

LLM Function Calling: Function calling enhances control and reliability in extracting structured information compared to simple prompt-based methods. The LLM is trained to match a predefined parameter schema, ensuring a specific output format (JSON) and extracting only relevant information from the input text for the designated function parameters.

A function definition consists of the name of the function, a description of the purpose, and a parameter schema. We define a function to extract preferences and use the function parameter schema to represent our categories and their hierarchy as parameters. The parameter schema is defined with pydantic (Colvin et al., 2024) and presented in Appendix E. In the schema, we define every parameter, representing one category (favourite cuisine, preferred radio station, etc.), as `Optional` so that the LLM is not forced to extract a preference within that category. By using the extraction function on a conversation, the LLM fills in the values of the nested schema, effectively extracting preferences according to the predefined categories and their hierarchy. Out-of-category preferences are either ignored by the LLM as there is no fitting function parameter or extracted in our designated `no_or_other_preference` parameter within the sub- and detail categories which are later discarded.

3.2 Preference Maintenance

Once extracted, it is essential to maintain the preferences by checking for redundancy or contradictions before storage. Following Bae et al. (2022), we have implemented three maintenance functions to account for this: **`Pass`**: The incoming preference already exists in the storage and is not inserted again; **`Update`**: The incoming preference updates an existing preference. The new preference is inserted, and the corresponding existing preference is deleted; **`Append`**: The incoming preference is new and not present in the storage. These functions are again used with LLM function calling, defined with a name, description, and parameter schema. The `append` function requires no parameters, while the `pass` and `update` functions need specification of the existing preference causing the call. To streamline comparison, we use the structured storage and present the LLM only existing preferences in the same detail category. Some detail categories allow only a single preference (cf. Figure 2) - in these cases, we disable the `append` function if a preference already exists.

3.3 Preference Retrieval

After maintaining an up-to-date database, the next step is to ensure that relevant preferences are retrieved during future interactions. To achieve this, we generate an embedding representation from a concatenated string of the detail category, preference attribute, and the sentence revealing the preference. Embeddings capture semantic relationships between preferences and context, enabling robust, low-latency retrieval, even with varied user phrasing. We retrieve the most relevant preferences by embedding similarity with the user utterance.

4 Data

This section outlines the construction of our synthetically generated dataset CARMEM . To evaluate the reliability of the category-bound extraction, the dataset features realistic multi-turn in-car *Extraction Conversations* where the user reveals exactly one given preference. Additionally, the dataset includes, in a second session, *Retrieval Utterances* for recalling preferences, and *Maintenance Utterances* for benchmarking maintenance scenarios. Figure 3 shows an example.

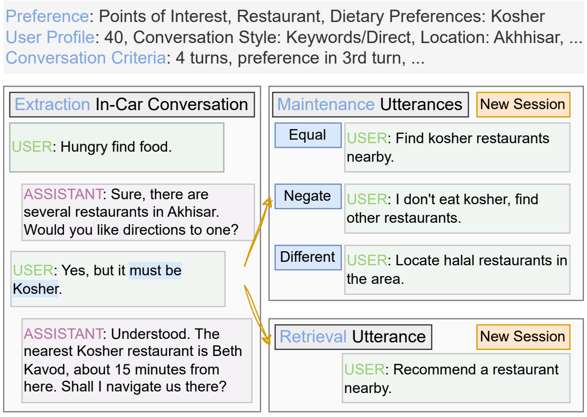

Preference: Points of Interest, Restaurant, Dietary Preferences: Kosher
User Profile: 40, Conversation Style: Keywords/Direct, Location: Akhhisar, ...
Conversation Criteria: 4 turns, preference in 3rd turn, ...
Extraction In-Car Conversation
USER: Hungry find food.
ASSISTANT: Sure, there are several restaurants in Akhisar. Would you like directions to one?
USER: Yes, but it must be Kosher.
ASSISTANT: Understood. The nearest Kosher restaurant is Beth Kavod, about 15 minutes from here. Shall I navigate us there?
Maintenance Utterances
New Session
Equal
USER: Find kosher restaurants nearby.
Negate
USER: I don't eat kosher, find other restaurants.
Different
USER: Locate halal restaurants in the area.
Retrieval Utterance
New Session
USER: Recommend a restaurant nearby.

Figure 3: Example data point of the synthetically generated CARMEM dataset showing the three different parts.

Statistics	
Extraction Conversations	$1,000$
Avg. tokens for generation	976
Avg. turns per conversation	5.08
Avg. words per conversation	80.78
Retrieval Utterances	$1,000$
Avg. tokens for generation	353
Avg. words per utterance	8.34
Maintenance Utterances	$3,000$
Avg. tokens for generation	357
Avg. words per utterance	12.06

Table 1: Statistics of our CARMEM dataset.

To generate the dataset, we use the LLM `GPT-4-1106-preview` (OpenAI, 2024d) with temperature 0.7, balancing creativity and coherence (cf. Appendix B). To ensure realistic conversations, we prompt the LLM with an elaborate input framework. For this, we have created 100 user profiles with varying characteristics in age, technological proficiency, user location, and conversation style. The latter, derived from real-world in-car conversations, ranges from commanding, keyword-only, questioning, to conversational and significantly influences the generated text. As seen in Figure 3, this can result in grammatically incorrect, but realistic interactions. Each profile is assigned 10 preferences, uniformly sampled across the predefined detail category level (cf. Figure 2). The categories are based on the most used car functionalities in the currently deployed voice assistant. For each preference, we create one *Extraction Conversation* where the user reveals the given preference. While the user characteristics remain consistent across the 10 generated conversations, the conversation criteria (e.g. conversation length (2-8 turns), position of preference-reveal, preference strength) are randomly sampled for increased diversity (cf. Appendix C.2). Additionally, we provide a real, topic-dependent conversation turn as a few-shot example for each generation. Strict guidance, random sampling, and the LLM's natural language generation create realistic, controlled, yet diverse dataset entries reflecting preferences relevant to the automotive domain. The resulting dataset contains 1,000 *Extraction Conversations*, 1,000 *Retrieval Utterances*, and 3,000 *Maintenance Utterances*, detailed statistics are shown in Table 1. Human evaluation results, showing the dataset's high quality and realism, are in Appendix C.1.

