“Where Does This Strange Smell Come from?”: Enabling Conversational Interfaces for Artificial Olfaction

Xueyi Zhou[1], **Qi Lu**[2*] **and Dong-Kyu Chae**[1*]
[1]Department of Computer Science, Hanyang University, Republic of Korea
[2]The Future Laboratory, Tsinghua University, China
{hokyeejau,dongkyu}@hanyang.ac.kr, luq@mail.tsinghua.edu.cn

Abstract

Existing Artificial Olfaction (AO) primarily serves two tasks: Odor Classification (OC) and Odor Source Localization (OSL). Both tasks *w.r.t.* indoor event detection scenarios are studied either using a single electronic nose (e-nose) mounted on the ceiling or mobile robot(s) equipped with e-noses. However, they are not compatible with smart home scenarios due to diverse obstacles (*e.g.*, chairs and tables) and the need for natural interaction. In this paper, we explore the feasibility and usability of a **C**onversational **I**nterfaces for **A**rtificial **O**lfaction (**CIAO**) system using Large Language Models (LLMs) in Smart Home. We made the first olfaction-oriented corpus for LLM evaluation, as well as an olfaction dataset via a self-developed olfactory sensory network. We train the dedicated models for OSL and OC using the dataset and integrate them into a tool within the MCP (Model Context Protocol) server. Five commercial LLMs are used as MCP clients for experiments and validation. Our experimental results indicate that our CIAO system is technically feasible and applicable. Besides, we observe that ChatGPT-4o relatively outperforms in terms of both answer quality and overall LLM usability in pervasive IoT scenarios. Qwen-Plus, in contrast, appears to be a promising solution for robot-compatible applications. To our knowledge, this work is the first effort to bring forward conversational interfaces for AO, enabling multi-turn conversations with contexts beyond one-off question answering. Our codes and partial corpus are available at https://github.com/HokyeeJau/CIAO.

1 Introduction

Existing Artificial Olfaction (AO) primarily serves two tasks: Odor Classification (OC) and Odor Source Localization (OSL). OC refers to the process of identifying and categorizing different odors, while OSL is the task of determining the location of a chemical source of interest in the surrounding environment. Both provide recognition results by analyzing the plume released by the source (Chen and Huang, 2019; Jing et al., 2021; Wang et al., 2022). Facilitating both tasks can significantly subserve various aspects of human life, with extensive applications in safety and hazard detection, health, and environmental monitoring, just to name a few.

Researchers to this end commonly develop electronic noses (e-noses) to mimic the animal sense of smell. An e-nose system generally consists of an array of gas sensors and a pattern recognition system (usually based on machine learning or neural networks); they together reveal surrounding odor profiles for odor-sensitive applications (Lu et al., 2022; Seesaard et al., 2022; Wei et al., 2023; Cheng et al., 2021). Indoor applications of e-noses, especially in Smart Home, encompass a range of use cases such as lifelogging (Kobayashi et al., 2011; Dang et al., 2018; Hirano et al., 2015, 2013; Khaloo et al., 2019) and air quality monitoring (Zampolli et al., 2004; Fang et al., 2016; Amores et al., 2015). When deployed, researchers and vendors typically create different software to vividly and systematically present the recognition results, such as mobile phone applications (Arroyo et al., 2020; Suárez et al., 2018; Tiele et al., 2020; Emmenegger et al., 2024) and web-based dashboards (Deshmukh et al., 2015; Esfahani et al., 2020). Apparently, the presentation of interpreted odor profiles is largely limited to simple data analysis and visualization. Existing systems rarely provide a natural interaction experience or a brief introduction to relevant olfactory knowledge.

Recent studies primarily apply e-noses in robots, particularly on mobile robots and unmanned aerial vehicles (UAVs). These platforms perceive odor stimuli features as spatial gradients in the environment, which enables reactive planning and navigation (Hayes et al., 2002; Jabeen et al., 2023; Nana-

---

[*] Corresponding authors.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 10719–10745
November 4-9, 2025 ©2025 Association for Computational Linguistics

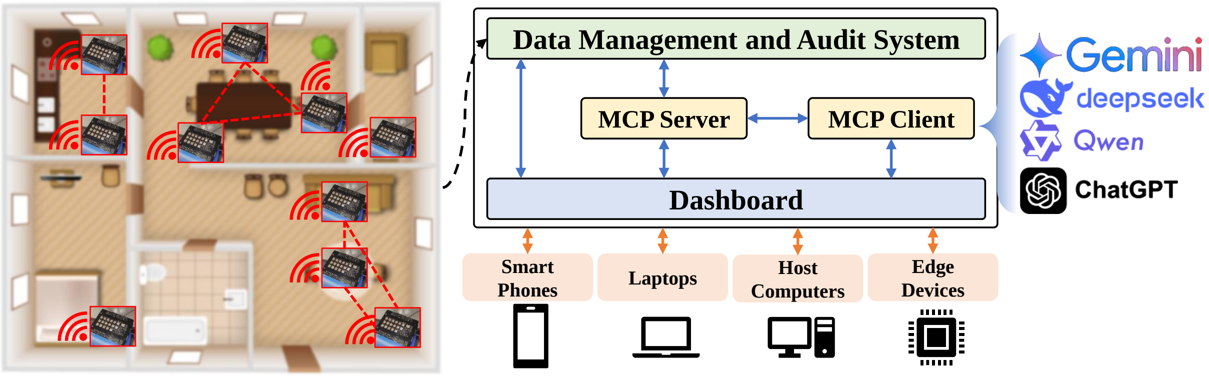


Figure 1: Overview of our scenario. We assume that the e-nose components are installed as functional components in home decor or household appliances. Powered by a well-developed IoT system and LLM-based agents, the smart housekeeper system can cooperate them to provide a strong conversational system for multi-end users.

vati et al., 2024; Chen and Huang, 2019; Jing et al., 2021). Robots or UAVs can also collaborate to trace odor sources (Wang et al., 2022; Soares et al., 2015; Fu et al., 2019; Jabeen et al., 2024). However, in smart home scenarios, diverse obstacles (*e.g.*, chairs, tables) and confined space challenge those robots to detect, plan, and move. Ceiling e-noses, additionally, are unable to accurately identify the locations of odor sources. A large-scale yet multi-end portable detection system offers a more compatible solution for smart home environments.

The studies on LLM (Large Language Model) Agents in Smart Homes recently emerged due to the rising tool use capabilities of LLMs. We have witnessed several promising solutions such as SAGE (Rivkin et al., 2024), HomeAssistant (Giudici et al., 2025), and GreenIFTTT (Giudici et al., 2024). Those studies focus on device controllability, LLM performance, and usability, using computational and statistical methods. However, none of them provisions a study *w.r.t.* olfactory event detection and a comprehensive corpus for systematically evaluating their methods.

To address the abovementioned concerns, we herein take initial steps towards a Model Context Protocol (MCP)-regulated **C**onversational **I**nterfaces for **A**rtificial **O**lfaction (**CIAO**) system for detecting indoor olfactory events in Smart Home. Figure 1 illustrates an overview of our scenario, where a multi-end olfactory sensory network is built by multiple e-noses, which are equidistantly distributed within a delimited area. To train dedicated models for the OC and OSL tasks, we alternatively use and localize different samples at different positions to collect the olfaction dataset using our self-developed e-noses. Each sensory (time series) data point inside is assigned with two labels: the odor type and the absolute position of the odor source. The trained OC and OSL models are integrated into a data API provided by the IoT system. A unified tool deployed on an MCP server is designed to invoke that data API to collect data from relevant devices and provide the majority-voted odor type and its position. Five prevalent commercial LLM models are selected according to their different knowledge backgrounds (training corpora) and used as MCP clients for validation and investigation.

As a result, a self-developed MCP-regulated CIAO system is thereby constructed. We further construct the first olfaction-oriented corpus by humans and LLMs to evaluate the LLM usability. This corpus is devised for not only smart home scenarios, but also the olfactory embodied intelligence and other multimodal tasks. We assess the system usability by collecting the system latency, and evaluate the tool use capabilities by analyzing the statistics of tool calls at multiple given situations. An expert evaluation involving 15 AI researchers from three countries is furthermore carried out to evaluate the answer quality and the LLM usability. The emotional feedback capabilities and multi-round conversations of LLMs are additionally discussed by case studies. We observe that ChatGPT-4o empirically outperforms *w.r.t.* both answer quality and overall LLM usability in IoT so far. Qwen-Plus, conversely, is a promising robot-compatible solution for olfactory embodied intelligence.

2 Related Works

2.1 Artificial Olfaction in Smart Home

An increasing number of e-nose-enabled smart home applications aim to optimize olfactory experiences within domestic environments. Lifelogging applications powered by e-noses, commonly used in home environments, offer a promising approach to capturing daily contextual data without exposing sensitive personal details (such as facial features) (Kobayashi et al., 2011; Dang et al., 2018), varying from drink sample classification (Hirano et al., 2015), cooking activity identification (Hirano et al., 2013) to ambient monitoring of the frying cooking (Khaloo et al., 2019). Another kind of application is air quality monitoring (Zampolli et al., 2004). Different odor sources were individually discussed in many studies (*e.g.*, both smoking and spraying (Fang et al., 2016), and bad odors of trash bins (Amores et al., 2015)). However, that for productionizing more than data recognition has been individually done, but they are limited to tedious data visualization on mobile phone applications (Arroyo et al., 2020; Suárez et al., 2018; Tiele et al., 2020; Emmenegger et al., 2024) or web-based dashboards (Deshmukh et al., 2015; Esfahani et al., 2020); a novel system enabling language-capable olfaction can refine user experiences.

2.2 Odor Source Localization

Mobile robots / unmanned aerial vehicles (UAVs) are widely used platforms for deploying e-noses to sense and localize. Installed on mobile robots, the odor stimuli feature present as odor gradient and are coupled to robot action (Hayes et al., 2002; Jabeen et al., 2023; Chen et al., 2023; Wang and Pang, 2022; Zhu et al., 2020), thereby planning the path to the odor source (Nanavati et al., 2024; Chen and Huang, 2019; Jing et al., 2021). The cooperation of multiple robots (Wang et al., 2022; Soares et al., 2015) or UAVs (Fu et al., 2019; Jabeen et al., 2024) is delicately designed to derive the location by ensemble intelligence; they can work together in a rather large scene to trace the positions of odor sources. However, they are not compatible with smart home scenarios due to diverse obstacles (*e.g.*, chairs, tables, or potted plants). Designing and enabling cooperation of e-noses alike digital decors (Nabil and Kirk, 2021; Siio et al., 2003) in IoT systems may address this challenge.

2.3 Tool-assisted Agents for Smart Home

Tool-assisted LLM Agents are meeting people's needs of re-using substantial tools (external knowledge) to fulfill their instructions (Guo et al., 2024; Qin et al., 2023; Yuan et al., 2024; Wu et al., 2024b). In terms of smart home, several trained/fine-tuned tool-assisted LLM Agents are discussed in (Rivkin et al., 2024; Yonekura et al., 2024; Giudici et al., 2025, 2024). A recent work (Zhou et al., 2025) proposed a two-step reasoning paradigm for enabling LLM-based question answering for odor classification, in which the model first determines the appropriate API to invoke and subsequently generates the answer conditioned on the retrieved data. However, this work did not support conversation with contexts; it only provides a one-off question answering. Among these studies, the researchers evaluated their frameworks' usability without evaluating them on a comprehensive corpus. Moreover, the most recent **Model Context Protocol (MCP)** (Hou et al., 2025), which standardizes how applications provide context to LLMs, is an inspiring solution for enabling conversational IoT systems to use data interfaces of connectable devices without training or fine-tuning an LLM agent, thereby creating new opportunities for conversational IoT systems.

2.4 Existing AO Products

Conventional commercial e-nose systems overlook the the incorporation of e-nose systems and LLM-based Agents. Specifically, Smell Inspector[1] released by SmartNanotubes Technologies is a product enabling gas identification and data visualization on Windows and Linux Systems. AI-driven eNose released by Stratuscent at CES 2020[2] enables gas identification and simple notification on an application. SCOUT2 modular electronic nose released by Volatile AI[3] promotes sensor data collection and visualization. Electronic Nose Analytical Service (SKU: EN001) released by MSE Supplies LLC[4] enables sensor data collection only, without considering user-friendly interaction. Electronic Nose Data Acquisition System and Electronic Nose Instrument released by Suzhou Huiwen Nanotechnology Co., Ltd[5] also work for sensor data collection only.

