Turn-taking in VUIs: Design strategies for achieving entrainment using audio cues

A thesis submitted in partial fulfillment of the requirements
of the degree of

Doctor of Philosophy

by

Mridumoni Phukon
(Roll No. 186105102)

Supervisor
Dr. Abhishek Shrivastava

Department of Design
INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI
2025

Dedicated to my beloved parents.

Department of Design
Indian Institute of Technology Guwahati
Guwahati - 781039 Assam India

Certificate

This is to certify that the work reported in this thesis with the title, ***"Turn-taking in VUIs: Design strategies for achieving entrainment using audio cues"*** is a bonafide research carried out by Mrs. Mridumoni Phukon, Roll No. 186105102 under my supervision.

This work, submitted for the award of the degree of *Doctor of Philosophy*, is original. It contains no materials previously published or written by any other person for the award of any degree or diploma at the Indian Institute of Technology (IIT) Guwahati or any other institute or university. Any prior workdone by other researchers is sufficiently referenced as well. All the requirements (including mandatory coursework) as per the rules and regulations prescribed in the Ph.D. ordinance for submitting the thesis for the Ph.D. degree of the Indian Institute of Technology (IIT) Guwahati have been fulfilled by the candidate.

Date: August 21, 2025

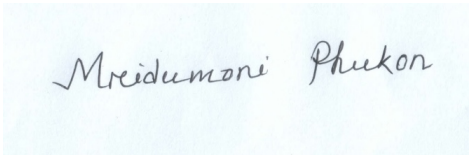
Place: Guwahati

Dr. Abhishek Shrivastava
(Thesis Supervisor)
Associate Professor
Dept. of Design
IIT Guwahati

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Date: August 21, 2025 Place: Guwahati



Mridumoni Phukon

Roll No. 186105102

Acknowledgements

First and foremost, I wish to convey my heartfelt gratitude to my supervisor, Dr. Abhishek Shrivastava, whose guidance, encouragement, and critical insights have shaped this research at every stage. I am deeply appreciative of his patience and the freedom he granted me to pursue my ideas. I also extend my sincere thanks to the members of my Doctoral Committee — Dr. Sougata Karmakar (Chairperson), Dr. Priyankoo Sarma (External Committee Member), and Dr. Sharmistha Banarjee (Committee Member) — for their invaluable support and for fostering a stimulating research environment. Their unwavering encouragement, patience, and constructive guidance were pivotal to the successful completion of my work. The continuous feedback, positive reinforcement, and thoughtful suggestions I received from them enabled me to grow not only as a researcher but also as an individual. I will always remain indebted for the privilege of working with them.

I am also grateful to Bruce Balentine and the entire team at Intelligently Interactive, with whom a formal research partnership was established. This collaboration greatly enriched my research by facilitating the exchange of ideas and expertise, contributing significantly to its advancement.

I extend my sincere thanks to all the participants who generously gave their time and perspectives, which were invaluable to this research.

I am also indebted to all the faculty members, staff, and security personnel of the Department of Design for their consistent assistance and support. My gratitude also goes to my colleagues and friends who accompanied me on this Ph.D. journey. I am particularly thankful to my fellow lab mates in the DFS Lab — Dr. Sandesh, Lipsa, Chinmoy, Shiva, Saurabh, Shyamala, and many others — for creating a vibrant and collaborative environment. The engaging discussions, collective problem-solving, and teamwork have greatly shaped my development as an independent and motivated researcher.

I owe my deepest thanks to my family, the true pillars of my strength — my parents (Mrs.

Renu Raj Kumari Phukon and Ajit Kumar Phukon), my extended family members, my brother and brother-in-law (Dayananda Phukon and Dr. Parikshit Gogoi), my sister and sister-in-law (Dr. Bornali Phukon and Sabita Borah), and our little angels (Aradhya, Aaron, and Tejaswani). Their unconditional love, constant support, warmth, and encouragement have defined who I am today. I remain forever indebted to them.

My heartfelt gratitude goes to the three most important anchors in my journey. First, my husband Ankur Dhekial Phukan, whose steadfast support has been the foundation of this achievement — I owe this accomplishment to him. To my daughter Driti, my bundle of joy, I am endlessly grateful. Her laughter, love, and presence brought light even to the darkest moments and inspired me to move forward with renewed strength. Lastly, to my sister Bornali, I am profoundly thankful for being my unwavering source of encouragement and companionship throughout both the highs and lows. Her presence in my life has been truly invaluable, and I feel blessed to have her by my side.

Finally, I am thankful to IIT Guwahati for providing an inspiring campus and excellent facilities that supported my academic journey. My gratitude also extends to the administrative staff, medical staff, security personnel, and cleaning staff of IIT Guwahati, whose contributions ensured a comfortable and safe environment during my time here.

Abstract

Spoken interactions with machines have become increasingly common in everyday contexts, from virtual assistants to voice-operated service kiosks. While full-duplex technology—allowing simultaneous speaking and listening—has shown promise in research and select commercial deployments, it remains technically demanding, resource-intensive, and not always the most practical choice for many real-world scenarios. In contrast, half-duplex turn-taking, where the system alternates between listening and speaking, continues to be a highly effective model for enabling interactive conversation—a structured, multi-turn exchange in which the user and system take sequential turns to share information, clarify intent, and progress toward a task goal. Its clear turn boundaries, predictable coordination, and resilience in noisy environments make it particularly well-suited for applications that benefit from such structured engagement without the complexity of managing overlapping speech. Domains such as prompt-response tasks, in-vehicle voice assistants, customer service helplines, and healthcare screening tools often favor half-duplex turn-taking, where clear turn-taking boundaries help reduce ambiguity and improve timing accuracy. These environments typically require fast, efficient exchanges—where one party speaks, and the other listens—without overlapping turns. Rather than seeing half-duplex as a limitation, it can be leveraged as a design advantage. By explicitly structuring turn-taking and integrating well-timed feedback cues, such systems enable users to entrain to the machine's turn-taking rhythm over time. This entrainment—where users gradually entrain their behavior to the system's timing—not only reduces errors but also improves coordination and task success.

This PhD Thesis investigates human user's turn-taking behaviors with a Voice User Interface (VUI) equipped with a novel turn-taking protocol that addresses the inherent constraints of half-duplex systems. By leveraging structured auditory feedback, including non-speech audio cues and explicit system prompts, the protocol sonifies the invisible seam of half-duplex turn-taking, encouraging users to entrain to the system's turn-taking rhythm and thereby enhancing

performance.

The study controls the VUI's turn-taking behavior by manipulating the temporal parameters of the automatic speech recognition (ASR) system. Through behavioral analysis of human participants, the research demonstrates that users not only respond to the artificial turn-taking cues but also gradually entrain to the VUI's turn-taking patterns. Key metrics, including task completion time, task error rate, and subjective user feedback, are utilized to evaluate the protocol's effectiveness.

The findings indicate that human users can temporally entrain to the turn-taking behavior of a Voice User Interface (VUI) when it is guided by a well-designed turn-taking protocol. This temporal alignment enhances coordination during interaction, leading to fewer task errors, faster task completion, and improved overall task success.

The mechanomorphic approach, which emphasizes system transparency and user adaptation, represents a critical advancement in the design of intuitive and effective VUI interactions. This research has practical implications for voice-based systems operating under half-duplex constraints, including customer service bots, virtual assistants in low-resource settings, and voice-based accessibility tools. By demonstrating that users can entrain to machine-imposed turn-taking protocols through sonified feedback tones, the study offers a low-overhead, self-teaching solution to improve timing, reduce errors during task completion, and increase interaction fluency. Systems that adopt a mechanomorphic design—presenting themselves as machines with clear, learnable rules—can set accurate user expectations and improve usability without relying on anthropomorphic cues. This approach is particularly valuable for structured, task-oriented applications such as different form-filling applications, helplines, healthcare surveys, and educational voice interfaces.

The study's scope is deliberately focused on half-duplex voice-based conversation, leaving aside multimodal feedback systems and full-duplex environments to enable a deep, targeted examination of turn-taking under constrained-channel conditions. The promising results from controlled experiments lay a strong foundation for future work, with real-world validation across diverse user populations representing the next step in broadening applicability. By addressing the unique challenges of half-duplex systems, this research contributes a scalable, low-overhead solution that bridges the gap between current technological constraints and task success in conversation.

Contents

भारतीय प्रौद्योगिकी संस्थान गुवाहाटी
Indian Institute of Technology Guwahati

List of Figures

List of Tables

List of Abbreviations

HCI	Human-Computer Interaction
AI	Artificial Intelligence
ANOVA	Analysis of Variance
ASR	Automatic Speech Recognition
HHI	Human-Human Interaction
LED	Light Emitting Diode
NLP	Natural Language Processing
PLM	Probabilistic Language Model
SD	Standard Deviation
STS	Spoke-Too-Soon
STSTone	Spoke-Too-Soon tone
SW	SmartWindow
TRPs	Transition Relevance Places
TTS	Text-to-Speech
VAD	Voice Activity Detection
VUI	Voice User Interface
VUIs	Voice User Interfaces
CA	Conversation Analysis

Chapter 1

Introduction

1.1 Background and motivation

Voice User Interfaces (VUIs) enable users to interact with digital systems through spoken language rather than traditional input methods like typing or tapping. They have become integral to everyday life through systems such as Amazon Alexa, Google Assistant, and Apple's Siri, as well as embedded applications in smart homes, vehicles, and customer service platforms. Their appeal lies in the convenience and accessibility they offer, supporting hands-free, screen-free interactions and providing assistance in contexts ranging from routine tasks to accessibility support. According to Strategy Analytics, the smart home market alone is projected to exceed $170 billion by 2025, with more than 400 million households adopting VUI-integrated systems (see Figure 1.1).

Despite growing adoption, VUIs still struggle to manage the fluid dynamics of human conversation. A persistent usability challenge is *turn-taking* the coordination of when the user speaks and when the system responds. This coordination is achieved through subtle multimodal cues, including timing, prosody, gaze, and body language in human dialogue. These cues are

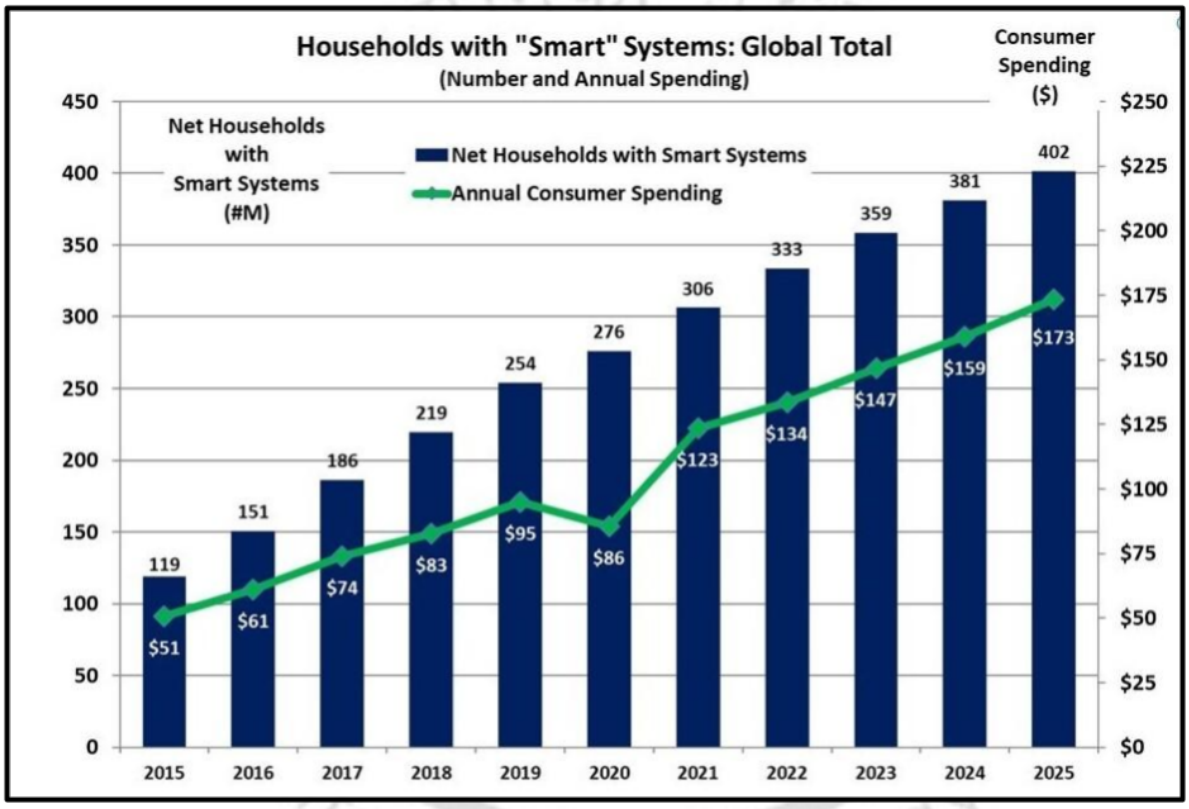


Figure 1.1: Projected global growth of households with smart systems and associated consumer spending from 2015 to 2025. Source: Strategy Analytics (2021) Strategy Analytics (2021).

largely absent in current voice interfaces. Most commercial VUIs use fixed silence thresholds or keyword detection to infer turn boundaries. These approaches are error-prone: if the silence threshold is too short, the system interrupts; if it is too long, the interaction becomes stilted.

A deeper problem lies in the underlying architecture of many VUIs, which operate in either *full-duplex* or *half-duplex* modes. Both parties can speak and listen simultaneously in full-duplex systems, mirroring natural human conversation. However, such systems require complex audio processing and are often sensitive to background noise. In contrast, half-duplex systems more commonly deployed in consumer VUIs allow only one direction of communication at a time. The system must either be speaking or listening, but not both. This design introduces an "invisible seam" in conversation, leaving users uncertain about when to take their turn (as discussed in chapter 4) rarely made transparent to users. An example, as shown in figure 1.2, illustrates how this constraint leads to classic turn-taking breakdowns during a human–VUI exchange in a navigation scenario.

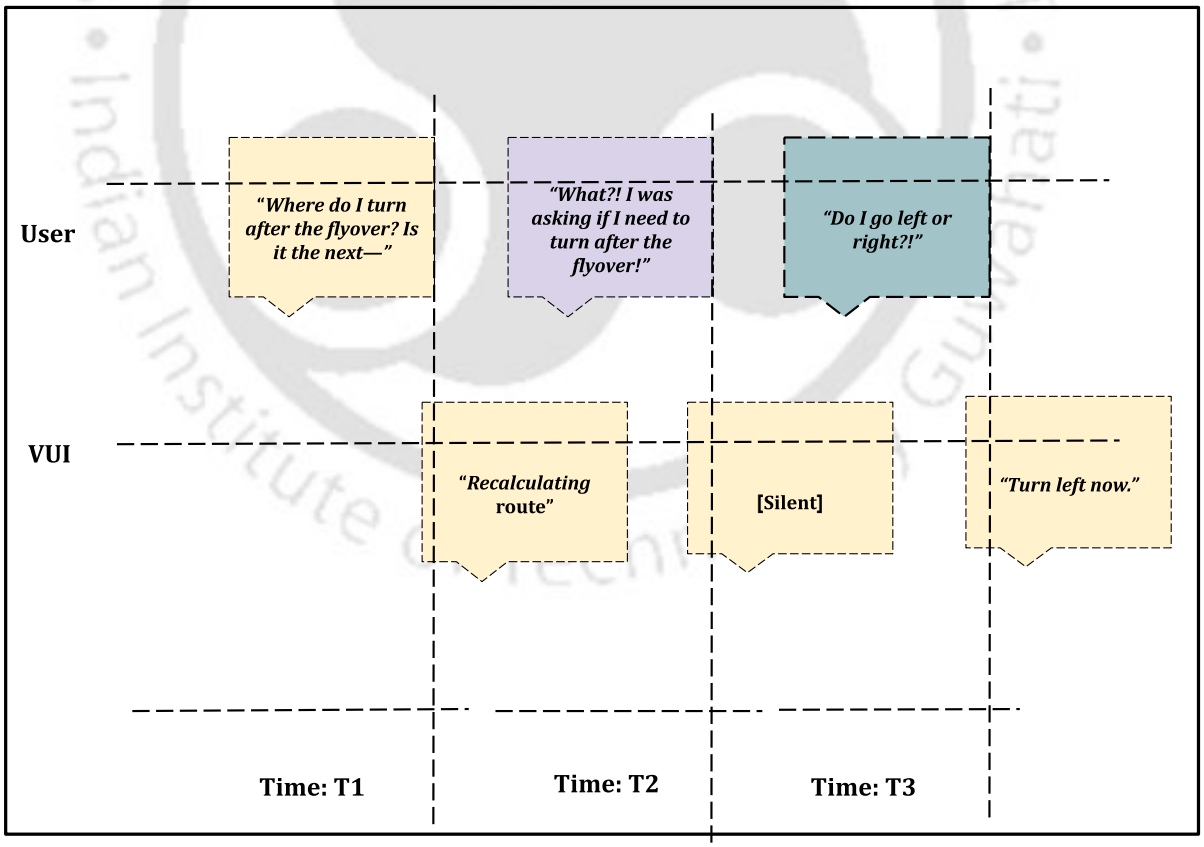


Figure 1.2: Example of a turn-taking breakdown in a half-duplex navigation scenario.

This figure 1.2 illustrates how turn-taking misalignment emerges in current VUI systems

when the user begins a request while the system simultaneously initiates its own output (T1). Because half-duplex designs allow only one side to speak at a time, the user is interrupted and becomes uncertain whether their input was captured. At T2, the system falls silent, providing no cue about whether it is listening, processing, or waiting forcing the user to repair or repeat the request. By T3, the user attempts again just as the VUI delivers a delayed response, resulting in overlap and confusion. The coloured bubbles separate user and system turns, while the time markers highlight how the absence of clear listening cues and inconsistent timing create difficulty for users in knowing when to speak, when to pause, and whether the system has heard them.

The figure 1.2 depicts a turn-taking breakdown in a navigation scenario using a half-duplex VUI. At T1, the user's query overlaps with the system's "Recalculating route" response, causing the original question to go unresolved. By T2, the system is silent, prompting user frustration and repetition. At T3, the user requests clarification ("Do I go left or right?") just before the system finally issues the directive "Turn left now," highlighting delayed and misaligned turn exchanges.

In contrast, Figure 1.3 shows how turn-taking unfolds more flexibly in a full-duplex, human–human conversation. Participants can overlap, interrupt, or return to previous threads without conversational breakdown, as speaking and listening channels remain open. Full-duplex systems inherently support conversational fluidity, while half-duplex systems demand strict sequential coordination, yet they often fail to signal this clearly.

This figure 1.3 illustrates how two human speakers manage turns fluidly, even when their utterances partially overlap. Because both can speak and listen at the same time, each participant is able to follow the other's intent, maintain awareness of previously initiated topics, and return to unanswered questions naturally. The overlap at T2–T3 is handled without breakdown, as both speakers track the conversation jointly and coordinate their turns using timing, prosody, and shared understanding. The coloured bubbles distinguish the two participants, and the time markers (T1–T5) highlight how full-duplex capability and human turn-management skills enable a coherent, well-aligned interaction despite simultaneous speech.

The figure illustrates smooth turn-taking in a full-duplex human–human conversation. At T1, Participant 1 asks about a flight time while Participant 2 simultaneously begins a comment about a movie. Across T2–T4, both speakers interleave responses and reactions without disrupt-

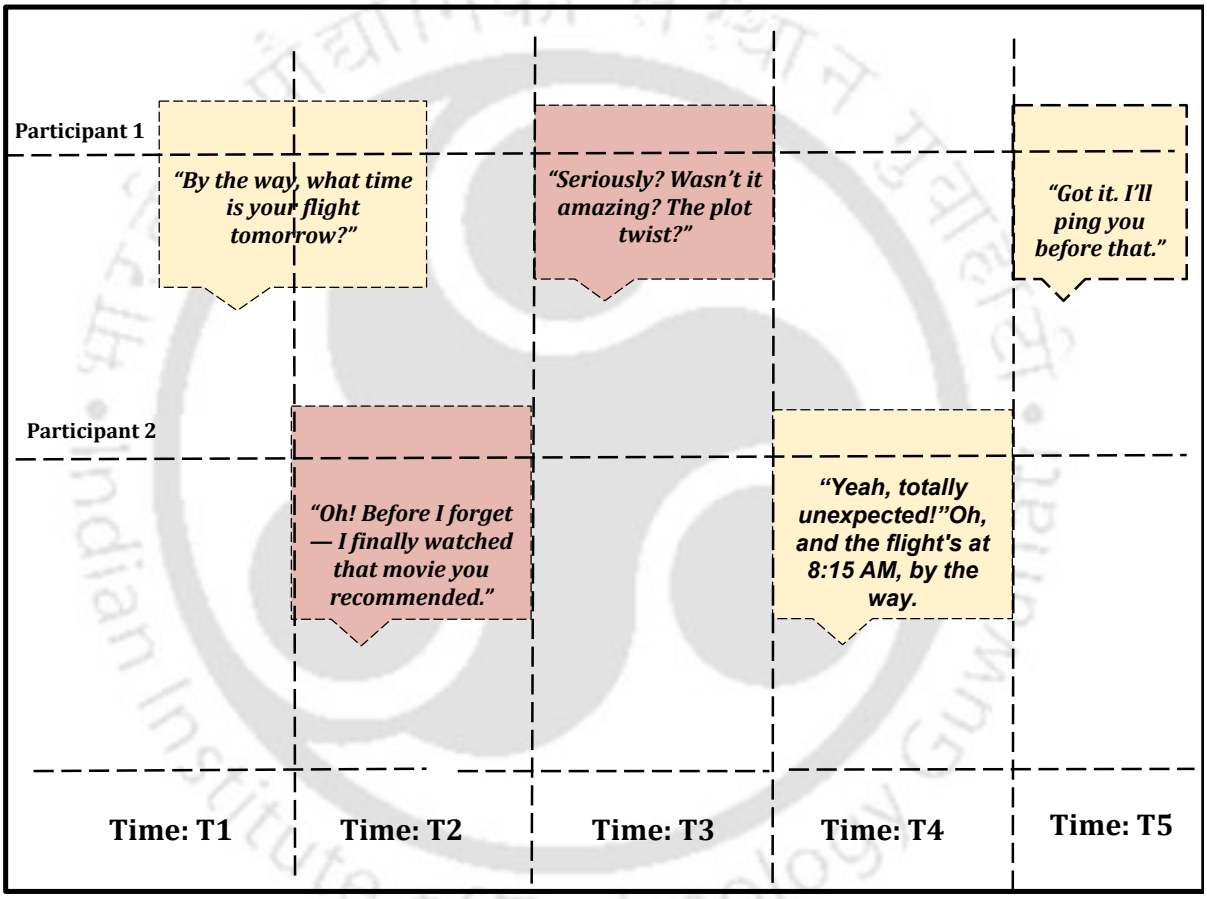


Figure 1.3: Example of smooth turn-taking in a full-duplex human–human conversation.

ing conversational flow, demonstrating the fluid overlap and rapid floor transitions characteristic of full-duplex dialogue. By T5, the exchange concludes with a coordinated wrap-up, reflecting natural timing and mutual responsiveness.

While full-duplex VUIs like Google Duplex aim to replicate natural conversation, they are often resource-intensive and fragile in noisy or dynamic environments. Though more robust and resource-efficient, half-duplex systems require a fundamentally different approach to manage turn transitions effectively. Yet, current systems rarely guide users explicitly on when to speak or wait. This ambiguity leads to errors, user frustration, and ultimately, reduced task success.

This thesis addresses these challenges by exploring how turn-taking can be redesigned for half-duplex VUIs using a *mechanomorphic* design philosophy. This approach treats the VUI as a transparent, cooperative machine rather than imitating human-like behaviour. The system explicitly communicates its internal states through auditory cues. By using carefully designed tones that provide perceptually salient feedback about turn transitions, the system helps users entrain to its rhythm.

Without clear turn-transition signals, users are left to infer system readiness based on trial-and-error, often leading to speaking too early, hesitating excessively, or talking over system responses. These breakdowns suggest a need for mechanisms that help users better align with the system's temporal rhythm. In human conversation, such alignment is achieved through a process known as *entrainment* a dynamic, often unconscious adjustment of speech timing, prosody, and phrasing between interlocutors. Entrainment contributes to smoother interaction, improved comprehension, and stronger interpersonal rapport. In the context of VUIs, particularly half-duplex systems, leveraging entrainment principles could help users synchronise with the system's turn-taking cues more effectively. In HCI, entrainment research has focused chiefly on prosodic or lexical alignment. Far less is known about temporal entrainment and its role in supporting smooth human–VUI interaction. This thesis extends this work by investigating how structured auditory feedback can be used to make human users entrain to the VUI's temporal turn-taking behaviour and provide successful conversations in half-duplex systems.