5 Experiments

The results are benchmarked on our dataset CARMEM. We applied a 50-50 split on validation and testing, resulting in 500 test entries. The experiments including an LLM, i.e. extraction and maintenance, were performed using function-calling with the LLM `GPT-4o` (2024-08-06) (OpenAI, 2024d) at a temperature of 0 to maximize deterministic output (cf. Appendix B).

5.1 Preference Extraction

We conducted two experiments to evaluate preference extraction from the *Extraction Conversations*:

1. **In-Schema**: Evaluates if the ground-truth preference can be extracted within the correct categories in the schema. An extraction is considered correct if the main-, sub-, and detail categories match those of the ground-truth preference.
2. **Out-of-Schema**: Evaluates if the ground-truth preference is not extracted when the corresponding subcategory is excluded from the schema, simulating a user opt-out. For the example "I want kosher food" the sub-category 'Restaurant' and corresponding detail categories would be excluded from the schema. A data point is considered correct if the ground-truth preference is not extracted.

Experiment Setting Both experiments were conducted on 500 *Extraction Conversations*, each containing exactly one ground-truth user preference.

The general extraction statistics in Table 2 show a low risk (6%) of non-extraction when a preference is present and represented in the schema. However, when excluding the subcategory from the schema, the non-extraction is desired and achieved 75% of the time, demonstrating strong boundness to the predefined categories. In general, we see an incorrect over-extraction with rates of 12% and 25%. The high number of valid structured outputs indicates the reliable adherence to the complex extraction schema, as misformatted JSON outputs and incorrect parameter ($\hat{=}$category) names and hierarchies are labelled as invalid.

Table 3 presents detailed extraction results for the In-Schema experiment. While recall for extracting the ground-truth preference and classifying it into the correct main category is high at .94, the performance declines with a deeper hierarchy level and an increasing number of categories. At the most detailed level (41 categories) precision is .75, which we see as a crucial score in an industry application, as it is better to not extract a preference than to extract an incorrect one. Appendix F.1 (Figure 5) includes the confusion matrix for the detail level of the In-Schema experiment, showing that most incorrect extractions occur in semantically closely related categories. This is further supported by the confusion matrix for the subcategory level of the Out-of-Schema experiment (Appendix F.1, Figure 6), which shows no incorrect extractions for semantically distinct categories like 'Climate Control' but significantly more confusions for closely related categories like 'Music' and 'Radio and Podcast'. These results indicate that defining clear and semantically distinct categories is crucial for achieving reliable category-bound extraction.

Extraction		In-Schema	Out-of-Schema
no	extraction	6%	**75%**
1	preference	**82%**	23%
2+	preferences	12%	2%
valid struct. output		99%	99%

Table 2: Statistics for the two *Extraction Conversation* experiments (1) In-Schema and (2) Out-of-Schema, with the ground-truth subcategory included (expects extraction of 1 preference, highlighted in bold) or excluded in the category schema (expects no extraction, highlighted in bold). The structured output is valid if the output JSON is parseable and matches the schema.

Level	#cat.	Prec. ↑	Rec. ↑	F1 ↑
Main	4	.93	.94	.94
Sub	11	.90	.91	.90
Detail	41	.75	.81	.78

Table 3: **In-Schema**. Performance scores (micro-averaged) for the *Extraction Conversations* and the ground-truth category included in the category schema. (#cat.) indicates the number of categories per level.

5.2 Preference Maintenance

Table 4 shows that each of the three *Maintenance Utterance* types is assigned a specific function call as its ground truth label. This mapping is based on the incoming preference from the *Maintenance Utterance*, the existing preference from the *Extraction Conversation*, and the detail category type. A data point is considered correct if the ground truth maintenance function is called.

Experiment Setting To ensure an independent evaluation, we perform the maintenance evaluation only on the dataset entries with perfect extraction

Type	Label
equal preference	→ `pass` (MP, SP)
negate preference	→ `update` (MP, SP)
different preference	→ `append` (MP)
	→ `update` (SP)

Table 4: Mapping of *Maintenance Utterance* type to maintenance function considering the detail category type (MP: multiple preferences allowed, SP: single preference allowed).

accuracy for both the original preference in the *Extraction Conversation* and the modified preferences in the *Maintenance Utterances*. The number of data points for each experiment is shown in Table 5. On average, each user has 7.02 existing preferences from the corresponding *Extraction Conversations*.

	#	Type	`pass`	`update`	`append`
MP	159	equal	**.86**	.03	.11
	143	negate	.00	**.87**	.13
	159	different	.03	.04	**.92**
SP	192	equal	**.68**	.32	-
	160	negate	.02	**.99**	-
	192	different	.01	**.99**	-

Table 5: Modified confusion matrix for the maintenance function calling task, segmented by categories that allow multiple preferences (MP) or a single preference (SP). Expected mapping (highlighted in bold) from *Maintenance Utterance* type to function is shown in Table 4. (#) indicates the number of data points used per type.

