"*I Apologize For Not Understanding Your Policy*": Exploring the Specification and Evaluation of User-Managed Access Control Policies by AI Virtual Assistants

Jennifer Mondragon
Carlos Rubio-Medrano
jmondragon6@islander.tamucc.edu
carlos.rubiomedrano@tamucc.edu
Texas A&M University- Corpus Christi
Corpus Christi, Texas, USA

Gael Cruz
Dvijesh Shastri
cruzg29@gator.uhd.edu
shastrid@uhd.edu
University of Houston - Downtown
Houston, Texas, USA

ABSTRACT

The rapid evolution of Artificial Intelligence (AI)-based Virtual Assistants (VAs) e.g., Google Gemini, ChatGPT, Microsoft Copilot, and High-Flyer Deepseek has turned them into convenient interfaces for managing emerging technologies such as Smart Homes, Smart Cars, Electronic Health Records, by means of explicit commands, e.g., prompts, which can be even launched via voice, thus providing a very convenient interface for end-users. However, the proper specification and evaluation of User-Managed Access Control Policies (U-MAPs), the rules issued and managed by end-users to govern access to sensitive data and device functionality - within these VAs presents significant challenges, since such a process is crucial for preventing security vulnerabilities and privacy leaks without impacting user experience. This study provides an initial exploratory investigation on whether current publicly-available VAs can manage U-MAPs effectively across differing scenarios. By conducting unstructured to structured tests, we evaluated the comprehension of such VAs, revealing a lack of understanding in varying U-MAP approaches. Our research not only identifies key limitations, but offers valuable insights into how VAs can be further improved to manage complex authorization rules and adapt to dynamic changes.

1 INTRODUCTION

Virtual Assistants (VAs) powered by Large Language Models (LLMs) [3, 16, 19] are becoming trendy, as they provide a highly convenient human interface, suitable for many different scenarios and application domains such as Smart Homes, in which they aid in controlling smart devices (TVs, Lights, Locks) [10], Smart Cars, in which they can be useful for giving driving directions and controlling functionality via voice commands [14], as well as in Electronic Health Records (EHR), as they can provide patient information in an expedite and efficient manner during mission-critical operations such as surgery or first-response emergencies [17]. Not surprisingly, several different commercial products are already in the market, either in a *hardware-based* mode, i.e., Amazon Alexa and Google Nest, as well as in an *online-based* approach, i.e., Apple Siri, ChatGPT, Google Gemini, Microsoft Copilot, etc. As of today, several other companies are actively working towards providing dedicated, efficient VAs for a variety of application domains [1].

In such a context, the management of User-Managed Access Control Policies (U-MAPs) [6], i.e., the rules governing access to *sensitive* data and functionality within computer systems, may be required in VA scenarios [12]. As an example, the correct specification, evaluation, and enforcement of U-MAPs may be crucial to mediate *who* is allowed to control devices (Smart Homes), *who* can give directions and change car settings (Smart Cars), and to mediate *who* can access private patient information in (EHRs). In these scenarios, not handling U-MAPs correctly can have serious consequences, e.g., thieves controlling a Smart Lock guarding the main door (Smart Homes), kids altering the course of action of a car and causing an accident (Smart Cars), and a surgeon missing important allergy information on a patient, causing unnecessary complications (EHRs).

However, despite the excitement of the possibilities of VAs for improving human-computer interactions, and the many different solutions that are becoming commercially available, it is not clear if publicly-available, general-purpose VAs can effectively and efficiently handle U-MAPs. More specifically, it is still unclear if the management of U-MAPs via VAs correctly assigns authorization privileges/rights, a.k.a., *permissions*, to protected resources, e.g., Smart Home devices and Smart Car functionality, thus potentially avoiding the introduction of security vulnerabilities otherwise, which could be exploited by malicious third parties to compromise the security of such systems. In addition, the current landscape of VAs provides no insights on whether the management of U-MAPs is convenient to humans, e.g., it avoids difficult interactions, delays in response, etc., while still preserving the security properties just discussed.

In order to address these concerns, this paper presents the very first exploratory investigation on whether current publicly-available and highly-popular VAs, namely, ChatGPT (Version GPT-4o), Google Gemini (Version 2024.09.04), Microsoft Copilot (Version 10.28), and High-Flyer Deepseek (Version 2025.01.20) can manage U-MAPs effectively across differing scenarios. Specifically, our study seeks to answer the following research questions:

- **RQ1**: *Can VAs handle U-MAPs effectively from a security perspective?*
- **RQ2**: *What recommendations can be given to future VAs to effectively and efficiently handle U-MAPs?*

Overall, this paper provides the following contributions:

- As a part of Sec. 4, we present a series of specification formats developed to better communicate U-MAPs to VAs, e.g., using plain natural language versus more *structured* formats, detailing their syntactic and semantic contents, as well as their theoretical background.

- Also, in Sec. 5, we present the results of an exploratory study assessing the performance of four publicly-available VAs when handling U-MAPs. Our results indicate that the VAs under study exhibit varying degrees of proficiency depending on the format used to communicate each U-MAP, e.g., certain VAs demonstrate strong performance with structured U-MAPs, but show limitations with natural language inputs. Conversely, other VAs perform well across a range of U-MAP formats, though specific areas require refinement.
- Finally, in Sec. 5, we provide a series of recommendations for enhancing the performance of VAs when managing U-MAPs. As an example, due to struggles with unstructured U-MAPs, integrating advanced natural language processing techniques could refine and assist in contextual understanding. Enhanced contextual understanding could empower VAs to better navigate inputs, thus improving their versatility and reliability.

This paper is organized as follows: we start in Sec. 2 by reviewing some relevant background and related work. Then, we dive into the problem statement we address in this paper in Sec. 3. We then move on to present the methodology we follow for our study in Sec. 4, and presents its results and subsequent recommendations in Sec. 5. Sec. 6 discusses the limitations of our work. Finally, Sec. 7 concludes this paper with some interesting topics for future work.

2 BACKGROUND AND RELATED WORK

We start by presenting some basic background on topics that we will further explore later. In Sec. 2.1 we explore the uprising of VAs as well as latest developments relating LLMs and security. Also, in Sec. 2.2, we explore a definition of U-MAPs and their relevance for handling emerging technologies.

2.1 LLM-based Virtual Assistants

Due to their remarkable performance, Large Language Models (LLMs) are rapidly gaining traction across a multitude of diverse domains, from finance and health care to education and software development [3, 5, 16, 19, 24, 29]. In particular, Virtual Assistants (VAs) leverage LLMs as their foundational model for the processing and retrieval of general and domain-specific knowledge, and can be further augmented with additional data processing and storage capabilities by pairing their LLM-based backend with traditional databases, ontologies, knowledge graphs, etc. [9], thus making them ideal for convenient user interfaces, through which emerging smart technologies such as Smart Homes, Smart Cars, and EHRs, can be operated naturally and intuitively.

However, despite the growing interest and emerging applications, the use of LLMs as a back-end module for VAs can pose security vulnerabilities, leading to undesirable outcomes for the users of these technologies. As an example, Yao et al. categorized LLMs's security and privacy vulnerabilities into five categories: hardware-level attacks, OS-level attacks, software-level attacks, network-level attacks, and user-level attacks [30]. Pearce et al. investigated GitHub Copilot's security vulnerability in computer code generation and reported that 40% of the 1,689 programs generated by Copilot in their experiment introduced security vulnerability [21]. Iqbal et al. discussed security, privacy, and safety challenges related to LLMs integration with third-party plugins. The challenges include injecting malicious descriptions, diverting prompts to another plugin, and stilling plugin data [15]. Although prior research has evaluated LLMs for security vulnerabilities and data privacy, there is no systematic examination of LLMs' ability to interpret U-MAPs necessary to operate smart technologies effectively.