Through three empirical studies, this research 1) examines turn-taking errors in commercial VUIs, 2) designs and evaluates a mechanomorphic half-duplex turn-taking protocol, and 3)compares user behaviour with and without non-speech corrective auditory feedback. The goal is to reduce turn-taking errors, task completion time, and improve conversation.

1.2 Aim of the thesis

The aim of this thesis is to investigate the effectiveness of the use of audio-based cues in providing opportunities for entrainment between the users and the voice-user interface in the context of half-duplex turn-taking conversation.

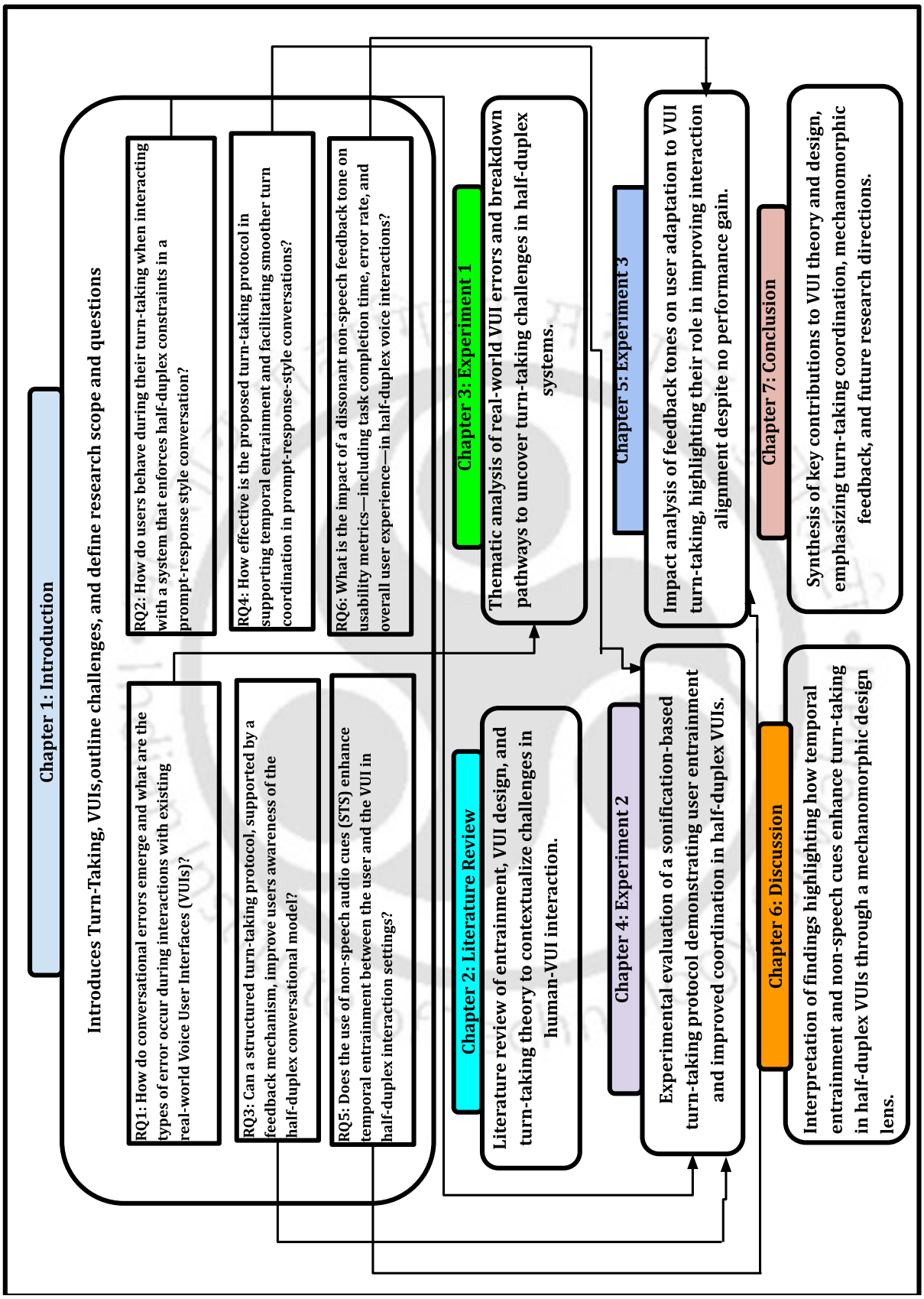


Figure 1.4: Chapter-wise mapping of the thesis in relation to the research questions.

The subject matter of the thesis is presented in the following seven chapters (as shown in Figure 1.4).

1.3 Research framing, questions, and contributions

This thesis investigates turn-taking breakdowns in contemporary Voice User Interfaces (VUIs), with a particular focus on half-duplex interaction constraints where the system cannot listen while speaking or playing auditory feedback. The central premise is that many everyday failures in task-oriented voice interaction are not only recognition errors, but timing and coordination failures that emerge from unclear turn boundaries and limited system feedback. To address this, the thesis adopts a mechanomorphic approach: instead of attempting human-like conversational smoothness under technical constraints, it designs explicit, machine-legible signals (non-speech auditory cues and structured prompts) that help users align with the system's turn-taking model.

1.3.1 Research objectives and how they are addressed

The following objectives guided the research. Each objective is addressed through a dedicated empirical or design-evaluation chapter, ensuring that the thesis progresses from diagnosing real-world failures to designing and validating an intervention.

1. **To investigate how conversational errors emerge and unfold in real-world VUI interactions, particularly in half-duplex systems.** This objective is addressed through a systematic analysis of task-oriented interactions with existing real-world VUIs, focusing on how user behaviour and system responses interact over turns to produce success, repair, or breakdown.

2. **To design and evaluate a structured turn-taking protocol incorporating a feedback system.** This objective is addressed by developing a protocol that explicitly marks turn boundaries and readiness-to-listen states using non-speech auditory cues and prompts, aiming to support temporal entrainment and reduce premature inputs and awkward silences.

3. **To evaluate the effectiveness of the proposed protocol using performance and experience metrics.** This objective is addressed through controlled user studies that assess task completion time, task error rate, and user experience, providing quantitative and qualitative evidence of protocol impact.

4. **To conduct a comparative study of a dissonant turn-taking cue as corrective feedback in urgency-driven interactions.** This objective is addressed by comparing interaction conditions with and without the corrective non-speech tone (STS), examining whether the cue reduces premature input and improves turn coordination under time-sensitive, prompt-response constraints.

1.3.2 Research questions and their role in the thesis

The research questions below translate the above objectives into specific, answerable inquiries. Together, they structure the thesis from (i) understanding error mechanisms in real-world VUIs, to (ii) characterising user behaviour under half-duplex constraints, to (iii) validating whether a structured, mechanomorphic protocol and feedback cue can improve turn coordination and entrainment.

1. **How do conversational errors emerge, and what types of errors occur during interactions with existing real-world VUIs?** This question motivates the exploratory analysis in Chapter 3, which identifies recurring error patterns and maps how breakdowns unfold over turns.

2. **How do users behave during turn-taking when interacting with a system that enforces half-duplex constraints in a prompt-response conversation?** This question motivates the behavioural characterisation reported in Chapters 4–5, focusing on timing, turn-entry attempts, and adaptation over repeated interactions.

3. **Can a structured turn-taking protocol, supported by a feedback mechanism, improve users' awareness of the half-duplex conversational model?** This question motivates the design rationale and evaluation of explicit system signalling (Chapter 4) and its effect on user understanding and interaction strategy (Chapters 4–5).

4. **How effective is the proposed turn-taking protocol in supporting temporal entrainment and facilitating smoother turn coordination in prompt-response-style conversations?** This question motivates the primary evaluation of interaction smoothness, coordination outcomes, and changes over time (Chapters 4–5), interpreted through the lens of entrainment and mechanomorphic design (Chapter 6).

5. **Do non-speech audio cues (STS) enhance temporal entrainment between the user and the VUI in half-duplex interaction settings?** This question motivates the comparative emphasis on the corrective cue (Chapter 5) and its relationship to behavioural alignment and reduced premature speech.

6. **What is the impact of a dissonant non-speech feedback tone on usability metrics—including task completion time, error rate, and user experience—in half-duplex voice interactions?** This question motivates the comparative analysis of performance and user experience across conditions (Chapter 5), providing evidence for when and why such cues are beneficial.

1.3.3 Scope and limitations

This research focuses on designing and evaluating a structured turn-taking protocol for *half-duplex*, speech-based, task-oriented VUI interactions (e.g., prompt-response and form-filling). The proposed approach is grounded in mechanomorphic design principles and investigates how explicit auditory feedback (including a dissonant corrective tone) and structured prompts can support user entrainment to system timing, reduce premature inputs, and mitigate awkward silences.

The research does not extend to multimodal turn-taking support using visual or haptic feedback, nor does it address full-duplex conversational systems. Although the protocol is evaluated through controlled user studies, further validation in in-the-wild contexts (e.g., diverse noise conditions, broader user populations, longer-term adoption) remains an important direction for future work. In addition, the thesis intentionally deprioritises anthropomorphic strategies and emotional/social dimensions of interaction, focusing instead on clarity, predictability, and user adaptation to system constraints.

1.3.4 Thesis roadmap

Figure 1.4 provides a chapter-wise mapping of the thesis in relation to the research questions. The thesis is organised into seven chapters. Chapter 1 motivates the problem and frames the objectives, questions, scope, and roadmap. Chapter 2 reviews entrainment, VUI interaction constraints, and turn-taking theory, establishing the theoretical grounding. Chapter 3 analyses real-world VUI conversational errors and presents common breakdown pathways. Chapter 4 introduces and evaluates a structured, sonification-supported half-duplex turn-taking protocol. Chapter 5 provides a comparative evaluation of corrective feedback tones (STS) and their impact on user behaviour and usability outcomes. Chapter 6 discusses the findings in relation to the literature and argues for mechanomorphic turn-taking design. Chapter 7 concludes with contributions, limitations, and future work.

1.3.5 Summary of contributions

This thesis makes the following contributions to research and design of half-duplex VUIs:

1. **An empirical account of turn-by-turn conversational breakdowns in real-world VUI use,** including a structured description of how errors emerge and unfold during task-oriented interactions (Chapter 3).

2. **A conversation breakdown mapping perspective** that characterises success, repair, and abandonment trajectories in VUI conversations, supporting diagnosis of timing-related failure points (Chapter 3).

3. **A structured, mechanomorphic turn-taking protocol for half-duplex VUIs** that uses explicit signalling of system state and turn boundaries via non-speech audio cues and prompts (Chapter 4).

4. **Controlled evidence on user adaptation and turn-taking behaviour under half-duplex constraints,** demonstrating how structured timing and feedback mechanisms influence coordination over repeated interaction (Chapters 4–5).

5. **Comparative evidence on a dissonant corrective auditory cue (STS)** as feedback for premature input in urgency-driven, prompt-response interactions, including its impact on behavioural outcomes and usability metrics (Chapter 5).

6. **Design implications for mechanomorphic turn-taking support,** translating findings into actionable guidance for feedback timing, turn-boundary signalling, and protocol transparency in constrained spoken interaction (Chapters 6–7).

This page was intentionally left blank.

Chapter 2

Review of Literature

2.1 Overview

This chapter reviews the theoretical foundations underpinning conversational dynamics in voice-based interactions, focusing on turn-taking mechanisms and speech entrainment for mechanomorphic design. It begins by exploring how timing and coordination function in spoken conversations, laying the groundwork for understanding communication flow in human-human and human-machine contexts. The discussion then turns to how users naturally align their speech patterns lexically and prosodically with voice-based systems, a phenomenon known as speech entrainment. Special attention is given to how these principles apply to mechanomorphic system design, where the VUI explicitly signals its state and constraints using non-speech audio cues instead of mimicking human-like behaviours. Through this review, the chapter identifies critical factors influencing conversation success and highlights existing limitations in current VUI designs, thereby establishing the key research gaps addressed in this thesis.

2.2 Introduction

This chapter reviews the body of scholarly work that informs and contextualises research on turn-taking in Voice User Interfaces (VUIs), with a particular emphasis on half-duplex systems, mechanomorphic interaction design, and the phenomenon of temporal entrainment. Turn-taking is a foundational mechanism in human conversation that ensures the orderly exchange of speaking roles, enabling fluid and cooperative interaction. While extensively studied in human-human interaction (HHI), implementing turn-taking in human-machine interfaces especially VUIs presents persistent challenges. These challenges arise not only from the technological limitations of current VUIs such as latency, speech recognition errors, and half-duplex constraints but also from the failure to incorporate well-established conversational principles, such as the human user's awareness about timing, the system design for turn predictability, and the use of feedback cues that humans rely on to coordinate speech.

The chapter begins by exploring foundational theories of conversational turn-taking and how these have shaped the design of dialogue systems. It then discusses how VUIs interpret and manage turn-taking, highlighting limitations specific to half-duplex communication, where the system cannot speak or listen simultaneously. This constraint introduces a distinct interaction dynamic often overlooked in contemporary voice assistant design, which increasingly favours full-duplex capabilities. Despite this trend, half-duplex systems remain widely deployed in many real-world applications such as form-filling, call centres, and smart appliances where clarity, control, and predictability are essential.

Next, the chapter critically examines the concept of entrainment the subconscious alignment of behaviour between interlocutors within both Human-human interaction (HHI) and human-computer interaction (HCI) contexts. It explores how speech entrainment, particularly in timing and prosody, can support smoother turn transitions and increase user satisfaction. This is followed by a discussion on design paradigms for VUIs, contrasting anthropomorphic approaches that attempt to emulate human-like conversation, with mechanomorphic approaches that emphasise transparency and machine-appropriate signalling.

This review identifies key research gaps by synthesising work across conversation analysis, dialogue systems, and VUI interaction design. These include the underrepresentation of

half-duplex systems in interaction research, the lack of structured turn-taking mechanisms informed by entrainment theory, insufficient exploration of non-speech auditory cues, and limited attention to mechanomorphic design strategies. These gaps directly inform the research questions and design interventions this thesis explores.

2.3 Theoretical foundations

The theoretical foundations of this research are rooted in the following key areas: Turn-taking during speech-based conversation, the necessity of regulated turn-taking protocols, half-duplex and full-duplex conversation, system feedback in turn-taking, speech entrainment in voice-based conversation, and the necessity of mechanomorphic design in voice interfaces. These provide a comprehensive understanding of human turn-taking behaviour and offer insight into designing an effective VUI turn-taking protocol that encourages temporal entrainment to achieve smooth turn-taking.

2.3.1 Turn-taking during speech-based conversation

Turn-taking refers to the coordinated exchange of speaking and listening roles between interlocutors, enabling the smooth conversation progression. This process is typically seamless in human-human interaction due to the inherent human ability to speak and listen simultaneously, allowing for dynamic negotiation of turns. In contrast, human-machine interactions particularly those constrained by half-duplex systems lack this simultaneity, making turn management more challenging. For effective communication, the user and the system must maintain a shared understanding of who holds the conversational floor and when a turn transition should occur. Despite its importance, turn-taking in most dialogue systems, including those employed in human-robot interaction, is often governed by simplistic heuristics. A common approach involves detecting a fixed period of silence to infer user turn completion, after which the system initiates input processing and formulates a response Raux (2008); Skantze (2007); Schlangen and Skantze (2011). While functional, such mechanisms fail to capture human conversational cues' nuanced, real-time nature, often leading to interruptions, delays, or overlap errors.

However, relying on silence alone is a flawed strategy. Users often pause mid-sentence for cognitive or expressive reasons, which does not necessarily indicate a willingness to yield the turn. As a result, systems based on this approach tend to either interrupt users prematurely or respond with noticeable delays Raux and Eskenazi (2009, 2012). Furthermore, distinguishing between actual interruptions and cooperative speech overlaps, such as brief affirmations like 'yeah' or 'mhm', remains a challenge, leading many systems to ignore any input while speaking.

This contrasts sharply with the finely coordinated nature of human conversation, where the gap between turns is often no more than a few hundred milliseconds. Humans rely on a rich set of cues to negotiate turn boundaries, including prosodic features such as intonation and rhythm, the presence of filled pauses (such as "uh" or "um"), syntactic completeness, body gestures, and eye gaze. Rather than waiting passively, listeners actively anticipate the end of a speaker's turn, enabling them to respond promptly and appropriately.

In recent years, full-duplex conversational systems have garnered significant attention, owing to advances in low-latency speech recognition, real-time processing, and incremental dialogue models. Unlike half-duplex systems, full-duplex architectures support simultaneous speaking and listening, allowing for more fluid, overlapping, and human-like exchanges. These capabilities have made full-duplex systems attractive for applications such as intelligent assistants, telephony (e.g., Google Duplex; Leviathan and Matias (2018)), in-car voice control, customer service, and social robotics. Researchers have demonstrated the feasibility of incremental response generation and overlap handling in real-time interactionsVeluri et al. (2024). However, despite these advancements, full-duplex systems continue to face critical limitations. One core challenge is reliably managing turn overlaps, interruptions, and backchannels without misrecognition or inappropriate timing. Human conversations are rich with subtle cues prosodic, lexical, and contextual that guide turn-taking, and machines still struggle to interpret and generate these cues with sufficient accuracy. Furthermore, full-duplex systems require more computational resources, robust ASR models, and advanced latency control, complicating deployment in constrained or noisy environments. Skantze (2021) argues that even with full-duplex capabilities, successful interaction depends not just on simultaneity but also on mutual coordination a nuance that current systems often fail to achieve. Thus, while full-duplex architectures represent a significant step toward naturalistic dialogue, they are not a panacea and must be complemented by better turn-taking models and feedback strategies. .

2.3.2 Necessity of regulated turn-taking protocols

Although the maturity of AI and NLP (Natural Language Processing) has made it possible to develop full-duplex human-machine conversation Lin et al. (2022); Leviathan and Matias (2018), most of the existing VUIs are still based on a half-duplex system. Current half-duplex Voice User Interfaces (VUIs), which can speak or listen but not both simultaneously, manage turn-taking through rigid, system-dominated protocols primarily built upon finite-state dialogue management frameworks Raux and Eskenazi (2009). In such systems, the turn-taking flow is typically controlled by predefined prompts where the system issues a question or directive, waits for a fixed silence threshold to detect the user's response, processes the input, and then speaks again. This turn-taking model follows a strict "listen-then-speak" cycle with minimal flexibility, leading to a lack of naturalistic conversational dynamics Skantze (2021). Voice Activity Detection (VAD) and silence-based thresholds (commonly set between 500 and 800 ms) are the primary mechanisms to detect user input endpoints, but they often misinterpret brief pauses in user speech as turn completion, resulting in premature system responses Raux (2008). Conversely, if the system waits too long to respond, it leads to unnaturally long pauses and perceived sluggishness, disrupting the conversational flow Skantze (2021). Barge-in capabilities, wherein users can interrupt the system, are often disabled or limited in half-duplex VUIs, given the technical constraints of being unable to listen while speaking. Even when implemented, barge-in handling remains error-prone due to latency and processing delays, further compromising conversational smoothness Chang et al. (2022). To navigate these constraints, most systems rely heavily on wake words and explicit system prompts to initiate or return turns to the user, emphasising a command-response turn-taking model rather than collaborative dialogue Luger and Sellen (2016). Moreover, half-duplex VUIs generally lack real-time feedback mechanisms such as visual or auditory cues that would otherwise signal when the system is ready to listen or speak. This absence of transparent system state feedback contributes to user confusion and hesitation, particularly during more prolonged or multi-turn interactions Porcheron et al. (2018). While some recent systems, like Google Assistant or Alexa, offer limited visual or tonal cues to indicate readiness, these remain inconsistent and often fail to support adaptive or entrained turn-taking behaviour. However, in situations where users cannot use their hands or eyes, and where full-duplex systems are either unavailable or underutilised, there is a need for half-duplex

systems equipped with effective interactive turn-taking mechanisms.

Use cases such as form-filling tasks and rapid question–and–answer exchanges are susceptible to timing, coordination, and clarity of conversational roles. However, current half-duplex VUI designs are ill-suited to support these interactional demands, as they often lack mechanisms for timing flexibility, turn anticipation, and explicit floor management elements crucial for ensuring smooth exchanges in constrained environments. Unlike human-human conversations, where interlocutors can rely on subtle cues and simultaneous feedback to negotiate turns fluidly, half-duplex systems cannot process speech and playback concurrently, making naturalistic turn-taking unreliable and error-prone. In such contexts, relying on implicit or organic conversational strategies may lead to frequent interruptions, premature responses, or prolonged silences. This underscores the need for regulated turn-taking protocols that explicitly define when the user may speak, when the system will listen, and how turn transitions are managed. Particularly in domains like automotive interfaces or low-resource embedded systems, where hardware and cognitive constraints limit the feasibility of full-duplex interaction, structured turn-taking can serve as a compensatory design principle Sacks et al. (1974); Levitan et al. (2016); McWilliams et al. (2015); Chang et al. (2022). Thus, rather than emulating natural turn-taking, VUIs in these settings benefit from clear, mechanomorphic but unnatural turn-taking strategies that foreground coordination over natural turn-taking as human-human exchange.

2.3.3 Examples from contemporary VUIs and why mechanomorphic turn-taking support is still needed

A large portion of deployed Voice User Interfaces (VUIs) used in everyday settings (e.g., smartphone assistants and smart speakers) rely on interaction patterns that are *functionally half-duplex* in many task-oriented contexts, even when the underlying platform may support limited barge-in. In practice, these systems typically follow a familiar loop. A wake word (or button press) opens a listening window, the user speaks, endpointing is inferred primarily through Voice Activity Detection (VAD) and silence thresholds, the system enters a processing phase, and then delivers spoken output Raux (2008); Raux and Eskenazi (2009); Skantze (2021). This design works reasonably well for short, single-shot commands, but it becomes

fragile in prompt–response and form-filling interactions where users must respond quickly, coordinate timing precisely, and maintain conversational state across multiple turns Porcheron et al. (2018); Sciuto et al. (2018); Luger and Sellen (2016).

2.3.3.1 Illustrative working principles in mainstream systems

Although different VUIs vary in capability and implementation detail, the *user-facing* working principle is often similar. Users are expected to infer when the system is listening, when it has stopped listening, and when it is ready for the next input. In home and workplace settings, studies show that users frequently develop coping strategies such as rephrasing, repeating, simplifying language, and using shorter commands because system readiness and conversational capabilities are not consistently discoverable through feedback Sciuto et al. (2018); Porcheron et al. (2018); Luger and Sellen (2016). In other words, even popular VUIs may succeed at speech recognition in many cases but still fail to provide robust, transparent mechanisms for *turn coordination* and *timing clarity* during multi-turn tasks.

2.3.3.2 Why these systems can be inefficient in task-oriented turn-taking

The inefficiency of current VUIs in task-oriented interaction is often not only a matter of recognition accuracy, but of *turn-taking miscoordination*. Three recurring breakdown patterns reported across the VUI literature align directly with the design problem addressed in this thesis:

1. **Premature user input during non-listening windows:** Users may respond quickly (especially in urgency-driven prompts) but speak at moments when the system is effectively not listening, leading to missed input and repeated attempts. Such breakdowns increase interaction time and frustration, and encourage users to adopt cautious, unnatural pacing strategies Skantze (2021); Porcheron et al. (2018).

2. **Silence-threshold endpointing causes premature cutoffs or sluggishness:** When endpointing relies on short silence thresholds, brief cognitive pauses can be misinterpreted as turn completion, cutting the user off; when thresholds are longer, the system feels slow and unresponsive Raux (2008); Raux and Eskenazi (2012). Both cases disrupt the tight timing that characterises human turn-taking and increase perceived conversational awkwardness.

3. **Limited feedback and weak discoverability of system state:** Users often cannot reliably determine whether the VUI has heard them, is processing, or is ready for the next turn. Prior work shows that this lack of transparent feedback restricts exploration of system capabilities and makes users hesitant during multi-turn exchanges Sciuto et al. (2018); Luger and Sellen (2016); Kirschthaler et al. (2020).

These patterns are especially pronounced in structured interactions (e.g., form-filling, verification questions, digit/slot entry, and rapid question–answer sequences) where the user must place responses into a narrow timing window and where the system's inability to listen while producing output can create brief conversational "blind spots". In such contexts, attempting to emulate natural human conversation through anthropomorphic cues alone is often insufficient, because the core issue is not social expressiveness but *timing transparency and floor management* under constraints Skantze (2021).

2.3.3.3 Best-fit scenarios for voice-only interaction (and why timing cues matter)

Voice-only interaction is particularly valuable in scenarios where users' hands and eyes are busy or attention is divided, such as driving, cooking, operating appliances, or performing on-the-go tasks where visual scanning is costly. In these settings, audio becomes the primary channel for communicating not only content but also system *state* and *timing* Brewster et al. (1993); Gaver (1993). However, precisely because the interaction is voice-only and transient, failures in turn boundary signalling (e.g., when to speak, when the system is ready) can be more disruptive than in screen-based interfaces, motivating lightweight auditory mechanisms that make turn opportunities explicit.