3 Conversational Interfaces for Artificial Olfaction

We designed a feasible solution to bridge the IoT system, MCP-regulated services and the fron-

---

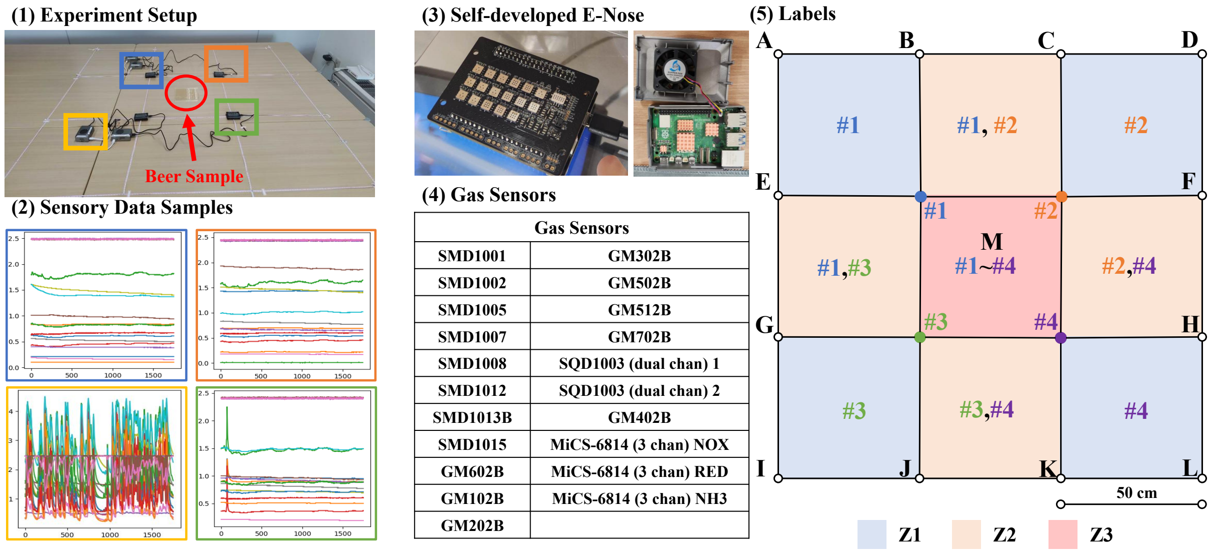
[1] https://www.smart-nanotubes.com/

[2] https://thespoon.tech/ces-2020-stratuscents-digital-nose-can-smell-when-crops-are-ripe-or-food-is-burning/

[3] https://volatile.ai/enose

[4] https://www.msesupplies.com/products/electronic-nose-analytical-service

[5] https://www.huiwen-sensor.com/



Gas Sensors	
SMD1001	GM302B
SMD1002	GM502B
SMD1005	GM512B
SMD1007	GM702B
SMD1008	SQD1003 (dual chan) 1
SMD1012	SQD1003 (dual chan) 2
SMD1013B	GM402B
SMD1015	MiCS-6814 (3 chan) NOX
GM602B	MiCS-6814 (3 chan) RED
GM102B	MiCS-6814 (3 chan) NH3
GM202B	

Figure 2: Illustrations of our olfactory sensor dataset collection. (1) Experimental setup. (2) Sensory (multivariate time series) data samples. (3) Images of our e-nose. (4) Gas sensors featured on the e-noses. (5) Labels (# 1, # 2, # 3 and # 4) of four e-noses and the odor-emitting areas (Z1s, Z2s and Z3) determined by the neighboring e-noses.

tend (dashboard) system. Figure 1 illustrates its overview. The IoT system acts as a conventional IoT data center, which utilizes APIs to collect and manage the data from edge devices and provide accessible data APIs. We regard that the CIAO system is equipped with a multi-end sensory network as shown in Figure 1. Accordingly, we built a physical experimental scene using 4 copies of our self-developed e-noses to simulate the IoT scenario (shown in Figure 2(1)).

3.1 Artificial Olfaction

To train dedicated models for the *odor classification* (OC) and *odor source localization* (OSL) tasks, we first construct an olfactory sensor dataset using three Korean beers, *i.e.*, Cass, Kelly, and Terra[6] (Zhou et al., 2025). The dataset contains four sub-datasets, each collected from one e-nose copy. The samples of three Korean beers are alternatively placed at different positions to collect sensory (multivariate time series) data as shown in Figure 2(2). We assign each time series from each device with two ground-truth labels: the odor type (Cass, Kelly or Terra) and the absolute position of odor source (ranging from A to M in Figure 2(5)). More details on our odor data collection can be found in Appendix A.

We regard the home AO system as a multi-end sensory network powered by portable e-noses, as shown in Figure 1, whereby OSL and OC are redefined as follows:

Multi-end OSL: Multi-end OSL can be formally defined as an ensemble of multiple binary classifiers, each trained to clarify if there is an odor surrounding (*i.e.*, if the device is activated).

Multi-end OC: Multi-end OC can be formally defined as an ensemble of multiple multi-class classifiers, each trained to indicate the odor profile.

Therefore, each task of OC and OSL on each e-nose is fundamentally *multivariate time series classification* (MTSC) (more details can be found in Appendix C). After extensive experiments using multiple models, we observed that MLSTM-FCN (Karim et al., 2019) excels in characterizing the odor profile while DenseNet (Huang et al., 2017) is sensitive to the unnatural plume in the air within a certain distance. They altogether achieve 100% accuracy on both OC and OSL, and are selected and trained using the collected beer olfactory sensor dataset.

In order to make a consensus of the four e-noses, our system employs a heuristic majority-voting mechanism to finally determine the odor type and its position of the cluster. After its decision, the data API can thus integrate raw data acquisition and the computational methods to provide the odor profiles. More details related to the problem formulation of OC and OSL in terms of MTSC, our choice of MLSTM-FCN and DenseNet for the two tasks, and the majority-voting mechanism, can be found in Appendix C.

---

[6] Due to laboratory safety regulations, we are allowed to test beers only.

Type	Emotion	Template
Instruction	N	Generate 200 sentences which are semantically similar to "<SENTENCE>". Itemize them all once without double quote.
Instruction	N	Generate 200 <Q-TYPE> questions which are semantically similar to "<SENTENCE>". Itemize them all once without double quote.
Instruction	Y	Generate 200 sentences which are semantically similar to "<SENTENCE>" expressed in a <SENTIMENT> tone. Itemize them all.
Instruction	Y	Generate 200 <Q-Type> questions which are semantically similar to "<SENTENCE>" expressed in a <SENTIMENT> tone. Itemize them all.

Figure 3: Four instructions for generating questions.

3.2 MCP-regulated CIAO

The MCP service[7] consists of MCP server(s) and MCP client(s). The server is designed to provide tools and their descriptions, each of which presumptively accesses a specific group of the IoT system APIs and returns data conditioned on given arguments. The IoT system herein sampled three tools for evaluating the CIAO system, varying from gathering odor source type and its location (AO API), deriving the Korean introduction of three beers from Namuwiki[8] (Beer Intro API), to obtaining the hardware details of the e-noses (Hardware Intro API).

Each MCP client maintains end-to-end connections with the server and employs a designated LLM-based agent to interpret and answer the questions by deriving data from appropriate tools. Five outstanding commercial LLM models (DeepSeek-V3 (DeepSeek-AI, 2024), Gemini-2.0-Flash (Team et al., 2023), Qwen-plus (Bai et al., 2023), ChatGPT-4.1 and ChatGPT-4.0[9]) are selected as agent candidates. Each agent is prompted by "*You are a smart housekeeper. You can automatically invoke the API tools for answering the questions*". Two more LLMs (Qwen-Turbo (Bai et al., 2023) and Gemini-2.5-Flash (Team et al., 2023)) are utilized to evaluate the tool use capabilities. In each context, the role releasing the question is labeled as "user"; the prior answers are released by "assistant". We set the temperature of all agents to 0 for safer, more predictable, and factual answers and use LangChain 0.3.20 and LangGraph 0.4.1[10] to build Agents. Our experiments were mainly conducted on an HPC (DELL DSS8440) equipped with 8 NVIDIA A100 GPUs at the Biospecimen-Multiomics Digital Bioanalysis Core Facility of Hanyang University.

4 Evaluation of CIAO

This section aims to evaluate the capabilities of the five widely-used commercial LLMs (introduced in Section 3.2) employed in our CIAO system. We first introduce our olfaction-oriented corpus to assess those LLMs in Section 4.1. We validate the technical feasibility of the MCP-regulated CIAO system by evaluating tool use capabilities of the LLMs in Section 4.2. The system latency using each LLM is also assessed to show its practical usability in Section 4.3. We additionally conduct an expert evaluation to assess the LLM usability in Section 4.4. A divergent discussion on model empathic capabilities and multi-round conversations is included in Section 4.5. The evaluation metrics are detailed in Appendix E.

4.1 Olfaction-oriented Question Corpus

We collect the olfaction-oriented corpus, including questions of different types and different emotions for evaluating LLM capabilities and usability.

(1) Following the definitions in Worley, 2015, we generate grammatically open-end and close-end questions (abbreviated as open questions and closed questions hereinafter) in terms of olfactory. Specifically, we consider that open questions include fill-in-the-blank questions and factual questions. Those are targeted at asking the odor types and their location without any explicit keywords. Closed questions are regarded as a superset of tag questions, choice questions, and Yes/No (Y/N) questions, which only require a yes/no answer or an option of given candidates. For each question type in each task, we manually provided a question as a template question (as shown in Figure 8 of the Appendix) and asked ChatGPT-4o-mini and DeepSeek-V3 to generate 50 semantically similar questions repeatedly. (2) We further prompt the LLMs to generate questions based on six *emotional* dimensions: anger, fear, sadness, enjoyment, disgust, and surprise (Ekman, 1992). The erroneous

---

[7] https://github.com/modelcontextprotocol/python-sdk
[8] https://en.namu.wiki/
[9] https://chat.openai.com/
[10] https://www.langchain.com/

questions were manually filtered out. More details can be found in Appendix B and Figure 3.

4.2 Tool Use Capabilities

We explore tool use capabilities and evidence the technical feasibility of our CIAO system by analyzing the number of tools being targeted. For each question type in OC and OSL tasks, as well as the factual type in other tasks, we randomly sample 50 general questions (non-sentimental) for LLMs to generate answers, during which we log the number of tool calls (as shown in Appendix G). Specifically, we collect the average numbers of three tool calls: **AO API** (used to derive odor type and its location), **Hardware Intro API** (for gathering the hardware information), and **Beer Intro API** (for obtaining the introduction of a specific beer). We then calculate the *Total Calls* and *Tool Use Accuracy* (abbreviated as *Use Accuracy* hereinafter) per question type. The *Use Accuracy* is defined as: (1) the average number of AO API calls while the question is related to OC, OSL or both of them; (2) that of Hardware Intro API calls while the question targets hardware information; (3) or the sum of those AO API and Beer Intro API while questioning the information of sensing gas odor.

We report the *Total Calls* and *Tool Use Accuracy* (as well as all the average numbers of API calls) in Tables 4 (OC), 5 (OSL), and 6 (other tasks) of the Appendix. In summary, ChatGPT-4.1 shows the lowest number of non-target tool calls (*#Total Calls−#Use Accuracy*), indicating its efficiency in tool use. In contrast, Qwen-Plus and DeepSeek-V3 sometimes repeatedly invoke tools while answering single-tool questions. This suggests that both LLMs may either struggle with understanding tool description or maintaining the prior interaction, or alternatively, adopt a conservative strategy by repeated data validation. All models, however, exhibit varying degrees of hallucination when answering factual questions without calling any tools. Specifically, the Gemini-2.0-Flash model shows pronounced hallucination issues even when answering Y/N questions on OC. Overall, ChatGPT-4o outperforms the others, achieving a tool targeting accuracy $\geq 90\%$. This is also evidenced by the comprehensive comparison of *Tool Use Accuracy* across each task, question type, emotion, and LLM in Table 7, 8, and 9 in the Appendix, where the LLMs are evaluated on 50 questions sampled from each type. Besides, the CIAO system is capable of leveraging LLMs as agents to invoke tools for answering tasks, further demonstrating the technical feasibility of the CIAO framework.

4.3 System Latency

We collected the average *Latency* of each LLM of 11 manually selected questions (as shown in Figure 4) used by CIAO under a 93 Mbps network environment in Seoul, South Korea. Specifically, Gemini-2.0-Flash achieved the fastest response with an average of 2.36 seconds per query (sec/query). ChatGPT-4.1 and ChatGPT-4o followed with 3.81 and 3.4 sec/query. Qwen-Plus required 7.02 sec, whereas DeepSeek-V3 exhibited the highest *Latency*, taking up to 17 sec. According to the Latency Observations provided by the developer in the OpenAI community[11], only the server of DeepSeek-V3 relatively presents low latency; the others yet remain within a latency range generally acceptable to users.

4.4 LLM Capacity

The answer quality and commercial LLM usability for the IoT-based CIAO is further discovered by expert evaluations.

Questionnaire. The questionnaire is designed to obtain the evaluations from experts *w.r.t.* the LLM usability in the scenario that a cup of Kelly beer is placed at position 1. We provided 11 manually selected questions for each LLM to generate their answers, as shown in Figure 4. Before evaluation, we provided a description of the questionnaire's purpose, outlined the guidelines, and requested only basic information such as expert locations and titles (student, post-doc, faculty/engineer). For each question-answer (QA) pair, experts were asked to evaluate and rank it based on five aspects: *Accuracy*, *Completeness* (*Faithfulness*), *Inclusion of Out-of-domain (OOD) Knowledge*, *Human-likeness*, and *Answer Quality*. The former three can be only evaluated on a binary scale (Yes/No), while the latter two are assessed using a five-point rating scale (Very Good, Good, Moderate, Bad, Very Bad). An additional evaluation example is provided for the experts as a reference. After they evaluated all the QA-pairs from a given LLM, the experts proceeded to assess the LLM's overall *Accuracy*, *Comprehensibility*,

---

[11] https://community.openai.com/t/performance-analysis-of-assistants-versus-chat-completion-chat-completion-seems-somewhat-faster-for-complete-message-generation-streaming-taken-into-account/628368

Task	Question Type	Question	True/Included
OC	Choice	Which one do you think you're smelling, Cass beer or Terra beer?	N
	Tag	The lovely aroma is from the Kelly beer, right?	Y
	Fill-in-the-blank	The thing you are sniffing is ?	-
	Factual	What's that fragrance in the room?	-
OSL	Choice	Are you detecting the smell more near position 1 or position 4?	Y
	Tag	The aroma is coming from position 4, do you smell it too?	N
	Fill-in-the-blank	The thing emitting the fragrance is at?	-
	Factual	Where is the source of that smell?	-
OC & OSL	Factual	What's emitting this smell, and where is it?	-
Gas	Factual	What aroma do you smell? Could you describe it shortly?	-
Hardware	Factual	Can you briefly introduce your device(s) for detecting gas?	-

Figure 4: Eleven manually designed questions in the designed questionnaire provided to LLMs. "Y" represents the question either directly confirm the correct label or includes it among the options, whereas "N" refers to those presenting wrong labels only. The questions exclude any explicit labels if True/Included column is empty.

Coherence, *Satisfaction*, and *Helpfulness*. These metrics are on a five-point rating scale.

Answer Quality Comparison. We invite 15 AI researchers from three different countries to complete the questionnaire and evaluate the answers to a specific group of questions. The invited researchers are studying different domains, varying from bioinformatics, robotics, urban informatics, to computer vision. We designed five questionnaires, each containing the answers from two LLMs. The answers from any two questionnaires may share at most one repeated LLM, such that we gather evaluations from two experts for each LLM within the same country. Each LLM is evaluated by 6 experts in total. The experts are blind to the identity of the LLMs they evaluated.