2.2 User-Managed Access Control Policies

For the purposes of this paper, we define User-Managed Access Control Policies (U-MAPs) to be the set of access control/authorization policies whose *lifecycle*, i.e., their specification, update, and removal over time, is handled mostly by *end-users*, who may have not received formal / informal training in cybersecurity/ access control topics [23]. End-users may also not receive dedicated advised from cybersecurity/access control experts when it comes to handling the lifecycle of U-MAPs, and may instead resort to a combination of domain-specific knowledge and common sense.

The correct management of U-MAPs may become central to the development and adoption of modern emerging technologies, which are fading away from expert-managed, *centralized* approaches, e.g., a single security officer handling all access control policies within an organization, to more *distributed* approaches in which end-users are given full control of the way they interact with a given technology, restricting access to both functionality and data at will. In such a scenario, U-MAPs are expected to provide a balance between *domain-specific* settings, e.g., conveniently restricting access to functionality, and *security-specifi*c settings, e.g., the collection and retrieval of data. Table 1 provides an example of a U-MAP handling a Smart Home environment consisting of a Smart Lock, a Smart TV, and a set of Smart Lights, all of them interconnected via an Internet of Things (IoT) setting, and controlled, e.g., access mediated, by means of a VA. In such a scenario, the U-MAP, stated in Natural Language, e.g., English, restricts access to each smart device to only a subset of users, identified by their *role*, e.g., *Homeowner*, *Partner*, etc., thus following an approach inspired by Role-Based Access Control (RBAC) [22], a well-known access control methodology.

For the purposes of this paper, we will consider a subset of the eXtensible Access Control Markup Language (XACML) [26], the *de facto* standard language for authorization/access control, in an attempt to provide a consistent foundation for the U-MAPs considered within our study, as it will be further described in Sec. 4. The U-MAP shown in Table 1 can be expressed in a subset of XACML which depicts a simplified syntactic structure in which U-MAPs are composed of *rules* consisting of a *target*, i.e., the resource whose access is mediated, a *condition* restricting under what circumstances access is to be allowed or denied, i.e., if the requester has a certain role, and a *rule decision*, which states if access is either allowed or denied in case the *condition* component is found to be true. Also, our subset of XACML includes only a single rule combining algorithm, i.e., *First Applicable*, which states that the overall result of evaluating the U-MAP against an access request will be the result of the first rule, starting from top to bottom, whose target is *equal* to the request's target, and whose condition is evaluated as true.

3 PROBLEM STATEMENT

As mentioned in Sec.2, VAs may offer significant convenience by allowing users to interact with emerging technologies through voice or text commands. However, integrating VAs into environments that handle sensitive data and functionality introduces notable security challenges, particularly concerning the specification and enforcement of U-MAPs, which are essential for ensuring that unauthorized individuals do not compromise security [18]. The challenge lies in ensuring that U-MAPs, as understood by the VAs, are robust enough to prevent security breaches, while maintaining usability. More specifically, in terms of *effectiveness*, the management of U-MAPs must correctly assign authorization privileges/rights, a.k.a., permissions, while avoiding the introduction of security vulnerabilities. Conversely, in terms of *efficiency*, the management of U-MAPs must be convenient to the end-user, e.g., it avoids difficult interactions, delivering incomplete information, etc.

With that in mind, this paper aims to evaluate the effectiveness of current VAs in handling U-MAPs across different scenarios, providing insights into their limitations and proposing new strategies for improving the secure management of sensitive data and functionality, such that future VAs can be designed to manage security while maintaining ease-of-use. Concretely, we consider the following research questions:

- **RQ1**: *Can VAs handle U-MAPs effectively from a security perspective?*
- **RQ2**: *What recommendations can be given to future VAs to effectively and efficiently handle U-MAPs?*

To further illustrate our problem statement, please consider the U-MAP described as a part of Sec. 2, also featured in Table 1. In such a scenario, VAs should be not only able to understand and enumerate the aforementioned U-MAP, but also to correctly answer to natural language questions representing access requests, which will subsequently trigger the evaluation of the U-MAP as described in Sec. 2. Such a task has interesting *usability* and *security* implications. For instance, access requests similar to the following: *Can Kids watch TV?* may certainly have a considerable impact in the overall usability of the whole Smart Home if they are processed incorrectly by the VA, e.g., Lights not working for the *Homeowner*, which represents a noticeable inconvenience. However, the security implications may not be relevant in this case, as access to any of those two devices may not be considered as highly *sensitive* for physical security purposes. On the other hand, questions similar to the following: *Can Visitors manipulate the Smart Lock?* can have non-trivial security consequences, as allowing for *Visitors* to access a security sensitive device such as a Smart Lock can certainly represent a serious risk. VAs not evaluating U-MAPs correctly, and therefore, granting unintended access as a result, may introduce a serious security vulnerability, which could be later exploited by malicious parties.

Table 1: Approaches for Encoding U-MAPs for Virtual Assistants (Smart Homes)

Approach	U-MAP
XACML **(Formal)**	<policy, CA=First-Applicable> <rule result=Allow> <target>Lock</target> <cond.>role=Homeowner</cond.> </rule> <rule result=Deny> ...</rule> </policy>
Informal **(INF)**	Only homeowners and partners are allowed to use the Lock, everybody is allowed to use the TV, everybody is allowed to use the Lights. Everything else is denied.
Modified Informal **(Mod-INF)**	Partners cannot use the lock, but visitors can use the lock now.
Semi-Formal **(SEMI-UNROLL)**	1. Homeowner can access Lock, 2. Homeowners can access Lights, ...10. Deny access to everything else.
Modified Semi-Formal **(Mod-SEMI-UNROLL)**	Remove Rule: Partner can access Lock. Create New Rule: Visitor can access Lock.
Semi-Formal-Rule-Based **(SEMI-RULE)**	1. if role = homeowner, Lock Access = allowed 2. if role = partner, Lock Access = allowed ... 5. Deny access to everything else.

Modified Semi-Formal-Rule-Based **(Mod-SEMI-RULE)**	if role = visitor: lock access = allowed,
	if role = partner: lock access = denied.

4 METHODOLOGY

To address our research questions, we conducted an exploratory study assessing the capabilities of four publicly available, general-purpose VAs: OpenAI ChatGPT [1] (Version GPT-4o), Google Gemini[2] (Version 2024.09.04), Microsoft Copilot[3] (Version 10.28), and High-Flyer Deepseek[4] (Version 2025.01.20) on tasks related to the specification and evaluation of a series of U-MAPs. The study began by selecting VAs that have gained significant popularity due to

[1] https://openai.com/chatgpt/
[2] https://gemini.google.com/app
[3] https://copilot.microsoft.com/
[4] https://www.deepseek.com

their recent advancements, accessibility, and relevance in general-purpose AI tasks. Next, we established the evaluation domains, providing a rationale for their selection. We then developed a series of U-MAPs, offering insight into the different policy formats used in the evaluation. Subsequently, we outline the methods for conducting interactive VA sessions, distinguishing between *Contextual* and *Non-Contextual* methods for prompting. Finally, we analyze VA interactions, concluding the experimental phase with an evaluation of the U-MAPs application within each VA.

4.1 Selecting VAs

ChatGPT, Gemini, Copilot, and Deepseek were chosen due to their broad user-base and frequent use, indicating that each VAs has likely accumulated knowledge from a wide variety of user interactions over its lifetime. These assistants are likely to have exposure to security-related questions and concepts making each of these VAs ideal candidates for evaluating, despite not being exclusively tailored to security tasks.