2.3.3.4 Scope alignment: why this thesis adopts a mechanomorphic, voice-only framing

Positioned within this literature, the present thesis intentionally focuses on **voice-only** turn-taking support for **half-duplex** and functionally half-duplex task-oriented interactions. Multimodal turn-taking aids (e.g., screen animations, LEDs, or haptic cues) may improve state awareness, but they are not consistently available across devices and are less suitable in eyes-busy scenarios. Accordingly, this thesis emphasises a **mechanomorphic** design approach. Rather than masking constraints behind human-like conversational behaviour, the system transparently sig-

nals listening state, turn boundaries, and timing constraints through structured non-speech audio cues and explicit prompts Kirschthaler et al. (2020); Furqan et al. (2017). This literature-backed framing motivates the thesis intervention, a regulated turn-taking protocol that prioritises predictability and coordination over human-likeness, and empirically evaluates whether such signalling supports user entrainment and reduces timing-related breakdowns in constrained voice interaction.

2.3.4 Half-Duplex and full-Duplex conversation

In the context of spoken dialogue systems (SDS), half-duplex and full-duplex refer to different modes of managing turn-taking based on communication flow. A half-duplex system allows only one party to transmit at a time either the system speaks or the user speaks, but not both simultaneously. This mode is typical of many commercial voice assistants, including Google Assistant in form-filling tasks, Siri during phone-based dictation, and many customer service IVR systems, where a structured "listen–then–speak" protocol governs interaction McWilliams et al. (2015); Zhao et al. (2015). In contrast, a full-duplex system supports simultaneous speaking and listening, enabling more fluid, overlapping conversational behaviour akin to human-human dialogue. Recent full-duplex systems such as Google Duplex and Alibaba's Duplex Conversation have demonstrated human-like responsiveness through capabilities like backchanneling, barge-in detection, and real-time user state monitoring Chang et al. (2022); Lin et al. (2022).

Each approach presents distinct trade-offs. Full-duplex systems offer greater naturalness and reduced response latency but require continuous speech processing, echo cancellation, and higher computational resources, making them expensive and less suitable for deployment in low-power or bandwidth-limited environments Skantze (2021). Moreover, managing interruptions in full-duplex setups remains challenging, as systems must accurately distinguish between genuine interruptions and non-disruptive speech phenomena such as backchannels (e.g., "mm-hmm") or environmental noise Skantze and Irfan (2025). Without robust turn-taking models, full-duplex systems risk frequent false triggers, resulting in abrupt cutoffs or misinterpretation of user intent, which can degrade user experience Skantze (2021); Skantze and Irfan (2025). In contrast, half-duplex systems constrain speakers to one speaker at a time, often relying on silence thresholds or explicit cues to manage turn exchanges. While this approach is less nat-

ural, it simplifies turn management and can be advantageous in constrained environments such as in-car interfaces or voice form-filling tasks where clear boundaries reduce cognitive load and ensure system responsiveness Phukon et al. (2022). Thus, while full-duplex systems aim to emulate natural conversation, half-duplex systems offer structured, reliable turn-taking mechanisms that remain relevant in task-oriented voice user interfaces. By clearly separating speaking and listening phases, half-duplex interaction avoids ambiguity, reduces cognitive load, and ensures clarity, especially in high-noise, multitasking, or assistive settings.

In this research, we deliberately focus on enhancing turn-taking in half-duplex systems, not as a fallback, but as a practical and scalable design choice. Our goal is to improve turn-taking during conversation in settings where resource constraints, deployment costs, and infrastructure limitations render full-duplex solutions impractical or overly complex. Half-duplex systems, while inherently limited to alternating speaking and listening modes, offer opportunities for more precise control and reduced ambiguity features advantageous in high-noise, multitasking, or low-resource environments. Our solution to the challenge is to accept half-duplex constraints "as they are" and attempt to turn them into HCI advantages, rather than masking or working around them. By designing around these constraints, we seek to leverage their predictability and transparency to make turn-taking that is both robust and understandable to users.

2.3.5 System feedback in turn-taking

User expectations tend to rise with the popularity of Voice User Interfaces (VUIs). However, humans must work to adapt their communication patterns to the needs of the machines rather than the machines adapting to humans Balentine (2007a); Jiang et al. (2013). It appears that the designers of the current VUI systems make no effort to create a conversational system where human users can adapt to the machine's behaviour during a conversation. To achieve that the designers should present the VUI system with discoverability features Kirschthaler et al. (2020); Jamshed et al. (2025); Furqan et al. (2017). However, the present VUI designs lag in providing appropriate feedback and incorporating relevant discoverability features about the system's capabilities and intelligence. Users do not find opportunities to experiment with newer tasks without appropriate feedback. Instead, they shorten their sentences or use simplified language with deliberate repetitions to get understood by the VUIs Jiang et al. (2013); Kennedy et al.

(1988); Lohse et al. (2008); Pelikan and Broth (2016). The interactions suffer from moments of frustration with no success in delivering the intended task. Luger and Sellen (2016), through semi-structured interviews with 14 regular users of VUIs, reports users feeling caught in a state of indecisiveness while interacting with the VUI. Users either feel overwhelmed (with the unknown potential of VUI) or assume several limitations. They express their desire to access the system's intelligence. They wonder whether it is the VUI which will adapt to their behaviour or the opposite. Subsequently, users misunderstand the context of their conversation with the VUI. They conclude that the VUI should communicate its system intelligence and capabilities effectively. Sciuto et al. (2018) conducted a study of specific interest about using VUIs in households. They reveal how VUI's lack of discoverability features limits users from experimenting with newer tasks in their interactions with VUIs. They believe that the lack of proper system feedback/affordances in the current VUI presents a challenge for users in discovering VUI features. Thus, to have a successful human-VUI conversation, there is an urgency to redesign the feedback system of VUIs to help users adapt or entrain to the VUI's behaviour. Beyond improving usability, system feedback also influences how users coordinate their speech patterns with the VUI. When feedback cues are clear, users can better time their responses, adjust their pacing, and align their speech rhythm with the system's turn-taking mechanisms. This phenomenon, entrainment in conversation, is crucial in facilitating smooth and fluent conversations with VUIs.

2.3.6 Speech entrainment in voice-based conversation

Speech entrainment is the conversational partners' tendency to become similar during conversation Beňuš (2014). Different studies have consistently shown that conversational partners often adjust their speaking style across multiple dimensions such as tone, linguistic style, speed of speech, pitch, voice quality, loudness, lexical choice, and phonetic attributes Tannen (2007); Levitan (2014); Brennan (1996). Natale (1975) discovered that an interviewer's vocal intensity can directly influence that of the interviewee. Heldner et al. (2010) observed that speakers align their pitch with conversational partners when providing a backchannel response. Similar alignment is also seen in lexical choices and syntactic structures during interactions Brennan (1996). Entrainment occurs in human-human interaction (HHI) and human-computer interaction (HCI),

influencing user experience, trust, and communication effectiveness. The following tables 2.1 and 2.2 summarise studies on speech entrainment in both HHI and HCI, highlighting its role in shaping conversational dynamics.

The figure 2.1 illustrates lexical and syntactic entrainment in human-human conversation: Speakers A and B align their language choices during conversation. The top exchange demonstrates lexical entrainment through repeated words like "amazing" and "immersive." In contrast, the bottom exchange showcases syntactic entrainment as both speakers adopt similar sentence structures when negotiating weekend plans.

The figure 2.2 demonstrates lexical and syntactic entrainment in human-VUI conversation: The top dialogue shows lexical entrainment as both the human (A) and the voice user interface (B) reuse terms like "showers" and "chilly." The bottom exchange highlights syntactic entrainment, where the VUI mirrors the user's sentence structure when offering response options, reflecting adaptation in conversational form.

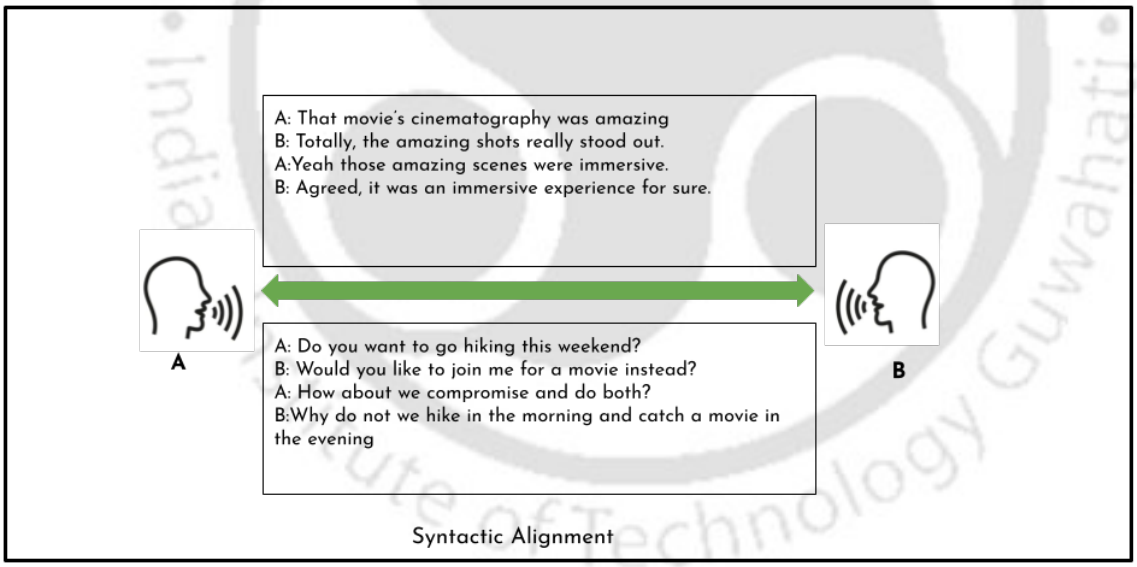


Figure 2.1: Entrainment in Human (A) -Human (A) conversation.

2.3.7 Effect of non-speech audio cues in human–computer conversation

Non-speech audio cues—short sounds that carry interface meaning without using linguistic content—have long been studied as a lightweight channel for communicating system state,

Table 2.1: Examples of acoustic-prosodic entrainment in human-human (HHI) and human-computer (HCI).

Type (HHI/HCI)	Reference	Effect on Conversation
HHI	Natale (1975)	Vocal intensity of speakers aligns, impacting conversational dynamics.
HHI	Heldner et al. (2010)	Pitch alignment occurs in backchannel responses, improving interaction fluidity.
HHI	Levitan and Hirschberg (2011)	entrainment in prosodic features enhances interaction smoothness.
HCI	Levitan et al. (2016)	Implementing prosodic entrainment in avatars improves naturalness of interaction.
HCI	Gálvez et al. (2020)	In spoken dialogue systems, acoustic–prosodic entrainment can shape user trust, with intensity entrainment boosting trust and pitch entrainment reducing it.
HCI	Benus et al. (2018)	Trust in VUI systems is enhanced through prosodic alignment.
HCI	Paletz et al. (2023)	Teams with higher agreeableness and lower neuroticism exhibited greater acoustic–prosodic entrainment, enhancing their coordination and task performance.
HCI	Cohn et al. (2021)	multimodal entrainment synchronizing speech, gestures, and other behaviors enhances rapport and communication in human–robot interaction.
HHI	Wynn et al. (2022)	Prosodic entrainment aids in successful speech interactions through rhythmic adaptation.

Table 2.2: Examples of lexical and syntactic entrainment in human-human (HHI) and human-computer (HCI)

Type (HHI/HCI)	**Reference**	**Effect on Conversation**
HHI	Brennan and Clark (1996)	Speakers naturally adopt similar word choices, enhancing communication efficiency.
HHI	Wynn et al. (2023)	Speech entrainment predicts conversational quality and efficiency in adolescents.
HHI	Giles (1975)	Syntactic mirroring facilitates smoother conversation flow.
HHI	Pickering and Garrod (2004)	Syntactic priming enhances mutual understanding in conversation.
HCI	Huiyang and Min (2022)	Virtual agents with lexical entrainment are perceived as more competent.
HCI	Huiyang and Min (2022)	Shifting from lexical alignment to non-alignment negatively affects perception of AI.
HHI	Wynn et al. (2023)	Speech entrainment predicts conversational quality and efficiency in adolescents.
HHI	Giles (1975)	Syntactic mirroring facilitates smoother conversation flow.
HHI	Pickering and Garrod (2004)	Syntactic priming enhances mutual understanding in conversation.

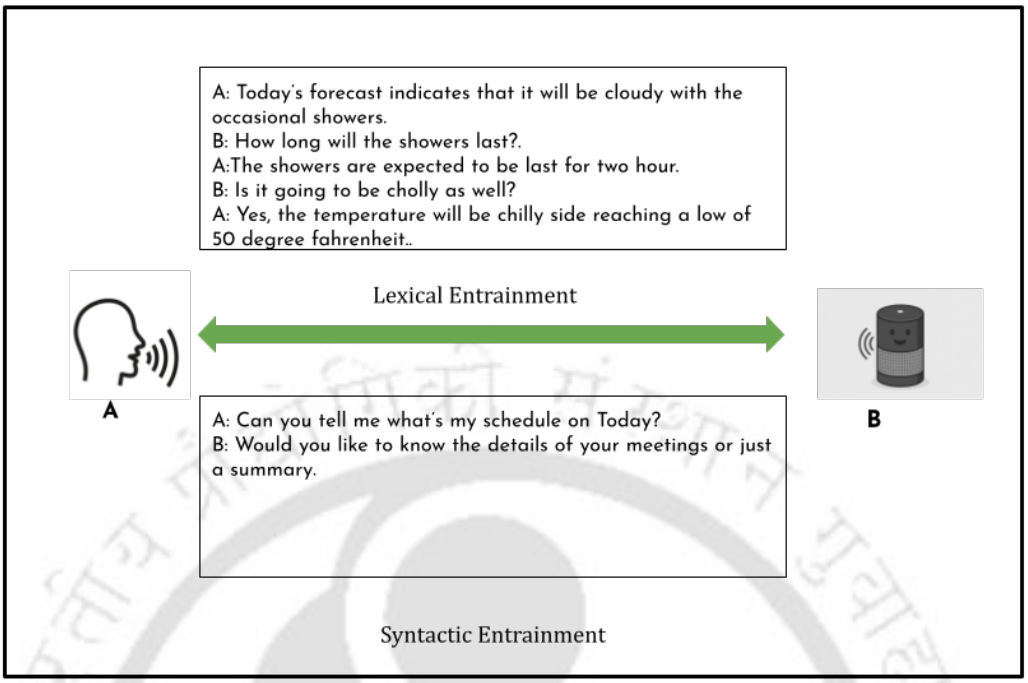


Figure 2.2: Entrainment in Human (A)-Machine (B) conversation.

events, and timing in interactive systems. In contrast to spoken feedback, non-speech cues can be language-independent, fast to render, and less disruptive when the primary task already demands speech, visual attention, or both. Classic HCI work frames these cues as a family of auditory signs that can support interaction by (i) signalling state transitions (e.g., *listening*, *processing*, *error*), (ii) providing confirmation and progress feedback, and (iii) shaping user timing and turn boundaries when interaction is time-critical or when the interaction channel is constrained Brewster et al. (1993); Gaver (1993).

2.3.7.1 Forms of non-speech cues: auditory icons, earcons, and related signals

Non-speech cues are often discussed through two foundational design metaphors: **auditory icons** and **earcons**. **Auditory icons** use everyday sounds mapped by analogy to events or actions (e.g., a crumpling sound for "delete"), leveraging listeners' ability to interpret familiar sound-producing events Gaver (1986, 1993). Because their meaning can be inferred from real-world associations, auditory icons typically require less explicit training, but they can be difficult to scale when a system needs a large vocabulary or when no clear real-world analogy exists Hermann et al. (2011).

Earcons, in contrast, are *abstract* musical or synthetic motifs whose meanings are learned rather than inherently inferred. Early work formalised earcons as structured non-verbal messages for representing interface entities and events, and proposed compositional approaches (e.g., hierarchical earcons) to scale to larger sets Blattner et al. (1989); Brewster et al. (1993). This structure makes earcons attractive for representing families of states (e.g., related system modes) and for enforcing consistency across an interface, but it also introduces classic challenges: learnability, confusability among similar motifs, and user annoyance if cues are frequent or poorly matched to context **?**Hermann et al. (2011).

Beyond these two core categories, later work explores additional cue types that sit between speech and non-speech, such as **spearcons** (time-compressed speech that becomes an identifiable auditory "fingerprint") and hybrid systems that combine musical structure with more iconic elements Dingler et al. (2008); Hermann et al. (2011). These alternatives are particularly relevant to conversational interfaces because they offer a spectrum of trade-offs: immediate interpretability (speech-like) versus compactness and low interference (more earcon-like).

2.3.7.2 Why non-speech cues matter for conversation and timing

In human–computer conversation, users continuously form expectations about *when* they should speak, *whether* the system is attending, and *what* will happen next. Non-speech cues can support these expectations by making temporal structure more explicit. This is especially valuable in voice-first contexts where audio is transient and cannot be visually scanned, increasing memory load and the cost of misunderstanding Chuklin et al. (2019). Even small, well-timed cues can function as *temporal anchors*—signalling boundaries such as "the system is ready now", "your input was captured", or "I'm still working"—and thereby reduce premature interruptions and unnecessary repetitions.

This timing role becomes even more central under half-duplex constraints (common in many deployed voice assistants and push-to-talk systems), where the system cannot listen while playing audio and where turn boundaries must be managed explicitly. In such settings, non-speech cues can provide a compact and reliable way to indicate "listening windows" and to prevent users from speaking into a conversational "blind spot" (i.e., speaking while ASR is unavailable).

2.3.7.3 Evidence from broader HCI domains that transfer to VUIs

Although many studies of non-speech cues originate outside VUIs, their findings generalise to conversational settings because they reveal how sound design affects recognition, response time, and trust in system state signalling. For example, Brewster et al. experimentally evaluated earcons as a method for communicating interface information and showed that structured earcons can outperform unstructured sound bursts, with additional design refinements improving recognition accuracy Brewster et al. (1993). This line of work supports a core thesis claim: *non-speech audio can be engineered as an interface language*, rather than treated as ad hoc beeps.

In safety-critical and time-critical contexts such as driving, the requirement is not merely recognition but rapid response under divided attention. Kutchek and Jeon examined non-speech auditory displays for takeover and handover requests in semi-automated vehicles and reported statistically significant differences in mean reaction time across display types, highlighting that detailed acoustic parameters (e.g., onset/interval patterns) can materially affect user responsiveness Kutchek and Jeon (2019). While automotive control transitions differ from conversational turn-taking, both share a key property: users must act at the right moment, under uncertainty, often while attention is split. This makes the automotive literature a useful analogue for designing timing signals in speech interfaces.

2.3.7.4 Prosodic and non-verbal audio modifications in voice-only interaction

Non-speech signalling in voice systems is not limited to "separate" tones; systems can also manipulate the acoustic form of their spoken output to improve comprehension and control pacing. Chuklin et al. investigated prosody modifications (e.g., pauses, speaking rate, pitch/emphasis changes) for voice-only question answering and found that some modifications improve comprehension and help users identify key answer parts, at the cost of slightly reduced perceived naturalness Chuklin et al. (2019, 2018). This result is important for VUI design: it shows that timing and emphasis cues can be integrated into speech output itself, offering a design space that complements (rather than replaces) discrete non-speech tones.

From a turn-taking perspective, such prosodic shaping can influence users' expectations about completion points and response timing (e.g., a pause before the crucial answer segment)—

a mechanism that can be aligned with structured turn protocols in constrained VUIs.

2.3.7.5 Learnability and the "cost" of sound vocabularies

A persistent concern in auditory interface design is whether users can reliably learn and retain cue meanings, especially when cue sets grow beyond a few items. Dingler and Lindsay compared learnability across cue types (auditory icons, earcons, spearcons, and speech) and discuss how different cue families impose different training and memory demands Dingler et al. (2008). The Sonification Handbook chapter on earcons synthesises broader evidence that training can substantially improve earcon identification, but also warns that without adequate onboarding and careful mapping, users may perceive frequent cues as annoying and may disable them Hermann et al. (2011). These findings translate directly to VUIs: a cue that is helpful during onboarding can become irritating in everyday use if it is too frequent, too loud, or not clearly tied to user goals.

More recent synthesis work continues to treat auditory icons, earcons, spearcons, and speech as a design space of recognisable auditory messages, arguing that each has characteristic strengths for speed, scalability, and interpretability depending on context and task Nees and Liebman (2023). For conversational systems, this implies that designers should avoid a "one cue fits all" approach; instead, the cue family and acoustic design should be chosen based on the interaction moment (e.g., error prevention vs. confirmation vs. progress vs. turn boundary marking).

2.3.7.6 Design implications for half-duplex VUIs and turn-taking protocols

Taken together, this literature supports three design implications relevant to half-duplex VUIs and the present thesis. First, non-speech cues can efficiently communicate state and timing when the speech channel is constrained, and they can reduce turn-taking errors by clarifying when users should speak. Second, cue design must treat sound as an interaction language: cue families should be systematic (not arbitrary), and their acoustic parameters should be tuned for discriminability and context-appropriate urgency Brewster et al. (1993); Kutchek and Jeon (2019). Third, training and long-term acceptability are first-order concerns: cue vocabularies should be kept minimal, reinforced through consistent mapping, and evaluated not only for

recognition but also for annoyance and habituation in repeated daily useHermann et al. (2011); Dingler et al. (2008).

Despite decades of auditory display research, two gaps remain especially relevant for VUI turn-taking. (i) Much of the evidence comes from GUI navigation, alarms, or non-conversational tasks; fewer studies directly test non-speech cues as turn-management mechanisms in real-world, task-oriented spoken interaction. (ii) There is limited consensus on *optimal timing* and *integration rules*—for example, how a tone should be placed relative to speech output and ASR availability to maximise alignment without increasing cognitive load. These gaps motivate the thesis focus on designing structured, mechanomorphic turn cues (including corrective tones) and empirically evaluating their effects on user entrainment and breakdown reduction.

2.3.8 Necessity of mechanomorphic design in voice user interfaces

Upon reviewing the literature on Voice User Interface (VUI) design, it became evident that a substantial portion of existing research and commercial implementation is rooted in anthropomorphic design philosophies. In this approach, interfaces are designed to mimic human-like characteristics, such as tone, emotional cues, or conversational styles, assuming that human-likeness fosters more engaging and natural interactions between users and systems. This perspective has gained traction due to its intuitive appeal: users may feel more comfortable and communicative if the interface behaves more like a person.

However, the anthropomorphic design paradigm also introduces notable challenges. One of the most critical issues is the emergence of a habitability gap, a mismatch between user expectations and the system's actual capabilities. As systems appear more human-like, users may overestimate their conversational intelligence or contextual awareness, leading to frustration when those expectations are unmet. This can ultimately result in disengagement or reduced task success, particularly in high-stakes or task-oriented settings. Furthermore, anthropomorphism carries the psychological risk of evoking the 'uncanny valley' effect, in which systems that appear almost human, but not quite, trigger discomfort or eeriness in users Mori et al. (2012).

In response to these challenges, a growing body of research advocates for a mechanomor-

phic design approach where the system explicitly presents itself as a machine rather than attempting to simulate human behaviour Takayama et al. (2011); Bretan et al. (2015). Unlike anthropomorphic systems, mechanomorphic VUIs are intentionally designed to reveal their machine-like nature, which can help align user expectations with system capabilities and foster more efficient, task-oriented interactions. Studies have shown that such systems avoid the uncanny valley effect and may, in fact, support better task performance and smoother coordination in constrained environments Lubold et al. (2015); Balentine (2007b). For example, research on prosodic entrainment has demonstrated that mechanomorphic designs can prompt users to naturally entrain to the system, without requiring the interface to simulate complex human-like behaviour Lubold et al. (2015). Desai et al. (2024) supports a mechanomorphic approach to VUI design by showing that non-human metaphors like calculators or encyclopedias can be equally effective and sometimes preferable, especially when their metaphorical nature is implicit, thus aligning more closely with users' mental models of machines rather than humans.

By avoiding the complications of human mimicry and instead emphasising transparent, structured, and machine-consistent interaction patterns, mechanomorphic design becomes a necessary and effective alternative, especially in contexts such as half-duplex systems, where conversational limitations are inherent. Rather than deceiving users into expecting more naturalness than the system can deliver, mechanomorphic VUIs aim to make system states explicit, encourage coordination, and ultimately enhance usability without overpromising capabilities.

2.4 Research gap and objectives

Based on the literature review, the research gap we found is-

2.4.1 Limited focus on half-duplex turn-taking for VUIs:

With the advancement of Artificial Intelligence (AI) and Natural Language Processing (NLP), contemporary research in Voice User Interfaces (VUIs) has increasingly focused on achieving naturalistic, full-duplex interactions that emulate human-human conversation. While this am-

bition is technologically commendable, our literature review reveals that such systems often encounter significant limitations, particularly in managing the complexity of human conversational dynamics. In striving to replicate natural dialogue, these systems risk becoming unwieldy, error-prone, and resource-intensive.

