POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Voice-based Direct Manipulation in Intent-driven User Interfaces

Laurea Magistrale in Computer Science and Engineering - Ingegneria Informatica

Author: Laura Colazzo

Advisor: Prof. Maristella Matera

Academic year: 2024-2025

1. Introduction

Recent advancements in Generative AI have contributed to the emergence of a new interaction paradigm centered on intent formulation. Despite the exceptional popularity Large Language Models (LLMs) have recently gained and the unprecedented opportunities they have unlocked, ensuring the user intent is accurately captured by the model, from a prompt expressed in natural language, remains challenging, especially when prompt refinements must be applied iteratively. Usability problems are even more critical when situational or permanent disabilities demand voice-based interaction, due to the lack of appropriate interaction mechanisms. Despite the effort directed toward improving usability in LLM Graphical User Interfaces (GUIs), it remains unclear whether analogous results can be achieved in voice-only interfaces. Given the key role Voice-based User Interfaces (VUIs) play in accessibility, ensuring their usability is fundamental.

Within this context, the thesis addresses the following Research Questions (RQs):

RQ1. *Can direct manipulation principles inspire the definition of new vocal interaction patterns to achieve more expressive and direct vocal interactions in LLM-powered VUIs?*

RQ2. *How do users perceive the usability and cognitive load of the new voice-based paradigm and what is the level of acceptance?*

To answer these questions, it surveys emerging human-LLM interaction techniques based on *direct manipulation principles* [5]. Then, inspired by the new emerging interaction techniques proposed for LLM GUIs, and informed by a prior user study with Blind and Visually Impaired (BVI) individuals whose findings revealed new design opportunities in voice-based human-LLM interactions [4], the thesis investigates how direct manipulation principles can be extended to voice-based LLM interfaces to inspire the definition of new vocal interaction patterns. To prove the feasibility of the proposed vocal interaction paradigm, the thesis also introduces the architecture of a prototype used to conduct validation activities with 9 participants and presents the collected results.

2. Related Work

Although text-based prompting is currently the most common interaction mechanism in Generative AI systems [1], novel approaches are emerging. In addition to chat-based interfaces, there are tools featuring a *canvas area* at the center of the screen where the majority of interactions oc-

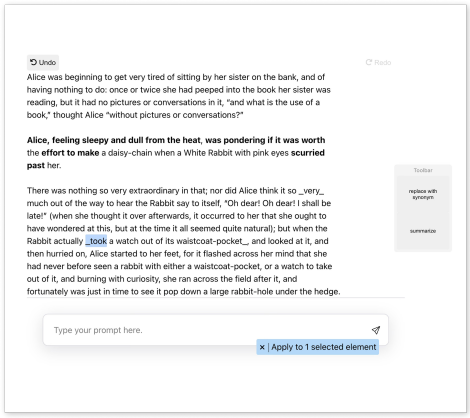

Figure 1: DirectGPT's UI [2]

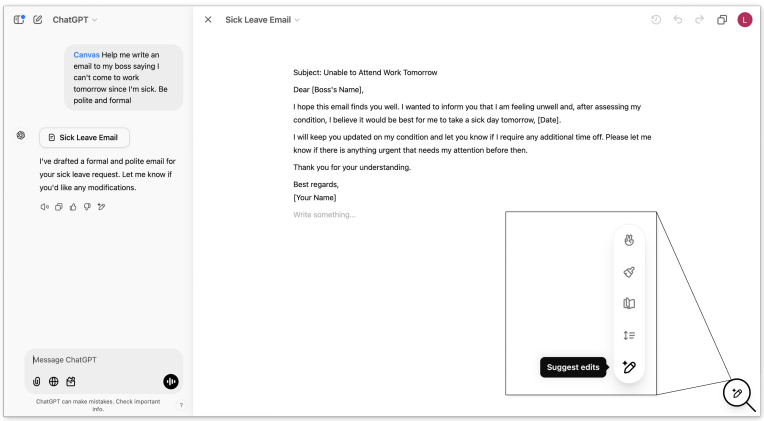

Figure 2: ChatGPT – Canvas' UI [3]

cur. In *Information visualization canvases*, then, elements inside the central area can be directly manipulated to alter the state of the system.