From the weighted average (MP & SP) in Table 5, 76% of equal preferences were correctly passed. Since updating an equal preference yields the same result as passing it, our maintenance method achieves a 95% reduction in redundant preferences. Additionally, contradictory preferences are reduced by 93% as negated preferences are updated. However, in 2%, preferences are still lost due to incorrect passes. In the MP case, 12% are still wrongly appended, similar to scenarios without maintenance.

5.3 Preference Retrieval

In the CarMem dataset, each *Retrieval Utterance* is designed to focus on the topic of the ground-truth subcategory, targeting the retrieval of the corresponding ground-truth preference. While the k for semantic retrieval is fixed in practice, we adapt it dynamically to provide more insightful results. Consequently, retrieval is considered optimal if the ground-truth preference is among the top-$n_{i,j}$ retrieved preferences, where $n_{i,j}$ represents the number of preferences stored for user i within subcategory j. On average, the parameter n is 1.57 and each user has 7.02 preferences stored.

Experiment Setting We perform the retrieval experiment on the 351 preferences with optimal extraction accuracy. Embeddings are generated using the OpenAI `text-embedding-ada-002` model.

Embedding	$k =$	n	$n+1$	$n+2$
Sentence only		.75	.88	.93
Detail Cat.+Attr.+Sent.		**.87**	**.94**	**.97**

Table 6: Top-k accuracy for retrieving the ground-truth preference based on the *Retrieval Utterance*. The parameter n is set dynamically to the number of preferences stored for the user i and subcategory j. Embeddings are created either (1) from the sentence where the preference was revealed or (2) enriched by the preference detail category and attribute.

Table 6 shows the results for two embedding approaches: (1) embeddings created solely from the sentence where the preference is revealed, and (2) embeddings enriched with the structured extraction, including the detail category and the attribute. Given that, on average, 7.02 preferences are stored and $\overline{n} = 1.57$, we can observe an effective retrieval. Furthermore, the enriched embedding outperforms the 'sentence only' embedding by .12 in accuracy for optimal retrieval. This improvement is evident in the following example:

- **Sentence only**: "I always find NavFlow to be reliable."
- **Detail Cat.+Attr+Sent.**: "traffic information source preferences: NavFlow. I always find NavFlow to be reliable."

We observe that categories clarify ambiguous sentences by providing additional context, and fixed category names help cluster preferences more closely in the embedding space.

6 Conclusion

We presented a structured, category-bound preference memory system capable of extracting, maintaining, and retrieving user preferences, while enhancing transparency and user control in privacy-critical contexts. Our approach utilizes a synthetic dataset grounded in real in-car conversations to ensure realism. Benchmarking the core components of the preference memory on this dataset demonstrated both the system's utility and strong performance. Future work could build upon the dataset, refine our baseline methods, and explore generalizing to other industry domains such as smart homes, further validating the approach's adaptability.

7 Limitations

The dataset contains exactly one preference per conversation, which is beneficial for evaluation but does not account for conversations containing no or multiple preferences. While we carefully simulated realistic in-car user-assistant interactions, we did not incorporate additional speech recognition errors or repeated user requests, both of which are common in real-world scenarios. Although LLMs often provide automatic corrections for such issues in practice, structural testing could yield further insights into robustness.

Moreover, the dataset represents interactions across only two timeframes, limiting our evaluation to the basic functionalities without testing the long-term ability to adapt to changing user preferences. Incorporating techniques such as temporal decay of memorized preferences (Zhong et al., 2024) or assigning importance ratings(Park et al., 2023) could improve our maintenance methods.

Although the preference extraction experiment adhered well to the category schema, incorrect over-extraction occurred at rates of 12% to 15%. To mitigate this, we propose to leverage in-context learning capabilities of the LLM and provide explicit few-shot examples where no preference should be extracted. Furthermore, we used OpenAI's JSON mode for data extraction. However, the just-released structured output mode by OpenAI (2024b) reportedly adheres 100% to the provided schema, which could further improve our preference extraction results.

8 Ethical considerations

Our dataset was synthetically generated and does not contain any personally identifiable information. The attributes for the categories such as 'favourite artist' or 'preferred radio station' were also generated, ensuring no real persons or brand names were included. For the user profiles used in dataset generation, we only incorporated neutral information such as age or conversation style, avoiding sensitive attributes like gender or ethnic background. However, since LLMs are trained on vast amounts of mostly online data, they inherit harmful social biases (Gallegos et al., 2024), which could be reflected in our dataset. By prompting the LLM with bias-neutral few-shot examples, we aimed to guide the model toward fairer extractions.

Our proposed preference memory system is designed to be transparent and explainable in its approach for extracting and managing user preferences. This aligns with emerging AI regulations such as the EU AI Act (Parliament and Council, 2024) which mandates transparency, and the General Data Protection Regulation (GDPR) (Commision, 2016), which emphasizes data protection and user consent. A key aspect of our system is category-bound extraction, which follows the principles of data minimization and user control. By aiming to extract and store only actionable information and allowing users to opt out of specific categories, we preserve user privacy while maintaining system intelligence.

However, despite our system's safeguards, it does not achieve perfect accuracy, and LLMs may hallucinate. This introduces potential risks, such as the extraction of false or irrelevant preferences. To mitigate this, integrating extracted data in the UX flow and transparently displaying them on the user interface, provides users with the ability to manually delete memories. Additionally, offering an interaction tool via voice allows users to review, edit, or delete preferences, maintaining system accuracy and trust. Future work may explore confidence thresholds that trigger user confirmation for uncertain extractions.

References

Amina Adadi and Mohammed Berrada. 2018. Peeking inside the black-box: A survey on explainable artificial intelligence (xai). *IEEE Access*, 6:52138–52160.