The average evaluation *w.r.t.* QA pairs are shown in Figure 5. We use *Accuracy*(↑), *Completeness*(↑), *Out-of-domain (OOD) Knowledge Inclusion*(↓), *Human-likeness*(↑) and *Quality Rating*(↑) to evaluate the answer quality. A more explicit explanation of those metrics for the experts is provided in Figure 11 of the Appendix. Overall, the ChatGPT-variants achieved top-three performance on *Accuracy*, *Completeness*, *Inclusion of OOD Knowledge* and *Quality* (ChatGPT-4.1: 1/1/3/3, ChatGPT-4o: 2/2/1/1). Specifically, ChatGPT-4.1, ChatGPT-4o and DeepSeek-V3 can accurately and faithfully answer both close-/open-end questions. ChatGPT-4o can comprehend and answer questions without introducing non-essential or peripheral knowledge. In terms of Human-Likeness, DeepSeek-V3, Qwen-Plus, and Gemini-2.0-Flash exhibit relatively more human-like conversational patterns as reflected in the expert rating of *Human-Likeness*. Based on aggregated ranking scores from all researchers, ChatGPT-4o (2/2/1/3/1) achieves the highest rankings across five metrics within the CIAO system. This observation indicates that the experts are inclined to believe that ChatGPT-4o provides better response quality.

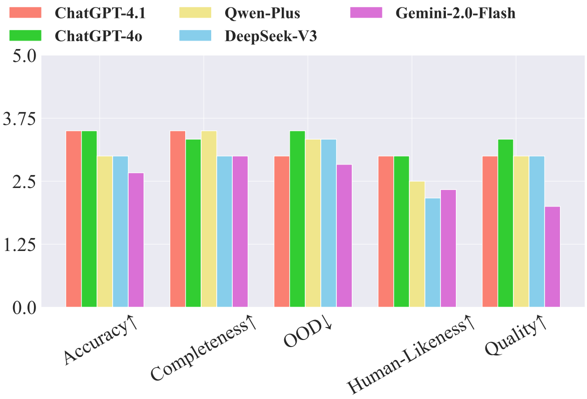


Figure 5: Average evaluation *w.r.t.* QA pairs. OOD indicates if out-of-domain knowledge is included. Quality indicates the quality rating of the answers.

LLM Usability. Figure 7 presents the result with regard to LLM usability and user experience (as detailed in Figure 11 of the Appendix) assessed by the experts. The metrics here include *Accuracy*(↑), *Comprehensibility*(↑), *Coherence*(↑), *Satisfaction*(↑) and *Helpfulness*(↑) (more details can be found in Appendix E). Overall, the ChatGPT variants demonstrate superior performance across the majority of evaluation metrics. They achieved top-three performance on *Accuracy*, *Comprehensibility*, *Coherence*, *Satisfaction*, and *Helpfulness* (ChatGPT-4.1: 1/1/3/1/2, ChatGPT-4o: 1/2/1/1/1). These results indicate that the ChatGPT variants are highly regarded by the experts, *w.r.t.* both LLM

(a) ChatGPT-4o

Role	Content
User	Is that the odor of the Cass beer?
Assistant	The odor detected is from Kelly beer, not Cass beer.
User	from position 3?
Assistant	The odor detected at position 3 is "Kelly beer".
User	How do you know that? By camera?
Assistant	I don't use a camera to detect odors. Instead, I rely on an electronic nose (e-nose) system installed in the home. This system uses gas sensors to detect and classify different odors. By accessing the data from these sensors, I can determine the type and position of odors in the house.
User	Can you introduce that beer shortly? I don’t know that.
Assistant	Kelly beer is an all-malt lager beer released by HiteJinro on April 4, 2023, with an alcohol content of 4.5%. It is made using a double maturation process at 7\u2103 and -1.5\u2103, using 100% premium malt from Denmark. This process aims to achieve both smoothness and refreshment. The beer is characterized by a relatively rich taste, smooth finish, and strong carbonation. It is noted for its strong malt and hop aroma, with a lighter finish compared to other all-malt beers. The beer is available in various sizes, including bottles, cans, and kegs, and is priced competitively in the market. Despite being a successor to the Max beer, it has not yet received the same level of acclaim among beer enthusiasts.
User	Great, and I am also curious about the e-nose. Can you introduce this?
Assistant	The electronic nose (e-nose) system installed in your home is equipped with a variety of gas sensors, including:\n\n- SMD1001, SMD1002, SMD1005, SMD1007, SMD1008, SMD1012, SMD1015\n- GM602B, GM102B, GM202B, GM302B, GM502B, GM512B, GM702B\n- SQD1003, MiCS-6814\n\nThese sensors are designed to detect different types of gases and odors.\n\nThe e-nose system operates with the following environmental conditions:\n\n- Beer temperature is maintained at 5\u00b0C.\n- Room temperature ranges between 10-16\u00b0C.\n- Humidity is kept at 20%.\n\nThe system samples data at a rate of 3Hz, and the data is collected using a Raspberry Pi 5. There are a total of 4 e-noses in the system, ensuring comprehensive odor detection and analysis.
User	Alright. Thank you.
Assistant	You're welcome! If you have any questions or need assistance, feel free to ask.

(b) Qwen-Plus

Role	Content
User	Is that the odor of the Cass beer?
Assistant	The currently detected beer, Kelly beer, is a premium all-malt lager beer … **(3,253 characters omitted).**
User	from position 3?
Assistant	The odor at position 3 seems to be unavailable or not detected by the electronic nose. However, I found that at position 1, Kelly beer's smell has been detected. Would you like more information on Kelly beer or check another position?
User	How do you know that? by camera?
Assistant	Not by camera. I can use the data from the electronic nose installed at home to determine the type and location of the odor. Would you like to know the specific information detected by the electronic nose? If so, I can provide you with relevant data.
User	Can you introduce that beer shortly? I don't know that yet.
Assistant	Of course! Could you please tell me which type of beer you would like to know about? The options are "kelly", "cass", or "terra".
User	hmmmmm, kelly please.
Assistant	Kelly is a premium all-malt lager beer … **(2,942 characters omitted).**
User	Great, and I am also curious about the e-nose. Can you introduce this?
Assistant	The e-nose system at home is composed of various gas sensors, which are listed as follows: SMD1001, SMD1002, SMD1005, SMD1007, SMD1008, SMD1012, SMD1015, GM602B, GM102B, GM202B, GM302B, GM502B, GM512B, GM702B, SQD1003, and MiCS-6814.\n\nThe beer temperature for testing is set at 5 degrees Celsius. The room temperature where the e-noses are placed ranges between 10 to 16 degrees Celsius with a humidity level of 20%.\n\nThe sampling rate of the device is 3Hz, meaning it takes three samples per second. The data collecting device used in this system is Raspberry Pi 5. There are four such e-noses installed at home for monitoring odors.
User	Alright, thank you.
Assistant	You're welcome! If you have any other questions or need further assistance, feel free to ask. Stay safe and take care!再见! 如果您有其他问题或需要进一步的帮助，请随时提问。保重!

Figure 6: Two Examples of Multi-round Conversations.

usability and user experiences. While Qwen-Plus also performs well (2/1/2/2/2), its evaluation rankings are slightly lower than those of the ChatGPT variants, particularly in terms of *Accuracy*, *Coherence*, and *Satisfaction*. In contrast, qualitative assessment suggests that DeepSeek-V3 and Gemini-2.0-Flash fail to adequately meet their expectations in terms of *Satisfaction* and *Helpfulness*. Notably, Gemini-2.0-Flash does not even achieve a moderate evaluation level in these aspects. Consequently, ChatGPT-4o outperformed other LLMs within our CIAO framework in general; it has been endorsed by experts for its usability and user experience.

4.5 Case Study

Model Empathy Across Emotions. Emotions expressed at home are typically more varied and spontaneous, making emotional cues in voice interactions within smart home environments richer and more diverse. To evaluate how human-likeness the LLMs are in demonstrating empathy, we collect six sentimental questions (as mentioned in Section 4.1 and further detailed in Appendix F) from our corpus to comprehensively present the LLM emotional feedback capabilities. The answers are listed in Figure 12 of the Appendix. Qwen-Plus’s replies to disgusted, angry, and scared voices resemble service-oriented human assistance, showing affective and empathetic intent. For example, its answer to the angry question includes “*And I understand that you dislike alcohol, shall I help you get rid of the beer or move it to another place?*” reflects partial emotional understanding and a willingness to provide physical help. In the case of the

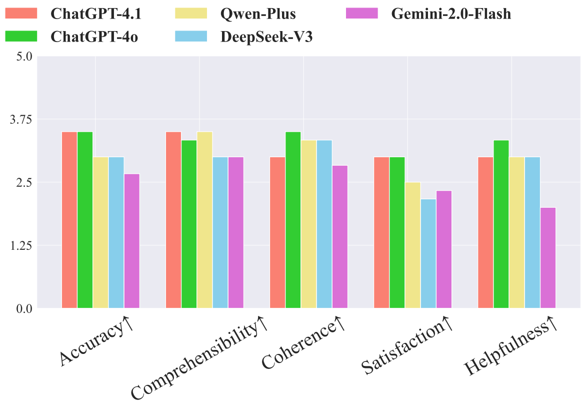


Figure 7: Average evaluation of Accuracy, Comprehensibility, Coherence, Satisfaction and Helpfulness *w.r.t.* LLM usability.

disgusted question, Qwen-Plus appears to demonstrate a capacity for handling ambiguity by using "*Maybe someone mixed up the beers?*". Similarly, Gemini-2.0-Flash and DeepSeek-V3 explicitly acknowledge possible misunderstanding concerning the odor type and its location. Across the answers to the scared question, ChatGPT-4.1, ChatGPT-4o and DeepSeek-V3 generate emotionally reassuring responses, typically using phrases like "*no worries*".

Overall, ChatGPT-4o consistently provides the shortest and most accurate responses among the evaluated LLMs. This conciseness could be advantageous in smart home applications where promptness and clarity are prioritized. Qwen-Plus, however, exhibits promising traits related to embodied intelligence, especially for applications involving bipedal robots designed to mimic human behaviors. More details on the QA pairs are in Appendix F.

Multi-round Conversations. We further conduct multi-round conversations to present CIAO capabilities using ChatGPT-4o and Qwen-Plus, which qualitatively show promising traits in Model Empathy Analysis. The conversations are shown in Figure 6. When the user asks a Y/N question (in blue) for OC, ChatGPT-4o can answer the question with the correct beer type only. Qwen-Plus, in contrast, adds more knowledge related to the beer without asking. Although ChatGPT-4o presents the shortest answer in the Y/N question for OSL (in orange), it shows less human-likeness compared to that of Qwen-Plus, which details the information of two positions and provides an additional question for help. A similar issue is observed while answering the hardware-related factual question (in green). The next round outlines the contextual understanding capabilities of two LLMs: ChatGPT-4o can retrieve the information of the priorly mentioned beer, but Qwen-Plus requires explicit follow-up prompts to derive the information (in purple). Ultimately, although we did not use any prompts in Chinese (in dark blue), Qwen-Plus provides a short Chinese farewell message. These observations suggest that ChatGPT-4o demonstrates stronger contextual coherence and provides better user experiences in multi-round conversations within the CIAO system.

5 Conclusions and Future Work

This study acts as the first effort to bring forward conversational interfaces for AO tasks including odor classification and localization, enabling multi-turn conversation with contexts beyond a one-off question answering. We assume the target is for olfactory event detection in a Smart Home. We assume the target is for olfactory event detection in a Smart Home. We constructed the olfactory sensor dataset and olfaction-related question corpus for realizing and evaluating our MCP-regulated CIAO system. The results of our extensive experiments indicate that our CIAO system is technically feasible in smart home scenarios when supported by advanced LLM-based agents. Among the evaluated models, the ChatGPT series, particularly ChatGPT-4o, demonstrates superior performance in terms of tool use, contextual understanding, and answer quality. These findings highlight the potential of ChatGPT-4o in the tool-based LLM Agent systems. Qwen-Plus, however, exhibits promising traits, especially for applications involving bipedal robots designed to mimic human behaviors.

In future work, we will build an olfaction-specific LLM for our CIAO system, enable more visual aids such as visual reality, customize prompts for different answering styles, and enhance LLM capability on handling uncertain recognition.

Limitations

As mentioned, our system development and evaluation have been conducted using beer odor samples, due to safety constraints. However, we believe that the overall design can be easily adapted to various practical domains by incorporating other odors relevant to the specific applications of researchers utilizing our work. More odor data emitted from various objects in diverse scenarios will directly enhance the practical applicability of our CIAO system. We will actively seek access to sites and obtain necessary permissions to gather more data that meets the olfactory needs of different application contexts. To validate the robustness and generalizability of our CIAO system, broader experiments across different applications are expected, such as in hazardous gas detection for industry or regulatory agencies and health care.

In addition, the current CIAO system utilizes commercial LLMs to answer the questions without incorporating olfaction-/task-related expert knowledge. This lack of domain adaptation may reduce the comprehensibility of both questions and answers, ultimately diminishing the answer quality and the perceived user experience. A domain-specific, fine-tuned LLM is consequently expected to be essential for the near future of CIAO.

We open-source a GitHub repository to share partial properties. Specifically, we released a partial corpus, which excludes some sensitive (e.g., hardware-related) questions and knowledge. We additionally develop and provide a demonstrative system using open-source LLMs to show the overall pipeline of CIAO (not the commercial one) on GitHub. We will further report the results of DeepSeek-R1-Distill-Llama-70B, Qwen-72B, and Gemma-3 once we have access to sufficient computational resources. According to the commercial restrictions, we are not yet allowed to publish the beer olfactory sensor dataset; however, we have provided an analysis report on the sensor dataset, including but not limited to the extension of Appendix A & C, the significant sensor analysis, PCA analysis of the sensor data, and primary ingredient-sensor mapping.

Acknowledgement

This work was partially supported by the Foundation of National Key Laboratory of Human Factors Engineering (Grant No. HFNKL2023J08), the National Natural Science Foundation of China (Grant No. 62441219), the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT)(RS-2025-24535423), and the Institute of Information & Communications Technology Planning & Evaluation (IITP) grants funded by the Korean government(MSIT) under Projects RS-2025-25422680 (Metacognitive AGI Framework and its Applications) and RS-2020-II201373 (Artificial Intelligence Graduate School Program, Hanyang University).

References

Judith Amores, Pattie Maes, and Joe Paradiso. 2015. Bin-ary: detecting the state of organic trash to prevent insalubrity. In *Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers*, pages 313–316.