DirectGPT [2] (Figure 1) is an example of a system featuring an information visualization canvas interface [1], implemented on top of an LLM. In DirectGPT, users are allowed to build prompts by manipulating visual components inside the UI, using mechanisms such as multimodal prompting and multi-selection [1]. The authors have demonstrated how direct manipulation principles can be effectively combined with traditional text-based prompting strategies to streamline the interaction with LLMs and make it more direct, while overcoming some of the limitations of purely conversational UIs. These mechanisms are in particular effective for the iterative prompt refinement, for which users must be able to reference specific elements from the model's previous response, something purely-conversational interfaces do not allow. DirectGPT, as opposed to strictly conversational interfaces, builds upon the idea that to enable direct manipulation the objects of interest must be continuously accessible inside the UI. For this reason, the long history of messages is replaced by a fixed area at the center of the screen, where the model output is continuously displayed after each modification. Furthermore, the dynamic population of a toolbar of commands makes available the most recent user prompts in the form of buttons, fostering their re-usability. Increased directness is also achieved through physical actions, such as dragging or highlighting, augmenting the expressivity and the flexibility of purely textual prompts. Additionally, error recovery is achieved through undo and redo commands. The system also highlights the elements affected by the LLM-generated modifications, allowing users to quickly assess the effect of their prompts and immediately revert them if needed.

Content creation is one of the primary applications of Generative AI. As Luera et al. [1] point out in their survey, the most common UI layouts for AI-assisted content creation tasks, involving the generation or editing of visual, written or audio content, are conversational and canvas UIs, with the latter being more frequently adopted for the generation of visual content. However, this pattern is not absolute and canvas UIs have also been developed for AI-assisted writing tasks. An example is given by Canvas (Figure 2): a tool integrated in ChatGPT, introduced in October 2024 and designed to support users in their writing and coding projects, with a UI that goes beyond the traditional chat layout. Indeed, as emphasized in the article about its launch [3], working with a chat can be limiting for tasks that require editing and revisions. The tool, instead, introduces a canvas area, separate from the chat, that ChatGPT users can leverage to directly engage with the text or code to be edited, using direct manipulation mechanisms analogous to those available in DirectGPT. These features facilitate the collaboration between the user and ChatGPT, with greater flexibility compared to standard UIs used for the same tasks.

3. Approach and Prototype

The discussion presented in this section is informed by a user study [4] that aimed to explore the accessibility potential and challenges of voice interaction when accessing LLMs. The study revealed a range of design opportunities in voice interaction, emphasizing the need for an improved conversation management and for better ways to consume and manipulate of the content generated as output by the LLM.

In the following, a discussion is provided regarding the way each of the principles of *direct manipulation* might be extended to LLM-based VUIs. The analysis presented is inspired by the previously-discussed examples of canvas UIs and considers some well-known challenges of VUI design from the literature.

Continuous representation of the object of interest. The system could let the user hear the most updated version of the output—the LLM-generated text—aloud. In this way, the effect of user-requested commands could be immediately perceived, without the need for the user to explicitly ask to have the full text read out loud any time a new transformation is applied. However, while this mechanism could be effective when the model's response is relatively short, for more lengthy outputs, providing a summary of the applied changes might be more appropriate, while still preserving the chance for the user to listen to the full modified text upon request.

Physical actions or labeled button presses instead of complex syntax. While the definition of visual abstractions acting like shortcuts is relatively easy in GUIs, in voice interfaces it requires careful consideration. Such an abstraction could be implemented as pre-defined vocal commands that act as proxies for more complex and longer prompts. Identifying a voice-based version of gestures that in GUIs are naturally associated with a semantic meaning, such as a selection mechanism to restrict the context of a given prompt to a smaller portion of text, is instead more challenging. It requires the ability to physically point to a specific element of text—an action that does not have a direct equivalent in the voice domain. However, equipping the user with a fine-grained navigation mechanism, through which the text can be linearly explored and potentially modified along the way, may give users an analogous degree of expressiveness in crafting prompts through speech. By considering the limitations of human short-term memory, this pattern might give users the chance to focus on smaller and more manageable portions of text for an easier editing and review experience. Furthermore, with this navigation mechanism users could gain finer-grained control over the text-generation process, having access to an in-place tool for a more precise and accurate definition of intents that removes the need for extensive descriptions. However, to help user form a mental model of the text being explored, this method should be combined with a suitable *location awareness* strategy, such as one that explicitly assigns each navigation node a progressive numerical identifier.