In contrast, half-duplex systems where interlocutors cannot speak and listen simultaneously present a pragmatic alternative that remains relatively underexplored in academic research. Despite their constrained communication model, half-duplex architectures continue to be employed in many real-world applications, including in-car voice assistants, smart appliances, and low-bandwidth environments. These implementations are often driven by practical constraints such as limited processing power, cost considerations, or domain-specific simplicity.

The relative neglect of half-duplex systems in VUI research may stem from the perception that they lack the conversational naturalness associated with full-duplex interactions. However, this perspective overlooks a critical opportunity: to reframe the half-duplex mode not as a deficiency, but as a design strength. By embracing a mechanomorphic interaction model where the system transparently communicates its machine-like operation and turn-taking boundaries designers can promote temporal coordination without the cognitive and technical burdens of human mimicry.

Moreover, the half-duplex format offers distinct interactional advantages, particularly in task-oriented domains. These include increased predictability, reduced ambiguity, and lower cognitive demand features especially valuable in contexts requiring safety, clarity, and user compliance. For instance, McWilliams et al. (2015) demonstrate how turn-taking delays in vehicular voice interfaces can detract from driver attention and performance, highlighting the importance of prompt and clearly demarcated system responses. Similarly, Zhang et al. (2024) integrate a half-duplex training phase within their OmniFlatten model to ensure modality alignment and stability before transitioning to more complex conversational structures.

Despite these compelling use cases, the systematic design and evaluation of half-duplex turn-taking protocols remain markedly underrepresented in the literature. There is a lack of cohesive frameworks or design guidelines that address how these systems might be optimised for usability, robustness, and task success. This thesis seeks to fill that gap by re-conceptualising duplex interaction as a viable and strategically advantageous design paradigm. Rather than treating half-duplex as a stopgap or constraint, we argue for its deliberate adoption in contexts

where structured, machine-led turn-taking can enhance the overall interaction experience.

2.4.2 Lack of systematic integration of entrainment principles in VUIs

While conversational entrainment is well-documented in human-human interaction (HHI), its potential application in Voice User Interfaces (VUIs), particularly under constrained conditions such as half-duplex systems, has not been systematically investigated. Existing studies have primarily examined entrainment across syntactic, semantic, lexical, and prosodic dimensions, demonstrating that speakers tend to align their language and speech patterns during interaction. However, these studies largely focus on natural, full-duplex conversation scenarios, often with anthropomorphic or open-domain systems.

There is a notable lack of research exploring temporal aspects of entrainment specifically how timing and turn-taking behaviors unfold between human users and VUIs. In half-duplex systems, where speech input and output occur sequentially rather than simultaneously, precise coordination of turn boundaries becomes essential. Yet, few studies have examined how users adapt to machine-imposed timing constraints, or how VUIs might be designed to support mutual temporal alignment. This gap is especially critical in task-based interactions such as form-filling or structured input, where breakdowns often occur due to mistimed responses or ambiguous turn-transition cues.

Therefore, the temporal dynamics of turn-taking and their relation to conversational entrainment in human-VUI interaction remain an underexplored area, revealing a significant research gap in both the theoretical understanding and practical design of VUIs.

2.4.3 Insufficient exploration of non-speech audio cues for turn-taking

While prior work acknowledges that tones, beeps, and structured pauses can assist in managing turn-taking, there is currently no established framework guiding the systematic use of non-speech cues in half-duplex VUIs. Studies included in the literature review suggest that non-verbal sounds can improve user awareness of system state, but their application has often been

limited to narrow use cases (e.g., accessibility, automotive alerts) rather than general-purpose conversational interfaces. The type, timing, and contextual appropriateness of these non-speech cues remain underexamined, especially in task-based dialogues where user expectations for fluid turn exchanges are high. Thus, the effectiveness of different non-speech cues in conveying system readiness, encouraging user pause, or preventing interruption is not well-documented.

2.4.4 Insufficient research on mechanomorphic turn-taking strategies in VUIs:

Most Voice User Interface (VUI) turn-taking strategies today are grounded in anthropomorphic design principles attempting to replicate the fluid, implicit, and often overlapping dynamics of human-human conversation. These systems rely heavily on human-like cues such as prosodic variation, verbal acknowledgements, and conversational backchannels to simulate natural interaction patterns Lin et al. (2022). The underlying assumption is that making the system "more human" enhances intuitiveness and trust. However, in practice, especially within half-duplex architectures or task-constrained domains, this anthropomorphic mimicry often leads to confusion, mistimed user responses, and ungraceful recovery from interruptions.

Despite the prevalence of anthropomorphic metaphors, there is a paucity of research directly comparing anthropomorphic and mechanomorphic approaches in turn-taking. Mechanomorphic design, which treats the VUI as a transparent, machine-like actor that communicates its constraints clearly rather than concealing them behind human-like behaviour offers an alternative paradigm that could yield higher predictability, user control, and efficiency in constrained interactions Kirschthaler et al. (2020). Yet, this paradigm remains largely under-theorised and empirically untested in VUI turn-taking design.

Furthermore, few studies have explored how users perceive, adapt to, or prefer one strategy over the other in different interaction contexts (e.g., goal-oriented tasks vs. casual chat). Even less is known about the long-term effects of these approaches on user experience, learnability, and task success. This lack of comparative analysis represents a significant research gap, especially when VUIs are being deployed in increasingly diverse and resource-constrained environments where design clarity and timing reliability matter more than social mimicry.

By synthesising findings from literature, this review underscores the importance of manipulating temporal aspects of VUIs to foster user entrainment and improve turn-taking reliability. The insights gained inform the development of a turn-taking protocol that strategically employs non-speech audio cues and explicit prompts.

2.5 Summary

The literature review underscores the necessity of a turn-taking protocol that leverages non-speech audio cues and explicit prompts to enable users to entrain to it. This approach is particularly effective in half-duplex VUIs, where the system's inability to handle simultaneous input and output disrupts natural conversational flow.

The decision to adopt a mechanomorphic design is driven by its cost-effectiveness and practicality for quick form-filling applications. Unlike anthropomorphic systems that mimic human conversation, this approach avoids the complexities and expenses of full-duplex interaction. While full-duplex platforms exist, they introduce distinct HCI challenges beyond this study's scope.

By focusing on clear, machine-like signals and structured turn-taking cues, this research aims to reduce cognitive load and improve task efficiency, thereby addressing the critical gaps identified in the literature.

Chapter 3

Interaction Failures in VUIs in Real-World Usage

3.1 Overview

This chapter serves as an initial, exploratory investigation into how conversational errors emerge during real-world interactions with widely used Voice User Interfaces (VUIs) such as Amazon Alexa, Google Assistant, Apple Siri, and Microsoft Cortana. The purpose of this exploratory study was to identify and categorise the different types of errors that occur in everyday use, and to understand their underlying causes. Since the thesis focuses on turn-taking in VUI interaction, a key aim was to determine whether turn-taking-related failures form a substantial proportion of overall conversational errors. The intention is not to evaluate the design of specific systems, but to reveal generalisable patterns of breakdown that can guide the design of more effective turn-taking strategies and feedback mechanisms in future VUI development.

3.2 Introduction

Voice interaction has increasingly become a primary modality for accessing digital services through devices such as smartphones, smart speakers, and home automation systems. Voice User Interfaces (VUIs), embodied in widely used systems like Amazon Alexa (Amazon Inc.), Google Assistant (Google LLC), and Apple Siri (Apple Inc.), promise hands-free, natural communication with technology. Framed as "conversational agents," these systems suggest an interaction style where users can "just talk" and expect efficient task fulfilment. Recent studies have highlighted the growing ubiquity of VUIs in everyday life and their expanding role across domestic, professional, and public environments Klein et al. (2024); Porcheron et al. (2018). In parallel, advancements in natural language processing, dialogue systems, and human–computer interaction research have fuelled optimism about the potential for increasingly sophisticated conversational agents Ni et al. (2023); Wu et al. (2020). However, ensuring quick and successful task completion remains a critical challenge, particularly in managing turn-taking and recovering from errors in real-time interactions.

Despite significant advances in Artificial Intelligence (AI), Automatic Speech Recognition (ASR), and Natural Language Understanding (NLU), Voice User Interfaces (VUIs) continue to face persistent challenges in achieving consistent task success Liesenfeld et al. (2023); Goetsu and Sakai (2020). Studies have documented recurring issues such as misrecognition, timing delays, premature interruptions, and breakdowns in conversational flow across a range of devices and contexts. While VUIs are now embedded in everyday life from smartphones and smart speakers to in-vehicle systems and wearables the majority of research has examined these systems through controlled experiments, technical benchmarks, or log-based error analyses. Although recent work has begun to capture real-world use, this body of evidence remains comparatively limited in scope and depth. In particular, there is a lack of detailed understanding of how users experience and adapt to turn-taking breakdowns, misrecognitions, and system responses in the fluid and unpredictable settings of everyday interaction. Addressing this gap requires identifying recurring breakdown patterns, categorizing error types, and uncovering their underlying causes, with the ultimate aim of linking these insights to improved VUI design and interaction strategies.

By systematically exploring conversation breakdown patterns, researchers can identify the underlying causes of interactional failures, moving beyond surface-level errors to deeper usability and design issues. This clarity enables targeted interventions, enhancing VUI effectiveness. Conversational breakdown patterns can reveal implicit user expectations and behaviours that users themselves might not articulate. Recognising these patterns helps designers better align system capabilities with user mental models, thereby improving the user experience.

Categorising errors into clearly defined themes provides a structured analytical framework. This approach supports rigorous and replicable analysis, facilitating clearer communication of findings within the scholarly community. Thematic categorisation also helps identify error clusters that are frequent, critical, or most detrimental to task success, enabling designers and developers to allocate limited resources to areas where intervention is most urgent. Grouping themes systematically allows insights from one study to be more readily applied across different VUI platforms or contexts, thereby maximising both academic and practical impact.

This chapter reports on an exploratory study systematically analysing real-world human–VUI conversations to identify recurring interaction breakdown patterns, categorise conversational errors into meaningful themes, and examine their implications for task success. Since the wider thesis focuses on turn-taking, a key aim here was to determine whether turn-taking-related failures form a substantial proportion of overall conversational errors observed in real-world contexts. The following research question guides this chapter

1. **RQ**: How do conversational errors emerge, and what are the types of errors that occur during interactions with existing real-world Voice User Interfaces (VUIs)?

Thus the two research objectives are

1. To determine interaction patterns that lead to the emergence of conversational errors in real-world VUI interactions.
2. To identify, categorize, and describe the different types of conversational errors that occur during user interactions with existing VUIs.

The primary contribution of this study is to provide a systematic, empirical understanding of conversational breakdowns within real-world interactions with Voice User Interfaces (VUIs).

By rigorously identifying recurring conversational patterns and categorising errors into coherent, actionable themes, the research offers practical insights into the critical points at which interactions fail. This thematic categorisation not only facilitates targeted improvements in VUI design but also advances theoretical understanding by highlighting user behaviours, expectations, and system limitations, with particular emphasis on the role of turn-taking-related failures

3.3 Related Study

Voice User Interfaces (VUIs) have become increasingly prevalent in consumer technology, yet their interaction paradigms often fall short of user expectations. A significant body of research has examined the shortcomings of VUIs, particularly focusing on inadequate feedback mechanisms and the anthropomorphic nature of these systems.

Sciuto et al. (2018) examined how everyday users understand conversational agents and found that people often hold fragmented and inaccurate mental models of system capabilities. This confusion is reinforced by the limited feedback provided during interactions, leaving users unsure of what the system can or cannot do.. Similarly, Porcheron et al. (2018) analyzed audio data from month-long deployments of Amazon Echo, emphasizing the importance of system response design as an interactional resource for users.

Pyae and Joelsson (2018) surveyed 114 Google Home users, identifying common issues such as misrecognition of non-English words, the necessity to repeat commands, and the inability to process multiple commands in a single transaction. These findings underscore the limitations of current VUIs in handling diverse user inputs effectively.

Recent studies have delved into the timing and nature of VUI feedback Wang et al. (2023) investigated the optimisation of feedback mechanisms based on time perception, finding that a feedback time of 750 ms yields the best user experience, while delays beyond 1,850 ms lead to negative emotional responses. This research highlights the critical role of timely feedback in user satisfaction.

Furthermore, studies have shown that users often struggle with the discoverability of VUI

capabilities due to the absence of visual cues and inadequate feedback. This limitation is particularly pronounced in multi-turn conversations, where the system's inability to track context and manage dialogue flow can lead to user confusion and task failure.

In summary, existing literature highlights the critical importance of effective feedback mechanisms in VUIs. The anthropomorphic nature of current systems, characterised by rigid turn-taking and a lack of adaptive feedback, contributes significantly to user dissatisfaction and task failure. Addressing these issues requires a concerted effort to develop VUIs that align more closely with human conversational behaviours and expectations. While prior research has extensively documented the limitations of VUIs such as poor user mental models, inadequate feedback, rigid turn-taking, and system design fragmentation few studies have offered a structured, real-world categorisation of how these interactional failures actually unfold during user-VUI exchanges. Existing work tends to focus either on system-level evaluations or retrospective user accounts, often lacking fine-grained, empirical analysis of naturalistic voice interactions.

Moreover, although issues like misrecognition, response delays, and overlapping speech constraints have been individually reported, there is little understanding of how these issues collectively manifest and compound during task-oriented VUI use. There is also limited evidence on how users attempt to recover from such breakdowns, or how conversational errors can be systematically classified in a way that informs feedback and turn-taking design.

This study addresses these gaps through an exploratory study, in-depth analysis of real-world human–VUI conversations, focusing on the interplay between system responses, user behaviour, and resulting task outcomes. The contribution is twofold. First, it generates a **conversation breakdown map** that traces how interactions progress toward either successful completion or failure, revealing common divergence points and recovery attempts. Second, it applies **thematic analysis** to systematically classify recurring error types into structured, actionable themes. Together, these outputs provide both a temporal view of how conversational failures unfold and a conceptual framework for understanding their root causes. This dual perspective offers a strong foundation for designing category-specific, mechanomorphically aligned feedback strategies that better manage turn-taking and improve user experience. In doing so, this work not only deepens our understanding of how VUI errors emerge in real-life settings but also provides a foundation for designing category-specific, mechanomorphically-aligned feedback

strategies that enhance turn-taking and improve user experience.

3.4 Task design and data collection procedure

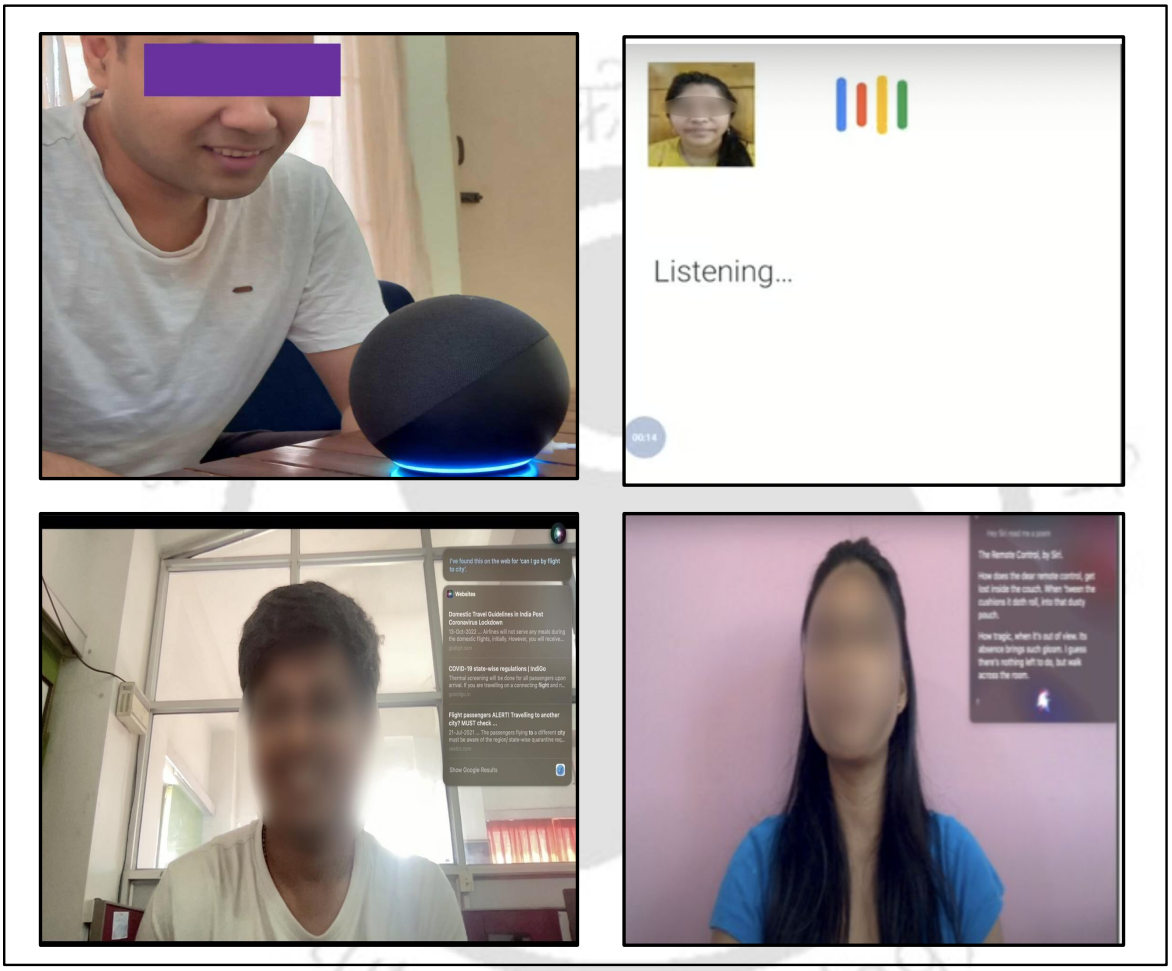


Figure 3.1: Participants taking part in conversation with different VUIs

To balance ecological validity with procedural control, participants were *explicitly instructed* to independently initiate and record naturalistic interactions with a Voice User Interface (VUI) integrated into their own mobile phones or laptops. Ecological validity was prioritised by allowing participants to select tasks they would normally perform in their everyday routines (e.g., setting reminders, asking for information, getting directions, or completing digital transactions) rather than following a researcher-provided script. This ensured that the captured interactions reflected genuine goals and spontaneous dialogue, while still adhering to consistent recording requirements.

Before any data collection, written informed consent was obtained from all participants. The consent form detailed the purpose of the study, the nature of the recordings, the storage and anonymisation process, and the participants' rights, including the ability to withdraw at any stage without penalty.

Participants were given two mandatory instructions: (1) to complete a minimum of three recording sessions at different times of day, and (2) to use at least two different task types across sessions. Beyond these constraints, task selection, phrasing, and interaction style were left entirely to participant discretion. This design minimised researcher influence while ensuring sufficient variation in both VUI behaviour and conversational context.

Each recording session was required to capture both *(a)* the participant's face and body language (via a front-facing camera) and *(b)* the VUI device screen (via built-in screen recording or an external camera). This dual-capture approach enabled fine-grained analysis of both verbal and non-verbal user behaviours (e.g., facial expressions, hesitations, repetitions) alongside system responses. An example of a participant engaging with a VUI is shown in Figure 3.1.

Participants were instructed to record in a quiet, familiar environment (e.g., home, office) to minimise background noise and reduce external distractions. After completing each session, participants uploaded the raw, unedited files directly to a secure, access-restricted online platform provided by the research team. All anonymisation including removal or blurring of personally identifiable visual elements and redaction of identifiable spoken content was performed by the research team prior to analysis.

The final dataset comprised 2 hours and 40 minutes of conversational video material across all participants, with individual session lengths ranging from 1–6 minutes. All recordings were transcribed verbatim by the research team, including annotations for relevant non-verbal cues such as pauses, overlapping speech, and facial expressions.

This task design produced a dataset that was both ecologically valid and methodologically robust. It captured the diversity and unpredictability of everyday VUI use while ensuring a level of procedural standardisation necessary for systematic comparison. The resulting recordings formed the empirical basis for constructing the *conversation breakdown map* and conducting the *thematic analysis* presented later in this chapter.

3.5 Sampling method

Participants were recruited through voluntary participation using a convenience sampling approach. This method aligned with the exploratory, naturalistic focus of the study, which aimed to capture real-world interactional breakdowns rather than achieve demographic representativeness. Although no age-based restrictions were imposed during recruitment, the final participant pool happened to fall within the 25–33 age range. This was simply a reflection of the individuals who responded to the open call within our community networks, where active VUI users are predominantly young adults. All volunteers who self-reported using a Voice User Interface (e.g., Alexa, Google Assistant, or Siri) at least three times per week were included. Recruitment announcements were shared through local community groups and personal networks, and interested individuals contacted the research team directly for screening. Participation was voluntary, with no monetary incentives; however, a brief summary of findings was provided after study completion. While this convenience-based approach naturally limited demographic diversity, it effectively supported the study's primary goal: identifying common categories of turn-taking errors and conversational breakdown patterns in everyday VUI use.

3.6 Participants

A total of 22 participants contributed to the study, generating 97 video recordings of real-world VUI interactions. Participants ranged in age from 25 to 33 years ($M = 28.4$, $SD = 2.5$). The narrow age range reflects the convenience-based recruitment process, which primarily drew from professional and university networks; as such, findings may not generalise to older or younger populations.

All participants self-reported fluency in English. While most participants were L2 English speakers, all reported regular interaction with English-language VUIs and demonstrated the ability to converse fluently during the recordings.

Inclusion criteria required participants to:

1. Use a Voice User Interface (e.g., Amazon Alexa, Google Assistant, Apple Siri) on a per-

sonal device at least three times per week for everyday tasks such as information retrieval, reminders, navigation, or entertainment.

2. Own a compatible device (smartphone, smart speaker, or laptop) with a functioning VUI.

3. Be aged 18 or older.

Exclusion criteria included:

1. Professional or academic background in speech technology, conversational AI, or related fields.

2. Any self-reported hearing or speech impairment likely to interfere with normal interaction with a VUI.

The sample consisted of 12 female and 10 male participants, with varied occupational backgrounds (e.g., postgraduate students, undergraduate students, educators, and corporate employees). Participants primarily used smartphones (n = 17) and smart speakers (n = 5) as their interaction devices. All participants used their own devices in familiar environments to preserve ecological validity.

While the sample reflects typical VUI users within the specified age range, it was not intended to be demographically representative. The recruitment approach and participant profile should therefore be considered when interpreting the transferability of the findings.

3.7 Data analysis

3.7.1 Overview of approach

Our analysis is organised to answer the chapter's research question: *How do conversational errors emerge and what types of errors occur during interactions with real-world VUIs?* To do so, we applied two complementary, purpose-built approaches to the 86 transcribed and time-aligned recordings.

Approach 1: Turn-by-turn Multimodal Observation The approach was conducted to address the first research objective mentioned earlier. We conducted a systematic turn-by-turn analysis of each interaction to identify the sequences leading to task success or failure. Using the multimodal transcripts (speech plus visible user behaviours), we traced the sequence of user actions and system responses, noted pivotal junctures (e.g., delayed feedback, premature cut-ins, context loss), and counted the outcomes of users' repair attempts. These observations were formalised into a general state–transition representation the *conversation breakdown path* that captures typical routes to task completion versus breakdown.

Approach 2: Thematic Analysis of Error Types. In parallel, we conducted an inductive thematic analysis to identify *what kinds* of errors recur across systems and tasks. Independent coders generated and consolidated codes describing breakdown phenomena (e.g., misrecognition, turn-taking failures, repair failures), which were then grouped into coherent higher-level themes. The resulting *thematic map* provides a structured taxonomy of error categories that complements the flow-oriented view from Approach 1.

Together, these analyses link interaction *dynamics* (Approach 1) with error *kinds* (Approach 2), allowing us to explain both the trajectories that lead to success or breakdown and the specific classes of failures that most often drive those trajectories. The next subsections detail each approach, followed by a Results section that presents the state–transition diagram and the final thematic map.

3.7.2 Approach 1: Turn-by-turn multimodal observation

To address the first part of the research question identifying how conversational errors occur in real-world Voice User Interface (VUI) use we conducted a turn-by-turn analysis of recorded interactions. This approach was informed by principles of conversation analysis Sacks et al. (1974) but extended beyond text transcripts to incorporate multimodal data. The aim was to preserve the sequential organisation of interaction while also capturing non-verbal and contextual signals that shape turn-taking and repair.

Rather than relying solely on verbal exchanges, each turn was analysed as a composite action containing multiple communicative channels, including speech, gaze direction, facial

expressions, hesitations, and timing of responses. This multimodal approach enabled us to detect subtle interactional cues such as user hesitation aligned with delayed system output that may not be visible in transcript-only analysis.