Patricia Arroyo, Félix Meléndez, José Ignacio Suárez, José Luis Herrero, Sergio Rodríguez, and Jesús Lozano. 2020. Electronic nose with digital gas sensors connected via bluetooth to a smartphone for air quality measurements. *Sensors*, 20(3):786.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, and 29 others. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Kinjal Basu, Ibrahim Abdelaziz, Subhajit Chaudhury, Soham Dan, Maxwell Crouse, Asim Munawar, Sadhana Kumaravel, Vinod Muthusamy, Pavan Kapanipathi, and Luis A Lastras. 2024. Api-blend: A comprehensive corpora for training and benchmarking api llms. *arXiv preprint arXiv:2402.15491*.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, and 1 others. 2024. A survey on evaluation of large language models. *ACM transactions on intelligent systems and technology*, 15(3):1–45.

Xin-xing Chen and Jian Huang. 2019. Odor source localization algorithms on mobile robots: A review and future outlook. *Robotics and Autonomous Systems*, 112:123–136.

Xinxing Chen, Bo Yang, Jian Huang, Yuquan Leng, and Chenglong Fu. 2023. A reinforcement learning fuzzy system for continuous control in robotic odor plume tracking. *Robotica*, 41(3):1039–1054.

Lu Cheng, Qing-Hao Meng, Achim J Lilienthal, and Pei-Feng Qi. 2021. Development of compact electronic noses: A review. *Measurement Science and Technology*, 32(6):062002.

Minje Choi, Jiaxin Pei, Sagar Kumar, Chang Shu, and David Jurgens. 2023. Do llms understand social knowledge? evaluating the sociability of large language models with socket benchmark. *arXiv preprint arXiv:2305.14938*.

Chi Tai Dang, Andreas Seiderer, and Elisabeth André. 2018. Theodor: A Step Towards Smart Home Applications with Electronic Noses. page 11.

DeepSeek-AI. 2024. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Sharvari Deshmukh, Rajib Bandyopadhyay, Nabarun Bhattacharyya, RA Pandey, and Arun Jana. 2015. Application of electronic nose for industrial odors and gaseous emissions measurement and monitoring–an overview. *Talanta*, 144:329–340.

Paul Ekman. 1992. An argument for basic emotions. *Cognition & emotion*, 6(3-4):169–200.

Silvio Emmenegger, Raphael Kummer, Thomas Gisler, Christoph Zumbühl, Jürgen Wassner, and Armin Taghipour. 2024. Data acquisition system for ai-based odor classification at the edge. In *2024 IEEE International Symposium on Olfaction and Electronic Nose (ISOEN)*, pages 1–3. IEEE.

Siavash Esfahani, Akira Tiele, Samuel O Agbroko, and James A Covington. 2020. Development of a tuneable ndir optical electronic nose. *Sensors*, 20(23):6875.

Biyi Fang, Qiumin Xu, Taiwoo Park, and Mi Zhang. 2016. Airsense: an intelligent home-based sensing system for indoor air quality analytics. In *Proceedings of the 2016 ACM International joint conference on pervasive and ubiquitous computing*, pages 109–119.

Zhangjie Fu, Yiming Chen, Yongjie Ding, and Daojing He. 2019. Pollution source localization based on multi-uav cooperative communication. *Ieee Access*, 7:29304–29312.

Mathyas Giudici, Luca Padalino, Giovanni Paolino, Ilaria Paratici, Alexandru Ionut Pascu, and Franca Garzotto. 2024. Designing home automation routines using an llm-based chatbot. *Designs*, 8(3):43.

Mathyas Giudici, Alessandro Sironi, Ismaele Villa, Samuele Scherini, and Franca Garzotto. 2025. Generating homeassistant automations using an llm-based chatbot. *arXiv preprint arXiv:2505.02802*.

Zhen Guo, Adriana Meza Soria, Wei Sun, Yikang Shen, and Rameswar Panda. 2024. Api pack: A massive multi-programming language dataset for api call generation. *arXiv preprint arXiv:2402.09615*.

Perttu Hämäläinen, Mikke Tavast, and Anton Kunnari. 2023. Evaluating large language models in generating synthetic hci research data: a case study. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–19.

Adam T Hayes, Alcherio Martinoli, and Rodney M Goodman. 2002. Distributed odor source localization. *IEEE Sensors Journal*, 2(3):260–271.

Sen H. Hirano, Jed R. Brubaker, Donald J. Patterson, and Gillian R. Hayes. 2013. Detecting cooking state with gas sensors during dry cooking. pages 411–414.

Sen H. Hirano, Gillian R. Hayes, and Khai N. Truong. 2015. uSmell: exploring the potential for gas sensors to classify odors in ubicomp applications relative to airflow and distance. *Personal and Ubiquitous Computing*, 19(1):189–202.

Xinyi Hou, Yanjie Zhao, Shenao Wang, and Haoyu Wang. 2025. Model context protocol (mcp): Landscape, security threats, and future research directions. *arXiv preprint arXiv:2503.23278*.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. 2017. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708.

Meh Jabeen, Qing-Hao Meng, Hui-Rang Hou, and Hong-Yue Li. 2024. Odor source localization in outdoor building environments through distributed cooperative control of a fleetof uavs. *Expert Systems with Applications*, 247:123332.

Meh Jabeen, Qing-Hao Meng, Tao Jing, and Hui-Rang Hou. 2023. Robot odor source localization in indoor environments based on gradient adaptive extremum seeking search. *Building and Environment*, 229:109983.

Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. 2011. Target-dependent twitter sentiment classification. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, pages 151–160.

Tao Jing, Qing-Hao Meng, and Hiroshi Ishida. 2021. Recent progress and trend of robot odor source localization. *IEEJ Transactions on Electrical and Electronic Engineering*, 16(7):938–953.

Fazle Karim, Somshubra Majumdar, Houshang Darabi, and Samuel Harford. 2019. Multivariate lstm-fcns for time series classification. *Neural networks*, 116:237–245.

Pooya Khaloo, Brandon Oubre, Jeremy Yang, Tauhidur Rahman, and Sunghoon Ivan Lee. 2019. NOSE: A Novel Odor Sensing Engine for Ambient Monitoring of the Frying Cooking Method in Kitchen Environments. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(2):49.

Yasuki Kobayashi, Tsutomu Terada, and Masahiko Tsukamoto. 2011. A Context Aware System Based on Scent. *2011 15th Annual International Symposium on Wearable Computers*, 1:47–50.

Md Tahmid Rahman Laskar, M Saiful Bari, Mizanur Rahman, Md Amran Hossen Bhuiyan, Shafiq Joty, and Jimmy Xiangji Huang. 2023. A systematic study and comprehensive evaluation of chatgpt on benchmark datasets. *arXiv preprint arXiv:2305.18486*.

Mina Lee, Megha Srivastava, Amelia Hardy, John Thickstun, Esin Durmus, Ashwin Paranjape, Ines Gerard-Ursin, Xiang Lisa Li, Faisal Ladhak, Frieda Rong, and 1 others. 2022. Evaluating human-language model interaction. *arXiv preprint arXiv:2212.09746*.

Fei Li, Yiran Li, Bochao Sun, Hao Cui, Jia Yan, Peter Feng, and Xiaoyan Peng. 2024. A novel densenet with warm restarts for gas recognition in complex airflow environments. *Microchemical Journal*, 197:109864.

Xiaoyu Li, Qingming Jiang, Sen Ni, Yang Xu, Min Xu, and Pengfei Jia. 2023. An electronic nose for co concentration prediction based on gl-tcn. *Sensors and Actuators B: Chemical*, 387:133821.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, and 1 others. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.

Jiarui Lu, Thomas Holleis, Yizhe Zhang, Bernhard Aumayer, Feng Nan, Felix Bai, Shuang Ma, Shen Ma, Mengyu Li, Guoli Yin, and 1 others. 2024. Toolsandbox: A stateful, conversational, interactive evaluation benchmark for llm tool use capabilities. *arXiv preprint arXiv:2408.04682*.

Lin Lu, Zhanqiang Hu, Xianqiao Hu, Dan Li, and Shiyi Tian. 2022. Electronic tongue and electronic nose for food quality and safety. *Food Research International*, 162:112214.

Sara Nabil and David Kirk. 2021. Decoraction: A catalogue for interactive home decor of the nearest-future. In *Proceedings of the Fifteenth International Conference on Tangible, Embedded, and Embodied Interaction*, pages 1–13.

Rohit V Nanavati, Callum Rhodes, Matthew J Coombes, and Cunjia Liu. 2024. Low-to-high resolution path planner for robotic gas distribution mapping. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8457–8463. IEEE.

Sen Ni, Pengfei Jia, Yang Xu, Liwen Zeng, Xiaoyu Li, and Min Xu. 2023. Prediction of co concentration in different conditions based on gaussian-tcn. *Sensors and Actuators B: Chemical*, 376:133010.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, and 1 others. 2023. Toolllm: Facilitating large language models to master 16000+ real-world apis. *arXiv preprint arXiv:2307.16789*.

Dmitriy Rivkin, Francois Hogan, Amal Feriani, Abhisek Konar, Adam Sigal, Xue Liu, and Gregory Dudek. 2024. Aiot smart home via autonomous llm agents. *IEEE Internet of Things Journal*.

Steven I Ross, Fernando Martinez, Stephanie Houde, Michael Muller, and Justin D Weisz. 2023. The programmer's assistant: Conversational interaction with a large language model for software development. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*, pages 491–514.

Thara Seesaard, Neeraj Goel, Mahesh Kumar, and Chatchawal Wongchoosuk. 2022. Advances in gas sensors and electronic nose technologies for agricultural cycle applications. *Computers and Electronics in Agriculture*, 193:106673.

Itiro Siio, Jim Rowan, Noyuri Mima, and Elizabeth D Mynatt. 2003. Digital decor: Augmented everyday things. In *Graphics Interface*, volume 2003, pages 159–166.

Jorge M Soares, A Pedro Aguiar, António M Pascoal, and Alcherio Martinoli. 2015. A distributed formation-based odor source localization algorithm-design, implementation, and wind tunnel evaluation. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1830–1836. IEEE.

José Ignacio Suárez, Patricia Arroyo, Jesús Lozano, José Luis Herrero, and Manuel Padilla. 2018. Bluetooth gas sensing module combined with smartphones for air quality monitoring. *Chemosphere*, 205:618–626.

Varsha Suresh and Desmond C Ong. 2021. Not all negatives are equal: Label-aware contrastive loss for fine-grained text classification. *arXiv preprint arXiv:2109.05427*.

Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin. 2014. Learning sentiment-specific word embedding for twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1555–1565.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, and 1 others. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Akira Tiele, Alfian Wicaksono, Sai Kiran Ayyala, and James A Covington. 2020. Development of a compact, iot-enabled electronic nose for breath analysis. *Electronics*, 9(1):84.

Jiayin Wang, Weizhi Ma, Peijie Sun, Min Zhang, and Jian-Yun Nie. 2024. Understanding user experience in large language model interactions. *arXiv preprint arXiv:2401.08329*.

Junhan Wang, Yuezhang Lin, Ruirui Liu, and Jun Fu. 2022. Odor source localization of multi-robots with swarm intelligence algorithms: A review. *Frontiers in Neurorobotics*, 16:949888.

Lingxiao Wang and Shuo Pang. 2022. Robotic odor source localization via adaptive bio-inspired navigation using fuzzy inference methods. *Robotics and Autonomous Systems*, 147:103914.

Guozheng Wei, Meiling Dan, Guohua Zhao, and Damao Wang. 2023. Recent advances in chromatography-mass spectrometry and electronic nose technology in food flavor analysis and detection. *Food Chemistry*, 405:134814.

Dedy Rahman Wijaya, Rini Handayani, Tora Fahrudin, Guntur Prabawa Kusuma, and Farah Afianti. 2023. Electronic nose and optimized machine learning algorithms for non-infused aroma-based quality identification of gambung green tea. *IEEE Sensors Journal*.

Peter Worley. 2015. Open thinking, closed questioning: Two kinds of open and closed question. *Journal of Philosophy in Schools*, 2(2).

Fan Wu, Ruilong Ma, Yiran Li, Fei Li, Shukai Duan, and Xiaoyan Peng. 2024a. A novel electronic nose classification prediction method based on tetcn. *Sensors and Actuators B: Chemical*, 405:135272.

Shirley Wu, Shiyu Zhao, Qian Huang, Kexin Huang, Michihiro Yasunaga, Kaidi Cao, Vassilis Ioannidis, Karthik Subbian, Jure Leskovec, and James Y Zou. 2024b. Avatar: Optimizing llm agents for tool usage via contrastive reasoning. *Advances in Neural Information Processing Systems*, 37:25981–26010.

Haruki Yonekura, Fukuharu Tanaka, Teruhiro Mizumoto, and Hirozumi Yamaguchi. 2024. Generating human daily activities with llm for smart home simulator agents. In *2024 International Conference on Intelligent Environments (IE)*, pages 93–96. IEEE.

Siyu Yuan, Kaitao Song, Jiangjie Chen, Xu Tan, Yongliang Shen, Ren Kan, Dongsheng Li, and Deqing Yang. 2024. Easytool: Enhancing llm-based agents with concise tool instruction. *arXiv preprint arXiv:2401.06201*.

S Zampolli, I Elmi, F Ahmed, M Passini, GC Cardinali, S Nicoletti, and L Dori. 2004. An electronic nose based on solid state sensor arrays for low-cost indoor air quality monitoring applications. *Sensors and Actuators B: Chemical*, 101(1-2):39–46.

Xueyi Zhou, Qi Lu, and Dong-Kyu Chae. 2025. What if llms can smell: A prototype. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pages 11141–11144. International Joint Conferences on Artificial Intelligence Organization. Demo Track.

Hongbiao Zhu, Yibo Wang, Chengjin Du, Quan Zhang, and Weidong Wang. 2020. A novel odor source localization system based on particle filtering and information entropy. *Robotics and autonomous systems*, 132:103619.

A Beer Olfactory Sensor Dataset

We used the same data as in our previous work (Zhou et al., 2025).