Nicole Alea and Susan Bluck. 2003. Why are you telling me that? a conceptual model of the social function of autobiographical memory. *Memory*, 11(2):165–178.

Sanghwan Bae, Donghyun Kwak, Soyoung Kang, Min Young Lee, Sungdong Kim, Yuin Jeong, Hyeri Kim, Sang-Woo Lee, Woomyoung Park, and Nako Sung. 2022. Keep me updated! memory management in long-term conversations. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3769–3787. Association for Computational Linguistics.

Robin N. Brewer, Meredith R. Morris, and Siân Lindley. 2017. How to remember what to remember: Exploring possibilities for digital reminder systems. *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies*, 1(3):1–20.

Samuel Colvin, Eric Jolibois, Hasan Ramezani, Adrian Garcia Badaracco, Terrence Dorsey, David Montague, Serge Matveenko, Marcelo Trylesinski, Sydney Runkle, David Hewitt, and Alex Hall. 2024. Pydantic. (accessed March 12, 2024).

European Commision. 2016. Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec (general data protection regulation) (text with eea relevance). *Official Journal of the European Union*, (119):1–88.

Emily Dinan, Varvara Logacheva, Valentin Malykh, Alexander Miller, Kurt Shuster, Jack Urbanek, Douwe Kiela, Arthur Szlam, Iulian Serban, Ryan Lowe, Shrimai Prabhumoye, Alan Black, Alexander Rudnicky, Jason Williams, Joelle Pineau, Mikhail Burtsev, and Jason Weston. 2020. The second conversational intelligence challenge (convai2). In *The NeurIPS '18 Competition*, pages 187–208. Springer International Publishing.

Isabel O. Gallegos, Ryan A. Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K. Ahmed. 2024. Bias and Fairness in Large Language Models: A Survey. *Computational Linguistics*, 50(3):1097–1179.

Aldren Gonzales, Guruprabha Guruswamy, and Scott R. Smith. 2023. Synthetic data in health care: A narrative review. *PLOS Digital Health*, 2(1):e0000082.

Mohammadreza Heydarian, Thomas E. Doyle, and Reza Samavi. 2022. Mlcm: Multi-label confusion matrix. *IEEE Access*, 10:19083–19095.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Lian Meng and Minlie Huang. 2018. Dialogue intent classification with long short-term memory networks. In *Natural Language Processing and Chinese Computing*, pages 42–50, Cham. Springer International Publishing.

OpenAI. 2024a. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

OpenAI. 2024b. Introducing structured outputs in the api. `https://openai.com/index/introducing-structured-outputs-in-the-api/`. (accessed 04-September-2024).

OpenAI. 2024c. Memory and new controls for chatgpt. `https://openai.com/index/memory-and-new-controls-for-chatgpt/`. (accessed 04-September-2024).

OpenAI. 2024d. Models. `https://platform.openai.com/docs/models`. (accessed 04-September-2024).

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. 2023. Memgpt: Towards llms as operating systems. *Preprint*, arXiv:2310.08560.

Joon S. Park, Joseph C. O'Brien, Carrie J. Cai, Meredith R. Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *In the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)*, pages 1–22. Association for Computing Machinery.

European Parliament and European Council. 2024. Regulation (eu) 2024/1689 of the european parlament and of the council laying down harmonised rules on artificial intelligence and amending regulations (ec) no 300/2008, (eu) no 167/2013, (eu) no 168/2013, (eu) 2018/858, (eu) 2018/1139 and (eu) 2019/2144 and directives 2014/90/eu, (eu) 2016/797 and (eu) 2020/1828 (artificial intelligence act). *Official Journal of the European Union*, OJ L.

Goran Paulin and Marina Ivasic-Kos. 2023. Review and analysis of synthetic dataset generation methods and techniques for application in computer vision. *Artificial intelligence review*, 56(9):9221–9265.

Endel Tulving. 1972. Episodic and semantic memory. In *Organization of Memory*, pages 381–403. Academic Press, New York.

Bing Wang, Xinnian Liang, Jian Yang, Hui Huang, Shuangzhi Wu, Peihao Wu, Lu Lu, Zejun Ma, and Zhoujun Li. 2024. Enhancing large language model with self-controlled memory framework. *Preprint*, arXiv:2304.13343.

Jian Wang, Yi Cheng, Dongding Lin, Chak Leong, and Wenjie Li. 2023a. Target-oriented proactive dialogue systems with personalization: Problem formulation and dataset curation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1132–1143. Association for Computational Linguistics.

Weizhi Wang, Li Dong, Hao Cheng, Xiaodong Liu, Xifeng Yan, Jianfeng Gao, and Furu Wei. 2023b. Augmenting language models with long-term memory. *Advances in Neural Information Processing Systems*, 36.

Derong Xu, Wei Chen, Wenjun Peng, Chao Zhang, Tong Xu, Xiangyu Zhao, Xian Wu, Yefeng Zheng, and Enhong Chen. 2023. Large language models for generative information extraction: A survey. *Preprint*, arXiv:2304.13343.

Jing Xu, Arthur Szlam, and Jason Weston. 2022a. Beyond goldfish memory: Long-term open-domain conversation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5180–5197. Association for Computational Linguistics.

Xinchao Xu, Zhibin Gou, Wenquan Wu, Zheng-Yu Niu, Hua Wu, Haifeng Wang, and Shihang Wang. 2022b. Long time no see! open-domain conversation with long-term persona memory. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2639–2650. Association for Computational Linguistics.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 2204–2213. Association for Computational Linguistics.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. 2024. Memorybank: Enhancing large language models with long-term memory. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17):19724–19731.

A Prompts

The prompts for dataset generation, preference extraction, and maintenance function calling are available in our released code on https://github.com/johanneskirmayr/CarMem.