4.1.1 OpenAI ChatGPT. ChatGPT, developed by OpenAI, is one of the most widely used conversational AI models. The latest version, GPT-4o, is designed to handle a broad range of tasks, including answering questions. ChatGPT has been trained on a vast dataset, enabling it to generate detailed and coherent responses across various domains. The model employs Reinforcement Learning from Human Feedback (RLHF), where developers provide desired outputs to guide the model, enhancing the structure of responses ideal for this study [4]. While ChatGPT does not learn from real-time user interactions, its extensive pre-trained knowledge base includes information about access control models, though inconsistencies or challenges may arise due to incorrect yet plausible responses [4].

4.1.2 Google Gemini. Gemini, developed by Google DeepMind, is a virtual assistant that integrates advanced reasoning with internet retrieval, providing an advantage when handling up-to-date queries. It leverages Google's powerful machine learning models, which have been trained on diverse datasets across various domains [13]. Its multi-modal processing capabilities enable it to analyze and adapt to various formats, including those in this study. However, its reliance on web-sourced data introduces potential inconsistencies, as it may provide responses based on publicly available but unverified information [13].

4.1.3 Microsoft Copilot. Copilot, developed by Microsoft, is a generative AI designed to assist with daily tasks across various platforms, including Office and GitHub, though this study evaluated with the general-use version to maintain consistency [8]. Built on the ChatGPT 4o model, Copilot is optimized for productivity and contains integration with Microsoft's security and compliance tools, making it particularly relevant for tasks involving access control and security policy interpretation in various environments [20]. Although primarily focused on other uses, Copilot was included to evaluate security-handling capabilities, offering insights into its application beyond general productivity tasks.

4.1.4 High-Flyer Deepseek. Deepseek, developed by High-Flyer, is an emerging VA with a focus on reasoning and structured data processing, using a Mixture-of-Experts (MoE) language model [25]. While it is less documented compared to the other VAs in this study, its emphasis on logic makes it an interesting candidate for evaluating security policy interpretation [25]. Deepseek may be able to process structured access control rules more effectively than general-purpose assistants, but its performance on natural language-based security policies is less certain. Given its relatively recent development, it is unclear how well it aligns with practices in cybersecurity or whether it introduces reasoning inconsistencies when handling access control scenarios.

4.2 Determining Application Domains

We selected three application domains where VAs could significantly enhance user experience through natural language interactions. These include Smart Homes, where VAs can manage IoT devices for seamless control [10], Smart Cars, where VAs can assist with tasks like navigation and comfort settings [2, 14], and EHRs, where VAs can support quick and accurate information retrieval in critical situations [11, 17].

4.2.1 Electronic Health Records Domain. EHR systems are well-established and have become a fundamental part of modern healthcare practices. They are primarily used by healthcare professionals to store, retrieve, and update patient records, making access control an integral component to ensure the protection of private information. Most EHR systems today rely on graphical user interfaces (GUIs), although voice-based systems are becoming integrated for tasks such as searching patient records or dictating notes [17]. Accuracy is another critical component in EHR systems, due to the severe or life-threatening consequences that could occur due to errors in these systems. While response times can be important, it may not be as urgent unless an emergency situation arises, where it will then become crucial for immediate access. Beyond that, EHR systems need to maintain an effective management system, to not only protect privacy and maintain regulations, but to improve the overall efficiency and reliability of healthcare delivery. Previous solutions include RBAC, which is widely used in this domain, as it ensures that different roles (e.g. doctor, nurse) have appropriate levels of access to sensitive medical data. This study explores EHRs due to the widespread adoption and the critical nature of accurate information access.

4.2.2 Smart Home Domain. Smart Home technologies are becoming increasingly common, though they are not yet universally adopted in all households. These systems primarily use GUIs, but there is an increasing preference for voice control to enhance ease of use [10]. While accuracy in controlling these devices is integral, minor errors are typically not disruptive, unless they relate to security devices (e.g., cameras, locks). Slow responses, however, can reduce the overall effectiveness of the system, as users expect quick responses and actions from the systems. While management is significant, poor management implementation will cause a smart home to quickly lose effectiveness, reliability, and security. Similarly to EHRs, RBAC is also a model that could be implemented, though issues with complexities may arise due to the specific needs of users (e.g., home owners, family members) and the specific devices (e.g. thermostats, lights). Therefore, the smart home domain was selected due to the growing popularity and the potential streamlining that VAs can bring to the domain.

4.2.3 Smart Car Domain. Connected Automated Vehicles (CAVs) are an emerging technology, that rely on voice control for tasks such as navigation or adjusting in-car settings (e.g. AC, radio). While some cars may have visual systems (i.e., multimedia receivers), the use of GUIs within the CAV context is not practical due to safety concerns. In terms of accuracy, critical vehicle functions could have serious consequences if errors were to occur due to incorrect policy applications. Response time is another essential component, as delays in control in a real-time environment could lead to safety risks beyond those in the CAVs. All of these components need to be effectively handled by a robust management system, ensuring that security protocols are followed to keep the system reliable and safe. Though RBAC is a possible solution, this model may not be ideal due to necessary flexibility that stems from the various users (e.g. driver, passenger) and dynamic contexts (e.g., autonomous vs. manual driving). The addition of the smart car domain was valuable for this study as the domain is still in its development stages, allowing an evaluation on the VAs ability to handle U-MAPs, with little to no background knowledge.

4.3 Establishing U-MAPs

In generating format-specific U-MAPs, this study employs a method that produced multiple structures of instructions from the same policy. This approach was used to evaluate VA capabilities across different formats, ranging from an informal, natural language format to a structured, formal code-like approach. Specifically, three formats were used: informal (natural language), semi-formal (concise statement-based), and semi-formal-rule-based (structured, code-like). Detailed descriptions of each U-MAP format, along with the steps taken to produce the U-MAPs are provided in the following sections. Table 1, as well as Tables 3 and 4 (listed as a part of Appendix A), provide a full listing of the U-MAPs.

4.3.1 Informal (INF). Developed as an initial format, is intended to exercise the capabilities of VAs for handling moderately descriptive U-MAPs in natural language. It contains different statement explaining the U-MAP rules, which in turn are composed of descriptions of roles, protected resources (targets), and quantity pronouns, i.e., everybody.

4.3.2 Formal (XACML). Developed from the INF format, leverages the subset of XACML in an effort to provide a formalization of each U-MAP from which other formats can be developed from. This format was not used directly in the procedures described here.

4.3.3 Modified Informal (Mod-INF). Developed from the INF format, the modified informal structure is intended to exercise the capabilities of VAs for handling reduced descriptions of U-MAPs in natural language. It contains a much brief statement describing the U-MAP rules using the same constructs as before.

4.3.4 Semi-Formal (SEMI-UNROLL). Developed from the XACML format by syntactically *unrolling* each XACML rule into a natural language one, in an effort to assess the capabilities of VAs for understanding U-MAP that are described as sequences of enumerated statements composed of a limited sequence of language constructs.

4.3.5 Modified Semi-Formal(Mod-SEMI-UNROLL). Developed from the SEMI-UNROLL format, adds and/or removes certain U-MAP rules from the original set, in an effort to assess the capabilities of VAs for handling dynamic updates in U-MAPs, which may constitute changes in the authorization decisions when the U-MAP is re-evaluated with respect to the one in the SEMI-UNROLL format.