Rapid, incremental, reversible operations whose impact on the object of interest is immediately visible. While reversibility can be easily achieved by introducing vocal commands that undo or redo recent modifications, performing incremental operations quickly is the most critical part of the interaction. This is because vocal commands can be relatively rapid to issue, but the efficiency with which the user is able to evaluate their effect on the system strongly depends on the way such actions are acknowledged. While assessing the impact of user requests is relatively easy when the target text can be visually inspected, in VUIs efficiency depends on the trade-offs discussed for the first principle.

3.1. Proposed Interaction

To assess the potential of voice-based direct manipulation mechanisms explored in the previous section, a new human-LLM vocal interaction paradigm is proposed. Being these mechanisms particularly advantageous in content creation tasks or, more in general, in any scenario that involves refining messages generated by the LLM through multiple iterations, the considered use case involves a user whose goal is to iteratively refine a message received by the model within an ongoing chat (e.g., the body of an LLM-generated email to adjust its tone and structure). Figure 3 presents the interaction flow, visually representing objects involved in the *vocal* interaction.

At first, the user selects a message from the chat as a target for editing actions. To support the

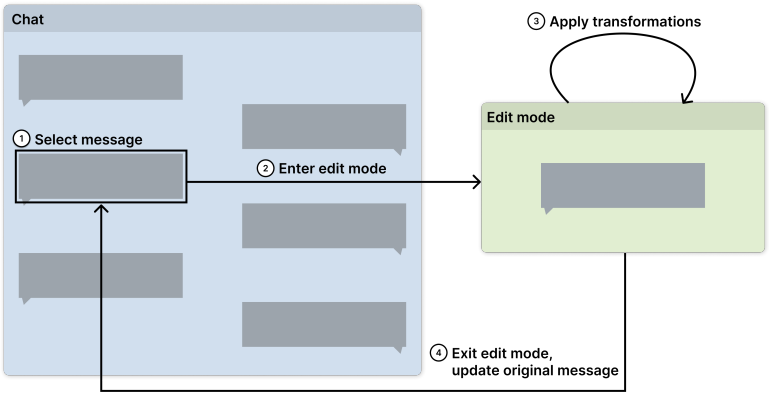

Figure 3: Interaction flow

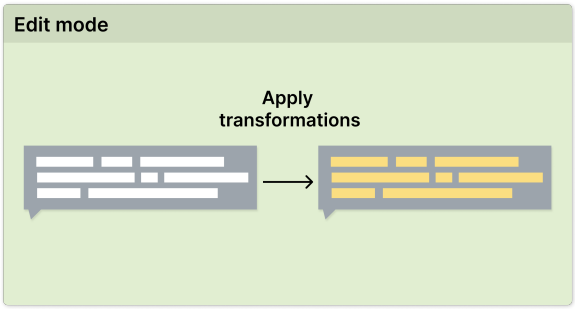

Figure 4: Global editing actions

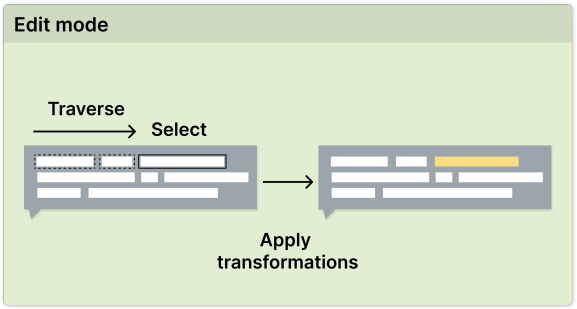

Figure 5: Localized editing actions

refinement process, an *edit mode* is introduced. This is conceptually analogous to the canvas area seen in some visual LLM interfaces and can be accessed using a dedicated vocal command (e.g., "*Enter edit mode*"). Being such a space virtually separated from the main chat with the model, it can be leveraged by the user to iteratively carry out a series of LLM-assisted transformations to the text, while keeping the main chat clean and concise. This can be particularly valuable if we consider that the previous study with BVI participants emphasized the need for users to revisit information stored in the chat more efficiently [4]. In light of these considerations, and given that in edit mode the exchange of messages between the user and the model only serves the purpose of producing a new, satisfactory version of the target, only the final output of such a procedure should persist to the main chat once the user is out of edit mode.