The recorded video of the conversation was essential because conversational breakdowns in everyday VUI use often become visible through behaviour rather than words. For example, users may pause mid-action, glance repeatedly at the device to check whether it is listening, show confusion or irritation after an undesired response, or silently abandon the task signals that are ambiguous or completely invisible in transcript-only or audio-only data. Including video therefore strengthened the validity of the turn-by-turn analysis by allowing us to connect system timing/events with observable user reactions and repair strategies.

Data preparation and segmentation. The video recordings were first segmented into discrete conversational turns, beginning with a user-initiated action (verbal command or question) and ending with a system response or user follow-up. Each turn was annotated using a structured coding template designed to ensure consistency across cases.

Multimodal coding framework. The framework comprised the following five components:

1. **Verbatim utterance:** Exact spoken words from the user or VUI.
2. **User facial expression:** Observed affective or cognitive states (e.g., confusion, frustration, neutral).
3. **Condensed meaning unit:** Short summary of the intended meaning behind the utterance.
4. **Category:** Functional label of the turn (e.g., clarification request, response type).
5. **Theme:** Higher-level interpretation linked to interactional outcomes (e.g., desired response, misrecognition, context loss).

Identification of breakdown junctures. Once all interactions were coded, we traced the sequential points at which breakdowns emerged such as turn-taking failures, misrecognition, unresponsive states, or explicit rejections and recorded subsequent user responses. These observations revealed consistent retry and repair patterns, which were aggregated across the dataset.

Verbatim Transcript	User Facial Expression	Condensed Meaning Unit	Category	Outcome
P: OK, I want to go to Gangtok this week.	Neutral	States travel destination	Statement	Desired Response
A: The best way to get to Gangtok by car is via National Highway 10...	Confident	Provides route	Task completion	Desired Response
P: OK, Google, can you show me public transport to Gangtok?	Neutral	Requests public transport info	Enquiry	Undesired Response
A: I'm afraid I can't find public transport directions to your destination.	Confused	Refuses request	Rejection	Undesired Response

Table 3.1: Example of multimodal coding for a travel-related VUI interaction

Derivation of the conversation breakdown path. The aggregated patterns were modelled into a generalised state-transition diagram, representing how conversations progressed towards task success or diverged into breakdown and abandonment. This diagram (Figure 3.2) reflects the observed decision points, retry thresholds, and error types that most frequently contributed to task failure. It serves as a structural framework for understanding breakdown trajectories in VUI use and provides the empirical basis for the thematic categorisation presented in the second analysis approach.

3.7.3 Approach 2: Thematic analysis

Following the turn-by-turn multimodal analysis in Approach 1, we conducted a thematic analysis to systematically categorise the different types of conversational errors observed in Voice User Interface (VUI) interactions. This step moved the analysis from a micro-level focus on specific breakdown instances to a macro-level understanding of recurring error patterns. The method followed Braun and Clarke's six-phase framework Braun and Clarke (2006), adapted to the unique challenges of analysing turn-taking, timing, and system feedback in half-duplex

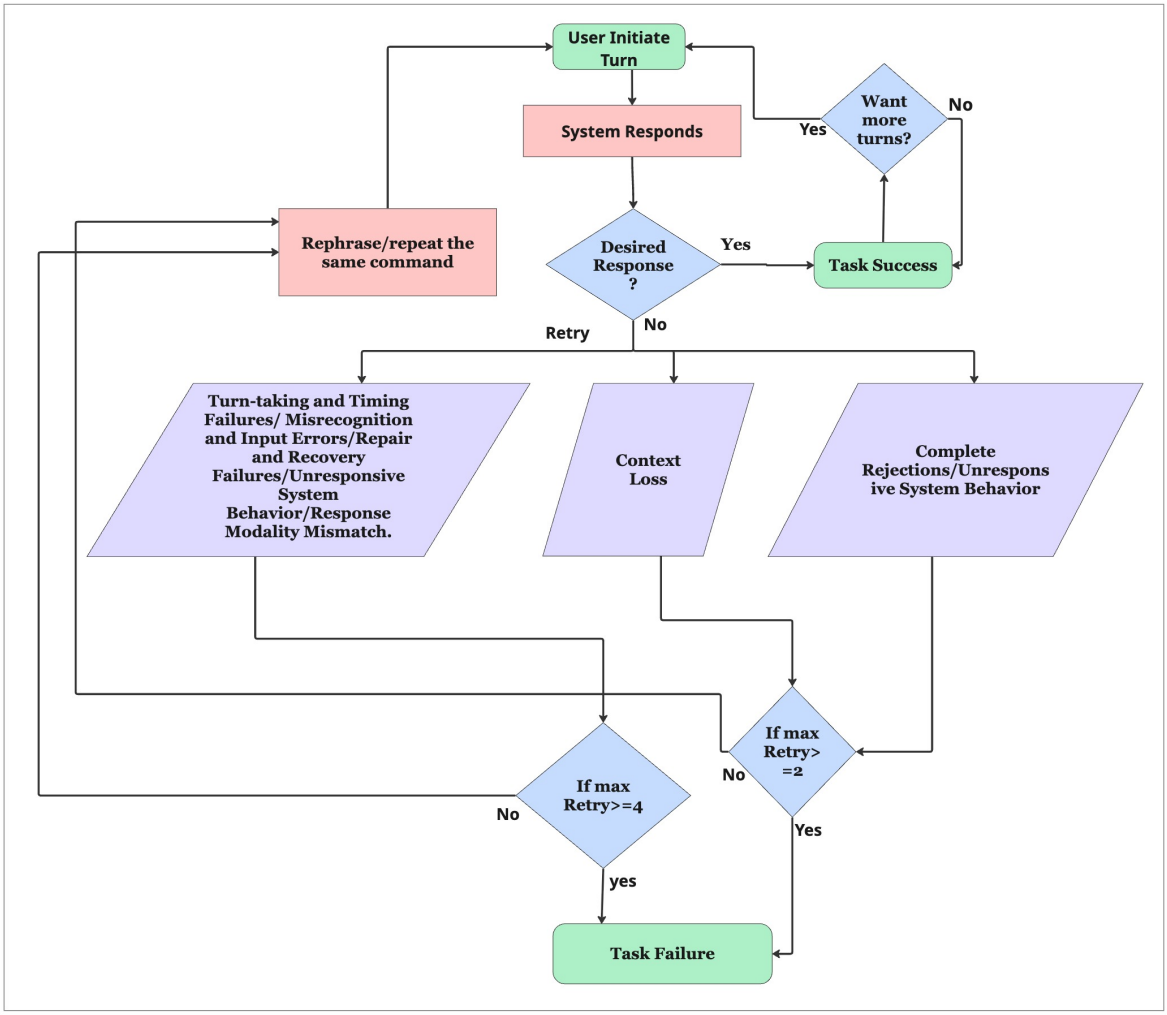


Figure 3.2: Conversation Breakdown Path

speech-based systems.

Phase one: Data Familiarisation The familiarisation phase for thematic analysis was inherently embedded within the earlier stage of this research namely, the multimodal analysis undertaken to construct the conversational breakdown path (Approach 1). In that process, each of the 86 recorded VUI interactions was examined in full using a multimodal coding framework, capturing verbal utterances, pauses, overlaps, gesture cues, facial expressions, and system feedback events.

All recordings were viewed repeatedly and transcribed orthographically, noting not only the spoken words but also non-verbal elements such as timing gaps, user gaze shifts, or manual interventions. These transcripts were further annotated with interactional events such as sys-

tem interruptions, ignored inputs, or delayed responses, allowing for fine-grained mapping of turn-taking sequences. The iterative review process meant that, by the completion of the conversational breakdown path, the research team had already achieved deep familiarity with the dataset's linguistic, paralinguistic, and behavioural dimensions.

Building on this existing immersion, the thematic analysis phase did not require a separate, isolated familiarisation stage; rather, it drew directly on the prior in-depth engagement with the data. Nonetheless, transcripts and multimodal annotations were revisited to refresh contextual understanding and to note preliminary ideas relevant to the identification of patterns for thematic coding. This integrated approach ensured that the thematic analysis was grounded in a rich, multi-layered understanding of each interaction, consistent with Braun and Clarke's (2006) recommendation to consider the breadth and depth of the dataset before formal coding.

Phase Two: Generating Initial Codes In this phase, each dialogue turn produced by the Voice User Interface (VUI) served as the primary unit of analysis. A dialogue turn was defined as a single system output bounded by a change in speaker. In instances where multiple consecutive turns formed a single coherent interactional episode (e.g., the system producing successive outputs in response to a reformulated user request), these were coded together to preserve the contextual meaning of the exchange. This decision prevented fragmentation of error patterns and ensured that system behaviours were interpreted within their immediate conversational context.

Coding was conducted inductively, following Braun and Clarke's Braun and Clarke (2006) guidelines, without reference to any pre-existing coding framework. Each system turn was reviewed for its semantic content, functional role in the conversation, and relevant multimodal cues (e.g., latency before output, overlapping audio, system tones) captured during observation.

The coding in this phase was informed by the multimodal framework developed in Approach 1 (see Table 3.1). While that framework does not include a dedicated *code* column, it provides the raw analytic fields verbatim utterance, facial expression, condensed meaning unit, sub-category, category, and provisional interpretive label from which we inductively generated initial codes. To make this transition explicit, we created a separate coding sheet with an added *Initial Code* column; an excerpt is shown in Table 3.2.

Codes were short, descriptive labels that could stand alone without the transcript, and

were designed to capture the nature of the system's behaviour or error. Where relevant, both surface-level (semantic) meaning and underlying (latent) implications were recorded. Our focus remained exclusively on system actions and errors, rather than user repair strategies.

For example, in one instance, the user explicitly instructed Siri to stop an ongoing action; however, Siri continued without acknowledging the request. This episode was assigned two codes: *Undesired response* (mismatch between intended and actual system output) and *System did not allow barge-in* (failure of turn-taking control, preventing interruption of ongoing output).

In another case, the system partially misrecognised an utterance and produced an irrelevant output. This was coded as *Speech misrecognition* (ASR error) and *Inappropriate response* (content unrelated to the request). A further instance involved delayed system readiness after a weather enquiry, where silence was followed only by a system tone with no speech output. This was coded as *Turn-taking error – delayed prompt*.

Three independent coders participated in the initial coding process, each working on a subset of transcripts. Coding decisions were compared and refined through iterative discussions until consensus was reached. This collaborative process reduced individual bias and enhanced reliability.

Table 3.2 and Table **??** shows excerpts from the coding process, illustrating how raw multimodal data fields were transformed into succinct analytical codes.

Dialogue Turn (Verbatim)	**Facial Expression / Behaviour**	**Condensed Meaning Unit**	**Initial Code**
P: Siri, stop the news.	Neutral	Request to stop ongoing playback	Stop request – playback interruption
A: (Continues reading the news without acknowledging the stop command)	Neutral	Ignores stop request	Undesired response – ignored stop command
A: (Finishes the current segment and then stops)	Neutral	Delayed stop after output completion	System did not allow barge-in

Table 3.2: Example of undesired response combined with barge-in failure in a Siri interaction.

Dialogue Turn (Verbatim)	**Facial Expression / Behaviour**	**Condensed Meaning Unit**	**Initial Code**
P: Hey Cortana. Play the song *Memories* from Maroon 5.	Neutral	Music request – correct song title	Music request – correct keyword provided
A: [Detects as: "Play the song *mein movies* from Maroon 5"] [Shows web result]	Neutral	Misrecognition of song title; returns irrelevant web result	Speech misrecognition – incorrect keyword detection; Undesired response – irrelevant output
P: Memories. Memories.	Neutral	Repeat request – same correct title given	Music request – correct keyword repeated
A: [Detects as: "Mein movies"] I found this for you.	Neutral	Repeated misrecognition; irrelevant web output	Speech misrecognition – persistent; Undesired response – irrelevant output
P: Hey Cortona play the song "Memories." (Frustrated)	Detects wrong keywords Asking to play a certain song	Undesired response	Speech misrecognition – persistent; Undesired response – irrelevant output
A: [Detect Play the song mein movies][Show web result]	Frustrated Detects wrong keywords Asking to play a certain song	Undesired response repeat command	repeat
P: Hey, Cortana. Play the song memories from maroon 5	Frustrated Asking to play certain song	Undesired response	Rephrase Repeat

Table 3.3: Example of repeated speech misrecognition and irrelevant output in Cortana interaction.

This systematic process was applied recursively across the dataset, resulting in a comprehensive code set that captured the range of system errors observed. These codes formed the foundation for identifying broader themes in Phase 3.

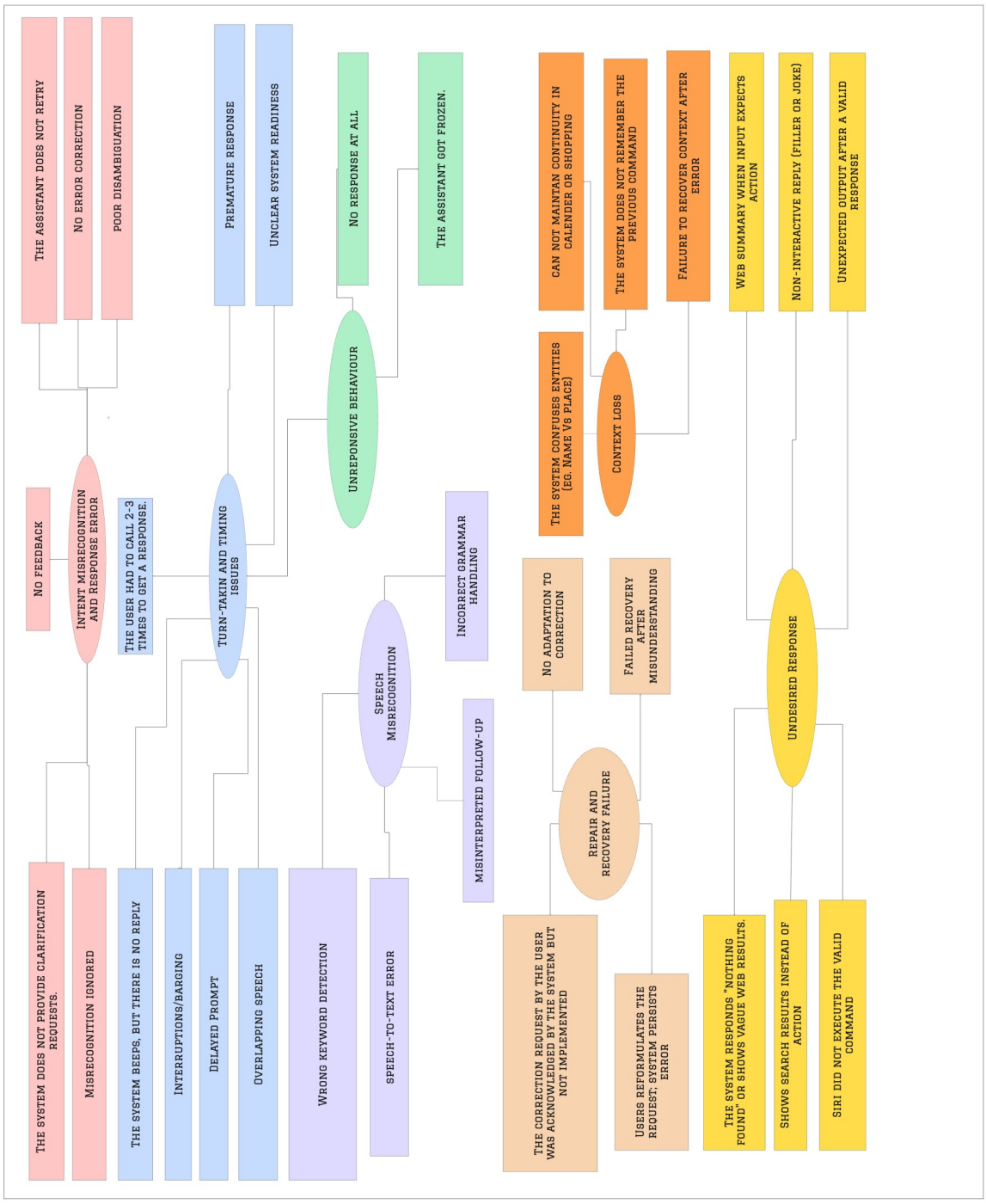


Figure 3.3: Initial Thematic Map showing 7 themes and 33 subthemes

Phase Three: Developing the Initial Thematic Map In this phase, the set of initial codes generated in Phase Two were systematically examined to identify broader patterns of meaning across the dataset. Following Braun and Clarke's Braun and Clarke (2006) guidance, the goal was to move from a descriptive list of system-level conversational errors to an organised set of potential themes that captured the underlying mechanisms and relationships among those errors.

The process began with an iterative sorting of codes into candidate themes and subthemes. This grouping was guided by two main principles: *internal coherence* (codes within a candidate theme should share a clear conceptual relationship) and *external distinction* (different themes should capture different aspects of the phenomenon). Each code was written on an individual sticky note and physically placed on a large board, allowing for easy rearrangement and visual comparison. These sticky notes were moved, clustered, and regrouped multiple times until stable and meaningful groupings emerged.

A key analytic decision at this stage was to treat multiple dialogue turns as a single coding unit when they formed one coherent *interactional episode* (e.g., repeated misrecognition of the same entity name across several turns). This allowed the thematic structure to reflect the conversational flow and preserved the causal link between an initial system error and its downstream effects. For example, a persistent misrecognition of "Memories" as "Mein movies" by the VUI across three consecutive turns was grouped as one episode under the theme *Speech Misrecognition*.

During the clustering process, several patterns became evident:

- Codes describing delayed prompts, overlapping speech, and frozen outputs clustered under the candidate theme *Turn-Taking and Timing Issues*.
- Misrecognition-related codes (e.g., incorrect keyword detection, grammar handling failure) grouped naturally under *Speech Misrecognition*.
- Codes referring to irrelevant or mismatched outputs were combined under *Undesired Response*.
- Errors involving poor disambiguation or ignored recognition errors were grouped under *Intent Misrecognition and Response Error*.

- Failures to remember prior commands or maintain task continuity formed the candidate theme *Context Loss*.
- Failed attempts to adapt to user correction or to recover after misunderstanding were placed under *Repair and Recovery Failure*.

The outcome of this phase was the *initial thematic map* (Figure **??**), which displayed seven candidate themes and 33 subthemes. This visual representation served as a working model for the subsequent phase of theme refinement, where overlaps were addressed and the thematic structure was streamlined.

Phase Four: Reviewing and Refining Themes In this phase, the initial thematic map generated in Phase Three was critically reviewed to ensure that each theme accurately captured the patterns in the data and that the overall thematic structure was coherent, distinct, and analytically meaningful.

Level 1: Within-Theme Coherence.

All coded data extracts within each candidate theme were re-examined to verify that they formed a coherent and internally consistent set. This review was conducted by revisiting each extract in its original conversational context (i.e., the full interactional episode) to ensure that the assigned theme genuinely reflected the meaning and function of that data. In some cases, codes were reallocated to more fitting themes when closer semantic or functional alignment was identified. For example, several instances initially coded under *Repair and Recovery Failure* were reclassified as *Context Loss* when the breakdown stemmed from the system's inability to recall prior turns rather than a failed repair attempt. Similarly, *Unresponsive Behaviour* was found to be a direct manifestation of *Turn-taking Issues*, arising when delays, silences, or missed cues disrupted the conversational flow.

Further, redundant or overlapping codes were consolidated to remove duplication. For instance, within *Turn-taking Issues*, the codes "assistant got frozen" and "no response" were merged, as both described scenarios where users had to repeat their request multiple times due to unclear or absent system feedback. In *Speech Misrecognition*, the code "misunderstood follow-up" was shifted to *Intent Misrecognition*, reflecting that these cases were rooted in failures to interpret user intent rather than lexical inaccuracies. *Context Loss* was also refined to explicitly

include cases where the system failed to remember the previous command, often preventing successful recovery. Moreover, in both *Intent Misrecognition* and *Response Error*, breakdowns were frequently traced to poor disambiguation, which meant the assistant did not attempt a repair.

Level 2: Across-Theme Distinction.

The thematic map was then reviewed holistically to ensure that each theme was conceptually distinct from the others. Overlaps were reduced by merging themes that addressed the same underlying mechanism. This stage was guided by three refinement criteria:

1. *Clarity of theme boundaries* themes had to be clearly delineated in scope and focus.
2. *Analytic depth* themes needed to move beyond surface description to capture underlying mechanisms or conversational processes.
3. *Representative coverage* themes had to adequately represent the diversity of system-level breakdowns in the dataset.

Applying these criteria, *Unresponsive Behaviour* was subsumed under *Turn-taking Issues*, as its occurrence could be causally explained by failures in turn-taking coordination. The label *Turn-taking and Timing Issues* was refined to *Turn-taking Issues* to better reflect its analytic focus and avoid redundancy.

Final Refinement Outcome.

Through this process, the thematic structure was streamlined from seven themes and 33 subthemes to four main themes and 14 subthemes (Figures **??**, 3.4, and 3.5). This reduction improved conceptual clarity, eliminated redundancies, and embedded causal linkages between themes while preserving the richness of the data.

Phase Five: Defining and Naming Themes Following the refinement of themes in Phase Four, the next stage involved defining the essence of each theme that is, identifying precisely what each theme captures and how it relates to the overall research question. This phase required moving beyond a descriptive account of the data to provide an interpretive narrative of how each theme functions in explaining system-level breakdowns in Voice User Interface (VUI) interactions.

For each theme, we:

1. Reviewed all collated extracts to ensure that they collectively reflected the central organising concept of the theme.
2. Identified the scope and boundaries of the theme, ensuring that it was analytically distinct from other themes.
3. Developed a concise working definition that encapsulated the key mechanism or phenomenon captured by the theme.
4. Selected a theme name that was both descriptive and accessible, avoiding overly technical terminology where possible.

The final thematic structure comprises four main themes, each supported by multiple subthemes. These themes are:

1. **Turn-Taking Issue** Breakdowns arising from the system's inability to manage conversational timing, including delayed responses, premature interruptions, and failure to allow barge-in.
2. **Context Loss** Failures where the system is unable to retain or utilise relevant conversational context, resulting in disjointed or irrelevant responses.
3. **Speech Misrecognition** Errors stemming from incorrect speech recognition or misinterpretation of user input, leading to inappropriate or unintended system actions.
4. **Intent Misrecognition** Cases where the system technically recognises the speech input but misinterprets the intended goal or meaning, producing an output that does not align with the user's intended action.

The themes were named to make their meaning clear at a glance. Each name was chosen to be descriptive enough so that readers could understand the type of breakdown without reading the full definition, while still being broad enough to cover all the different examples included under that theme.

Phase Six: Producing the Report The final phase focused on moving from the refined thematic structure to a coherent and persuasive analytic narrative. At this stage, the 4 final themes developed in Phase Five (Figure 3.5) were not treated as static labels, but as organising concepts around which the findings would be communicated. The objective was to integrate these themes into a written account that demonstrated both the breadth of system-level breakdowns in real-world VUI interactions and the depth of their underlying mechanisms.

For each theme, a dedicated subsection was created in the Results section. These subsections followed a consistent structure:

1. A brief definition of the theme's central organising concept.
2. An outline of the scope of the theme, explaining how it relates to the research question.
3. Representative extracts from the multimodal coding framework (Approach 1) illustrating the theme in action.
4. An interpretive commentary linking the extract to the wider dataset and theoretical context.

Extracts were carefully selected to be vivid and self-contained, enabling readers to grasp the nature of the breakdown without consulting the full transcript. Wherever possible, they were drawn from interactional episodes that typified the theme while also reflecting variation in how it appeared across different participants, systems, and task types.

The write-up of each theme balanced descriptive and analytic elements. Descriptively, we reported the forms in which the breakdown appeared; analytically, we interpreted what these breakdowns reveal about the conversational design and technical constraints of current VUIs. For example, in *Turn-taking and Timing Issues*, delays, premature cut-offs, and blocked interruptions were described as observable patterns, and then analysed in terms of how they disrupted user expectations and reflected the limitations of half-duplex systems.

Each thematic subsection concluded with a brief note on its implications for VUI design, linking back to relevant literature in conversational analysis, HCI, and speech interface research. This ensured that the findings were not presented in isolation but situated within the broader scholarly discourse.

The completed write-up thus served a dual function: it documented the themes as the final product of a rigorous six-phase analysis, and it conveyed a clear, evidence-based account of how conversational errors occur and manifest in real-world VUI interactions. In doing so, it provided the necessary bridge to the discussion chapter, where the implications of these findings are examined in greater depth.