B LLM Temperature Settings

The temperature parameter controls the randomness and creativity of the generated text. We used different settings of temperature depending on the task:

- **Dataset Generation:** According to GPT-4 technical report, a temperature of 0.6 is recommended for free-text generation (OpenAI, 2024a). Considering the need for creativity and diversity in dataset generation task, and referencing related work by Wang et al. (2023a), which employs a temperature of 0.75, we decided on a temperature setting of 0.7.
- **Extraction and Maintenance Function Calling:** For the tasks of extraction and maintenance function calling, we set the temperature to 0. These tasks require precise and consistent outputs without creativity, maximizing deterministic and reproducible results.

C CarMem Dataset

C.1 Human Evaluation

In this section, we present the results of the human evaluation conducted to assess the quality and relevance of the dataset. A subset of 40 data points, systematically selected from 40 users in the CarMem dataset, was evaluated by three human judges. The preferences, which are ordered correspondent to the category list, were chosen in a repeating pattern from the first to the tenth preference. This approach ensured a representative coverage of all preference categories and user profiles. To ensure high intercoder reliability , the judges were provided with detailed instructions. The instructions included the goals for each dataset component, an explanation of the dynamic inputs (user profile, conversation criteria), the evaluation criteria, and guidelines for the different evaluation values. Furthermore, one independent data point was evaluated collaboratively to establish a consistent evaluation standard.

The evaluation criteria for the *Extraction Conversation* part of the CarMem dataset are as follows:

1. **Realism of User Behavior**: Does the simulated user behave and communicate in a manner that reflects how real users would act in a similar in-car situation?
2. **Realism of Assistant Responses**: Are the assistant's responses contextually appropriate, relevant, and reflective of a natural understanding of human speech patterns?
3. **Organicness of User Preference Revelation**: Is the user preference revealed naturally within the flow of the conversation without being forced or out of place?
4. **Clarity of User Preference**: Is the user preference communicated clearly, making it distinct from a temporary wish or a one-off statement?
5. **Environment Understanding**: Does the model demonstrate an understanding of the context in which the conversation is taking place?

Each criterion was assessed on a Likert scale from 1 (worst) to 3 (best). Additionally, each *Extraction Conversation*, *Retrieval Utterance*, and *Maintenance Utterance* is assessed for appropriateness within the dataset and scored for subjective quality on an overall Likert scale rating (1-3). A data point should be scored inappropriate if, for example, the user preference is unclear, the conversation contains multiple preferences, the retrieval utterance already included the ground-truth preference or the maintenance utterances do not fulfil the intended purpose. The majority vote was taken in discordant situations.

Human Evaluation Results Table 7 details the results of the human evaluation on 40 datapoints for the CarMem dataset. The results indicate that the *Extraction Conversations* were generally realistic, with high scores in realism and environment understanding. The reveal of user preferences was mostly natural and clearly identifiable. However, nine conversations were classified as inappropriate: in six cases, the user preferences were not identifiable, and in three cases, multiple preferences, including the ground truth preference, were revealed. Both *Retrieval Utterances* and *Maintenance Utterances* showed high overall subjective quality and a high ratio of appropriate utterances. For the *Retrieval Utterances*, one instance was classified inappropriate due to the utterance not being related to the user preference, and one because of 'other' reason -

Criteria	**Average Score** $[1, 3]$↑	**Ratio ’Appropriate’** $[0, 1]$↑
Extraction Conversations		
Realism of User	2.73	
Realism of Assistant	2.93	
Organicness of User Preference	2.67	
Clarity of User Preference	2.47	
Environment Understanding	3.0	
Overall Subjective Quality	2.18	
Appropriate Conversation for Dataset		$31/40 = 0.78$
Retrieval Utterance		
Overall Subjective Quality	2.75	
Appropriate Question for Dataset		$38/40 = 0.95$
Maintenance Utterances		
Overall Subjective Quality	2.71	
Appropriate Maintenance Questions for Dataset		$40/40 = 1.0$

Table 7: Results of Human Evaluation based on 40 Data Points of the CARMEM dataset.

here the utterance contradicted the user preference.

C.2 Increased Diversity through User Profiles and Conversation Criteria

As detailed in Section 4, dynamic prompt inputs are sampled for generating each conversation. We hypothesize that this variation in user profiles and conversation criteria will result in increased diversity in the generated text.

Experiment Setting To test our hypothesis, we randomly sampled four different user preferences. For each preference, we "regenerate" the conversations 10 times with 2 methods: (1) regenerate with varying dynamic inputs, and (2) regenerate with non-varying fixed inputs. To compare the diversity of the generated conversations, we increasingly concatenate (from 1-10) the regenerated conversations for both methods and calculate the Distinct-1, Distinct-2, and Distinct-3 scores. Calculating the three Distinct-N scores allows for a comprehensive assessment of text diversity across varying levels of lexical and syntactic granularity. The same prompt was used for both methods. The dynamic inputs to the prompt are: User Profile Data (Age, Technological Proficiency, Conversation Style, Location), Conversation Criteria (Position User Preference, Preference Strength Modulation, Level of Proactivity Assistant), and the Few Shot Example. Note: To mitigate the issue of unequal evaluation due to varying text lengths, the conversation length was fixed to six messages for both methods. For the fixed input method, the dynamic inputs were sampled once at the beginning and kept constant across the 10 conversations. For the dynamic input method, inputs were resampled for each conversation. Since the conversation style was found to have a significant influence on the generated text, we exceptionally manually set the conversation style to the four possible values for the four different user preferences in the fixed input method to ensure more representative results. The temperature of each LLM is set to 0.7. The averaged Distinct-N scores across the four preference generations can be seen in Figure 4.