In edit mode, user requests to modify text can be classified into two categories: ***(i) Global editing actions*** (Figure 4): they affect the target message in its entirety (e.g., change its tone). Any time a request of this kind is issued to the LLM, the full modified message is read aloud. In the case of particularly long messages, instead, only a summary of the modifications applied is provided. ***(ii) Localized editing actions*** (Figure 5): they affect only a restricted portion of the message (e.g., replace a word with a synonym). To perform this class of actions in an unambiguous way, the user must be able to both *traverse* the message using a finer-grained and adjustable granularity (e.g. at sentence level) to reach the portion of interest, and to *select* it as target for transformations. This behaviour can be achieved through the previously-discussed navigation pattern, implemented as a *navigation mode* that can be activated using a vocal command (e.g., "*Enter navigation mode*"). Landing on a given node triggers a series of actions: the node's identifier is read aloud followed by its content; then, the user receives suggestions from the system for possible actions to take. At this stage, any editing request considers the node currently in focus as the target.

3.2. Prototype

Initially, a fast-prototyping approach based on OpenAI's custom GPTs was selected to translate the proposed interaction into a tangible ar-

tifact to carry out validation sessions with users. However, the transition to a more sophisticated, yet low-cost architecture to facilitate the next design iterations was deemed necessary to overcome the limitations faced with GPTs, especially in terms of scarce determinism, where a consistent behaviour was desirable.

The final prototype has a traditional client-server architecture. Communication between the two relies on WebSockets, ensuring low-latency real-time exchanges—essential in VUIs. The client is a React application and acts as the front-end of the system. To support the core functionalities of a VUI, it embeds: a Speech-To-Text module to convert spoken utterances in text-based requests to be sent to the server, and a Text-To-Speech module to convert the output of server-side computation in audible responses for the user. To let future participants focus on vocal manipulation mechanisms only, the final prototype's GUI was kept minimal, with a single push-to-talk button. Implemented in Python using Flask, the server instead serves the role of the back-end, integrating a predefined logic with calls to OpenAI's models via the Responses API. Every text-based user request reaching the server is processed by the *request dispatcher* module, which orchestrates the application logic based on a state machine, routing each request to a relevant *state handler*. Each handler processes incoming requests based on a defined code-based logic that may involve one or more interactions with an LLM. Depending on the instructions ("system prompt") provided, LLMs are leveraged in three ways: (i) as classifiers; (ii) for managing multi-turn interactions; and (iii) for managing multi-turn interactions with access to local functions. Before forwarding a reply to the client, whenever applicable, handlers also implement state transitions.

4. Validation

From May to June 2025, we conducted a study to validate the new voice interaction mechanisms. Nine (9) participants, recruited from the network of friends and acquaintances, were observed while using the prototype, following a thinking-aloud protocol. The study was authorized by the ethical board of Politecnico di Milano (authorization no. 58/2024). In total, the 66.7% self-identified as female, while the 33.3% as male. Ages ranged from 24 to 55 years (M=30, SD=10). None of the participants reported visual disabilities. While the proposed paradigm was originally motivated by a study involving BVI individuals, the procedure focused on the evaluation of usability and cognitive effort of the proposed voice interaction mechanisms independently of sole voice reliance.

4.1. Procedure

Each session lasted approximately 1.5 hours. Before starting, participants signed an informed consent form. Then, they completed an online demographic questionnaire, designed to collect data about participants' age, gender, and technological expertise, as well as to investigate their prior experience with vocal assistants (e.g., Alexa) and tools powered by LLMs (e.g., ChatGPT). In the core part of the procedure, participants were given time to familiarize themselves with the prototype's interface and were later observed while completing a set of tasks involving the composition of an email through voice using the prototype introduced in Section 3.2. After completing tasks, a post-session survey was filled out by each participant. This included a collection of standardized questionnaires—System Usability Scale (SUS), NASA Task Load Index (NASA-TLX), and Technology Acceptance Model (TAM)—to assess the perceived usability, workload, and technology acceptance, respectively. At the end of the survey, two open-ended questions asked about the participants' most liked and disliked aspects of the experience. Finally, a semi-structured interview guided by open-ended questions elicited discussion on key aspects of the experience.