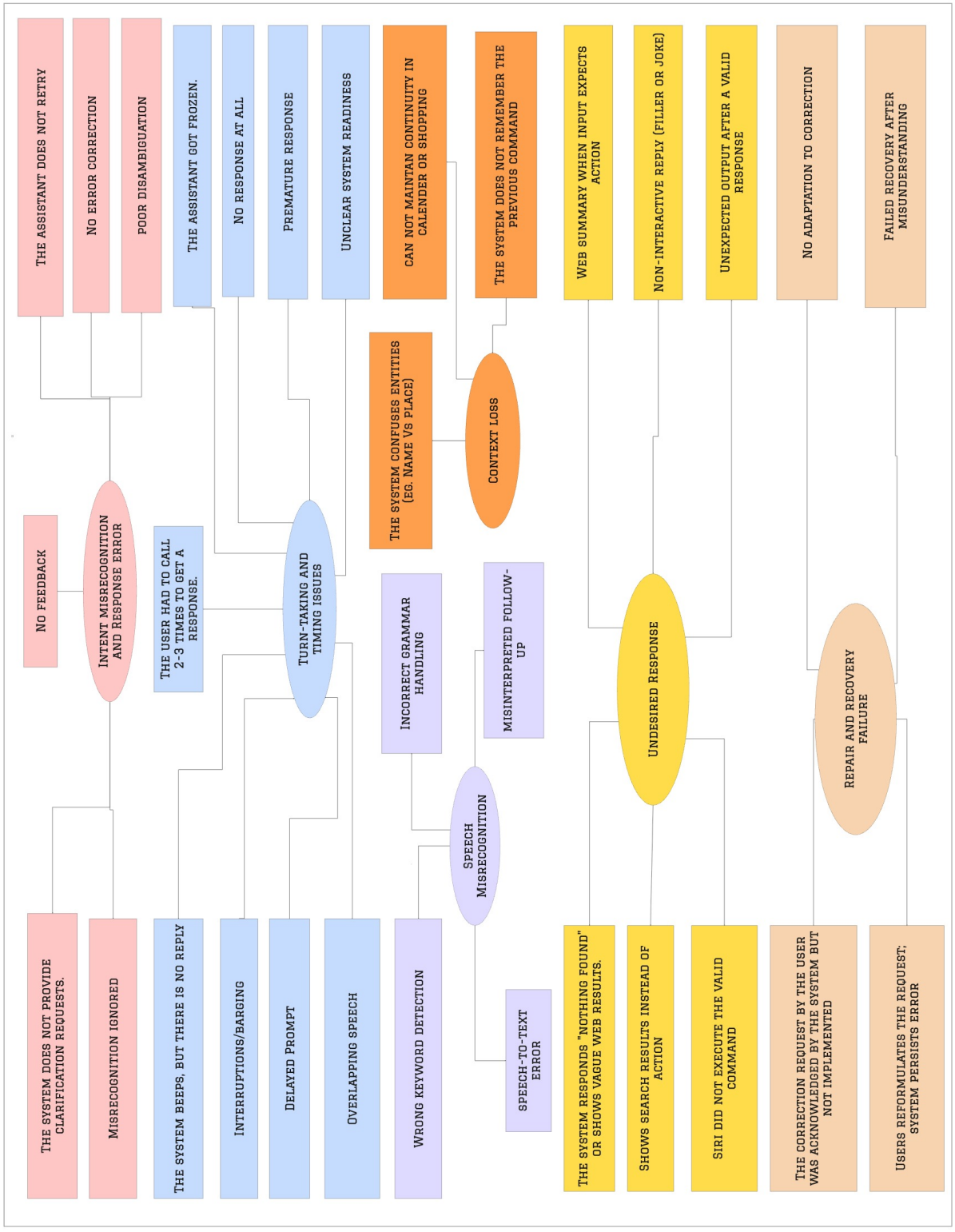


Figure 3.4: Developed(pre-final) Thematic Map showing 6 themes and 32 subthemes

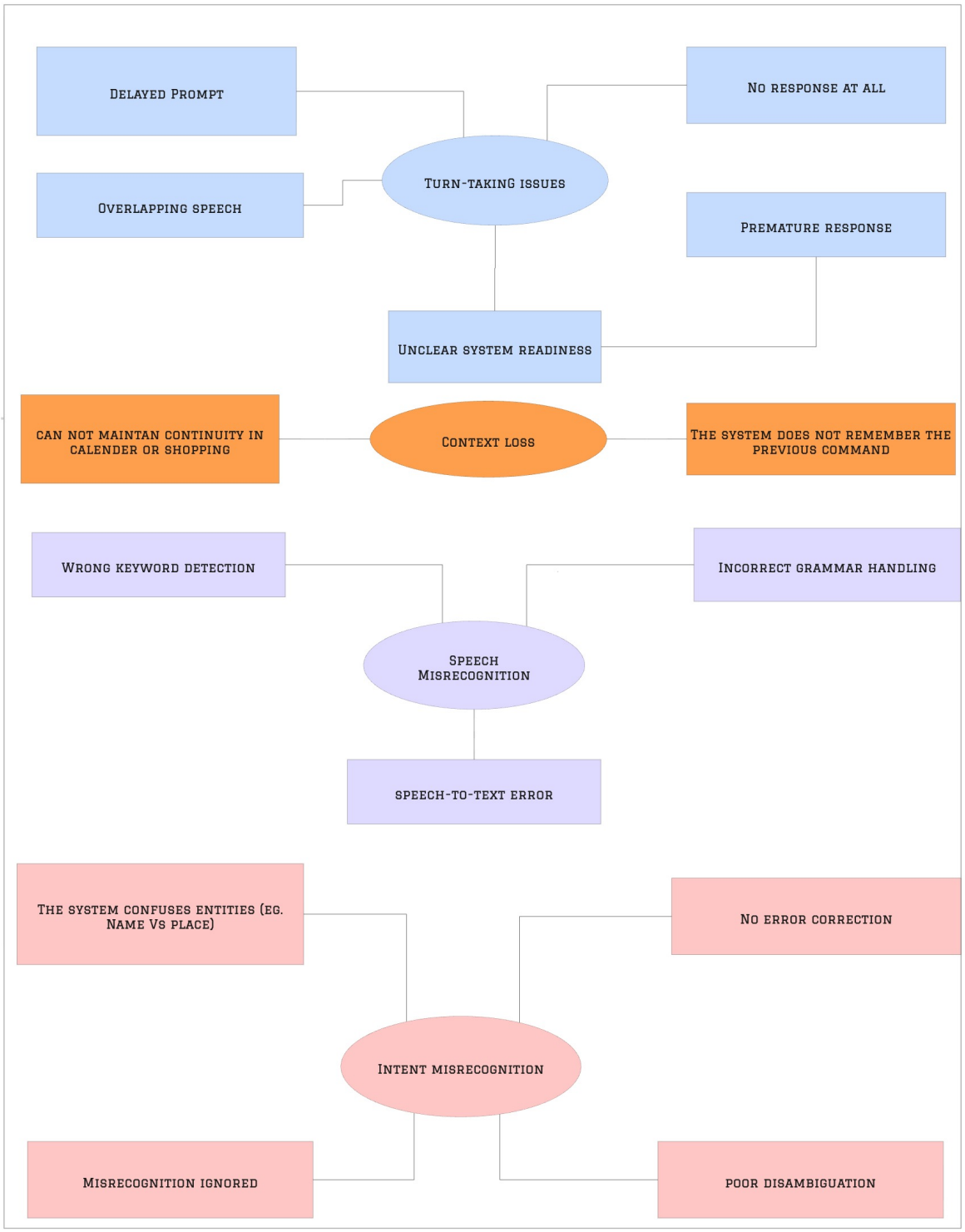


Figure 3.5: Final Thematic Map showing 4 themes and 14 subthemes

3.8 Results

The analysis revealed two complementary perspectives on breakdowns in real-world Voice User Interface (VUI) interactions. The first is a conversational breakdown path (3.2), derived from turn-by-turn multimodal observation, that maps the typical conversational trajectories that lead to either successful task completion or breakdown and abandonment. The second is a thematic map 3.5, based on thematic analysis, identifies the recurring categories of system-level errors that drive these trajectories.

3.8.1 Conversation breakdown path

We analysed the full set of recorded interactions turn by turn and observed that, despite variation in tasks and devices, users' exchanges tended to progress through a small number of recurring trajectories from initiation to completion or abandonment. These recurring sequences motivated the development of the state–transition diagram in Figure 3.2, which was constructed to capture, in a compact and faithful form, how interactions typically unfold in practice namely, the decision points where breakdowns occur, the repair actions users attempt (e.g., repetition or rephrasing), and the conditions under which the interaction returns to success or transitions to failure.

Figure 3.2 illustrates the most common way a voice interaction progresses from start to end in our dataset, and is intended to represent what users typically have to navigate in order to reach either task success or task failure. The interaction begins when the user initiates a turn and the system responds, after which the user implicitly evaluates whether the response matches their goal. If the response is satisfactory, the interaction reaches task success, and the user may either stop or continue with additional turns. If the response is not satisfactory, the interaction enters a retry phase in which users typically repeat or rephrase the same command. The diagram groups these breakdowns into three broad types observed in the conversations: errors that users often perceive as "fixable" through repetition or rephrasing (such as turn-taking/timing failures, misrecognition and input errors, repair/recovery failures, intermittent unresponsiveness, or response-modality mismatch), context loss, and complete rejections or strongly unrespon-

sive system behaviour. A key pattern captured in the model is that these breakdown types lead to different levels of persistence: users tend to tolerate more retries for the first group (often up to four attempts), whereas context loss and complete rejection behaviours typically trigger earlier abandonment (often within two attempts). When the interaction exceeds these retry thresholds, it transitions to task failure, thereby capturing the practical "cost" users incur across retries, rephrasing, and recovery attempts before they either achieve the intended outcome or disengage.

Across the 86 recorded interactions, user persistence varied systematically by breakdown type. When failures involved *context loss* or *complete rejections/strongly unresponsive system behaviour*, users seldom persisted beyond two attempts, indicating that these breakdowns were quickly perceived as difficult to recover from. In contrast, breakdowns that appeared more amenable to user-led repair such as *turn-taking/timing failures*, *speech misrecognition and input errors*, *repair/recovery failures*, *intermittent unresponsiveness*, and *response-modality mismatch* typically prompted longer sequences of repetition or rephrasing, with persistence observed up to a maximum of four retries. These empirically observed retry thresholds and decision points are captured in the conversation breakdown path (Figure 3.2), a state-transition model that summarises how interactions progress toward either task success or abandonment.

An example of task failure as a result of speech misrecognition by the assistant is shown in the table 3.3 exemplifies a breakdown trajectory driven by persistent *speech misrecognition* and the system's repeated production of an *undesired response*. The participant issues a clear music-playback command ("Play the song *Memories* from Maroon 5"), yet the assistant repeatedly mishears the key entity ("Memories") as a phonetically similar but incorrect phrase ("mein movies") and responds by presenting irrelevant web results rather than initiating music playback. Importantly, the user's subsequent turns demonstrate a canonical repair strategy observed throughout the dataset: repetition and rephrasing of the same request ("Memories. Memories."; "Hey Cortana, play the song 'Memories' "), accompanied by escalating frustration. Despite these repair attempts, the assistant continues to detect the wrong keyword and returns the same type of irrelevant output, indicating a failure to recover from the initial ASR-level error.

Within the conversation breakdown path model (Figure 3.2), this interaction follows the "repairable breakdown" branch associated with *misrecognition and input errors*. After the system response fails to match the user's goal ("Desired response? = No"), the user enters the retry

loop and repeatedly *rephrases/repeats the same command*. Because misrecognition is typically perceived by users as potentially correctable through clearer articulation or repetition, this category often elicits higher persistence than context loss or outright rejection; accordingly, the excerpt illustrates multiple successive retries before resolution. In other words, this example provides concrete turn-level evidence for the left-hand pathway in the transition diagram *misrecognition* triggering repeated repair attempts and prolonged interaction while also motivating why repeated misrecognition was coded as a salient subtheme under the broader error structure in the thematic map (Figure 3.3).

This process view emphasises that breakdowns are not isolated events but part of a dynamic sequence. The point in the trajectory where an error occurs and whether the system supports effective recovery has a clear impact on whether the user continues or leaves.

3.8.2 Thematic Map

The thematic analysis distilled the dataset into four main categories of system-level errors (Figure 3.5), each encompassing a range of subthemes. These themes represent the most frequent and disruptive breakdown patterns observed across systems and tasks.

Turn-Taking Issues Breakdowns stemming from the system's inability to manage conversational timing, including delayed prompts, premature interruptions, and blocked barge-in attempts. Such failures often disrupted the rhythm of interaction and created uncertainty about system readiness.

Context Loss Failures where the system could not retain or apply relevant conversational context, leading to disjointed or irrelevant responses. This was particularly evident in follow-up queries that referred back to information from previous turns.

Speech Misrecognition Errors arising from incorrect speech recognition or misinterpretation of user input, resulting in unintended or irrelevant actions. Persistent misrecognition was a major driver of user retries.

Intent Misrecognition Cases where the lexical content of the input was correctly recognised but the intended action was misinterpreted. This often produced plausible but incorrect outputs. as generic web search results.

3.8.3 Integrated View

The combination of the two analyses offers a layered understanding of VUI breakdowns. The conversation breakdown path reveals the process dynamics how interactions evolve and where they fail while the thematic structure explains the nature of the failures at each point. Together, they show that breakdowns are both patterned and multi-causal: a single trajectory may involve several error types, and the likelihood of recovery depends as much on error type as on timing within the exchange.

3.9 Discussion

The findings of this study, visualized through a structured conversation breakdown path map and a developed thematic analysis map, reveal deep-rooted challenges in the way current Voice User Interfaces (VUIs) manage user interaction. By analyzing real-world interaction failures, this study offers a layered understanding of how specific user requests escalate into frustration, embarrassment, or complete task failure.

The thematic map, developed through qualitative coding of user behavior and system output, reinforces this interpretation by organizing failures into four major themes. Each of these themes reflects a core breakdown in conversational flow. For example, Turn-Taking Failure is driven by overlapping turns, delayed prompts, and excessive user repetition a pattern directly visible in the conversation breakdown flow. Similarly, context loss Breakdown emerges from the system's inability to handle previous context of the same conversation.

The maps show how user-initiated escalation (e.g., repeating commands, rephrasing, or increasing urgency) often fails to yield adaptive system behavior. Instead, these efforts are met with static or generic responses, reinforcing the "mechanomorphic" nature of current VUI

systems. This lack of transparency and responsiveness is especially problematic in sensitive scenarios such as emergency help, where the system defaults to a generic web result instead of initiating a direct action a clear case of Context-Inappropriate Response, as mapped in both diagrams.

One striking insight from the combined analysis is how early-stage errors often compound rather than self-correct. A misrecognition (system misunderstanding) not only leads to user repetition, but also primes the system for further misinterpretation, eventually pushing the interaction into a failure loop. This layering of small errors into larger breakdowns suggests that current VUIs lack effective mid-conversation repair mechanisms a limitation echoed in studies like Porcheron et al. (2018); Myers et al. (2018).

The maps also expose vulnerabilities in user mental models of the system. For instance, in the "Ask for Reminder" example, the user attempts overlapping correction, assuming the system can parse real-time feedback but the system misinterprets it, leading to a breakdown in turn-taking. This reflects findings by previous studies Luger and Sellen (2016); Sciuto et al. (2018), who noted that users often overestimate system capabilities due to human-like presentation. Without visible cues or explicit state indicators, users default to conversational norms that VUIs cannot yet accommodate.

In sum, the findings of the chapter not only validate known VUI challenges but also present a cohesive, empirical framework for understanding how conversation breakdowns unfold and cluster. They highlight the need for feedback systems that are context-aware, repair-sensitive, and designed around the realities of user behavior not idealized models of command recognition.

3.10 Conclusion

This chapter presented a foundational investigation into real-world conversational failures in commercial Voice User Interfaces (VUIs), with the aim of investigating whether turn-taking errors contribute to a majority portion of the VUI errors in real-world conversations. Motivated by the persistent usability challenges associated with timing, feedback, and adaptation in VUIs, the study sought to identify not only the types of errors users encounter, but also the trajectories

through which seemingly minor misalignments escalate into complete conversational failures.

Through a detailed analysis of video-recorded conversations, the study yielded two key contributions: a Conversation Breakdown Path Map illustrating the sequential progression of user-system misalignments, and a thematic analysis map capturing the underlying causes of these breakdowns in structured, higher-order categories. The breakdown path revealed recurring conversation path such as speech misrecognition, Turn-taking errors, context loss, etc, each contributing to mounting user frustration or eventual task abandonment. Thematic coding further synthesised these observations into 4 final themes of errors.

Importantly, the findings underscore that many VUI breakdowns arise not solely from technical limitations like speech recognition errors, but from fundamental design choices,particularly in how systems handle turn coordination and provide feedback. Current commercial VUIs often rely on anthropomorphic cues that imply conversational competence, yet fail to offer transparent, adaptive responses that help users manage errors or align their input with system timing. This mismatch fosters unrealistic expectations and erodes trust when the system does not behave intelligibly.

The diagnostic tools developed in this study the breakdown path and thematic maps offer a practical framework for both evaluating current systems and informing the design of next-generation VUIs. These findings serve as a foundational motivation for the subsequent experiments presented in this thesis, which explore targeted interventions such as non-speech auditory cues to address the specific breakdown patterns observed here.

This study provides a real-world, video-based, turn-by-turn evidence base of how conversational breakdowns and repair unfold in current VUIs, and translates these observations into a structured error taxonomy and breakdown-path model that directly informs the mechanomorphic turn-taking design evaluated in subsequent chapters.

Limitations of this study include its reliance on observational data from existing commercial systems, which do not permit experimental manipulation or insight into internal decision-making processes. Future work should extend these findings through controlled studies, user interviews, and participatory design methods to probe user expectations and test more explicit turn-taking supports.

In sum, this study reveals that successful human-VUI conversation depends not only on

recognition accuracy, but on the design of the conversation itself including how the system frames its capabilities, supports repair, and facilitates temporal coordination. Addressing these elements is essential for creating voice interfaces that are not only usable, but intelligible, co-operative, and resilient in the face of real-world conversational demands.

This page was intentionally left blank.

Chapter 4

Half-Duplex Turn-Taking Protocol for Prompt-Response Conversation

4.1 Overview

Building on the insights gained from the analysis of real-world conversational errors in VUIs, this chapter presents an empirical evaluation of a structured turn-taking protocol designed to reduce timing-related breakdowns in half-duplex voice interactions. By examining user behavior across repeated interactions, the study investigates whether users adapt to the system's temporal cues and how such adaptation impacts task success. Special attention is given to the role of system feedback in guiding user timing and improving interaction fluency. The findings contribute to a growing body of work on temporal entrainment in human–machine communication and provide evidence for the value of mechanomorphic interaction design.

This chapter directly addresses the second and third objectives of the thesis, focusing on the design and evaluation of a feedback-supported turn-taking protocol.

4.2 Introduction

This study introduces a turn-taking protocol that utilizes different auditory cues, in the form of tones and system prompts, to signify the otherwise invisible seam of half-duplex turn-taking. By sonifying this seam, the protocol encourages users to entrain to the turn-taking pattern that may initially feel unnatural. The intention is to enable a more reliable turn-taking performance, required for task success during human-VUI conversation.

To control the VUI's turn-taking behavior, the protocol manipulates specific temporal parameters applied to the automatic speech recognition (ASR) within the speech platform. Through a comprehensive analysis of human participants conversing with the VUI, the study illuminates how users respond to these artificial turn-taking behaviors. Furthermore, it reveals a fascinating trend: over time, users tend to entrain to the VUI's turn-taking behaviour.

The implications of this research extend meaningfully to the design of VUIs, offering practical insights for improving fluency in half-duplex, prompt–response interactions. The findings help clarify how users adapt to system constraints and point toward design directions that enhance transparency and turn-taking coordination. These contributions support the broader aims of the thesis by deepening our understanding of human–VUI interaction and informing more effective design strategies.

This research tackles a core challenge in VUI interaction through an innovative lens, advancing both the technical implementation of conversational systems and our theoretical insight into human turn-taking behavior during human-VUI exchanges. It marks a meaningful advancement toward designing VUIs that engage users in conversations that are not only responsive and natural but also conducive to successful task completion.

4.3 Turn-Taking in voice-based conversation

Turn-taking, the process of exchanging speech and silence between participants during conversation, is a crucial aspect of spoken interaction. This seemingly simple behaviour is a complex, dynamic, multi-levelled cognitive process that generates "when to speak" decisions by con-

versational partners. Automating this process is one of the relevant topics of human-computer conversational systemsRaux and Eskenazi (2012, 2009); Razavi et al. (2019); Zhao et al. (2015). Researchers have designed 1) various end pointing solutions such as silence-based, IPU based, continuous models, and 2) use of explicit cues to influence turn-taking behaviours in artificial-speech systems. Although both methods can manage turn-taking to some extent, researchers have found that settings for various thresholds result in awkward silences on the one hand or overlap and confusion on the other Heldner and Edlund (2010). Alternatively, using explicit cues (e.g., use of wake words, push-to-talk, visual indicators) is both inconvenient and unnatural Woodruff and Aoki (2003). As a result, modelling turn-taking in conversational systems remains a work in progress with certain specific known challenges.

4.3.1 Blind spot and invisible seam

The disallowance of concurrent speech creates two related HCI implications, shown in Figure 4.1. At the start (A) and the end (F) of a conversational sequence, the ASR is OFF and must remain so whenever the machine is talking or playing a tone (B). This unique property of half-duplex turn-taking leads to a conversational blind spot, (shaded area B), during which the user's speech cannot be detected. Users may only speak during the listening window (D), but they do not understand this rule, which is unnatural. So, it often happens that their speech hits the invisible seam (dashed vertical line at C). This leads to a complete or partial loss of human speech. In a prompt-response dialogue, turns are exchanged frequently through a conversational loop (E). The interactivity of alternating turns exacerbates the impact of this invisible seam, leading to several emergent errors during human-computer conversation. This diagram 4.1 shows why half-duplex systems frequently miss user speech. When the system is speaking or playing a tone, the ASR must remain OFF, creating a blind spot where any user speech goes unheard. Users naturally attempt to speak whenever they are ready, but if their speech overlaps with the system's switch from speaking to listening (the invisible seam), their words are partially or completely lost. In a prompt–response interaction with rapid turn exchanges, this seam appears repeatedly, increasing the likelihood of timing-related errors and conversational breakdowns.

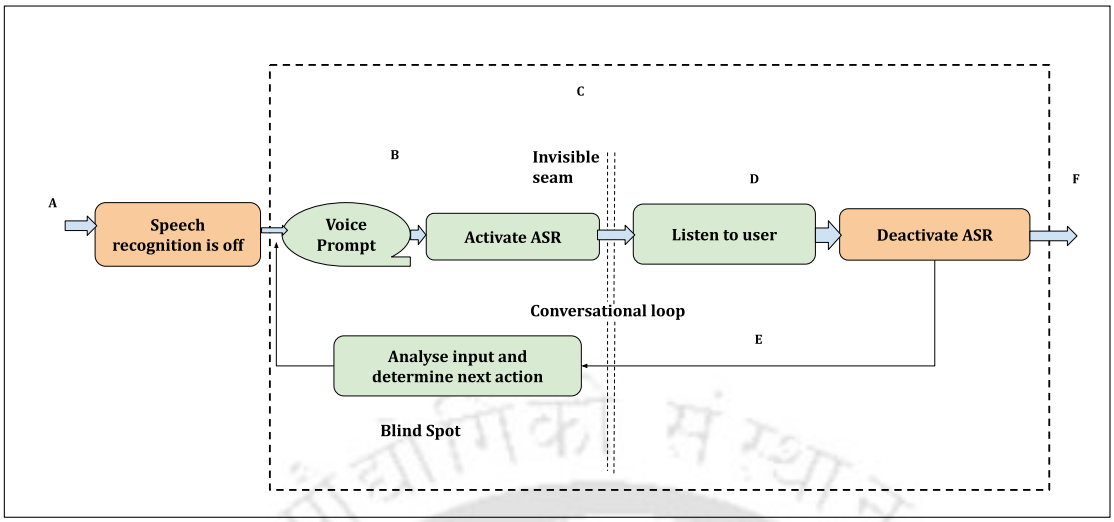


Figure 4.1: Blind spot and invisible seam in half-duplex VUI turn-taking.

4.3.2 The need for unnatural turn-taking

The work described here applies only to half-duplex voice interfaces. It should be noted that half-duplex turn-taking is intrinsically awkward for humans, making it impossible to pursue “natural" interactions (in the sense of human-to-human behaviours). In a natural (full-duplex) conversation, human speakers are accustomed to overlapping speech. One party takes a turn before the other party has finished speaking. This overlapping is well-known in the field. But certain technology and platform combinations (including ours) require half-duplex without recourse, creating a challenge for designers. Our solution to the challenge is to accept half-duplex constraints “as they are,” attempting to turn them into HCI advantages. The key is to adopt a mechanomorphic — rather than an anthropomorphic — design philosophy.