4.3.6 Semi-Formal-Rule-Based (SEMI-RULE). Developed from the XACML format, provides a shorter version of the unrolling with respect to SEMI-UNROLL using a stricter natural language syntax in a rule form, in an effort to provide a more concise description of U-MAPs for evaluation purposes.

4.3.7 Modified Formal (Mod-SEMI-RULE). Finally, this format was developed from the SEMI-RULE format by *compressing* several rules into a shorter statement, in an effort to assess the capabilities of VAs for handling succinct U-MAP rule-based descriptions that may lack extended explanations.

4.4 Constructing Inquiries

To assess the ability of the VAs to interpret and apply U-MAPs, we designed a set of structured questions aimed at evaluating each VAs capabilities and limitations, which are visible in Table 2. The questions were constructed to test two primary aspects of the VA performance: policy retrieval accuracy, which measures how well the VA extracts and applies explicit rules from a given policy, and logical consistency, which evaluates whether the VA demonstrates any underlying rationale rather than arbitrary responses.

Each of these questions were developed to be straightforward and directly answerable based on the provided U-MAP within each of the application domains. For example, questions such as *"Can partners manipulate the Lock?"*, or *"Can visitors watch TV?"*, were used to determine whether the VA accurately retrieved and applied the U-MAPs rules. In addition to the U-MAPs rules, we tested the VAs ability to handle implicit reasoning and ambiguous cases by including questions where the policy did not directly state an answer. For instance, if a policy specified that *" Only Admin Staff is allowed to access Personal Identifiable Information (PII) Data."*, we prompted the VA inquiring if nurses could access PII Data. These cases required the VA to infer whether such roles fell under the definition of "admin staff" based on the given context.

While responses were primarily recorded as *true* or *false*, additional notes were taken for instances where a VA struggled to provide clear reasoning or justification for its answer. The results were analyzed based on correctness and reasoning quality. Correctness was determined by comparing the VAs response to the explicit policy statement, while reasoning quality was noted in cases where the VA deviated from the correct answer or provided an unclear justification.

4.5 Conducting Interactive Sessions

To evaluate the capabilities of VAs in interpreting and applying U-MAPs, we designed two distinct methods: *Contextual* and *Non-Contextual* methods. These approaches were developed to assess

Table 2: Access Request Questions for Three Different U-MAP Scenarios

Smart Homes		Smart Cars		EHRs	
ID	**Question**	**ID**	**Question**	**ID**	**Question**
Q-SM-1	*Can Visitors manipulate the Lock?*	Q-SC-1	*Can Drivers give Directions?*	Q-EHR-1	*Can Physicians access Medical Data?*
Q-SM-2	*Can Visitors watch TV?*	Q-SC-2	*Can Co-Pilots give Directions?*	Q-EHR-2	*Can Staff access Medical Data?*
Q-SM-3	*Can Homeowner turn on the Lights?*	Q-SC-3	*Can Passengers set the Radio?*	Q-EHR-3	*Can Nurses access PII-Data?*
Q-SM-4	*Can Partner manipulate the Lock?*	Q-SC-4	*Can Drivers set the Radio?*	Q-EHR-4	*Can Staff access PII-Data?*
Q-SM-5	*Can Homeowner access a Meter?*	Q-SC-5	*Can Kids give Directions?*	Q-EHR-5	*Can First Responders access PII-Data?*

whether prior contextual information influences a VAs ability to retrieve and apply policy information accurately. The Contextual Method simulated scenarios where VAs have domain-specific knowledge before policy evaluation, while the Non-Contextual Method assesses how well a VA performs when given policies without any prior contextual grounding. By comparing these methods, we aim to determine the extent to which contextual grounding affects policy comprehension and retrieval accuracy.

The interactive sessions were conducted during two periods: the first two weeks of September 2024 and the first two weeks of March 2025. Three research team members with moderate training in U-MAP theory and VA-related technologies interacted independently with each VA. The session structure is shown in Figure 1. Afterward, two additional team members – one expert in U-MAPs and one expert in VA technologies – collaborated to compile and compare the results against ground truth, ultimately deriving the recommendations to be presented in Sec. 5.

4.5.1 Contextual Method. In the Contextual method, VAs were provided with background information before being presented with U-MAP-related queries. This step aimed to simulate a scenario where the VA had access to domain-specific knowledge that could aid in reasoning about policies. The process included:

(1) Prompting the VA to explain RBAC, based on its pre-existing knowledge and training data, including definitions and examples.
(2) Asking the VA about application domains (e.g., Smart Homes or Smart Cars) and relevant security considerations.
(3) Requesting the VA to generate a sample RBAC-like U-MAP, from instructions, specific to the given domain.
(4) The generated U-MAP was evaluated using predefined questions from Table 2, with responses compared against ground truth values.
(5) Each response was recorded, including any assumptions, inconsistencies, or deviations.
(6) After recording the results, provide the VA with the modified rules and repeat the process.
(7) Once the modified rules were recorded, the session was closed and memory was cleared.

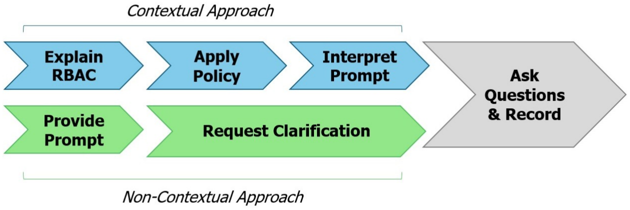

Figure 1: Contextual and Non-Contextual Approach Steps for VA Interaction Described in Sec. 4.5.1 and Sec. 4.5.2.

4.5.2 Non-Contextual Method. In contrast, the Non-Contextual Method involved directly presenting the U-MAP within a fresh chat session, without any background information, ensuring that responses relied solely on its pre-existing knowledge. The VA received the policy statement and was immediately tested on its ability to apply the rules correctly. The procedure followed:

(1) Provide the U-MAP to the VA within in a new chat session.
(2) Evaluate the VA by prompting the predefined questions from Table 2, with responses compared against ground truth values.
(3) Record responses and in some cases, VAs responded with apologies when questioned about the incorrect responses [28].
(4) After recording the results, provide the VA with the modified rules and repeat the process.
(5) Once the modified rules were recorded, the session was closed and memory was cleared.

5 RESULTS

The evaluation of Gemini, ChatGPT, Copilot, and Deepseek revealed distinct performance characteristics across various domains. The VAs were assessed in five key areas: **Effectiveness** (Sec. 5.1), **Consistency** (Sec 5.2), **Perception** (Sec. 5.3), **Reasoning** (Sec. 5.4), and **Usability** (Sec. 5.5). While each VA had strengths and weaknesses, all struggled with inference-based tasks and understanding default permissions, highlighting areas for improvement, though longer interactions generally led to performance gains. Across domains, VAs faced challenges in accurately retrieving and applying U-MAPs, with some performed better in specific formats. Sections 5.1 to 5.5 are then focused on providing answers to **RQ1**, whereas Sec. 5.6 provides a set of interesting **Recommendations** to answer **RQ2**.