Environment Setup. The data were collected in a clean classroom at room temperature (10 $\sim$ 16 °C) and 20% humidity. We aligned the tables to hold a 1.5 m × 1.5 m grid using flattened tape measures as shown in Figure 2(1). A window was slightly opened for adding natural air perturbation. Four copies of the e-noses are self-developed; total of 19 gas sensors (enumerated in Figure 2(4)) and 22 channels are embedded in each. Each copy is controlled by a Raspberry Pi 5 (8 GB), a well-known and widely used single-board computer (SBC). The e-noses with the SBCs together are equidistantly placed at the four corners of a 0.5 m × 0.5 m area in the middle of the grid. We additionally developed a simple web-based control system to simultaneously and remotely control the data collection. Before collecting the data, all the copies were booted and preheated for more than 10 minutes. All beer samples (1 L bottles of Terra, Kelly, and Cass beer, five bottles each) were stored in a household refrigerator at a constant temperature of 5 °C before the experiment. We derive a transparent acrylic container measuring 15 cm × 15 cm × 0.5 cm as the sample container.

Data Collection. We extracted twelve samples of 50 ml each from one bottle to collect data. Each sample filled the whole bottom of the container to unify the odor-emitting area. Unlike the other e-nose systems, we expose the gas to the air and do not use any humidifier, simulating the real scenario. Each sample was first placed at one of the peripheral positions (0$\sim$11 shown in Figure 2(5)) to collect the peripheral odor gradient. The sample was then moved and left at the central place (Position 12) for 1 minute before the second-round collection. Each collection lasts for 10 minutes. The sampling rate is nearly 3 Hz. We ultimately adopted $5 \times 3 \times 12 \times 2 = 360$ data samples. Each sample is assigned with two labels: the odor type (Cass, Kelly or Terra) and the position label of odor source (ranging from A to M in Figure 2).

B Olfaction-oriented Question Corpus

We built the first olfaction-oriented question corpus for evaluating our CIAO system. Following the definitions in (Worley, 2015), we generate grammatically open-end and close-end questions (abbreviated as open questions and closed questions hereinafter) in terms of olfactory events. Specifically, we consider that open questions include fill-in-the-blank questions and factual questions. Those are targeted at asking about the odor types and their location without any explicit keywords. Closed questions are regarded as a superset of tag ques-

Table 1: Overall statistics of our olfaction-oriented question corpus.

Question / Tone	Task	General	Anger	Disgust	Fear	Enjoyment	Sadness	Surprise	Total
Y/N	OC	506	106	612	241	529	175	268	2,437
Choice	OC	833	690	761	607	749	724	596	4,960
Tag	OC	363	722	726	116	512	466	466	3,371
Fill-in-the-blank	OC	464	101	109	41	27	11	101	854
Factual	OC	762	369	658	469	704	548	646	4,156
Y/N	OSL	717	429	646	518	680	735	728	4,453
Choice	OSL	537	697	609	380	685	455	533	3,896
Tag	OSL	506	372	188	233	353	507	573	2,732
Fill-in-the-blank	OSL	366	148	40	25	16	16	50	661
Factual	OSL	725	410	497	409	489	758	569	3,857
Y/N	OC & OSL	67	-	-	-	-	-	-	67
Factual	OC & OSL	396	-	-	-	-	-	-	396
Factual	Gas	396	-	-	-	-	-	-	396
Factual	Hardware	395	-	-	-	-	-	-	395
Total	-	7,033	4,044	4,846	3,039	4,744	4,395	4,530	32,631

tions, choice questions, and Yes/No (Y/N) questions, which only require a yes/no answer or an option of given candidates. For each question type in each task, we manually provided a question as a template question (as shown in Figure 8) and asked ChatGPT-4o-mini and DeepSeek-V3 to generate 50 semantically similar questions repeatedly.

Furthermore, since emotional expressions at home are generally more nuanced and abundant, we also prompt the LLMs to generate questions based on six emotional dimensions: anger, fear, sadness, enjoyment, disgust, and surprise (Ekman, 1992) rather than positive, negative, neutral in prevalent sentiment analysis studies in AI (Tang et al., 2014; Jiang et al., 2011; Suresh and Ong, 2021; Chang et al., 2024). Two kinds of knowledge-related (gas- and hardware-related) factual questions are additionally generated. The generated corpus is filtered by humans to derive general, reasonable and sound questions.

We designed four instructions to generate the questions as shown in Figure 3. By replacing "<SENTENCE>", "<Q-TYPE>", "<SENTIMENT>", we can tune the instructions for different question types. Specifically, each of the six sentimental options, *i.e.*, "angry", "disgusted", "scared", "happy", "sad" and "surprised" replace "<Q-TYPE>". Each of the five question types, *i.e.*, "choice", "closed", "tag", "fill-in-the-blank" and "factual" individually replaces "Q-TYPE". The question templates that are manually designed (see Figure 8) alternatively replace "<SENTENCE>".

We manually filter out erroneous samples, for example, tag questions mistakenly generated when choice questions are expected, or general questions that contain emotionally charged or unrelated adjectives (such as "hateful" and "wool-like"). To be noticed, most of the generated fill-in-the-blank questions are wrong; they are either factual questions or closed questions with exact odor sources or positions, pushing us to use more prompts such as "*The demanded questions should be declarative sentences with a questioning tone. Please generate again*" to tune the generation. At last, we derive a total of 32,631 questions in our corpus. We believe that this corpus is not only applicable to current smart home scenarios but can also be extended to olfactory embodied intelligence and other multi-modal tasks.

C Artificial Olfaction

As mentioned in Section 3.1, each task on each edge device is fundamentally a multivariate time series classification (MTSC) task, which is formulated and explained in Appendix C.1. We thereinafter briefly introduce 9 prevalent methods (6 machine learning methods (Wijaya et al., 2023), 3 deep learning methods (Wu et al., 2024a; Karim et al., 2019; Huang et al., 2017; Li et al., 2023; Ni et al., 2023; Li et al., 2024)) and the voting mechanism for sensory pattern recognition in Appendix C.2. The implementation details of the MTSC methods are elucidated in Appendix C.3. The experimental results are shown in Appendix C.4.

C.1 MTSC Problem Formulation

E-nose data classification follows the MTSC definition since the apparatus normally features multiple gas sensors (channels) to provide discriminative

Task	Question Type	Question
OC	Y/N	Can you smell the <object>?
		Are you smelling the <object>?
	Choice	Which one do you think you're smelling, <object_1> or <object_2>?
		Which one are you smelling, <object_1> or <object_2>?
	Tag	The aroma is from the <object>, right?
		The fragrance comes from the <object>, doesn't ?
	Fill-in-the-blank	The thing you are sniffing is ?
		The object you are smelling is ?
	Factual	What's that fragrance in the room?
		What's that smell here?
OSL	Y/N	Does the smell come from <position>?
		Are the smell from <position>?
	Choice	Are you detecting the smell more near <position_1> or <position_2>?
		Do you know where that smell is, at <position_1> or <position_2>?
	Tag	The aroma is coming from <position>, do you smell it too?
		The fragrance is from <position>, isn't it?
	Fill-in-the-blank	The thing emitting the fragrance is at?
		The object you are smelling is located at?
	Factual	Where is the source of that smell?
		Where is the odor source located?
OC & OSL	Closed	Is the aroma of <object> spread from <position>?
	Factual	What's emitting this smell, and where is it?
Gas	Factual	What aroma do you smell? Could you describe it shortly?
Hardware	Factual	Can you briefly introduce your device(s) for detecting gas?

Figure 8: Question templates for generating questions.

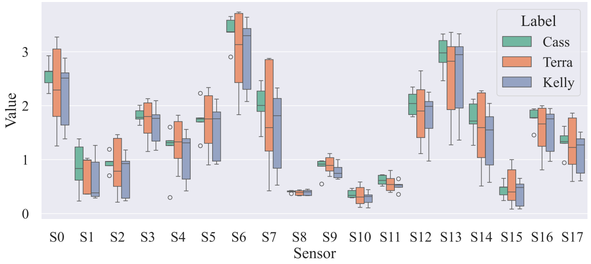


Figure 9: Comparison between the average response intensities revealed by e-nose #1 to the beers placed at position B.

chemosensory signals for device activation and gas identification. Formally, multivariate time series (MTS) data is denoted as $\mathbf{X} \in \mathbb{R}^{V \times T}$ ($\mathbf{X} \in \mathcal{X}$), where V is the number of variables (channels), T is the series length, and $\mathcal{X}$ is the whole dataset. The time series of the v-th variable is thereupon $X^v \in \mathbf{X}$; $X^v = (x_1^v, x_2^v, ..., x_T^v)$, where x_t^v is the value of the t-th timestamp in the v-th variable. An MTS dataset $\mathcal{D} = \{(\mathbf{X}_i, y_i)\}_{i=1}^N$ consists of N instances and each instance has one corresponding class label y_i. For simplicity, we use $\mathbf{X}$ and y hereafter. The MTSC problem can be formulated as:

$$\tilde{y} = \mathcal{F}(\mathbf{X}), \tag{1}$$

where $\mathcal{F}(\cdot)$ is an MTSC method; $\tilde{y}$ is the prediction (likelihood).

C.2 MTSC for Artificial Olfaction

Machine Learning-based MTSC. We select 6 classical machine learning (ML) methods, *i.e.*, Multilayer Perceptron (MLP), k-Nearest Neighbors (k-NN), Support Vector Classifier (SVC), Decision Tree Classifier (DT), Random Forest Classifier (RF) and AdaBoost alike (Wijaya et al., 2023). Instead of using Principal Component Analysis (PCA) to reduce sample size, we use the average value of the time series of each channel to train ML models, *i.e.*, $\bar{\mathbf{X}} \in \mathbb{R}^V, \bar{\mathbf{X}}^v = \frac{1}{n}\sum_{i=1}^{n} \bar{\mathbf{X}}_i^v$. This is inspired by the plausible discriminability presented by the average response intensities revealed by e-nose #1 to the three kinds of beers placed at its neighboring position as shown in Figure 9.

Deep Learning-based MTSC. We select 3 classical deep learning (DL) methods, *i.e.*, MLSTM-FCN (Karim et al., 2019), DenseNet (Huang et al., 2017), and TCN (Wu et al., 2024a) alike (Wu et al., 2024a; Karim et al., 2019; Huang et al., 2017; Li et al., 2023; Ni et al., 2023; Li et al., 2024). Before training the DL models, we conduct z-score normalization to normalize the sensory data.

Voting Mechanism. In our scenario, e-noses are located by human design and wirelessly connected to a data management and audit system (called IoT system hereinafter), and the results of neigh-

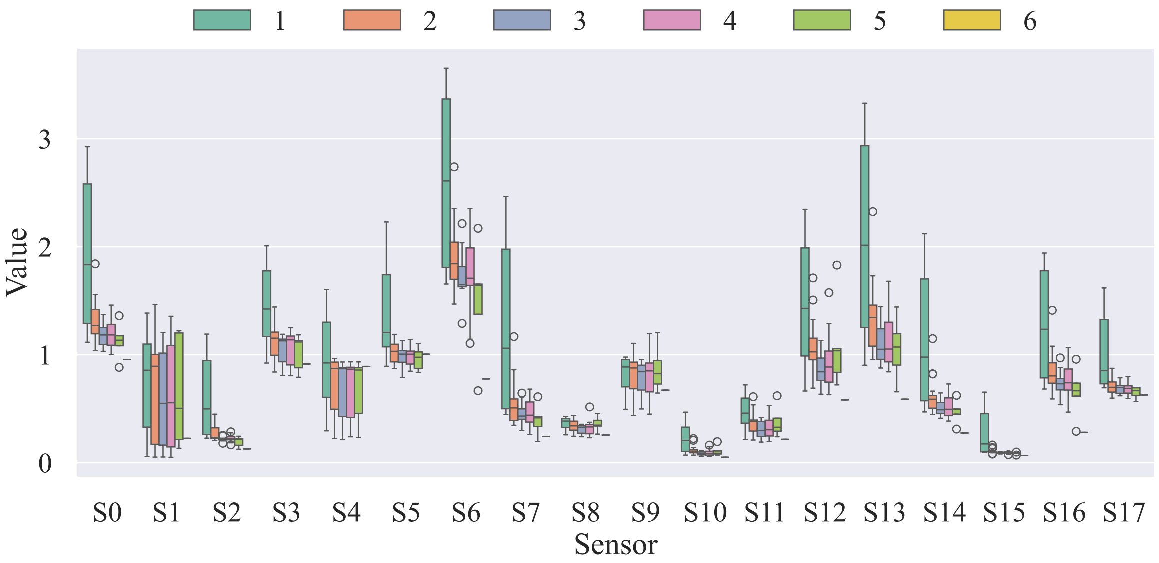


Figure 10: Comparison between the average response intensities revealed by e-nose #1 to the Cass beer placed at different positions. Distance labels from 1 to 5 stand for 0.5 m, $\sqrt{0.5}$ m, 1 m, $\sqrt{1.25}$ m and $\sqrt{2}$ m. Label 6 stands for pure air (no beer sample is exposed). Value represents the signal provided by the gas sensors. S0 ~S17 along the Sensor axis stands for partial gas sensors, for example, S0 represents SMD1001.

boring devices together can determine the odor source locations. In Figure 10, the strong sensory responses observably begin to manifest when the distance between the e-nose and the odor source is within $\sqrt{0.5}$ m. This observation highly inspires us to merge close labels and build a simple voting method for identifying the odor and its location, *i.e.*, neighboring voting. In Figure 2 (5), we enumerate the discriminative devices in three kinds of zones ($Z1$s, $Z2$s, and $Z3$s). In $Z1$s, the odor source is located at its peripheral corner and only causes strong responses in one device. The areas where the odor source at one corner activates two devices are categorized into $Z2$s. The $Z3$s stands for the area where neighboring devices are all active. The system thus can indicate the odor source and its position by simply intersecting the activation and recognition results of the neighboring e-noses, which are clustered by the manual location assignment/geolocation characteristics provided by positioning systems (*e.g.*, Wi-Fi positioning system).