4.2. Quantitative Results

The analysis of quantitative results revealed a mean SUS score of 86.9 (SD=14.2) on a 0-100 scale, indicating a high level of perceived usability. The mean NASA-TLX score was 35.9 (SD=12.6) on a 0-100 scale, suggesting participants found the tasks moderately demanding. From the plot in Figure 6, it emerges that *mental demand* was the most critical during task execution, given both the higher scores received—indicating a greater perceived cognitive workload—and the greater variability of responses—suggesting that participants experi-

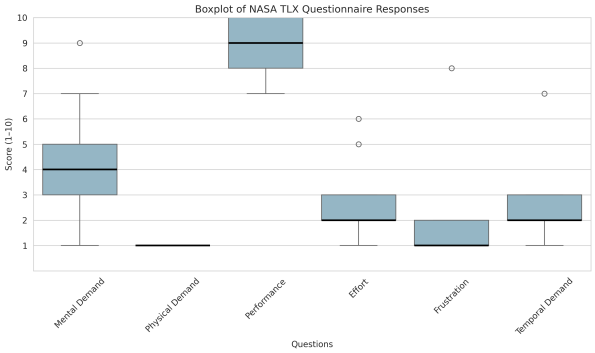

Figure 6: Distribution of NASA-TLX scores on a 10-point scale per dimension across participants

enced mental demand differently. The analysis of the TAM questionnaire responses highlighted an overall high level of perceived acceptance of the technology, with notable results in terms of system *learnability* and *ease of use*. In fact, questions addressing these aspects received a mean score of 4.8 and 4.7 on a 5-point scale, respectively, with very low standard deviations.

4.3. Qualitative Results

A thematic analysis of the gathered users' opinions highlighted the following themes.
Saving voice shortcuts. Convenient for recurrent tasks, but naming/recall can be challenging; some prefer not to save them. **Numbered options.** Effectively serves as an anchoring mechanism. **Feedback & summarization.** Preferences for strategies (full read-backs vs. summaries) are subjective and vary with the task criticality. **Voice vs. screen.** Voice is fine for quick actions; screen is preferred for long or critical texts. **Delegation & mixed modality.** Delegating tasks is accepted only with user confirmation; voice-plus-visual interaction valued. **Edit mode.** Edit mode welcomed for safe, focused edits; brief orientation still needed by some. **Navigation mode.** Rarely invoked; many assume its functions are implicit or forget it exists. **Mode announcements.** Distract goal-oriented users; silent switch preferred, but a single confirmation reassures some. **Ease of use.** Overall the system is perceived as simple, intuitive, and well-designed.

4.4. Discussion

The validation study highlighted relevant design implications and directions for future work.
Concerning the prototype, a major improvement could involve experimenting with strategies to reduce nondeterministic system behaviours, thus satisfying the need for controllability and reproducibility.
Concerning the procedure, to reduce the novelty effect of users with limited experience with LLM-powered VUIs and allow a more genuine evaluation of the proposed voice-based manipulation mechanisms, a comparative study might be carried out, involving a commercial LLM-powered VUI (e.g., ChatGPT's voice mode).

5. Conclusions

To address **RQ1**, the thesis has surveyed emerging human-LLM interaction techniques based on direct manipulation principles. Informed by a previous user study with BVI individuals, the thesis has introduced a new interaction paradigm to enable the manipulation of LLM outputs through voice. To prove the feasibility of the proposed interaction mechanisms and answer **RQ2**, the architecture of a prototype and its validation with 9 participants has been presented. Although further investigation is required to fully prove the validity of the new paradigm, the findings revealed positive perceptions concerning usability, workload, and technology acceptance. The validation also elicited genuine user needs and expectations, informing future work and the next design iterations.

References

[1] R. Luera, R. A. Rossi, A. Siu, F. Dernoncourt, and et al. Survey of User Interface Design and Interaction Techniques in Generative AI Applications, 2024.

[2] D. Masson, S. Malacria, G. Casiez, and D. Vogel. DirectGPT: A Direct Manipulation Interface to Interact with Large Language Models. In *Proc. of CHI '24*, 2024.

[3] OpenAI. Introducing Canvas, October 2024.

[4] E. Pucci, L. Piro, S. Andolina, and M. Matera. From Conversational Web to Inclusive Conversations with LLMs. In *Proc. of AVI '24*.

[5] Shneiderman. Direct Manipulation: A Step Beyond Programming Languages. *Computer*, 16(8):57–69, 1983.