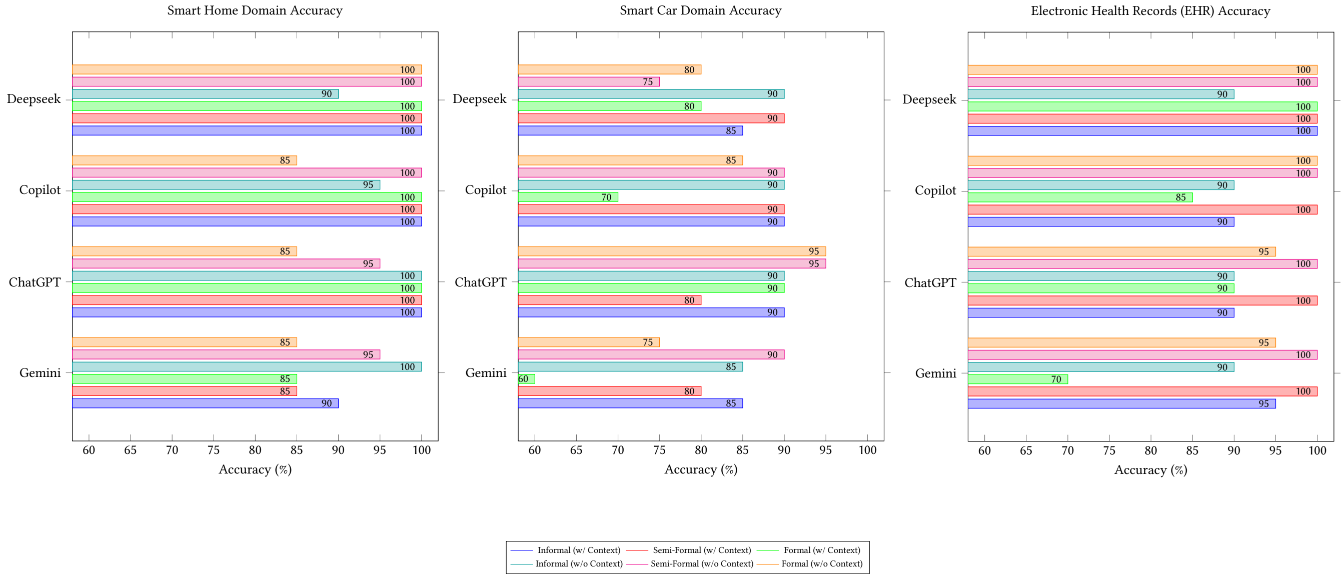

Figure 2: Overall Session Accuracy for Specific Domains (Smart Home, Smart Car, Electronic Health Records) - With and Without Context.

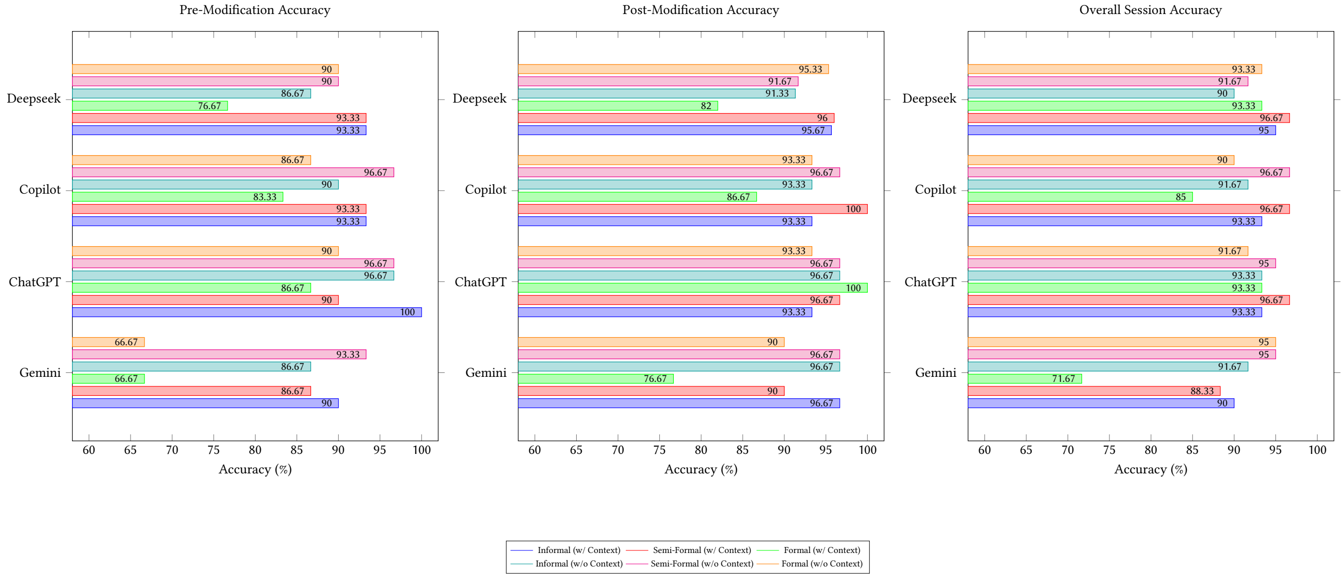

Figure 3: Overall Session Accuracy, Pre-Modified Session Accuracy, & Post-Modified Session Accuracy for All Scenarios(Smart Home, Smart Car, Electronic Health Records) - With and Without Context.

5.1 Effectiveness

Effectiveness measures how well a VA can apply U-MAPs correctly with pre-defined rules or ground truth values. An important and primary function of VAs in this context is to make decisions based on security policies, as incorrect decisions can cause the system and users to be at risk. This study aims to evaluate the overall effectiveness of VAs, which directly impacts their security.

ChatGPT performs well across all formats and methods. Accuracy ranges from 91.67% to 96.67%, with the lowest at 80% in the Smart Car domain and the highest at 100% across various others, as seen in Figure 2. This consistent accuracy positions ChatGPT as a strong contender for future VA advancements.

Gemini shows promise, specifically in usability, but lags behind other VAs in overall accuracy, with scores ranging from 71.67% to 95%. Its lowest score is 60% in the Smart Car domain, while it achieves 100% in other domains. These mixed results suggest areas for improvement, especially in the pre-training process.

Copilot, built on ChatGPT, ranges from 85% to 96.67% in accuracy, but drops to 70% in the Smart Car domain. While it performs similarly to ChatGPT, it may not be as robust for handling U-MAPs, especially when compared to other specialized VAs like Deepseek.

Deepseek, an emerging VA, demonstrates strong performance with accuracies ranging from 90% to 96.6%. This reasoning-based model allows it to produce results comparable to ChatGPT, with a low of 75% and a high of 100% across various domains. Deepseek and ChatGPT are among the top performers for applying U-MAPs, making them contenders for VA and security applications.

5.2 Consistency

Consistency is a critical measure of VAs reliability in recalling and applying U-MAPs in repeated interactions. Inconsistent behavior can lead to confusion or misinterpretation, particularly in scenarios where access control rules must be applied uniformly. In real-world applications, security policies must be consistently enforced to maintain integrity and prevent vulnerabilities. Notably, longer interaction sessions generally led to improved consistency, suggesting that extended engagement allows VAs to refine their responses and better adhere to previously established policies. As shown in Figure 3, the improvement in post-modification accuracy highlights the benefits of restating rules, which enables the system to better retrieve and apply information.

ChatGPT, like most VAs, showed a steady increase in accuracy as sessions progressed, even when re-prompted with information that adjusted the original rules. Starting with an average accuracy of 93.34% in the pre-modification phase, ChatGPT achieved an average accuracy of 96.11% in the post-modification phase.

Gemini, though scoring low in pre-modification, with an average accuracy of 81.67%, was able to redeem its score through the longer interactive sessions. Once the rules were restated and modified, Gemini was able to increase its average accuracy to 91.11%, though that is one of the lowest of the post-modification scores when compared.

Copilot, though able to improve its average accuracy in the post-modification phase, had the lowest pre-modification score at an average of 90.56%. This suggests that longer interactions may be necessary for information retention. After rule modifications and restatements, Copilot's accuracy increased by three percent, reaching an average of 93.89%.