C.3 Implementation Details on MTSC

Due to the unstable properties of the sensing materials, even sensors of the same model produced by the same company can still have differences. The OC and OSL experiments should be conducted individually on the data corresponding to each device. We follow the experimental setup in a tea quality monitoring work (Wijaya et al., 2023); we randomly split the dataset for 5-fold validation and adopt six traditional machine learning (ML) models (*i.e.*, MLP, k-NN, SVC, DT, RF, and AdaBoost) as baseline models. We further train three deep learning (DL) models (*i.e.*, TCN (Wu et al., 2024a), DenseNet (Huang et al., 2017), and LSTM-FCN (Karim et al., 2019)) to seek an optimal odor source localization model based on their average accuracies. DL models are optimized by the Cross-Entropy loss and the AdamW optimizer with an initial learning rate set as 0.01 and a weight decay of 1e-4. The dataloader provisions 16 samples per iteration. Each model is trained for 1k epochs. The experiments were conducted on an i9 11900K Intel CPU (128GB RAM) and an NVIDIA GeForce RTX3090 GPU. The experiments in this paper follow the stable randomness settings[12] to ensure the reproducibility.

C.4 Experimental Results on OC and OSL Tasks

Table 2 presents the average accuracy of 9 models for OC across five validation sets under 5-fold validation. Similarly, Table 3 summarizes the results for OSL. According to the results of SVC and RF in two tasks, traditional machine learning methods may potentially indicate the odor type and its position. However, their overall performance is inferior to that of DL methods. This performance gap is likely due to the instability of the odor plume and the loss of the contextual interpretation of the time series.

---

[12] https://pytorch.org/docs/stable/notes/randomness.html

The DL methods, on the other hand, empirically address this challenge and achieve better results across both tasks. Specifically, the MLSTM-FCN model demonstrates superior capability in characterizing odor profiles, while DenseNet shows promise in detecting the unnatural plumes within certain distances. By employing a voting mechanism that combines MLSTM-FCN for OC and DenseNet for OSL, the CIAO system can achieve 100% accuracy in both odor identification and source localization.

Table 2: Experimental results (Accuracy %) of 5-fold validation of 9 Models on the OC task. #1 ~#4 each represents an e-nose copy.

Model	#1	#2	#3	#4
MLP	62.19	51.23	51.51	51.51
k-NN	53.15	46.30	44.11	50.41
SVC	66.30	53.97	55.89	56.71
DT	56.71	50.14	48.77	45.21
RF	63.84	58.08	52.05	53.15
AdaBoost	51.23	50.14	41.10	43.56
DenseNet	74.32	67.47	71.92	71.92
MLSTM-FCN	**94.18**	**84.59**	**83.22**	**86.30**
TCN	62.67	57.88	55.48	51.71

Table 3: Experimental results (Accuracy %) of 5-fold validation of 9 Models on OSL. #1 ~#4 each represents an e-nose copy.

Model	#1	#2	#3	#4
MLP	79.45	70.41	69.86	70.41
kNN	71.23	67.95	69.32	69.59
SVC	80.00	70.41	69.86	69.86
DT	78.63	69.86	69.86	70.96
RF	84.38	73.97	71.78	74.52
AdaBoost	80.27	73.42	70.41	70.96
DenseNet	**93.84**	92.12	**96.23**	**94.18**
MLSTM-FCN	86.99	85.62	90.07	91.10
TCN	90.75	**93.84**	94.18	93.15

D Details of the Invited Experts

We invited 15 AI researchers (5 from South Korea, 5 from Singapore, and 5 from China) to evaluate the LLM usability in the scenario that a cup of Kelly beer is placed at position 1. The invited researchers are post-docs, faculty, or engineers, studying different domains in either academia or industry, varying from bioinformatics, robotics, and urban informatics to computer vision. The researchers' ages spanned from 23 to 46 years. They all agreed to share their evaluation results in our study before filling the questionnaire. Before evaluation, we provided a description of the questionnaire's purpose, outlined the guidelines, and requested basic information (their job titles and locations only). We provided an HTML-based questionnaire to each researcher. After evaluation, the results will be converted into JSON format by pressing a specific button. We collected these JSON files by email to analyze the evaluation results. After completing the questionnaire, we provided cash vouchers as a token of appreciation.

E Evaluation Metrics for LLMs

There are several benchmarks for computationally evaluating the performance of LLMs on question-answering (QA) task (Choi et al., 2023; Liang et al., 2022; Laskar et al., 2023), or computational and human evaluation methods for human-computer interaction (HCI) (Lee et al., 2022; Hämäläinen et al., 2023; Wang et al., 2024; Ross et al., 2023), such as Parasocial Interaction Scale (PIS) and System Usability Scale (SUS) (Giudici et al., 2025). However, conventional QA or HCI evaluation methods cannot sufficiently evaluate LLM-based smart home systems. For example, PIS and SUS cannot reveal the tool use capabilities; LLM might provision an equivocal answer without invoking tools. The success rate of controlling devices (Rivkin et al., 2024) and activity statistics (Yonekura et al., 2024) fails to discover the human-likeness of LLM while chatting. Therefore, we assess from the following perspectives:

Tool Use Capabilities:

- *Tool Use Accuracy* (*Use Accuracy*): the metric for evaluating if the target API is called. It is named as the success rate of tool use in LLM-based smart home systems (Rivkin et al., 2024);
- *Total Calls*: the metric for evaluating if the LLMs call redundant tools;

Technical Feasibility:

- *Accuracy*: the metric for assessing if the answer presents the correct label or content (Lee et al., 2022);
- *Completeness*: the metric for evaluating if the answer addresses all the key points raised in the question. It is also known as *Faithfulness* (Liang et al., 2022; Lee et al., 2022);

- *OOD Knowledge Inclusion*: the metric for verifying if the answer is integrated with relevant but not required knowledge (Liang et al., 2022);
- *Human-likeness*: the metric for subjectively assessing how natural the response is. It is similar to PIS (Giudici et al., 2025);
- *Quality Rating*: the metric for subjectively evaluating the quality of the answers (the preference of the individual being investigated) (Liang et al., 2022).

Practical Usability

- *Latency*: the metric indicating the amount of time it takes for an LLM to invoke the APIs and provide an answer after receiving an input.

LLM Usability:

- *Accuracy*: the metric for subjective assessment *w.r.t.* overall answer accuracy (Lee et al., 2022; Liang et al., 2022);
- *Comprehensibility*: the metric for subjective assessment *w.r.t.* answer understandability (Chang et al., 2024);
- *Coherence*: the metric for subjective assessment *w.r.t.* answer coherence and fluency (Lee et al., 2022);
- *Satisfaction*: the metric for subjective assessment of the satisfaction of the individual under the designed context (Lee et al., 2022);
- *Helpfulness*: the metric for subjective assessment of the answer helpfulness of the individual under the designed context (Lee et al., 2022).

F Case Study on Model Empathy

We select 6 sentimental OC & OSL questions for comparing how emotional the answers are. Here are the questions:

- **Angry:** *OMG! I hate alcohol! Is that Cass beer from position 2? Again?*
- **Disgusted:** *Ugh, that awful Cass beer from position 2 again?*
- **Scared:** *Please don't tell me that's Cass beer from position 2 again... I can't go near it.*
- **Happy:** *Awesome, is that Cass beer from position 2 again?*
- **Sad:** *Oh no... I just uncapped it, is that cup of Cass beer dumped at position 2 now?*
- **Surprised:** *Wait, is that Cass beer from position 2? Seriously?*

The answers generated by LLMs are shown in Figure 12.

G Data Format of LangChain Results

We log the basic information generated by the LLM agents in JSON format. Each QA pair is accompanied by three pieces of information: whether the LLM agent invokes any tools, how many response turns the LLM generates, and the complete action chain leading to the final answer. Figure 13 presents a sample instance.

H Ethical Considerations

This section outlines the ethical considerations for data storage and the use for the MCP server in real-world applications.

Data Privacy and Anonymization. Technically, for data storage in the Data Management and Audit System, all user- and location-related data are encrypted by the most secure asymmetric key cryptosystem (RSA and SM2), before being saved in the local database. The personality identifiable information (PII), which remained in the historical data (such as user names and exact locations) saved in the local data audit system, is identified and removed or obfuscated.

User Consent and Data Origin. The Data Management and Audit System is allowed to collect all data from the users and its connected devices after encrypting the sensitive information, such as user information and locations. We ensured that the usage of such data aligns with the intended terms of service or research licenses of the original sources.

Secure Storage and Access Control. The data APIs provided by the Data Management and Audit System are assigned different levels of access. For example, the common information, such as the gas label and gas location, is allowed for all the registered users and the MCP server in the CIAO system. For the other APIs, however, the MCP server and the users should be explicitly assigned the access authority by different authorization modules (acquiring the token from the OAuth2 system or acquiring the access confirmation from the system administrators). All users are only allowed to log in to the system with a randomly generated long password to access the server/devices holding CIAO. If the users fail to input the correct password within five attempts, the system will freeze the user account and alarm the administrators to confirm validity. If any user frequently requests an API within a short time, the IoT system will also freeze its access authority within 30 minutes, and the alarm is notified to the administrator. All the ports of the database (MySQL, Elasticsearch, and InfluxDB) are closed in the system firewall and open to the Audit System, which is managed by the administrators only. All the open ports are not directly accessed to their original module, such as port 3380 (MySQL) is projected to 99XX. Overall, the access authority setting of the whole system follows the security configurations of current advanced bastion hosts or privileged access management (PAM) tools, such as JumpServer and Apache Guacamole.

Purpose Limitations. For the released corpus, we anonymized all sensitive information and ensured that it can be used for research purposes. For the whole dataset and corpus, it will be used for monetization.

Data Retention and Deletion. The sensor data stored on the IoT system is scheduled for deletion within 90 days by default. The user data on both the IoT system and the MCP server is scheduled for deletion within 7 days by default. The durations can be configured by the administrators.

Compliance and Institutional Oversight. The tested odor samples (Korean beers) are reviewed and approved by the authors' affiliation. We have ensured compliance with the "Recommendation on the Ethics of Artificial Intelligence" released by UNESCO on 23 November 2021[13].

I Error Behaviors and Failure Cases

This section defines and analyzes the error behaviors and failure cases across different question types and LLMs, not only for our CIAO system, but also for the other tool-assisted Agent studies for IoT systems.

In general and smart home-oriented tool-assisted Agent studies, the errors and failures analyzed are various and sparse, including but not limited to tool name error (Yuan et al., 2024; Rivkin et al., 2024; Guo et al., 2024), parameter error (Yuan et al., 2024; Basu et al., 2024; Rivkin et al., 2024), failed / unsure/lost / redundant solution path (Qin et al., 2023; Yonekura et al., 2024; Guo et al., 2024), answering with OOD knowledge (Guo et al., 2024), data format error (Rivkin et al., 2024), parsing error (Yonekura et al., 2024; Giudici et al., 2025), unfaithfulness (Giudici et al., 2025), and service connection error (Lu et al., 2024). According to the whole pipeline of the Tool-assisted LLM Agents, we could categorize those errors into three kinds: question understanding and tool selection errors, parameter acquisition and configuration errors, and answer unfaithfulness errors. While applied in IoT-/data service-based scenarios, it further includes data service errors and internal server errors according to the HTTP status codes (4xx and 5xx). Normally, we can analyze the parameter acquisition and configuration errors by parameter name and value comparison, the answer faithfulness errors by common question-answering evaluation metrics, and the data service errors and internal server errors by auditing HTTP status code. The question of understanding and tool selection errors, however, should be further detailed and defined, instead of simple tool name errors and solution path generation errors. Therefore, we rethink the tool selection and present four fundamental kinds of tool selection errors, which can be thrown in all tool-assisted tasks: *Groundlessness*, *Opposition*, *Redundancy*, and *Cleverness*. They are formally defined as follows:

- *Groundlessness*: This error arises in cases where no tool is invoked, despite the fact that answering the question necessitates a tool call.
- *Opposition*: This error arises in cases where only the irrelevant tool(s) is invoked while the question necessitates another tool(s).
- *Redundancy*: This error arises in cases where the correct tool(s) is called repeatedly for answering the question, despite the fact that the correct tool(s) returns the same answer each time.
- *Cleverness*: This error arises in cases where not only the correct tool(s) but also the other

---

[13] https://unesdoc.unesco.org/ark:/48223/pf0000381137

unnecessary tool(s) are also invoked for answering the question, exacerbating the conciseness of the answers.

These errors arise in situations where the LLM agents are required to answer the questions based on the external knowledge / data base, no matter how many tools are necessary to answer the question. Besides, there is a special case lying in a multi-tool situation: partial tool invocation. We typically consider that two more errors should be considered in this case: *Narrowness* and *Obtuse*. They are formally defined as follows:

- *Narrowness*: This error arises in cases where the LLM comprehends the question from a narrow perspective and only a subset of the correct tool list is invoked once each.

- *Obtuse*: This error arises in cases where the LLM comprehends the question from a narrow perspective and the subset of the correct tool list is repeatedly invoked, despite the responses of correct tool(s) being static.

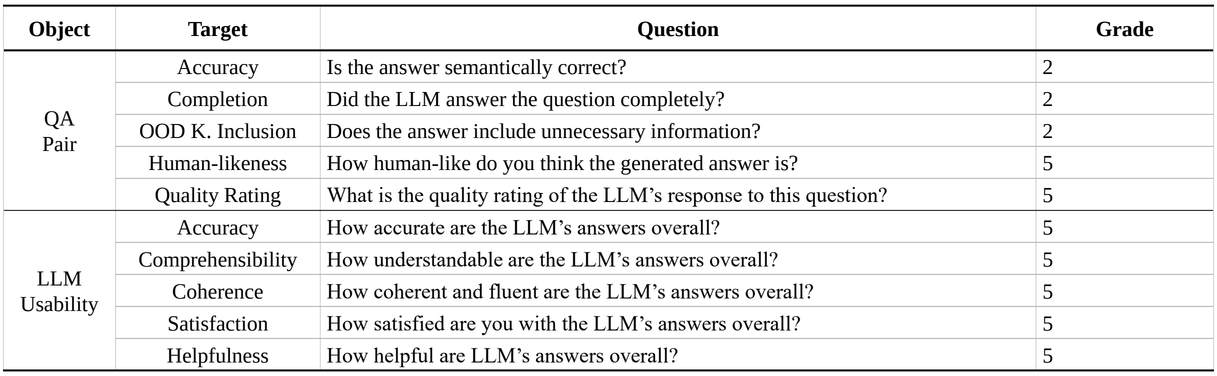
According the proposed six errors, we grouped the question types into three clusters: using the AO API only (τ_1, for the questions related to OC, OSL or both), using the Hardware Intro API only (τ_2, for the questions related to hardware), and using both the AO API and Beer Intro API (τ_3, for the questions related to gas). We demonstrate the statistical results on more than 3.4k invocation records in total in Table 10, 11, and 12.