As the number of regenerations increases, diversity tends to decrease for both methods. However, the results indicate that conversations with dynamic inputs consistently achieve higher diversity scores across all Distinct-N metrics compared to those without dynamic, but fixed inputs.

D Predefined Categories

D.1 Full List of Preference Categories with Attributes

In the following, the full list of preference categories with attributes is shown. From this list, every user profile gets sampled 10 preferences.

1. Points of Interest
   - (a) Restaurant
     - i. MP: Favorite Cuisine
       - Attributes: Italian, Chinese, Mexican, Indian, American
     - ii. MP: Preferred Restaurant Type
       - Attributes: Fast food, Casual dining, Fine dining, Buffet
     - iii. MP: Fast Food Preference
       - Attributes: BiteBox Burgers, GrillGusto, SnackSprint, ZippyZest, WrapRapid
     - iv. SP: Desired Price Range
       - Attributes: cheap, normal, expensive
     - v. MP: Dietary Preferences
       - Attributes: Vegetarian, Vegan, Gluten-Free, Dairy-Free, Halal, Kosher, Nut Allergies, Seafood Allergies
     - vi. SP: Preferred Payment Method
       - Attributes: Cash, Card
   - (b) Gas Station
     - i. MP: Preferred Gas Station
       - Attributes: PetroLux, FuelNexa, GasGlo, ZephyrFuel, AeroPump

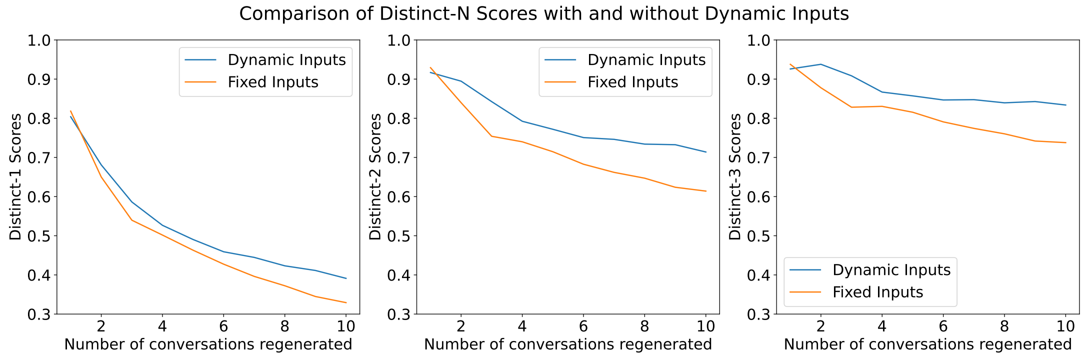


Figure 4: The figure shows the diversity evaluation (Distinct-1, Distinct-2, Distinct-3) (y-axis) with dynamic and fixed inputs. The scores were calculated and then averaged for four different user preferences, with each preference's conversations being regenerated 1 to 10 times (x-axis).

ii. SP: Willingness to Pay Extra for Green Fuel
- Attributes: Yes, No (cheapest preferred)

iii. SP: Price Sensitivity for Fuel
- Attributes: Always cheapest, Rather cheapest, Price is irrelevant

(c) Charging Station (in public)

i. MP: Preferred Charging Network
- Attributes: ChargeSwift, EcoPulse Energy, VoltRise Charging, AmpFlow Solutions, ZapGrid Power

ii. SP: Preferred type of Charging while traveling
- Attributes: AC, DC, HPC

iii. SP: Preferred type of Charging when being at everyday points (e.g., work, grocery, restaurant)
- Attributes: AC, DC, HPC

iv. MP: Charging Station Amenities
- Attributes: On-site amenities (Restaurant/cafes), Wi-Fi availability, Seating area, Restroom facilities

(d) Grocery Shopping

i. MP: Preferred Supermarket Chains
- Attributes: MarketMingle, FreshFare Hub, Green-Groove Stores, BasketBounty Markets, PantryPulse Retail

ii. SP: Preference for Local Markets/Farms or Supermarket
- Attributes: Local Markets/Farms, Supermarket

2. Navigation and Routing

(a) Routing

i. MP: Avoidance of Specific Road Types
- Attributes: Highways, Toll roads, Unpaved roads

ii. SP: Priority for Shortest Time or Shortest Distance
- Attributes: Shortest Time, Shortest Distance

iii. SP: Tolerance for Traffic
- Attributes: Low, Medium, High

(b) Traffic and Conditions

i. SP: Traffic Information Source Preferences
- Attributes: In-car system, NavFlow Updates, Route-Watch Alerts, TrafficTrendz Insights

ii. SP: Willingness to Take Longer Route to Avoid Traffic
- Attributes: Yes, No (traffic tolerated for fastest route)

(c) Parking

i. SP: Preferred Parking Type
- Attributes: On-street, Off-street, Parking-house

ii. SP: Price Sensitivity for Paid Parking
- Attributes: Always considers price first, Sometimes considers price, Never considers price

iii. SP: Distance Willing to Walk from Parking to Destination
- Attributes: less than 5 min (accepting possible higher cost), less than 10 min (accepting possible higher cost), not relevant (closest with low cost)

iv. SP: Preference for Covered Parking
- Attributes: Yes, Indifferent to Covered Parking

v. SP: Need for Handicapped Accessible Parking
- Attributes: Yes

vi. SP: Preference for Parking with Security
- Attributes: Yes, Indifferent to Parking Security