Deepseek, despite being an emerging VA, surprisingly had the second-lowest pre-modification average accuracy at 88.33%, suggesting that longer sessions may be necessary for establishing a clearer U-MAP. Similar to Copilot, Deepseek improved in the post-modification phase, increasing its average accuracy to 92%.

5.3 Perception

Perception, in the context of this study, refers to how the instructions are formatted and the accuracy of interpretation from each VA. Each format, ranging from informal, natural language to semi-formal, rule-based, code-like structures, will indicate varying level of understanding that are dependent on the pre-training process. Some VAs were able to produce results across all formats, while others excelled or struggled with specific formats. Perception aims to understand the best format to provide U-MAPs for higher accuracy.

Each VA (ChatGPT, Gemini, Copilot, and Deepseek) achieved relatively high accuracy when handling the *Informal* format, with scores ranging from 86.67% to 100%. This indicates that most VAs effectively process natural language requests. While the majority performed well, as seen in Table 5, Deepseek consistently produced lower results in this format, which may be attributed to its pre-training process and how it interprets informal inputs. Since U-MAPs will likely be provided in natural language from users, ensuring strong performance in this format is crucial for real-world applications.

While the VAs performed well in the natural language (*Informal*) format, the *Semi-Formal* format yielded slightly higher scores, ranging from 88.33% to 100%. This difference may be attributed to the structured nature of U-MAPs in this format, where rules are explicitly stated with clear roles and conditions. Notably, each VA either matched or outperformed its informal format score, suggesting that the semi-formal structure may be more effective for providing VAs with specific rules to apply accurately.

While the VAs performed well with the *Informal* and *Semi-Formal* policy formats, the *Semi-Formal-Rule-Based* format resulted in a notable decline in accuracy. Scores ranged from 66.67% to 100%, with this wide gap likely attributed to the code-like structure of the U-MAP. Despite being trained on large-scale datasets, these models are primarily optimized for natural language processing and may have less exposure to structured, rule-based formats. However, ChatGPT and Deepseek performed significantly better, in comparison, in this format likely due to differences in the training processes.

5.4 Reasoning

Reasoning evaluates the logical structure of responses, particularly when it comes to justifying decisions based on U-MAPs. The ability to reason effectively in dynamic environments is essential where new or complex policies may need to be applied.

Each VA produced structured responses, highlighting either the potential for future reasoning capabilities or the current implementation of logical reasoning, as seen with Deepseek. While Deepseek benefits from built-in reasoning, other VAs struggled with inference-based questions. For instance, when asked, *"Can homeowners access a meter?"* a question relying on the U-MAP statement *"everything else is denied"*, the VA must infer that the meter falls under "everything else" and is therefore not accessible. Since the meter is not explicitly mentioned, some VAs failed to recognize this inference and incorrectly marked it as accessible.

Though inference was a consistent struggle across all domains, all VAs provided responses with relatively similar structures, with only a few exceeding the basic format. As expected, both Copilot and ChatGPT followed the same response pattern, which includes an immediate answer (i.e., *"No, the Homeowner cannot access a Meter"*), followed by a justification (i.e., *"Your policy states that everything else is denied unless explicitly allowed"*). These responses were generally brief and accurate, but in undetermined answer cases an additional apology was included, (i.e., *"I apologize for not understanding your policy"*), which inspired the title of this paper.

In comparison, Gemini uses a similar structure, but adds an additional component to the end, which includes a summary of the U-MAP. This approach allows the VA to restate the U-MAP without user prompting, possibly reinforcing the U-MAPs. However, as shown in Figures 2 and 3, reinforcing incorrect policies can negatively impact the accuracy of the VA (i.e., *"We cannot determine if a homeowner can access a meter"*).

Deepseek, following a similar structure to Gemini, includes an additional component in its responses by providing a suggestion. These suggestions vary depending on the U-MAP, but may include recommendations for resolving denied access issues or improving the overall model (i.e., *"Example modifications"*).

5.5 Usability

Usability measures how easy and efficient it is for users to interact with a VA. This includes the response time, quality of interaction, and any potential issues such as blocking or delays from the VA. While each VA offers varying levels of response times and message limits, each VA evaluation is necessary to understand the underlying potential solutions and limitations.

ChatGPT demonstrated consistent usability across various contexts. Its response times were relatively quick, averaging approximately 3.15 seconds across different scenarios. While response generation was immediate, we observed that ChatGPT imposed a message limit, particularly when multiple messages were sent within a one-hour period. We found that limit was triggered after approximately sixty to sixty-five messages were sent, by notifying the user that they *"reached our [ChatGPT] limit of messages per hour."* Notably, this threshold was recorded in September 2024, and with each new version release, there appears to be a corresponding increase in the number of messages allowed within the hour.

Gemini, another strong contender in the VA environment, demonstrated quick response times across various contexts. Like ChatGPT, Gemini delivered responses promptly, averaging approximately 3.21 seconds across the differing formats. Although response times were fast, Gemini also imposed a message limit when multiple messages were sent within a one-hour window. This message limit was triggered after approximately one hundred messages were sent, with users being notified that *"Gemini is on a break."*

Copilot, a VA built on ChatGPT, performed slightly better in response time but has a much smaller message threshold. It generates responses to prompts, whether questions or instructions, relatively quickly, averaging approximately 2.88 seconds. However, Copilot has a much smaller message limit, restricting entire chat sessions to only thirty messages before requiring the user to open a new chat. Additionally, Copilot imposes a message limit within the one-hour window similar to ChatGPT, typically around the sixty-message mark, informing users that it is *"time for a new topic"* or that the user has *"reached the limit of messages in one hour"*.

Deepseek, an emerging VA, incorporates reasoning into its base model, resulting in longer response times, as observed in the study. Although Deepseek provides detailed and clear responses, it averages a generation time of approximately 14 seconds, which is significantly longer than other VAs. This extended response time is likely correlated with the reasoning aspect of the model. Additionally, Deepseek's message threshold is smaller but not implemented in the same way as Copilot, Gemini, or ChatGPT. Deepseek imposes a temporary barrier after approximately twenty messages, requiring users to pause for five to ten minutes before they can send messages again by notifying the user that there is a *"temporary server error"*. This temporary restriction occurs three times, after which the user is blocked for the remainder of the hour by notifying that the user has *"reached the message limit for the hour"*.

5.6 Recommendations

Each of these previously mentioned components (effectiveness, consistency, perception, reasoning, usability) highlight the strengths and limitations of each VA. The results from these sections point to a key underlying issue of a lack of inference and logical structure. At the time of testing, most models lacked a dedicated reasoning framework. As a result, incorrect responses often stemmed from prompts requiring inference whether it was related to specific devices or roles. Beyond that, pre-training information also plays a part in the accuracy of these models, as domain results are representative of how widespread the domain is.

5.6.1 Implementing Reasoning into VAs.

In order to address the lack of reasoning in models, one notable solution is to implement a form of reasoning. As demonstrated by Deepseek, this approach can significantly enhance the quality of responses. However, it may lead to a trade-off in usability, specifically in terms of response time. To avoid compromising usability of VAs, it is crucial to incorporate reasoning without negatively impacting response speed. Deepseek, while incorporating reasoning, experiences much longer response times compared to other VAs. Although the models previously evaluated lacked reasoning integration, ChatGPT has since released a reasoning model which adds approximately four-to-six seconds to the average response time [27]. This makes its response time closer to Deepseek's but still retains a reasonable balance between logic and speed. These changes could improve the effectiveness, consistency, and clarity of each VA, allowing the VA to be utilized across various domains.