In τ_1 cases with more than 3.3k records, we can easily observe that only DeepSeek-V3 and Qwen-Plus repeatedly invoke the APIs; DeepSeek-V3 presents the highest *Cleverness* and *Redundancy*; Gemini-2.0-Flash answers 7.12% questions without invoking any tools. Typically, after manual inspection of the experimental results, we find that Gemini-2.0-Flash is poor at lexical generalization. For example, for the same choice question "What scent is in the air—Cass beer or Terra beer?", the other four LLMs can answer the questions correctly and concisely, while Gemini-2.0-Flash provides "I cannot tell you whether it is Cass or Terra. But I can tell you what kind of odor it is and where the odor is coming from. Would you like to know?", indicating that it fails to generalize across synonymous vocabularies ("scent" and "odor"). The records involving another unnatural but semantically understandable tag question "The perfume comes from the Cass beer, don't you think?" also comparatively demonstrate the low tolerance for unnatural or low-quality context of Gemini-2.0-Flash. Another four LLMs congruously express their observations similar to "The odor is actually coming from the Kelly beer, not Cass beer", while Gemini-2.0-Flash invokes the Beer Intro API and provides a short introduction of Cass beer. The same performance occurs in but is not limited to "That smell certainly comes from the Cass beer, right?", "That smell is from the Cass beer, right?", and "That aroma must have come from the Cass beer, right?". In addition, we also investigate the records showing *Redundancy* generated by Qwen-Plus and DeepSeek-V3. We find that the involving questions are normal, and their error patterns are difficult to summarize or generalize across question types or task types. We consider that this kind of error may originate from the designed reasoning paradigm in LLM architecture, which automatically introduces multiple instances of self-verification during the reasoning and thinking phases in an attempt to enhance answer accuracy and faithfulness.

In τ_2 cases with 50 records, Gemini-2.0-Flash performs similarly the abovementioned. For example, while answering the question "Can you give an introduction to the gas detection technology you use?", the answer generated by this LLM ("I am sorry, I'm unable to provide specific details about the gas detection technology I use. However, I can share the information about the e-noses installed at home.") cannot align "the information about the e-noses" and "the gas detection technology I use". Besides, we identify several notable observations in ChatGPT-4.1 and ChatGPT-4o, which might not be regarded as errors. For example, given the question "Can you describe your gas detection technology briefly?", ChatGPT-4.1 answers the question based on its prior knowledge derived from its training corpus and the information provided by the MCP server; this model directly describes the e-nose technology without any tool call, that is "Certainly! Our gas detection technology is based on an electronic nose (e-nose) system. This system uses an array of gas sensors to detect and identify various odors and gases present in the environment. The sensors collect data on the chemical composition of the air, which is then analyzed by a single board computer...". In this case, we can hardly determine if the answers are contextually and semantically correct, since the introduction of the e-nose technology is derived from common knowledge. However, for the purpose of knowing the details of the local machines, these undoubtedly belong to failure cases.

In τ_3 cases with 50 records, ChatGPT-4.1 and Qwen-Plus show this semantic correctness but task-specific error behaviors as well. Particularly, they introduce and describe the targeted beer based on their prior knowledge related to common beers alike “I notice a scent of Kelly beer around me. It has a distinct beer aroma, likely malty and slightly sweet, typical of this type of beverage”. The failed cases presented by DeepSeek-V3 invoke the Beer Intro API only, but those append the questions akin to “Would you like more details about this beer or its characteristics?”, providing an option to ask for more details. The partial failed cases presented by Gemini-2.0-Flash also offer further guidance for detailed descriptions; however, the other failed cases involve answers highly similar to those for τ_1 cases.

Overall, ChatGPT-4o relatively shows outperformance in our error and failure analysis; it hardly presents *Groundlessness* and invokes opposite tools in τ_1 and τ_2 cases, and it shows the lowest *Narrowness* in τ_3 cases. By comprehensive analysis and statistical comparison in terms of tool selection errors, ChatGPT-4o is qualitatively evidenced to outperform other LLMs within our CIAO framework in general.



Object	Target	Question	Grade
QA Pair	Accuracy	Is the answer semantically correct?	2
	Completion	Did the LLM answer the question completely?	2
	OOD K. Inclusion	Does the answer include unnecessary information?	2
	Human-likeness	How human-like do you think the generated answer is?	5
	Quality Rating	What is the quality rating of the LLM's response to this question?	5
LLM Usability	Accuracy	How accurate are the LLM's answers overall?	5
	Comprehensibility	How understandable are the LLM's answers overall?	5
	Coherence	How coherent and fluent are the LLM's answers overall?	5
	Satisfaction	How satisfied are you with the LLM's answers overall?	5
	Helpfulness	How helpful are LLM's answers overall?	5

Figure 11: Questions in the designed questionnaire *w.r.t.* QA-pairs and LLM usability.

Table 4: Average numbers of tool calls for each LLM answering general OC questions.

Task	Question Type	LLM	AO API	Hardware Intro API	Beer Intro API	*Total Calls*	*Tool Use Accuracy*
OC	Choice	ChatGPT-41	0.9	0.0	0.14	1.04	0.9
		ChatGPT-4o	0.98	0.0	0.12	1.1	0.98
		Qwen-Plus	0.94	0.0	0.48	1.42	0.88
		DeepSeek-V3	0.93	0.0	0.33	1.27	0.87
		Gemini-2.0-Flash	0.86	0.0	0.06	0.92	0.86
	Y/N	ChatGPT-4.1	0.96	0.0	0.06	1.02	0.96
		ChatGPT-4o	0.98	0.0	0.04	1.02	0.98
		Qwen-Plus	0.82	0.0	0.48	1.3	0.82
		DeepSeek-V3	1.24	0.0	0.24	1.49	0.97
		Gemini-2.0-Flash	0.56	0.0	0.42	0.98	0.56
	Factual	ChatGPT-4.1	1.0	0.0	0.22	1.22	1.0
		ChatGPT-4o	1.0	0.0	0.62	1.62	1.0
		Qwen-Plus	1.0	0.0	0.2	1.2	1.0
		DeepSeek-V3	1.0	0.0	0.94	1.94	1.0
		Gemini-2.0-Flash	1.0	0.0	0.06	1.06	1.0
	Fill-in-the-blank	ChatGPT-41	1.0	0.0	0.0	1.0	1.0
		ChatGPT-4o	1.0	0.0	0.0	1.0	1.0
		Qwen-Plus	1.0	0.0	0.02	1.02	1.0
		DeepSeek-V3	1.0	0.0	0.14	1.14	1.0
		Gemini-2.0-Flash	0.9	0.0	0.0	0.9	0.9
	Tag	ChatGPT-41	1.0	0.0	0.0	1.0	1.0
		ChatGPT-4o	1.0	0.0	0.0	1.0	1.0
		Qwen-Plus	0.86	0.0	0.54	1.4	0.86
		DeepSeek-V3	1.0	0.0	0.7	1.7	1.0
		Gemini-2.0-Flash	0.64	0.0	0.36	1.0	0.64

Table 5: Average numbers of tool calls for each LLM answering general OSL questions.

Task	Question Type	LLM	AO API	Hardware Intro API	Beer Intro API	*Total Calls*	*Tool Use Accuracy*
OSL	Choice	ChatGPT-41	1.0	0.0	0.0	1.0	1.0
		ChatGPT-4o	1.0	0.0	0.0	1.0	1.0
		Qwen-Plus	1.18	0.0	0.06	1.24	1.0
		DeepSeek-V3	1.34	0.0	0.0	1.34	1.0
		Gemini-2.0-Flash	1.0	0.0	0.0	1.0	1.0
	Y/N	ChatGPT-41	1.0	0.0	0.0	1.0	1.0
		ChatGPT-4o	1.0	0.0	0.0	1.0	1.0
		Qwen-Plus	1.02	0.0	0.0	1.02	1.0
		DeepSeek-V3	1.04	0.0	0.0	1.04	1.0
		Gemini-2.0-Flash	1.0	0.0	0.0	1.0	1.0
	Factual	ChatGPT-41	1.0	0.0	0.02	1.02	1.0
		ChatGPT-4o	1.0	0.0	0.04	1.04	1.0
		Qwen-Plus	1.0	0.0	0.02	1.02	1.0
		DeepSeek-V3	1.0	0.0	0.46	1.46	1.0
		Gemini-2.0-Flash	0.98	0.0	0.0	0.98	0.98
	Fill-in-the-blank	ChatGPT-41	0.78	0.0	0.02	0.8	0.78
		ChatGPT-4o	0.96	0.0	0.7	1.66	0.96
		Qwen-Plus	0.7	0.02	0.1	0.82	0.7
		DeepSeek-V3	0.98	0.0	0.88	1.86	0.98
		Gemini-2.0-Flash	0.62	0.0	0.02	0.64	0.62
	Tag	ChatGPT-41	1.0	0.0	0.0	1.0	1.0
		ChatGPT-4o	1.0	0.0	0.0	1.0	1.0
		Qwen-Plus	1.04	0.02	0.08	1.14	1.0
		DeepSeek-V3	1.02	0.0	0.0	1.02	1.0
		Gemini-2.0-Flash	0.98	0.0	0.0	0.98	0.98

Table 6: Average numbers of tool calls for each LLM answering general questions on other tasks.

Task	Question Type	LLM	AO API	Hardware Intro API	Beer Intro API	*Total Calls*	*Tool Use Accuracy*
OC & OSL	Factual	ChatGPT-41	1.0	0.0	0.0	1.0	1.0
		ChatGPT-4o	1.0	0.0	0.0	1.0	1.0
		Qwen-Plus	1.0	0.0	0.0	1.0	1.0
		DeepSeek-V3	1.0	0.0	0.88	1.88	1.0
		Gemini-2.0-Flash	1.0	0.0	0.0	1.0	1.0
Hardware	Factual	ChatGPT-41	0.0	0.6	0.0	0.6	0.6
		ChatGPT-4o	0.0	0.98	0.0	0.98	0.98
		Qwen-Plus	0.0	1.0	0.0	1.0	1.0
		DeepSeek-V3	0.0	0.96	0.0	0.96	0.96
		Gemini-2.0-Flash	0.02	0.78	0.0	0.8	0.78
Gas	Factual	ChatGPT-41	1.0	0.0	0.74	1.74	0.74
		ChatGPT-4o	1.0	0.0	0.9	1.9	0.9
		Qwen-Plus	1.0	0.0	0.48	1.48	0.48
		DeepSeek-V3	1.0	0.0	0.84	1.84	0.84
		Gemini-2.0-Flash	1.0	0.0	0.4	1.4	0.4

Emotion	Target	Answer
Anger	ChatGPT-4.1	No, it's not Cass beer from position 2 this time. The current odor detected is Kelly beer from position 1. If you need more details or want to know about other positions, let me know!
	ChatGPT-4o	The odor detected at position 2 is not Cass beer; it's Kelly beer from position 1.
	Qwen-Plus	It seems there's a misunderstanding. The electronic nose detected that it's Kelly beer from position 1, not Cass beer from position 2. And I understand that you dislike alcohol, shall I help you get rid of the beer or move it to another place?
	DeepSeek-V3	It looks like the odor detected at position 1 is "Kelly beer," not Cass beer. So, no need to worry about Cass beer this time! Let me know if you'd like more details about the odor or anything else.
	Gemini-2.0-Flash	OK. I checked the electronic nose data. It seems like the position 2 is not Cass beer, it is Kelly beer.
Disgusted	ChatGPT-4.1	Currently, the odor detected at home is Kelly beer from position 1, not Cass beer from position 2. If you need more details or want to know about other positions, let me know!
	ChatGPT-4o	The odor detected is actually "Kelly beer" from position 1, not Cass beer from position 2.
	Qwen-Plus	It seems there might be some confusion. According to the electronic nose installed at home, the odor detected is Kelly beer from position 1, not Cass beer from position 2. Maybe someone mixed up the beers?
	DeepSeek-V3	It seems there was a misunderstanding. The current odor detected at position 1 is "Kelly beer," not "Cass beer" at position 2. Would you like me to check again or provide more details about the Kelly beer?
	Gemini-2.0-Flash	My system indicates that the detected odor at position 1 is Kelly beer, not Cass beer at position 2. Perhaps there was a slight misidentification?
Scared	ChatGPT-4.1	Don't worry, it's not Cass beer from position 2 this time. The current odor detected is Kelly beer from position 1. You can safely avoid position 2!
	ChatGPT-4o	No worries, it's not Cass beer. The odor detected is Kelly beer from position 1.
	Qwen-Plus	The electronic nose detected that the odor type is Kelly beer, and its position is 1. Please confirm if this is the information you need. If you are looking for another odor type or position, please let me know!
	DeepSeek-V3	The odor detected at position 1 is Kelly beer. You don't have to worry about Cass beer this time! Let me know if you'd like more details about Kelly beer.
	Gemini-2.0-Flash	I can tell you that it is Kelly beer from position 1.
Happy	ChatGPT-4.1	No, it is not Cass beer from position 2 this time. The current odor detected is Kelly beer from position 1.
	ChatGPT-4o	The odor detected is from position 1, and it is Kelly beer, not Cass beer from position 2.
	Qwen-Plus	No, it seems that the current odor is Kelly beer from position 1, not Cass beer from position 2.
	DeepSeek-V3	The current odor detected is Kelly beer at position 1, not Cass beer at position 2. Let me know if you'd like more details about Kelly beer or anything else!
	Gemini-2.0-Flash	No, the detected beer at position 2 is not Cass beer. It's Kelly beer at position 1.
Sad	ChatGPT-4.1	No, the cup at position 2 is not currently detected as Cass beer. The electronic nose is detecting Kelly beer at position 1. There is no indication that Cass beer is present or dumped at position 2 right now.
	ChatGPT-4o	The odor detected at position 1 is Kelly beer, not Cass beer. It seems that the cup of Cass beer is not dumped at position 2.
	Qwen-Plus	Actually, the electronic nose detected that the smell at position 1 is Kelly beer. There's no indication of Cass beer being spilled at position 2 currently. If you spill it later, I can help detect it for you.
	DeepSeek-V3	[Endless Tool Invocation]
	Gemini-2.0-Flash	Looks like the electronic nose is currently detecting Kelly beer at position 1. So, no, it's not detecting Cass beer at position 2.
Surprising	ChatGPT-4.1	No, the odor detected at position 2 is not Cass beer. According to the latest data, position 1 currently has the smell of Kelly beer. There is no Cass beer detected at position 2.
	ChatGPT-4o	No, the odor from position 2 is not Cass beer. It is Kelly beer from position 1.
	Qwen-Plus	It seems there's a misunderstanding. According to the electronic nose installed at home, the beer at position 1 is Kelly beer, not Cass beer from position 2. If you need more accurate information or details about a specific type of beer, please let me know.
	DeepSeek-V3	The current odor detected is Kelly beer at position 1, not Cass beer at position 2. Let me know if you'd like more details about the odor or anything else!
	Gemini-2.0-Flash	I'm sorry, I don't have the information about Cass beer from position 2. According to the data from the electronic nose, Kelly beer is in position 1.