4.4 Research overview

To investigate possible solutions to the above challenges, we developed an experimental design aimed at answering the following questions:

- Can we design a half-duplex turn-taking protocol that detects timing errors, and in turn adopts machine behaviors likely to prevent or correct those errors?

- Can such an artificial turn-taking protocol entrain users to half-duplex behaviours?
- More specifically, can we teach users to 1) delay their spoken response briefly after hearing a machine prompt (error prevention), and/or 2) stop speaking and restart when speech onset is too early?
- What user behaviours do we expect before and after exposure to the protocol?
- Assuming success, can such a protocol be effective, efficient, and satisfying when completing tasks?

If the answer to any of these questions is "no," then we must conclude that half-duplex interfaces are inherently unstable and erroneous. Therefore, our study aims to develop a system that can make human participants aware of the invisible seam to respond on time during an ongoing conversation and thus resulting in smooth conversation with fluent (albeit unnatural) turn-taking. To implement this system, we exploited the social phenomenon called entrainment, also known as accommodation Giles et al. (1991), which occurs when dialogue partners adapt their behaviour to each other during an interaction Lubold et al. (2015) . There exists a considerable amount of evidence for entrainment from laboratory experiments such as gaze or facial expression Levitan (2020), acoustic-prosodic Levitan et al. (2012); Moore (2017a) word, lexical based Gustafson et al. (1997); Lakin et al. (2003), phonetic Gessinger et al. (2019); Pardo (2006) or speech Beňuš et al. (2018); Oviatt et al. (2004).

4.5 Related studies

This section synthesizes existing research that provides the conceptual and methodological grounding for the present work. The literature reviewed encompasses key themes such as system feedback mechanisms, the comparative challenges of full-duplex and half-duplex architectures, issues of blind spots and invisible seams in interaction, and the problem of unnatural turn-taking in voice user interfaces. Through a critical engagement with these studies, this section highlights influential theories, prevailing design paradigms, and unresolved gaps that have informed the trajectory of this research. Each subsection explores contributions that directly or peripherally advance our understanding of the nuanced dynamics in human-VUI interaction.

4.5.1 System feedback and discoverability features in VUI design

User expectations tends to rise with a rise in the popularity of Voice User Interfaces (VUIs). However, humans must work to adapt their communication patterns to the needs of the machines rather than machines adapting to humans Balentine (2007b); Jiang et al. (2013) . It appears that the designers of the current VUI systems make no effort to make a conversational system where human participants can adapt to the machine's behaviour during a conversation. The present VUI designs lags in terms of providing appropriate feedback and incorporating relevant discoverability features about the system's capabilities and intelligence. In absence of the same, users do not find opportunities to experiment with newer tasks. Rather, they shorten their sentences or use simplified languages with deliberate repetitions to get understood by the VUIs Jiang et al. (2013); Kennedy et al. (1988); Lohse et al. (2008); Luger and Sellen (2016); Pelikan and Broth (2016). Apparently, the interactions suffer from moments of frustrations with no success in delivering the intended task.

Recent work Baughan et al. (2023); Klein et al. (2024) found that users of VUIs frequently feel uncertain about the system's intelligence and boundaries — whether the system will adapt to them or they must adapt to it — and this uncertainty relates strongly to the need for clearer system feedback and turn-taking cues

Sciuto et al. (2018) conducted a study of specific interest about the use of VUIs in households. They reveal how does the lack of discoverability features in VUI limits users from experimenting with newer tasks in their interactions with VUIs. They believe that the lack of proper system feedback/affordances in the current VUI presents challenge before the users in discovering VUI features. Thus, to have a successful human-VUI conversation, there is an urgency to redesign the feedback system of VUIs to help users adapt or entrain to the VUI's behaviour.

4.5.2 Entrainment in human-human interaction

Human participants frequently adapt their speech patterns during conversations with each other Tannen (2007). This entrainment Huggins-Daines et al. (2006) occurs at various levels viz. acoustic-prosodic, linguistic style, speech rate, pitch, voice quality, intensity, and phonetic. For

example, in an investigation of relevance of vocal intensity, Natale (1975) finds that lowering or raising interviewer's vocal level brings a corresponding change in the vocal intensity of the subject during an interview. Likewise, Heldner et al. (2010) show that a speaker's pitch matches that of his or her partner when producing a backchannel. Conversational partners communicating over a computer chat Niederhoffer and Pennebaker (2002) tend to use similar linguistic style such as same word count, verbs, prepositions and social categories. Pardo (2006) also finds conversational partners converging on phonetic features (related to speech sound) producing imitation of pronunciation of their partners. . Levitan and Hirschberg (2011) while investigating entrainment for speech-rate both in turn-level and conversation level during a conversation, found that speaking rate in syllable calculated per second of interlocutors converges both at turn-level and conversation level. In an investigation of entrainment in terms of intensity, pitch, voice quality and speech rate, Levitan et al. (2015) discovers that at the turn level, all features show proximity and synchronization. Whereas at the session level, certain aspects display proximity, and some converge later in the conversation. In another context of playing a computer game by combinations of male and female pairs, Levitan et al. (2012) find that entrainment on acoustic-prosodic variables is most relevant in case of mixed-gender. They report entrainment occurring for all the features such as intensity, pitch, voice quality and speaking rate at turn level or session-level during the conversation. Lubold and Pon-Barry (2014) confirms a similar finding with their claim of acoustic prosodic entrainment existing in the collaborative learning dialogue corpus. They argue that interlocutors often entrain to the acoustic feature intensity during collaborative learning.

Available literature is univocal in stating association between entrainment and a successful and fluent conversation between humans. It examines various features of such a conversation. Let us now extend this discussion from human-human to human-computer interaction.

4.5.3 Entrainment in human-computer interaction

In HCI, we find a strong relationship between rapport perceptions and different types of pitch entrainment Lubold et al. (2015). Further it is seen that while taking part in a conversation with the computer, some participants do entrain to the speech output of the dialogue system. In a specific scenario of learning with a tutoring dialogue system, students are seen entraining

to the prosodic features of the system Thomason et al. (2013). Additionally, Gessinger et al. Gessinger et al. (2019) find that if the dialogue system appears less competent, i.e., if the system's outputs are judiciously introduced so that users do not feel these as interruptions, there is a chance of entrainment between the human and the computer. This is extended by Levitan et al. Levitan et al. (2016) who present an algorithm for creating dialogue systems that entrain to their user's speech. They investigated acoustic-prosodic entrainment. Their pilot study showed a positive association between entrainment and the system's perceived reliability. In the context of human participants conversing with an avatar,Benus et al. (2018) find users inclined to follow the avatar's advice. They conclude this behaviour as the result of users developing trust over the avatar. The trust builds when prosodic features are manipulated to generate entrainment.

In a recent study, Levitan (2020) criticize the approach of investigating entrainment on a broader scale. They argue that instead of investigating entrainment either at the global or local level, specific conversational segments be investigated for deriving relevant insights. Furthermore, as one can notice, most of the available literature including the studies we have mentioned, VUI interactions are designed to be anthropomorphic. The argument in vogue is to aim for achieving human-like qualities in VUI interactions. However, we also know that such a design fosters a mismatch between the capabilities and expectations of human users and the features and benefits provided by contemporary technology, producing a habitability gap Moore (2017b). Contrary to the anthropomorphic design approach, we see compelling arguments in favor of using rather a counter-intuitive approach in VUI designs. This approach is mechanomorphic in nature with potential for delivering improved task success Balentine (2007b); Lubold et al. (2015) and for avoiding the uncanny valley Mori et al. (2012) effect. Note that as computers assume a stance which is increasingly about being human-like, the users' experience of interacting with such systems varies drastically over time. The system allures users into interaction up to a certain extent only to induce revulsion later. Mechanomorphism, in an investigation of prosodic entrainment between humans and machines Bell et al. (2003), has enabled users to entrain to the speaking rate of the simulated dialogue system. The focus of this work is entrainment on turn-taking segments of conversing with VUIs. In these segments, interlocutors converge on temporal aspects of turn-taking. In other words, speakers adjust their turn-taking behaviour in response to the behaviour of their conversational partner Lubold and Pon-Barry (2014). We argue in favor of enhanced effectiveness and fluency in conversations with VUIs if turn-taking entrainment is appropriately exploited. Our approach leans towards

being mechanomorphism. Our test application and SmartWindow™ algorithm is designed to communicate VUI capabilities to its users over choosing to become 'human-like'.

4.6 Method

This section outlines the methodology employed in our experiment, detailing the participants, task, experimental setup, prototype used, and procedure. Our approach is designed to systematically evaluate the effectiveness of various feedback mechanisms in enhancing turn-taking behavior within Voice User Interfaces (VUIs). By providing a thorough description of each methodological component, we aim to ensure the replicability of our study and the validity of our findings. The following subsections will comprehensively cover each aspect of our experimental design, offering insights into how we conducted the research and gathered data to support our conclusions.

4.6.1 Experimental materials

1. **Platform and software:** The platform is a mobile device (iPhone SE smartphone) running PocketSphinx, an open-source ASR speech engine. The technology is a small-footprint offline recognizer that supports small-to-medium vocabularies. The ASR is known as a connected-speech recognizer. It accepts multiple-word, fluently co-articulated phrases (joined together without pauses). Such a platform is very useful for certain applications, but exhibits the unique limitations of half-duplex, small-vocabulary, prompt-response dialogues. Further, such platforms do not support fully natural-language conversational AI.

   Running on the platform is SmartWindow (an off-the-shelf turn-taking algorithm). It insulates low-level ASR from upper-level interactive software by inserting a half-duplex engine between them. The low-level ASR incorporates a voice activity detection (VAD) algorithm — to detect onset and offset of speech — and a segmentation algorithm that determines the boundaries between words. The half-duplex engine is under parametric control. Our experimental VUI application TS+SW manipulates these parameters with

an Event-Script that introduces highly interactive prompt-response exchanges of turns designed to expose turn-taking rhythms and entrainment. Even though participants all spoke with strongly accented Indian dialects, we used the US English generic acoustic model that comes with SmartWindow because we were interested only in turn-taking, and not in speech recognition accuracy per se.

2. **SmartWindow algorithm**: The algorithm is explained in algorithm1.

Algorithm 1 SmartWindow Turn-Taking Algorithm

Set ASR to **OFF**
    (a) Play Prompt-Tone pair
    (b) Set `STS_count` = 0
    (c) Start `WindowTimer`
Start `SilenceTimer`
Set ASR to **ON**
One of the following may occur:
    Case 1: If speech is detected **before** SilenceTimer expires:
        (a) Set ASR to **OFF**
        (b) Increment `STS_count` by 1
        (c) Play `STS_Tone`
        (d) Go to Step 5
    Case 2: If SilenceTimer expires and no speech is detected:
        (a) Wait for speech onset
        (b) If WindowTimer expires and no speech occurred:
            - Retry and go to Step 2(a)
    Case 3: If speech is detected after SilenceTimer (correct timing):
        (a) If more conversational turns remain:
            - Go to Step 2(a)
        (b) Else:
            - Speech successfully detected and session complete

At the beginning of a conversational sequence of the SmartWindow algorithm (please refer to algorithm 1), the ASR is off. This allows the app to talk to the user, taking its turn by presenting data, instructions, or a prompt to speak . The last thing the machine

"says" is the PromptTone, after which it sets up the user turn. Parameters relevant to this discussion include a counter that keeps track of spoke-too-soon (STS) occurrences, a window timer that specifies the total allowed duration for the user's turn, and a silence timer (1) that specifies the maximum width of the invisible seam — a time duration that we call the STS Zone. The user turn begins when ASR is turned on . The user turn must begin with silence.

The STS Zone allows detection of early user speech — almost always caused by the natural tendency to overlap (the user is responding to the machine's prompt with a reply that comes a little too soon). This is disallowed in a half-duplex interface. For this study, we used a value of 175 milliseconds for the silence timer, meaning that the user is expected (required) to remain silent for this brief amount of time. If user speech overlaps with the end of the prompt/tone pair, the speech is detected. ASR must be turned off to allow playback of the STS tone , which serves as an alert to the user that speech is too early. In this way, we can sonify the invisible seam, making the user aware of it. Before resuming ASR, the silence timer is restarted — we must have a full 175 milliseconds of silence before the speech recognition window can be considered open.

3. **Voice-activity detection**:Voice-Activity Detection (VAD) is performed by PocketSphinx under parametric control. The four VAD parameters and their settings are shown in Table 4.1. These parameters remained the same throughout the experiment. VAD is responsible for detecting the onset and offset of user speech.

Table 4.1: VAD Parameters

Name	Description
VAD Threshold	Energy above this threshold is speech; below the threshold is silence.
StartSpeech	50 ms — time in milliseconds that energy must remain above the VAD threshold before speech onset is declared
PostSpeech	750 ms — time in milliseconds that energy must remain below the VAD threshold before speech offset is declared
Prespeech	100 ms — additional low-energy speech that is prepended to the beginning of a captured utterance after onset to include starting consonants

It should be noted that VAD detects an entire user utterance from silence to silence. In

this study, that single utterance contains one or more digits that are spoken fluently. When the entire utterance is captured cleanly, the likelihood of accurately segmenting and recognizing those digits goes up.

4. **VUI application (Event-script):** The VUI application we developed is known as an Event-Script, which repeatedly calls the SmartWindow subroutine with a sequence of prompt-response turns. The Event-Script is contained in a text file that is read and acted upon by the SmartWindow software, thus making SmartWindow a type of authoring system. The vocabulary is a probabilistic language model (PLM) that recognizes English digits (zero through nine plus "oh") uttered by the human participant. We used the text-to-speech (TTS) synthesis available on Apple iOS, plus tones developed for this purpose to play the voice prompts and tones.

Table 4.2: Event-Script Parameters (Prompts & Tones)

Name	**Description**
Introduction	"Thanks for doing the experiment." Below the threshold is silence.
Prompt	Please say the first number.", "Say the next number.", "Next number?"
PromptTone	PromptTone.m4a — a musical tone pair moving from ii7 to V7 (dominant)
STSTone / QRPrompt	STSTone.m4a — a dissonant triple-tone. "Say that again?", "Sorry, once again?", "Again?"
QRTone	QRTone.m4a — a musical tone pair moving from V to V7 (dominant 7th)
FRPrompt	Full Retry Prompt — "Sorry, say the number?"
AckTone	AckTone.m4a — acknowledgment tone, a musical tone, serves as reward/success

To tailor the turn-taking behaviour of Event-Script interactions, we specified the values for various PocketSphinx and SmartWindow parameters (please see tables 4.1,4.3) . The primary function of the subroutine is to detect and report a spoke-too-soon condition. STS occurs whenever the participant speaks before the time required to open the window

has elapsed. The goals of playing this so-called "STS tone" are: 1) make the participant aware of the invisible seam, and 2) entrain the participant to the turn-taking expectations of the machine (i.e., wait for a brief amount of time, or stop and repeat speech).

Table 4.3: SmartWindow Timing Parameters

Name	**Description**
WindowTimer	6000 ms — the amount of time the window for accepting user speech will remain open.
SilenceTimer	175 ms — the amount of time that silence must precede the start of user speech; aka the "width of the invisible seam," or the "duration of the STS Zone
MaxSTS	8 — the maximum number of spoke-too-soon occurrences before declaring noise

5. **Prompt-script:** Participants know what to say by reading from a Prompt-Script(Appendix A), as shown in dialogue 2 (4.2). A very short introduction is followed by a sequence of prompt-response events as specified in the Event-Script. The first two and final two events (out of a total of 12) are shown.

6. **Tapering and prompt-response rhythm:** Notice in Table 4.2 that the prompt changes from event to event. Prompt is a sequential parameter, which is stored as an array (each element in the Prompt array of table (4.2) is separated by a comma-space pair). The use of a sequential parameter for the prompt allows the conversational loop to vary each time through. We took advantage of this SmartWindow feature to decrease the length of each prompt sequentially.

   This means that the first time a prompt is needed, it will say "Please say the first number." For the second prompt, it will say, "Say the next number." For the third prompt, it will say, "Next number?" Since the third prompt is the last prompt in the prompt array, it will be used for all subsequent passes through the conversational loop. The algorithm is known as prompt tapering. By shortening each occurrence, the dialogue is less repetitive and begins to move more quickly — a phenomenon that has the effect of pushing the

user forward in the multi-turn sequence. Our goal in tapering is twofold: 1) to observe the start time of user speech for the first three iterations, and 2) to determine if, once the steady-state rhythm of unchanging prompts ("next number?") becomes predictable, users begin anticipating the question and therefore move the start time of their speech leftward. The internal mental goal might be to get as close to the invisible seam as possible without triggering STS. The effect should be amplified by the variable length of digit string inputs (the length of each series varied in the range of 2-6 words long). One of the experimental goals is to observe participant behaviour in response to this tapered, prompt-response rhythm.

App: Thanks for doing this experiment. Please say the first number <prompt tone>

User: "Four-Three-Six-Zero-Three"

App: <ack tone> Say the next number <prompt tone>

User: "Eight-Two-Two"

. . .

App: <ack tone> Next number? <prompt tone>

User: "Zero-Four-Seven"

App: <ack tone> Next number? <prompt tone>

User: "Six-One"

Figure 4.2: Dialogue 2: Typical Pass Showing Variable-Length Digits & Tapering

User: Six-Two-Eight-Three

User: Two-Eight-Three

Figure 4.3: Dialogue 3: Deletion error caused by speaking too soon

7. **The log file:** The SmartWindow subroutine generates a log file that contains a listing of turn-taking events. The experimenter saves the log file after each pass and thus has a total of three log files for each participant. The comparison of the log file against the video is focused mainly on two things: 1) the occurrence of spoke-too-soon events, and 2) the presence of deletions at the start of the recognized number. Except for deletions,

digit recognition accuracy is not of particular importance. The association of STS with deletion errors is a known effect 4.3. The log file also contains time stamps that allow investigation of tapering and rhythm.

8. **Interviews and questionnaire:** We conducted an interview for participants once they finished the experiment. See Appendix D for the questionnaire that guided the interview. We recorded the interviews for analysis, coded them to do a qualitative analysis of the data, and compared them against the quantitative analysis.

4.6.1.1 Experiment setup

We planned to set the experiment with some predicted behaviours and thus the application was set accordingly.

Predicted Behaviours and Hypotheses The primary goal of the experiment was to investigate participant reactions to STS conditions — the important software feature that sonifies the invisible seam. Predicted user behaviours are shown in Appendix F Table F.1. SmartWindow is based on these behaviours as design assumptions. A secondary goal, the effect of tapering, is discussed in the section "*Tapering and prompt-response rhythm*". All behaviours are numbered in the column, and we will refer to them by number throughout this thesis. General Expectations/Predictions regarding STS are:

- There should be a large number of STS occurrences on Pass 1 for almost every user. This reflects the tendency for conversants to overlap their speech as a natural talking behaviour.
- There should be fewer STS on Pass 2 and fewer still on Pass 3 for all users as a result of entrainment.
- An exception to #2 is those who converge to behaviour #6, for whom we expect an increase in STS.

Behaviours that are desired (e.g., those that contribute to the effectiveness and efficiency of the interface) are shaded in grey. Expected reactions to tapering can be considered “desired” to the extent that they contribute to convergence on behaviours #1 & #6 or to long-term synchronization. These behaviours demonstrate entrainment. Users that converge to desired

behaviours are expected to experience upticks in ASR accuracy, improved prompt-response rhythms, higher throughput and greater satisfaction. Behaviours #3–5 are undesired — it is the job of SmartWindow to extinguish these three behaviours. All of the anomalous errors are technology-caused and will be removed from the HCI calculations.

Description of Behaviour

- The first behaviour is normal capture — users hear Prompt followed by PromptTone, pause briefly, and then speak their response. The STS tone is designed to be somewhat annoying (a sort of punishment), and certainly noticeable. So at least some participants are expected to conclude that a brief delay in speech prevents it from happening, adopting behaviour #1 over time.

- Behaviours 2a and 2b are both timeouts. They both are a silence timeout — the machine plays Prompt and PromptTone, waits for anything, but only silence is detected. The window eventually times out (for the experiment, WindowTimer parameter is set to 6 seconds). This occurrence is extremely rare, but the FRPrompt (full retry, "say the number?") is expected to recover it when it does happen. 2b is the same event as far as the machine is concerned. But — as viewed on the video — the participant actually spoke, but the speech was not detected. Either it was very early (before the invisible seam) or contained embedded silences that fell accidentally within the STS zone. Behaviours #3 through 6 are all user reactions to the spoke-too-soon alert (STSTone).

- The "plough-through" behaviour implies that the participant 1) does not notice the tone at all (unlikely), or 2) hears the tone but makes no connection with its meaning. "Noticing" the tone may be manifest by certain facial expressions.

- Abruptly stopping speech implies a reaction to the tone without necessarily imbuing it with any meaning. Waiting for something new to happen implies that the user has halted due to the perception that there was "some kind of problem," but not necessarily with any understanding. The quick retry (QRPrompt, "once again?") is designed to recover this condition.

- Abruptly stopping but then continuing is a "What was that?" reaction. We hypothesize that these participants are primed to learn responsive behaviours more quickly.

- This behaviour arises when the participant commits an STS error, hears the STS tone, stops reading a string, and then starts reading again from the beginning. Further, some users may infer that the "error tone" implies that speech was missed, and a repetition is called for. The result is behavioural conditioning (entrainment) that extinguishes the speech overlap that triggers spoke-too-soon conditions.

- The ideal behaviour is the ultimate goal of SmartWindow. We expect to see occasional spontaneous learning, but very little in early repetitions. We expect the error-recovery behaviours for the other conditions (#1–#4) to lead eventually to entrainment (converge to the most stable #6).

  In addition, we have the following hypotheses regarding tapering & prompt-response rhythm:

- We should see a slight but measurable difference between STS occurrences between the first three numbers (more STS on the first turn, fewer on the second, fewest on the third) due to tapering. This effect should be more pronounced on the first pass.

- We should see an increase in the occurrence of STS after the steady-state rhythm "Next number" proceeds. There are two reasons: 1) the user begins to predict (anticipate) the prompt, leading to earlier, overlapping responses, 2) those users who converge to stop-repeat behaviour #5) see a marked reduction the penalty for STS and therefore allow themselves to speed up. Both of these imply entrainment.

4.7 Execution of the experiment

4.7.1 Environment

The experiment was conducted in a well-lighted, soundproof recording room inside a lab. An iPhone installed with the VUI application was placed on a table with the help of a stand. A chair was placed in front of the table to allow the participant to sit comfortably while interacting with the mobile application.

A video camera equipped with a microphone was placed on a tripod to record the en-

Figure 4.4: (Left) A participant interacting with the VUI in the lab; (right) the lab environment in which the experiment was conducted.

tire interaction. The camera was set at a proper distance to capture the facial expressions of participants. Figure 4.4 shows the experimental set-up used for the experiment.

4.7.2 Procedure

Participants present themselves at the appointed time, waiting in the lab's common seating area. There they fill out appropriate forms and read printed information regarding the study's overall goals and procedures. Participants must then give written consent. The experimenter provides a printed script for the participant to read out loud during the experiment. The script contains 12 strings of digits. See Appendix A for details on the vocabulary. The experimenter takes the participant into the experiment room and seats her at the desk. The experimenter describes the setup and the task, also indicating availability to help in case of any problems or questions. After this short explanation, the experimenter leaves the room, observing the experiment through the glass pane of the recording room.

The participant starts the application by touching a big PLAY button displayed on the screen. After starting, there is no touching or viewing of the display — the task is voice-only. The participant hears each spoken prompt and replies by speaking each number. The application responds with an acknowledgement tone after each turn. The process continues until the participant has uttered all twelve numbers in the script. The application automatically stops once the complete script is finished. See Dialogue 2 (4.2) for an example of how the

sequence sounds. After each 12-turn pass, the experimenter re-enters the room to perform a brief interview. During the interview, an assistant stops and saves the video recording and the SmartWindow log file to an iOS folder on the iPhone. The same process is repeated for each pass. On completion of the third pass and its interview, the experimenter thanks the participant, and the participant leaves the lab.

4.7.3 Repeated passes for the experiment

To complete the experiment, participants must go through 3 different passes. Each pass has a total of 12 events (turn pairs). This offers ample opportunity to observe errors caused by highly interactive prompt-response turn-taking. The 12 events constitute a conversational loop as shown in Figure (1). In that figure, the loop from point B to point E will repeat 12 times, incrementing its prompt each time through, and falling into a rhythmic repetition of the shortest prompt, "Next number?" from the third turn on. The task is randomized for each pass. Note that the SmartWindow algorithm allows for a single retry based on two kinds of timeouts (more than WindowTimer seconds of either silence or speech). The retry is a second turn within the turn, meaning that a user might experience 24 separate turns in a single pass, for a total of 60 possible (albeit extremely unlikely) turn exchanges for any given participant.

4.7.4 Participants

A total of 13 individuals, (M=27 years; SD=3.28; Range=18–36 years) voluntarily participated in the study. Most of the participants were students from the authors' institute itself. The participant pool was created using the snowball sampling technique. All of the participants were capable of reading and writing English.