5.6.2 Importance of Pre-Training Information.

To enhance the quality of VAs responses, the training process plays a crucial role. Analyzing the results across various domains reveals that the EHR domain consistently achieved high accuracy. These scores may be attributed to the widespread use of access control within the healthcare field, which provides a solid foundation for training within this domain. On the other hand, the Smart Home domain scored decently across all VAs, but exhibited inference errors. Since smart homes are less widespread and not as extensively documented as EHRs, the lack of supporting documentation in the training process may point to the importance of background information. Specifically, for a VA to be integrated into a Smart Home domain, there may be necessary tailoring process to introduce specific access control concepts and applications. In contrast, Smart Cars face an even greater challenge. Not only is there limited documentation available when compared to EHRs, but the documentation on security, particularly access control, is scarce. This lack of security-related training materials may contribute to challenges VAs face when operating within the Smart Car domain. Each of these domains contribute to the overall conclusion that there is not enough supporting documentation of emerging technologies, as VAs are often updated with information at a later time.

5.6.3 Enhancing Real-Time Adaptability. A major limitation in current VAs is their inability to learn in real-time. Most VAs rely on pre-training data, meaning they cannot adapt to new information or refine their understanding based on user interactions. This is particularly problematic in domains that evolve quickly or lack comprehensive documentation, such as emerging technologies like Smart Cars. In these dynamic fields, the inability of VAs to update their knowledge can result in outdated or inaccurate responses, limiting their usefulness. Introducing real-time learning capabilities would enable VAs to continually improve and adjust their responses during interactions. This enhancement would make the VAs more dynamic and adaptable, offering more relevant responses while significantly improving accuracy and utility. Though, it is important to note that learning in real-time could allow unverified and incorrect information to be taught to the model, therefore, a real-time model would require a meticulous implementation.

6 LIMITATIONS

While this study provides valuable insights into the limitations and strengths of VAs, several limitations should be considered when interpreting the results.

6.0.1 Simplification of U-MAPs. This study, while exploring various domains, had U-MAPs that were based on simple case scenarios. While the U-MAPs were able to showcase underlying issues within the VAs, the policies were not exhaustive (e.g. no corner cases and/or complicated policies with combining algorithms). In order to address this limitation, a future study needs to be conducted to explore richer and more complicated U-MAPs that require multi-step decision-making. Further exploration into complex cases and U-MAPs would showcase the importance of reasoning within VAs, and how the usability will be impacted.

6.0.2 Narrow Range of Perspectives. Though the results were produced from few researchers, each of these researches that interacted with the VAs contained varying levels of expertise: expert, mid-level, and novice. In comparison, the end-users who may interact with the VAs will likely not have the same levels of expertise in security or access control methods. Therefore, it is worth noting that a future usability study will need to be conducted with participants who have limited security experience, to gauge how policies are understood and managed from an outside perspective.

6.0.3 Limitations of VAs. For this study, only a handful of the most popular VAs were chosen, as they would likely have the most human interaction, in turn, having the most training. Though these VAs would have the benefit of extensive training, only general-purpose, commercially available, non-specialized VAs were considered. In light of this limitation, future work for this study includes leveraging insight from the VAs strengths and weaknesses to develop a custom VA that focuses on handling U-MAPs consistently.

7 CONCLUSIONS AND FUTURE WORK

In this paper, we have presented an exploratory study to evaluate the effectiveness of the publicly-available VAs in managing U-MAPs for a set of representative domain case scenarios. Our results indicate the need to further customize VAs to effectively handle U-MAPs to prevent security vulnerabilities, while not infringing on the user experience. Overall, the VAs demonstrated varying results, where some excelled in formal approaches but declined in performance for informal ones. Each of the VAs showcased above-average comprehension in at least one of formatted U-MAP approaches, but also highlighted some possible limitations in other ones. Although these results are encouraging, their power is limited to the well-known RBAC approach. In the future, we plan to extend our investigation to other authorization approaches, i.e., Attribute-Based Access Control (ABAC) [7], which may be better suited for the representative domain scenarios we introduced for our study. Finally, we also plan to further implement our proposed recommendations in a series of custom-made VAs, which can effectively leverage these experience for enhanced security and usability performance.

ACKNOWLEDGMENTS

This work was partially supported by the National Science Foundation (NSF) under Grants No. 2232911 and No. 2131263, and by the CAHSI-Google Institutional Research Program, sponsored by Google Inc. and the Computing Alliance of Hispanic-Serving Institutions (CAHSI), a National Science Foundation (NSF) INCLUDES Alliance.

REFERENCES

[1] Android Authority. 2024. Siri vs Alexa vs Google Assistant vs Bixby: Which one reigns supreme? https://www.androidauthority.com/siri-vs-alexa-vs-google-assistant-vs-bixby-3192996/. [Online; accessed September-12-2024].

[2] Fabio Arena, Giovanni Pau, and Alessandro Severino. 2020. An overview on the current status and future perspectives of smart cars. *Infrastructures* 5, 7 (2020), 53.

[3] Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. 2024. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology* 15, 3 (2024), 1–45.

[4] OpenAI ChatGPT. 2022. Introducing ChatGPT. https://openai.com/index/chatgpt/

[5] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374* (2021).

[6] Chung, David Ferraiolo, and David Kuhn. 2006. Assessment of Access Control Systems. https://doi.org/10.6028/NIST.IR.7316.

[7] Chung, David Ferraiolo, David Kuhn, Adam Schnitzer, Kenneth Sandlin, Robert Miller, and Karen Scarfone. 2019. Guide to Attribute Based Access Control (ABAC) Definition and Considerations. https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=927500.

[8] Microsoft Copilot. 2025. Empower your organization with Copilot. https://www.microsoft.com/en-us/microsoft-copilot/organizations

[9] Xin Luna Dong, Seungwhan Moon, Yifan Ethan Xu, Kshitiz Malik, and Zhou Yu. 2023. Towards next-generation intelligent assistants leveraging llm techniques. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 5792–5793.

[10] Jide S Edu, Jose M Such, and Guillermo Suarez-Tangil. 2020. Smart home personal assistants: a security and privacy review. *ACM Computing Surveys (CSUR)* 53, 6 (2020), 1–36.

[11] R Scott Evans. 2016. Electronic health records: then, now, and in the future. *Yearbook of medical informatics* 25, S 01 (2016), S48–S61.

[12] Fortune. 2024. Apple, Google, and Amazon May Have Violated Your Privacy by Reviewing Digital Assistant Commands. https://fortune.com/2019/08/05/google-apple-amazon-digital-assistants/. [Online; accessed September-12-2024].

[13] Google Gemini. 2023. Introducing Gemini: our largest and most capable AI model. https://blog.google/technology/ai/google-gemini-ai/#sundar-note

[14] Jacopo Guanetti, Yeojun Kim, and Francesco Borrelli. 2018. Control of connected and automated vehicles: State of the art and future challenges. *Annual reviews in control* 45 (2018), 18–40.

[15] Umar Iqbal, Tadayoshi Kohno, and Franziska Roesner. 2023. LLM Platform Security: Applying a Systematic Evaluation Framework to OpenAI's ChatGPT Plugins. *arXiv preprint arXiv:2309.10254* (2023).

[16] Enkelejda Kasneci, Kathrin Seßler, Stefan Küchemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Urs Gasser, Georg Groh, Stephan Günnemann, Eyke Hüllermeier, et al. 2023. ChatGPT for good? On opportunities and challenges of large language models for education. *Learning and individual differences* 103 (2023), 102274.

[17] Yaa A Kumah-Crystal, Claude J Pirtle, Harrison M Whyte, Edward S Goode, Shilo H Anders, and Christoph U Lehmann. 2018. Electronic health record interactions through voice: a review. *Applied clinical informatics* 9, 03 (2018), 541–552.