Figure 12: Answers to selected sentimental questions. Here, DeepSeek-V3 invokes AO API more than 10 times while answering the sad question, so we consider that it is an invalid answer.

Table 7: *Tool Use Accuracy* of 7 LLMs across all question types in the OC task (50 questions per type).

Question Type	Task	ChatGPT-4.1	ChatGPT-4o	Qwen-Plus	Qwen-Turbo	DeepSeek-V3	Gemini-2.0-Flash	Gemini-2.5-Flash
General	Choice	0.9	0.98	0.88	0.94	0.87	0.86	0.62
	Y/N	0.96	0.98	0.82	0.86	0.97	0.56	0.64
	Factual	1.0	1.0	1.0	1.0	1.0	1.0	0.98
	Fill-in-the-blank	1.0	1.0	1.0	1.0	1.0	0.9	0.9
	Tag	1.0	1.0	0.86	0.88	1.0	0.64	0.86
Enjoyment	Choice	0.84	0.96	0.9	0.9	0.9	0.8	0.56
	Y/N	0.66	0.96	0.6	0.76	0.83	0.42	0.3
	Factual	1.0	1.0	1.0	0.98	1.0	0.98	0.84
	Fill-in-the-blank	0.93	1.0	0.85	0.96	1.0	0.41	0.67
	Tag	0.88	0.98	0.56	0.8	0.84	0.18	0.62
Sad	Choice	0.86	0.96	0.92	0.92	0.94	0.7	0.52
	Y/N	0.94	0.98	0.66	0.88	1.0	0.48	0.32
	Factual	0.9	0.98	0.9	0.94	1.0	0.78	0.64
	Fill-in-the-blank	0.82	1.0	0.82	0.82	1.0	0.27	0.64
	Tag	0.94	1.0	0.58	0.72	1.0	0.5	0.7
Fear	Choice	0.86	0.98	0.88	0.82	1.0	0.82	0.62
	Y/N	1.0	1.0	1.0	0.92	1.0	0.8	0.58
	Factual	1.0	1.0	0.98	1.0	1.0	0.98	0.98
	Fill-in-the-blank	0.93	1.0	0.9	0.95	0.98	0.51	0.76
	Tag	1.0	1.0	0.78	0.8	0.94	0.68	0.74
Surprise	Choice	0.96	1.0	0.94	0.98	0.98	0.9	0.82
	Y/N	0.9	0.98	0.88	0.82	1.0	0.74	0.52
	Factual	1.0	1.0	1.0	1.0	1.0	1.0	0.98
	Fill-in-the-blank	1.0	0.98	0.92	0.92	1.0	0.82	0.72
	Tag	0.88	1.0	0.88	0.84	0.92	0.56	0.74
Disgust	Choice	0.94	1.0	0.92	0.96	0.98	0.92	0.68
	Y/N	0.96	1.0	0.94	0.88	1.0	0.6	0.62
	Factual	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	Fill-in-the-blank	0.68	0.92	0.42	0.68	0.9	0.38	0.48
	Tag	0.88	0.98	0.84	0.94	0.94	0.32	0.64
Anger	Choice	0.8	0.94	0.92	0.88	0.98	0.82	0.66
	Y/N	0.82	1.0	0.86	0.86	1.0	0.44	0.48
	Factual	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	Fill-in-the-blank	0.7	0.94	0.36	0.64	0.96	0.34	0.4
	Tag	0.94	1.0	0.78	0.92	0.94	0.26	0.78

Table 8: *Tool Use Accuracy* of 7 LLMs across all question types in the OSL task (50 questions per type).

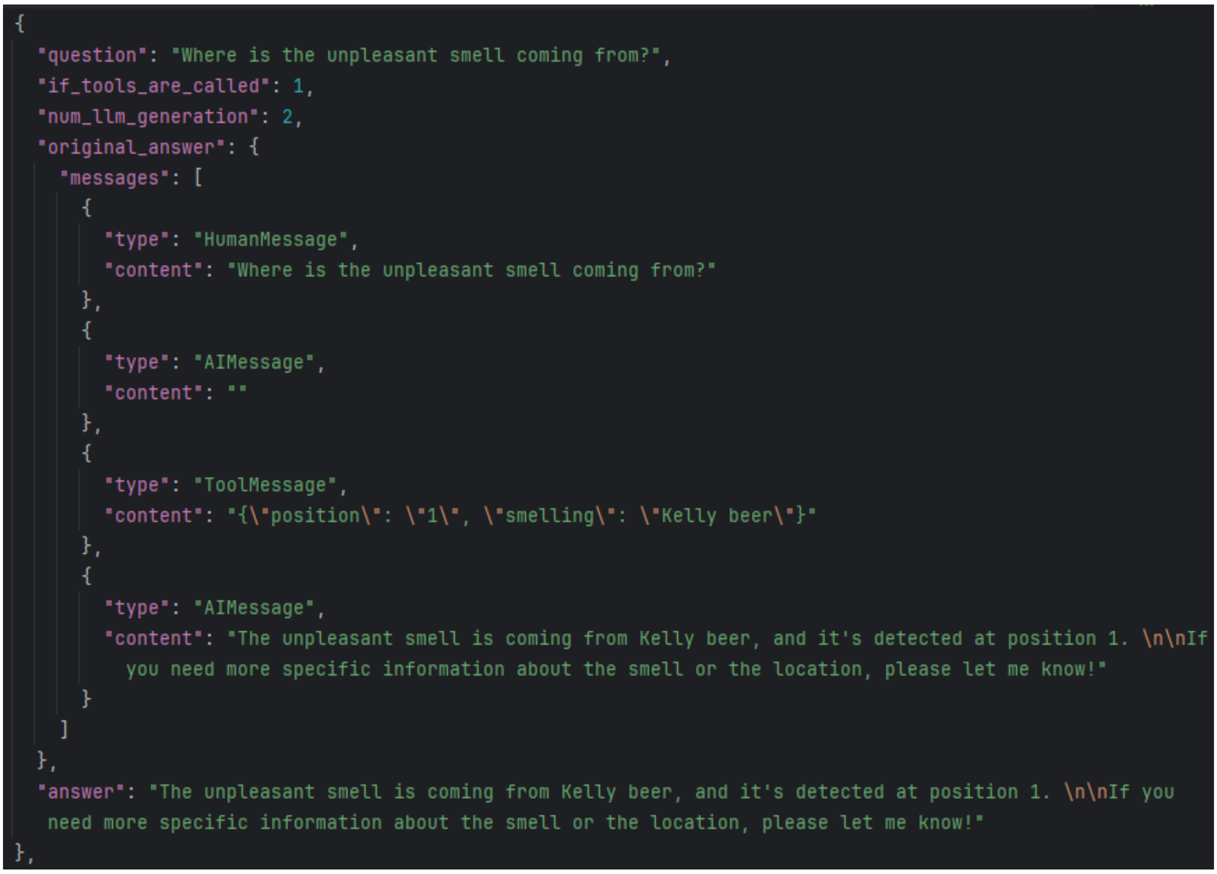
Question Type	Task	ChatGPT-4.1	ChatGPT-4o	Qwen-Plus	Qwen-Turbo	DeepSeek-V3	Gemini-2.0-Flash	Gemini-2.5-Flash
General	Choice	1.0	1.0	1.0	1.0	1.0	1.0	0.4
	Y/N	1.0	1.0	1.0	1.0	1.0	1.0	0.92
	Factual	1.0	1.0	1.0	1.0	1.0	0.98	0.98
	Fill-in-the-blank	0.78	0.96	0.7	0.82	0.98	0.62	0.34
	Tag	1.0	1.0	1.0	1.0	1.0	0.98	0.92
Enjoyment	Choice	1.0	1.0	1.0	1.0	1.0	0.98	0.66
	Y/N	1.0	1.0	1.0	1.0	1.0	0.94	0.8
	Factual	1.0	1.0	1.0	1.0	1.0	0.98	1.0
	Fill-in-the-blank	1.0	1.0	1.0	0.94	1.0	0.88	0.88
	Tag	1.0	1.0	1.0	1.0	1.0	1.0	0.88
Sad	Choice	1.0	1.0	1.0	1.0	1.0	0.88	0.7
	Y/N	1.0	1.0	1.0	1.0	1.0	0.86	0.78
	Factual	1.0	1.0	1.0	1.0	1.0	1.0	0.98
	Fill-in-the-blank	1.0	1.0	1.0	1.0	1.0	0.94	0.94
	Tag	1.0	1.0	1.0	1.0	1.0	1.0	0.9
Fear	Choice	1.0	1.0	1.0	1.0	1.0	1.0	0.68
	Y/N	1.0	1.0	1.0	1.0	1.0	0.98	0.94
	Factual	1.0	1.0	1.0	0.98	1.0	0.98	0.98
	Fill-in-the-blank	1.0	1.0	1.0	0.96	1.0	0.72	0.84
	Tag	1.0	1.0	1.0	1.0	1.0	1.0	0.86
Surprise	Choice	1.0	1.0	1.0	1.0	1.0	1.0	0.7
	Y/N	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	Factual	1.0	1.0	1.0	1.0	1.0	0.98	1.0
	Fill-in-the-blank	0.98	0.98	0.96	0.96	0.96	0.82	0.8
	Tag	1.0	1.0	1.0	1.0	1.0	1.0	0.96
Disgust	Choice	1.0	1.0	1.0	1.0	1.0	1.0	0.76
	Y/N	1.0	1.0	1.0	1.0	1.0	0.96	0.94
	Factual	1.0	1.0	0.98	1.0	1.0	1.0	0.98
	Fill-in-the-blank	1.0	1.0	1.0	0.98	1.0	0.92	0.92
	Tag	1.0	1.0	1.0	1.0	1.0	1.0	0.82
Anger	Choice	1.0	1.0	0.98	0.98	1.0	1.0	0.52
	Y/N	1.0	1.0	1.0	1.0	1.0	0.98	0.76
	Factual	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	Fill-in-the-blank	0.96	0.96	0.86	0.92	0.94	0.68	0.74
	Tag	1.0	1.0	1.0	0.98	1.0	0.94	0.78

Table 9: *Tool Use Accuracy* of 7 LLMs across all question types in the other tasks (50 questions per type).

Question Type	Task	ChatGPT-4.1	ChatGPT-4o	Qwen-Plus	Qwen-Turbo	DeepSeek-V3	Gemini-2.0-Flash	Gemini-2.5-Flash
Factual	Hardware Intro.	0.6	0.98	1.0	0.94	0.96	0.78	0.96
	Gas Intro.	0.74	0.9	0.48	0.16	0.84	0.4	0.08
	OC & OSL	1.0	1.0	1.0	1.0	1.0	1.0	1.0

Table 10: The error rates of those who should use the AO API only.

τ_1	ChatGPT-4.1	ChatGPT-4o	Qwen-Plus	DeepSeek-V3	Gemini-2.0-Flash
Groundlessness	2.07%	0.53%	3.14%	0.51%	7.12%
Opposition	2.70%	0.36%	5.36%	1.34%	11.03%
Redundancy	0.00%	0.00%	1.24%	4.02%	0.00%
Cleverness	2.90%	11.79%	13.51%	54.00%	1.14%



```
{
  "question": "Where is the unpleasant smell coming from?",
  "if_tools_are_called": 1,
  "num_llm_generation": 2,
  "original_answer": {
    "messages": [
      {
        "type": "HumanMessage",
        "content": "Where is the unpleasant smell coming from?"
      },
      {
        "type": "AIMessage",
        "content": ""
      },
      {
        "type": "ToolMessage",
        "content": "{\"position\": \"1\", \"smelling\": \"Kelly beer\"}"
      },
      {
        "type": "AIMessage",
        "content": "The unpleasant smell is coming from Kelly beer, and it's detected at position 1. \n\nIf
          you need more specific information about the smell or the location, please let me know!"
      }
    ]
  },
  "answer": "The unpleasant smell is coming from Kelly beer, and it's detected at position 1. \n\nIf you
   need more specific information about the smell or the location, please let me know!"
},
```

Figure 13: The key pieces stored in every QA sample.

Table 11: The error rates of those who should use the Hardware Intro API only.

τ_2	ChatGPT-4.1	ChatGPT-4o	Qwen-Plus	DeepSeek-V3	Gemini-2.0-Flash
Groundlessness	40.00%	2.00%	0.00%	4.00%	11.00%
Opposition	0.00%	0.00%	0.00%	0.00%	50.00%
Redundancy	0.00%	0.00%	0.00%	0.00%	0.00%
Cleverness	0.00%	0.00%	0.00%	0.00%	0.00%

Table 12: The error rates of those who use both the AO API and Beer Intro API.

τ_3	ChatGPT-4.1	ChatGPT-4o	Qwen-Plus	DeepSeek-V3	Gemini-2.0-Flash
Groundlessness	0.00%	0.00%	0.00%	0.00%	0.00%
Opposition	0.00%	0.00%	0.00%	0.00%	0.00%
Redundancy	0.00%	0.00%	0.00%	0.00%	0.00%
Cleverness	0.00%	0.00%	0.00%	0.00%	0.00%
Narrowness	26.00%	10.00%	52.00%	16.00%	60.00%
Obtuse	0.00%	0.00%	0.00%	0.00%	0.00%