3. Vehicle Settings and Comfort

(a) Climate Control

i. SP: Preferred Temperature
- Attributes: 18 degree Celsius, 19 degree Celsius, 20 degree Celsius, 21 degree Celsius, 22 degree Celsius, 23 degree Celsius, 24 degree Celsius, 25 degree Celsius

ii. SP: Fan Speed Preferences
- Attributes: Low, Medium, High

iii. SP: Airflow Direction Preferences
- Attributes: Face, Feet, Centric, Combined

iv. SP: Seat Heating Preferences
- Attributes: Low, Medium, High

(b) Lighting and Ambience

i. SP: Interior Lighting Brightness Preferences
- Attributes: Low, Medium, High

ii. SP: Interior Lighting Ambient Preferences
- Attributes: Warm, Cool

iii. MP: Interior Lightning Color Preferences
- Attributes: Red, Blue, Green, Yellow, White, Pink

4. Entertainment and Media

(a) Music

i. MP: Favorite Genres
- Attributes: Pop, Rock, Jazz, Classical, Country, Rap

ii. MP: Favorite Artists/Bands
- Attributes: Max Jettison (Pop), Melody Raven (Pop), Melvin Dunes (Jazz), Ludwig van Beatgroove (Classical), Wolfgang Amadeus Harmonix (Classical), Taylor Winds (Country/Pop), Ed Sherwood (Pop/Folk), TwoPacks (Rap)

iii. MP: Favorite Songs
- Attributes: Envision by Jon Lemon (Rock), Dreamer's Canvas by Lenny Visionary (Folk), Jenny's Dance by Max Rythmo (Disco), Clasp My Soul by The Harmonic Five (Soul), Echoes of the Heart by Adeena (R&B), Asphalt Anthems by Gritty Lyricist (Rap), Cosmic Verses by Nebula Rhymes (Hip-Hop/Rap)

iv. SP: Preferred Music Streaming Service
- Attributes: SonicStream, MelodyMingle, TuneTorrent, HarmonyHive, RhythmRipple

(b) Radio and Podcasts

i. SP: Preferred Radio Station
- Attributes: EchoWave FM, RhythmRise Radio, SonicSphere 101.5, VibeVault 88.3, HarmonyHaven 94.7

ii. MP: Favorite Podcast Genres
- Attributes: News, Technology, Entertainment, Health, Science

iii. MP: Favorite Podcast Shows
- Attributes: GlobalGlimpse News, ComedyCraze, SciEnceSync, FantasyFrontier, WellnessWave

iv. SP: General News Source
- Attributes: NewsNexus, WorldPulse, CurrentConnect, ReportRealm, InfoInsight

E Methodology: Preference Extraction

We define the LLM function for extracting user preferences as follows:

```
"type": "function",
"function": {
    "name": "extract_user_preference",
    "description": "A function that extracts
    ↪   personal preferences of the user ...",
    "parameters": "<nested parameter schema
    ↪   representing the hierarchical
    ↪   categories>"}
```

The parameter schema, defined using Pydantic, includes categories and their hierarchy. Below is a representative subset for the main category `Points of Interest`, sub-category `Restaurant`, and detail-category `Favourite Cuisine`:

```
class PreferencesFunctionOutput(BaseModel):
    points_of_interest:
    ↪   Optional[PointsOfInterest] =
    ↪   Field(default=None,
        description="The user's preferences in
        ↪   the category 'Points of
        ↪   Interest'.",)
    navigation_and_routing:
    ↪   Optional[NavAndRouting] = Field(...)
    ...

class PointsOfInterest(BaseModel):
    no_or_other_preference: ...
    restaurant: Optional[Restaurant] =
    ↪   Field(defualt=None, description="...")
    ...

class Restaurant(BaseModel):
    no_or_other_preference: ...
    favourite_cuisine:
    ↪   Optional[List[OutputFormat]] =
    ↪   Field(default=[], description="...",
    ↪   examples=["Italian", "Chinese", ...])
    ...

class OutputFormat(BaseModel):
    user_sentence_preference_revealed:
    ↪   Optional[str] = Field(default=None,
    ↪   description="user sentence where the
    ↪   user revealed the preference.")
    user_preference: Optional[str] =
    ↪   Field(default=None, description="The
    ↪   preference of the user.")
```

Each category is represented as a parameter with a type, default value, description, and optional examples. The nested schema represents the relationship of the categories. As every parameter is `Optional`, the LLM is not forced to extract a preference for every parameter within that category. We found that including the parameter `no_or_other_preference` within the sub- and detail categories reduces over-extraction, as the LLM must actively decide not to place a preference there if it intends to extract one. Through the `Output Format`, we can see, that the LLM should not only extract the preference itself, but also the sentence where the user revealed the preference.

F Additional Experiment Results

F.1 Confusion Matrices of Preference Extraction

Figure 5 shows the multi-label confusion matrix on the detail category level for the In-Schema experiment (refer to Section 5.1).

The strong diagonal in the confusion matrix indicates that the extraction process reliably adheres to the category schema. Most incorrect extractions occur in semantically related categories. After manual analysis, we found that the increased misclassifications in the detail category ’avoidance of specific roadtypes’, ’shortest time or distance’, and ’tolerance for traffic’ are mostly due to the dataset. During dataset generation, an extra preference is occasionally included in the user utterances within these categories, as in-car conversations often evolve toward these topics naturally.

Figure 6 shows the multi-label confusion matrix on the subcategory level for the Out-of-Schema experiment (refer to Section 5.1). As the category of the ground-truth preference is excluded in the schema for this experiment, we expect the system to perform no extraction. We see that we have few incorrect extractions when excluding semantically distinct categories such as ’Climate Control’ (0 incorrect extraction), but significantly more if there is still a closely related category like in ’Music’ and ’Radio and Podcast’. This indicates that the definition of clear and semantically distinct categories is key to a reliable category-bound extraction.