[18] Song Liao, Christin Wilson, Long Cheng, Hongxin Hu, and Huixing Deng. 2020. Measuring the Effectiveness of Privacy Policies for Voice Assistant Applications. http://arxiv.org/abs/2007.14570

[19] Amarachi B Mbakwe, Ismini Lourentzou, Leo Anthony Celi, Oren J Mechanic, and Alon Dagan. 2023. ChatGPT passing USMLE shines a spotlight on the flaws of medical education. , e0000205 pages.

[20] Microsoft. 2024. Microsoft Trustworthy AI: Unlocking human potential starts with trust. https://blogs.microsoft.com/blog/2024/09/24/microsoft-trustworthy-ai-unlocking-human-potential-starts-with-trust/

[21] Hammond Pearce, Baleegh Ahmad, Benjamin Tan, Brendan Dolan-Gavitt, and Ramesh Karri. 2022. Asleep at the keyboard? assessing the security of github copilot's code contributions. In *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 754–768.

[22] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman. 1996. Role-Based Access Control Models. *Computer* 29, 2 (Feb. 1996), 38–47.

[23] D. K. Smetters and Nathan Good. 2009. How users use access control. In *Proceedings of the 5th Symposium on Usable Privacy and Security* (Mountain View, California, USA) *(SOUPS '09)*. Association for Computing Machinery, New York, NY, USA, Article 15, 12 pages. https://doi.org/10.1145/1572532.1572552

[24] Zhongxiang Sun. 2023. A short survey of viewing large language models in legal aspect. *arXiv preprint arXiv:2303.09136* (2023).

[25] DeepSeek-AI Research Team. 2025. DeepSeek-V3 Technical Report. (February 2025). https://arxiv.org/pdf/2412.19437

[26] Fatih Turkmen, Jerry den Hartog, Silvio Ranise, and Nicola Zannone. 2017. Formal analysis of XACML policies using SMT. *Computers & Security* 66 (2017), 185–203. https://doi.org/10.1016/j.cose.2017.01.009

[27] Chris Varner. 2025. Understanding Different ChatGPT Models: Key Details to Consider. https://teamai.com/blog/large-language-models-llms/understanding-different-chatgpt-models/

[28] Joel Wester, Tim Schrills, Henning Pohl, and Niels van Berkel. 2024. "As an AI language model, I cannot": Investigating LLM Denials of User Requests. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 1–14.

[29] Binwei Yao, Ming Jiang, Diyi Yang, and Junjie Hu. 2023. Benchmarking llm-based machine translation on cultural awareness. *arXiv preprint arXiv:2305.14328* (2023).

[30] Yifan Yao, Jinhao Duan, Kaidi Xu, Yuanfang Cai, Zhibo Sun, and Yue Zhang. 2024. A survey on large language model (llm) security and privacy: The good, the bad, and the ugly. *High-Confidence Computing* (2024), 100211.

8 APPENDIX A: USER-MANAGED ACCESS CONTROL POLICIES AND SAMPLE RESULTS

Table 3: Approaches for Encoding U-MAPs for Virtual Assistants (Smart Cars)

Approach	U-MAP
XACML (Formal)	<policy, CA=First-Applicable> <rule result=Deny> <target>Directions</target> <cond.>role=Co-Pilot</cond.> </rule> </policy>
Informal **(INF)**	*Only drivers should be allowed to give driving directions to the smart car. Only drivers and co-pilots should be allowed to modify smart car settings. Other passengers should be only allowed to change the radio settings.*
Modified Informal **(Mod-INF)**	*Kids can't give directions, but co-pilots can.*
Semi-Formal **(SEMI-UNROLL)**	1. Co-Pilots cannot give Directions, 2. Passengers cannot give Directions, ... 4. Everything else is Allowed.
Modified Semi-Formal **(Mod-SEMI-UNROLL)**	Create new Rule: Kids cannot give Directions; Remove Rule: Co-Pilots cannot give Directions

Semi-Formal-Rule-Based
(SEMI-RULE)
	if Role = Co-Pilot:
	Directions = denied,
	if Role = Passenger:
	Settings = denied
	& Directions = denied,
	Default Radio
	& Directions
	& Settings = approved

Modified Semi-Formal-Rule-Based
(Mod-SEMI-RULE)
	if Role = Kids:
	Directions = denied,
	if Role = Co-Pilot:
	Directions = approved

Table 4: Approaches for Encoding U-MAPs for Virtual Assistants (EHRs)

Approach	U-MAP
XACML (Formal)	<policy, CA=First-Applicable> <rule result=Allow> <target>Medical</target> <cond.>role=Physician</cond.> </rule> <rule result=Allow>...</rule> </policy>
Informal **(INF)**	*Only Physicians, Nurses, and First Responders are allowed to access Medical Data.* *Only Admin Staff is allowed to access Personal Identifiable Information (PII) Data.*
Modified Informal **(Mod-INF)**	*First Responders can now access PII-Data and Staff can now access Medical Data.*
Semi-Formal **(SEMI-UNROLL)**	1. Physicians can access Medical Data, 2. Nurses can access Medical Data, ... 9. Everything else is denied.
Modified Semi-Formal **(Mod-SEMI-UNROLL)**	Create new Rule: "First Responders can access PII- Data"; Remove Rule: "First Responders cannot access PII-Data"; Create new Rule: "Staff can access Medical Data"

Semi-Formal-Rule-Based
(SEMI-RULE)
	if role = Staff:
	PII-Data = Approve;
	if role = Physician, Nurse,
	First Responders:
	Medical Data = Approved;
	Default Access = Denied

Modified Semi-Formal-Rule-Based
(Mod-SEMI-RULE)
	if role = Staff, First Responders:
	PII-Data = Approve;
	if role = Physician, Nurse,
	First Responders, Staff:
	Medical Data = Approved;
	Default Access = Denied

Table 5: Sample Results of Evaluating Different U-MAPs on a Series of VAs (Smart Homes)

		ChatGPT				Google Gemini				Microsoft Copilot				Deepseek			
		Context		No Context		Context		No Context		Context		No Context		Context		No Context	
Question	**GT**	**R-1**	**R- 2**	**R-1**	**R-2**	**R-1**	**R-2**	**R-1**	**R-2**	**R-1**	**R-2**	**R-1**	**R-2**	**R-1**	**R-2**	**R-1**	**R-2**
INF Format																	
Q-SM-1	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Q-SM-2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Q-SM-3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Q-SM-4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Q-SM-5	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Modified INF Format																	
Q-SM-1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓
Q-SM-2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Q-SM-3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Q-SM-4	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✓	✗
Q-SM-5	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
SEMI-UNROLL Format																	
Q-SM-1	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Q-SM-2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Q-SM-3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Q-SM-4	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓
Q-SM-5	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Modified SEMI-UNROLL Format																	
Q-SM-1	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Q-SM-2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Q-SM-3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Q-SM-4	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Q-SM-5	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗
SEMI-RULE Format																	
Q-SM-1	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
Q-SM-2	✓	✓	✓	✓	✗	✓	✓	✓	✗	✓	✓	✓	✗	✗	✗	✓	✓
Q-SM-3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓
Q-SM-4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Q-SM-5	✗	✗	✗	✗	✗	✗	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗
Modified SEMI-RULE Format																	
Q-SM-1	✓	✓	✓	✗	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Q-SM-2	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✓
Q-SM-3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓
Q-SM-4	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Q-SM-5	✗	✗	✗	✗	✗	✗	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