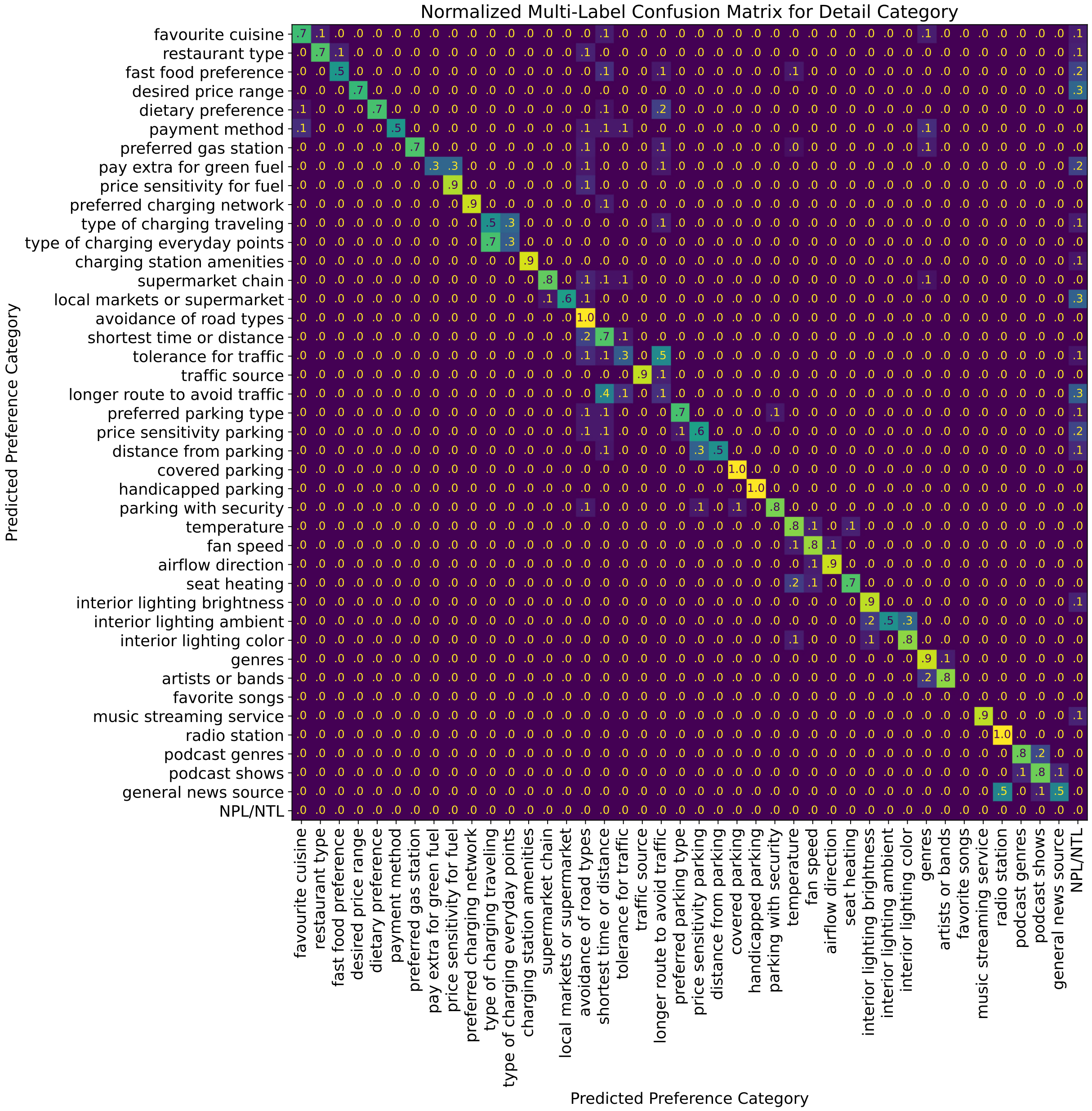


Figure 5: Multi-Label confusion matrix (Heydarian et al., 2022), normalized across the rows, on the detail category level for the In-Schema experiments (refer to Section 5.1). The last row represents data points with no true label (NTL), while the last column represents data points with no predicted label (NPL).

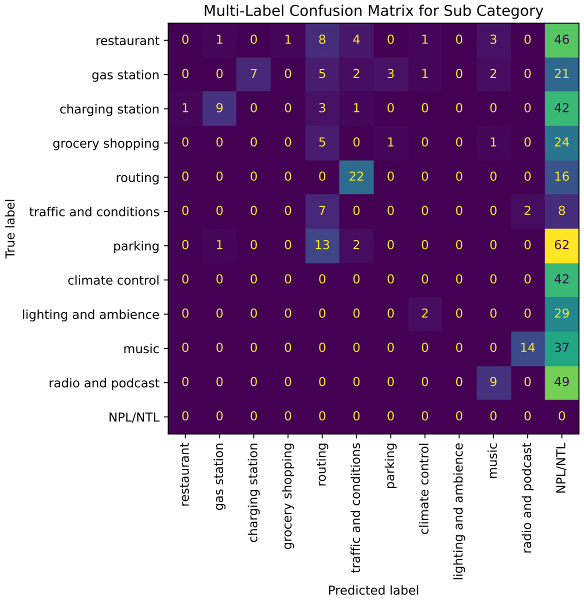


Figure 6: Multi-label confusion matrix (Heydarian et al., 2022) on the subcategory level for the Out-of-Schema experiment (refer to Section 5.1). The last row represents data points with no true label (NTL), while the last column represents data points with no predicted label (NPL). In this experiment, it is expected to have no predicted label for every data point.