Table 4.4: Summary of Turns

Participant ID	Total Turns 1st-try/2nd-try	Total STS Pass: 1/2/3	Recovered STS Self-/Algorithm	Failed Event
p31	36/0	0/0/0	0/0	0
p15	36/2	1/1*/0	2/0	0
p16	36/2	2/0/0	2/0	0
P08	36/4	3/1/0	4/0	0
P30	36/1	1/0/0	0/1	0
P26	36/3	1/1/1	0/3	0
P23	36/4	2*/0/0	2/0	0
P24	36/8	3*/3*/0	5/1	0
P33	36/4	2/4/1	0/4	4
P14	36/4	6/0/2	1/4	4
P10	36/10	5/7/10	2/12	8
P12	36/5	4/7/1	1/4	1
P17	36/2	7/3/3	0/2	11
TOTALS	517	82	19/31	28

4.8 Result and analysis

The experimenters collected a large quantity of data. This includes videos, log files, and interview responses. There is much to be discovered, so we have divided the analysis into three parts:

4.8.1 Summary of total turns

Table E.1 (Appendix E) summarizes turns taken by the thirteen participants across all three passes. All thirteen participants completed all twelve tasks for all three passes, representing a total of 468 separate turn exchanges between machine and user. In addition, all but one of the participants experienced at least one additional 2nd-try turn due to timeouts during the first turn (the WindowTimer was set to 6 seconds). The timeout was usually the result of silence following mistimed user speech, and occasionally was caused by the concurrent-timeout problem . Recovering errors with a 2nd-try represented an additional 49 turns. We therefore count 517 total turns exchanged between machine and user, shown in column two of table 4.4.

4.8.2 Divergent user behaviours: analysing types of populations in turn-taking Dynamics

The population of users can be divided into two groups — sheep and goats. Most users were sheep, as shown in the top eight rows of Table (Appendix E, Table E.1). Five users were goats, responsible for the large majority of errors . This bifurcation is typical in ASR research Doddington et al. (1998). The sheep population had a natural tendency to infer appropriate behaviours, which included listening for the prompt, hearing the tones, falling quickly into a prompt-response rhythm, and avoiding the invisible seam. The population of sheep in this study was surprisingly large. Goats, on the other hand, tended to form a different mental model of the intent and internal rules of the app, stepping on the invisible seam (generating a large number of STS errors), and perseverating despite error-correction efforts. They constitute the most

interesting group, and we will devote most of the remaining discussion to how SmartWindow handled them. One especially telling aspect of goats can be seen in the second column of Table 4.4 in the form of spoke-too-soon (STS) occurrences. Sheep generally saw few STS, and the number generally diminished over time. Goats experienced more — sometimes far more — including in second and third passes. This indicates little learning and less entrainment (although there are some surprises hidden in the data, as described in the discussion below).

SmartWindow is designed to give a second try to users who are unable to provide input. After the second try, SmartWindow gives up, passing a failure status to overlying software, which in turn is responsible for correcting the problem, re-synchronizing the dialogue, and trying the turn again. The 28 of such failures were all from the goats. That does not mean that a given application would fail on such circumstances, only that more extensive interaction would call for — extending the dialogue and affecting both effectiveness and efficiency.

4.8.3 Observed behaviour

In examining various turn-taking behaviors, the data reveal several distinct patterns and deviations. The data revealed are added in a listed table (Appnedix A). Firstly, the behavior of normal capture occurred 352 times, significantly more than anticipated. This indicates a higher frequency of this behavior in the observed interactions. In cases of silence timeout, two subcategories were noted: complete silence timeout (no speech), which occurred zero times as expected, and silence timeout due to mistimed speech, which happened 9 times, slightly exceeding expectations.

When analyzing reactions to specific turn-taking signals (STS), there were 55 instances of the “plough-through” response, slightly fewer than expected, observed among three participants. Abrupt stopping followed by waiting for the next step occurred twice, aligning with expectations. Brief stops followed by a continuation of the utterance and abrupt stops with subsequent restarts from the beginning were noted 3 and 8 times, respectively, both as expected.

Regarding reactions to tapering, certain patterns emerged, though data was insufficient to draw firm conclusions. Notably, the STS on the first event followed by no STS on the subsequent events lacked enough data for a definitive conclusion. Similarly, an increase in

STS from events 4 through 12 could not be conclusively analyzed due to limited data. The 1st-try/2nd-try STS pattern on quick retries occurred once, which was expected, and general synchronization on prompt-response rhythm and speech rate was observed 11 times, also as expected

4.8.4 Broad analysis

4.8.4.1 Long-Term Entrainment

We saw a great deal of evidence for three long-term adaptations that participants made as a result of repetition. Users were observed chunking their speech into separate units interspersed with the highly repetitive machine prompt. Some of this temporal adjustment appears to be the result of prompt tapering. As the prompt sequenced from long to short, users were encouraged to slightly delay onset of speech:

App: Please say the first number.
User: "One-Eight-Seven-Three."
App: Say the next number.
User: "Six-One."
App: Next number?
User: "Four-Five-Nine."
App: Next number?

Figure 4.5: Prompt tapering enhanced entrainment

After just a few prompt-response cycles, the pattern of question-tone-pause became imprinted. The urge to fall into a rhythmic (periodic) back-and-forth proved irresistible — almost like one pendulum entraining another. This was true of both sheep and goats, although adaptation was much quicker with sheep, and much slower with goats.

Users were observed adapting their talking tempo to that of the machine. In one example of this phenomenon, a participant began pass 1 with an especially slow and deliberate rate of speech — perhaps in the interests of "clarity" or "precision." The initial speech was sometimes

even hyper-articulated, leading to timeout problems with long numbers (five or more digits in length). However, after experiencing the machine's prompts and tones repeatedly, this user gradually presented digit strings at a faster and faster rate. We validated this observation from the total duration of each number reported by the recognizer in the log files for each pass. For example, it took 4.009 seconds to read out the number "Four-Three–Six-Zero-Three" the first time, whereas it took 2.047 seconds to read the same number during the third pass. Most users exhibited this behaviour to some degree, and the slowest of the speakers universally. We ascribe this adjustment to the repetitive periodicity of the prompt-response dialogue style, which led users to converge to the rate of speech exhibited by the machine — clearly a form of entrainment. Note that speech fluency and rate of speech always went up — there were no examples of participants with naturally-fast speech slowing down. All participants "settled into" a prompt-response rhythm approaching that of the machine prompt + tone tempo.

Users rarely spoke of tones during the post-experiment interview when asked how they knew what to say and when to speak. At least not at first. When probed, they indicated awareness of the tones, and correctly inferred the meaning of all four tones, but could not remember exactly the sound and sequence. They just "made sense" and "felt comfortable." We conclude that tones exist in the very short time zone just at the threshold of consciousness — as is the case with all turn-taking cues. They operated subliminally but effectively. (see the later discussion of the special-case issues of STS tone, which did not behave as predicted.

The above three adaptations happened quickly for sheep—within pass 1. Goats required more instances of the cycle, but converged eventually to the machine's natural rhythm, usually during pass 2 and certainly during pass 3.

4.8.5 Unexpected STS results

There were plenty of STS occurrences (68), but they were mostly exhibited by goats — sometimes accidentally due to ploughing-through behaviours, but often by one or two users who — although they encountered them on almost every turn — completely ignored them. In several cases, anomalous machine behaviours (e.g., embedded silence false accepts) obscured STS awareness. In some cases, they even caused false learning. One participant experienced the STS

tone, ploughed through, and was then rewarded with the Ack tone due to false acceptance. The result was a growing confidence in ignoring the STS tone as a proper behaviour. We were expecting to see more sheep encounter STS events simply due to the natural turn-taking behaviour of overlapping speech. But our sheep were so sheepish that they fell quickly into behaviour #1, and never had a chance to learn the more efficient behaviour #6.

4.8.6 Extinguishing behaviours #3, 4 and 5

We wanted to see how many of the 13 users adopted behaviours #1–3 early in the experiment, and of those, how many saw them extinguished? Our hypotheses expected a significant number of behaviours #3–5, at least for the first pass (when little learning has occurred). Such an occurrence corroborates the unnaturalness of a half-duplex turn-taking protocol. Both behaviours are indicated by an initial STS, followed by a retry after a silence timeout.

To answer this, we find that 5 out of 13 participants adopted the behaviours # 1-3 during the first pass. As we investigate Table 4.5, these participants committed STS error at least for 6 times, which is a noticeable number to assume the behaviour as an adoption. However, later in the second try, these participants committed appreciably fewer errors compared to the first try (look at column 4 of the Table 4.5). We ascribe this to the efficiency of our VUI application. It helped either reducing or extinguishing these errors to a greater extent by making participants aware of their erroneous behaviour using its tones and prompts.

4.8.7 Reactions to tones

Many participants said during the interview that they knew when to speak because of the PromptTone. But they didn't, according to the data, as they continued to commit STS errors. It is interesting that a user can form a cognitive understanding of the meaning of a tone without mapping that understanding onto behaviour. This is a question worthy of more detailed research.

During the video analysis we observed that participants noted on STS tone. This behaviour

Table 4.5: Response to Behaviours #1–2

Participant ID	1st-Try STS	2nd-Try?	2nd-Try STS	Recovered?
P16	2	N	0	-
P15	2	N	0	-
P14	7	Y	2	Y
P12	9	Y	2	Y
P24	1	Y	0	Y
P23	2	N	0	-
P26	3	Y	1	Y
P30	1	N	0	-
P33	6	Y	1	N
P31	0	N	0	-
P08	4	Y	0	Y
P10	18	Y	5	N
P17	11	Y	1	N

arises when the participant commits the STS error, and the application plays the STS tone. The participants exhibiting this behaviour were found to show some facial expression/ change in the facial reaction. On hearing the STS tone, some participants looked up from the script to the iPhone's screen with a confused face. Some participant looks up from the script, open their eyes wide/ raise their eyebrow, looking into the iPhone's screen with a confused face. Importantly, although several participants glanced towards the device, the screen did not display any task-related information and showed only a single large play button; therefore, these glances appeared to be attempts to seek confirmation rather than responses to any visual cue. It is also relevant to note that participants were not provided with any training on experiment, as the study was intentionally designed to observe their natural, first-time adaptation to the turn-taking constraints. During the video analysis, 10 out of 13 participants were found to be exhibit this behaviour. It is worth mentioning here that, although during the video analysis, we could not capture any reactions on the rest of the 3 participants, during the interview, these participants express their confusion about STS tone. This indicates that, all 13 participants somehow noted the STS alert. Out of 10 participants, who showed reactions on STS alert, 9 participants realized the STS alert on the first pass itself. Of these 9 participants, 5 participants found to show this

behaviour.

During the interview, while asking if they found the tone STS as an interruption, 10 out of 13 participants answered it as not interrupting. However, the remaining 3 participants mentioned the tone of STS as interrupting. We also noticed that while describing the process of "How did they know when to speak," we also noticed that these three participants mentioned the STS tone as an error tone that made them stop speaking before the "StartTone".

During the interview, the researcher asked each participant how they knew when to speak. In response to this question, 12 out of 13 participants revealed that for the 1st or 2nd string of numbers from the first pass, they followed only the spoken Prompt to begin speaking. However, from the third string of numbers onward, they realized the playing of PromptTone as an additional cue to start. Five participants used different words to describe the PromptTone: such as “beep”; “some tin-tin sound”; “tone with ting sound”; “that ting”; “sound cue”; “chime”; “smooth clang type noise”; “sound blink”; and “tuntun sound”. We found only one single participant who did not notice the PromptTone at all, explaining that he followed only the spoken Prompt as an indication to start speaking. He used the words "voice prompt and computer recorded voice" to describe the Prompt.

From the video recording, we observed that one of the 13 participants did not show any facial or bodily expression when the STS tone buzzed. Each time she commits the STS error, the STS tone buzzes and with no surprised look on her face, without looking to the iPhone screen she only waits for the machine to provide the next voice prompt instruction to read the next string. She never paused during the STS tone buzz and ploughed through the tone to finish speaking the entire string she was speaking. It was found that this participant never took any pause, looked surprised or looked onto the iPhone’s screen during both pass1 and pass2. However, it was observed that the participant looked into the screen only when there was no feedback tone from the VUI. During the 3rd pass also, her behaviour remained the same for the entire duration except for only one time. During the 3rd pass, she never took pause, looked surprised, or looked at the screen when the STS tone buzzed for all the strings except for the one. However, she never looked surprised on hearing the tone or took any pause; instead, she looked to the screen with a mild smile as if the VUI app installed on the iPhone was showing an appropriate behaviour by playing the STS tone. This participant was found to commit the STS error in the sequence of 09, 10, and 13.

4.9 Discussion

We observed that prompt-response dialogue style repeating periodicity made the users synchronize their speech rate to those of the VUI. We consider this as an exhibit for natural entrainment. Moreover, the behaviour of all the participants having constantly increased speech fluency rate during the experiment gave an impression of arise of more-settled behaviour with the VUI. This confirms presence of entrainment, as stated in earlier studies Gessinger et al. (2019); Lubold et al. (2015). In contrast, among 8 users (out of total 13 users) who adopted behaviour #6 – 8, 30, 15, 16, 12 constituted an interesting lot. 3 of these 8 users somehow learnt to delay the speech to avoid STS. They showed significant reduction in STS error in pass3 in comparison to pass1. The remaining 5 were sheep- who did not commit many errors in pass1 only and a little error in the pass3. We consider this as an exhibit for unnatural entrainment (desired STS behaviours). These two observations demonstrate the effectiveness of the interface, showing entrainment in the form of behavioural adjustment to the temporal patterns of the half-duplex protocol. For a course judgment, we simply compare the number of spoke-too-soon errors for pass 1 against pass 3 (see Appendix B,about the summary of observed behaviour). During the analysis we observed significant reduction in case of two participants. We conclude that these two participants learnt to delay their speech in order to avoid STS. This behaviour is the second most useful behaviour but still effective at reducing deletion errors.

Humans are known to adapt well to rhythmic (read: "periodic") repetitions, and indeed we saw such adaptation in this study. It was observed that the participants adapt to the temporal turn-taking behaviour of the VUI over the period. For example, participants who were committing more STS errors during the first and second pass, gradually started committing much fewer STS errors. The participants overcome the problem of committing STS error by waiting for a certain amount of time allowing the VUI to play the PromptTone as an indication to start. Thus, most of the participants started adapting the temporal behaviour of the VUI aligning with it to have fluent prompt-response conversation. While attempting to reason our observations, we believe that our observations were like those exhibited in case of interactive alignment, although with a difference. In interactive alignment, interlocutors subconsciously align themselves in a dialog Pickering and Garrod (2004). This alignment happens at different levels of linguistic variations and not because of low-level temporal features as shown in the current study.

Further, post-experiment interviews suggest that all (except two) of the participants used mechanomorphic references while speaking of their interactions with the VUI. Their responses to the question, "How did you know when to speak?" included mentions of interacting with a machine and not with a human-like conversation agent. For example, they said "there was a sound, the beep, and I got to know that I should speak now", "the devise said and there was a sound". We ascribe this to the use of appropriate tones indicating VUI's system state. These helped achieving an appropriate turn-taking protocol between the VUI and the participants.

4.10 Conclusion

This study investigated user turn-taking behavior during interaction with a Voice User Interface (VUI) governed by a custom-designed half-duplex turn-taking protocol. As the first empirical step toward theorizing mechanomorphic VUI design, the experiment examined how users adapt to structured temporal cues in the absence of anthropomorphic features. The findings suggest that even in a constrained half-duplex environment—where only one party can transmit voice at a time—users can gradually entrain to a fixed prompt-response rhythm, provided that the system offers clear, consistent auditory signals.

Over the course of repeated interactions, participants increasingly synchronized their responses with the system's timing cues, reducing premature speech and exhibiting smoother turn transitions. This behavioral adjustment occurred without any explicit anthropomorphic design, reinforcing the value of a mechanomorphic approach wherein the system communicates its internal state and constraints transparently, much like a machine rather than a human mimic. Post-study interviews confirmed that users perceived the interface as a non-sentient machine and relied on structured auditory cues—particularly the PromptTone and, implicitly, the Spoke-Too-Soon (STS) tone—to guide their turn-taking. These results support the hypothesis that well-designed, sonified boundaries in half-duplex VUIs can foster natural user entrainment and reduce conversational breakdowns.

One of the most compelling implications of this study is that half-duplex interaction—often viewed as a limitation—can become a strength if its temporal constraints are rendered perceptible through auditory cues. The STS tone, in particular, served to "sonify the invisible

seam," transforming an abstract system constraint into an actionable user signal. Moreover, anecdotal observations suggest that users who learn to halt and repeat after encountering the STS triple-tone (behaviour #6) tend to interact more efficiently. Over time, some even exploit the boundaries aggressively, achieving higher throughput by skirting close to the turn-taking seam—evidence of adaptive learning. These insights open the possibility for future systems to include built-in "hints" or "tips" triggered by user behavior (e.g., behaviours #3–#5), making such protocols self-reinforcing and potentially self-teaching.

However, the study has several limitations that offer directions for future work. First, participant recruitment was based on snowball sampling with a limited sample size, which restricts generalizability. Larger, randomized studies would help validate the observed behavioral patterns and support the formulation of more robust design hypotheses. Second, the auditory tone set—while effective—was based on musical syntax without deeper examination of its perceptual affordances. Whether alternative sounds (e.g., mechanical clicks, chimes, or culturally meaningful tones) would yield equal or greater effectiveness remains an open question. Exploring these variations could inform not just interaction design, but also opportunities for branding and user customization.

Third, the task vocabulary used in the experiment was limited to number strings of variable lengths. Expanding this to other domains—such as healthcare questionnaires, educational tasks, or language learning applications—would allow researchers to explore how vocabulary complexity and cognitive load interact with structured turn-taking protocols.

In sum, this study demonstrates that structured turn-taking mechanisms grounded in mechanomorphic design can facilitate smooth, intelligible interaction in half-duplex VUIs. By reducing reliance on anthropomorphic cues and instead capitalizing on clear, interpretable system feedback, such designs hold promise for improving user experience in constrained interaction environments. These findings lay the groundwork for further experimentation, particularly around the role of non-speech auditory cues, which will be explored in greater detail in the subsequent chapter.

Chapter 5

Effect of Non-Speech Feedback Tone on Entrainment During Prompt-Response Conversation

5.1 Overview

This chapter presents a comparative study evaluating the specific role of a dissonant, non-speech auditory cue—Spoke-Too-Soon (STS)—used as corrective feedback in a structured turn-taking protocol for half-duplex Voice User Interfaces (VUIs). While the prior experiment in Chapter 4 (Experiment 2) demonstrated that the full protocol supports user entrainment, it was unclear whether this effect stemmed from the protocol design or the corrective cue in particular. To investigate this, two groups were compared: one with the STS tone and one without, under identical task conditions. By analysing timing errors, user adaptation, and post-task reflections, the study assesses how the presence of corrective feedback influences temporal alignment and cue awareness.

5.2 Introduction

Timely turn-taking cues are essential for maintaining fluency and reducing misunderstandings in voice-based conversational systems. In human dialogue, even brief delays or overlaps can disrupt interactional flow; the same is true in voice-based systems. The absence of clear coordination can lead to premature user speech, missed input, and task failure—especially in interactive conversations like prompt-response exchanges. In such cases, turn-taking often follows a half-duplex model due to limitations in platform or technology (like the one we have used), making precise timing critical. In such settings, human users may struggle to infer when the system is ready to listen, particularly if feedback cues are ambiguous or delayed. Effective turn-taking cues are necessary for making human users aware of the machine's requirement for turn exchange. Many platforms attempt to simulate full-duplex interactions using turn-taking protocols, but these bring their own set of HCI challenges. This work does not focus on such approaches. Our focus is on those systems that depend on certain technology and platform combinations (like the one we have used) and require half-duplex without recourse, creating a challenge for designers. Moreover, in safety-critical contexts such as in-car voice interfaces with quick exchanges, half-duplex turn-taking remains essential, as it enables structured and predictable interaction. Studies have shown that in contexts such as in-car systems, if turn-taking is not quick and well-coordinated—as is often the case in half-duplex systems—it can increase driver stress, cognitive load, and visual distraction. In such scenarios, controlled, one-at-a-time interactions may be more suitable than overlapping dialogue models McWilliams et al. (2015).

Non-speech auditory cues—such as beeps or tones—offer a powerful mechanism to support user awareness. Unlike speech prompts, which are semantically rich but cognitively demanding, tones are fast, unambiguous, and more easily perceived as signals rather than content. Research has shown that users are more likely to respond accurately and quickly to tonal feedback, particularly in time-sensitive or repetitive tasks. Nees and Liebman (2023) conducted a systematic review on brief audio alerts in human–machine interfaces. Their findings underscore the importance of using perceptually salient audio cues—such as speech and hybrid sounds—in time-critical interactions like alerting systems and prompt-based tasks. However, their review does not specifically address conversation-based VUIs, and it remains unclear how well these

findings translate to supporting effective turn-taking in dialogic interactions.

Our previous study introduced (Chapter 4) a structured turn-taking protocol that incorporates different turn-taking cues for different purposes (please refer to table 4.2), ranging from system prompts asking the user to start to providing feedback to the user, along with an STS tone that makes the user aware of the system's temporal pattern for turn-taking. In this study, our objective is to examine the effectiveness of the STS tone as a turn-taking cue in promoting temporal entrainment with the system, an important factor contributing to successful and efficient task completion. Does the absence of such a dissonant tone really affect entrainment for prompt-response-type conversation?

5.3 Related study

5.3.1 The crucial role of timing awareness in voice User interfaces

Precise timing and robust handling of recognition errors are critical to the success of Voice User Interfaces (VUIs), particularly in systems that operate under turn-taking constraints. Deletion errors, where the system omits or fails to register parts of the user's utterance, have been shown to significantly contribute to task failure and user frustration.

Liesenfeld et al. (2023) emphasises that timing bottlenecks in real-time ASR, especially in overlapping speech or rapid turn exchanges, impair intent recognition and disrupt conversational flow. Their multilingual study demonstrates that poor timing—not just poor recognition—can be a fundamental barrier to smooth human-machine interaction, particularly in task-oriented scenarios like navigation or information retrieval.

In a similar vein, Sigtia et al. (2021) presents a voice trigger system that balances latency and accuracy. They show that introducing a brief, strategically delayed second-stage detection phase improves rejection of false triggers by up to 66%, with negligible impact on response time. This finding underscores the importance of timing calibration in preventing premature cutoffs or delayed turn initiation—both of which can manifest as deletion errors. Goetsu and Sakai (2020) further clarifies the emotional impact of such errors. They classify VUI failures

and find that utterance match failures—a type of deletion where the system does not recognise valid user input—cause greater user frustration than typical misrecognition. Unlike substitution errors, deletion errors give users no visible signal of failure, leaving them unsure if their input was heard or acted upon.

Trust is also adversely affected by deletion-style errors. Baughan et al. (2023), in their analysis of real-world voice assistant failures, show that input omissions and over-captures lead to a significant decline in user trust. After such failures, users tend to avoid relying on the assistant for important tasks, especially in time-sensitive contexts, reinforcing the real-world implications of deletion errors beyond momentary usability issues.

These studies highlight that achieving temporal alignment between user input and system response and designing systems that can robustly detect and recover from deletion errors is crucial for effective, trustworthy voice interaction. For half-duplex or constrained environments, where timing errors are magnified, these challenges become even more pronounced, necessitating careful design of turn-taking cues and error feedback mechanisms.

5.3.2 Role of non-speech sound in human-computer interaction

Non-speech audio cues—such as beeps, tones, earcons, auditory icons, and artificial vocalisations—play a vital role in Human-Computer Interaction (HCI) by supporting communication, guiding user attention, expressing affect, and enhancing usability, particularly in constrained settings. A recent study by Nees and Liebman (2023) conducted systematic review emphasing how brief audio cues enhancing alerting performance in human-computer interface, specially when visual resources are constrained. Zhang and Fitter (2023) shows that these cues are widely used to indicate system states (e.g., readiness, errors), regulate turn-taking, and facilitate user understanding in dialogue-based systems. In Human-Robot Interaction (HRI), non-speech sounds further serve emotional and social functions by expressing robot intent or mood, fostering trust and engagement, especially in contexts like child interaction or assistive environments Read and Belpaeme (2012). Some studies further demonstrate, through real-world evidence, that non-speech auditory cues can improve users' comprehension and foster trust in automated systems—even in the absence of direct interaction. Ko et al. (2022) investigated the

effectiveness of non-speech auditory cues in supporting control transitions in semi-automated vehicles. Through a driving simulator study, they found that high-frequency, repetitive tones (e.g., Hyundai's alert) resulted in the fastest driver response times during takeover scenarios, while also being rated as urgent and attention-capturing, though somewhat annoying. In contrast, soundscapes were perceived as pleasant but failed to convey urgency. Using the Queuing Network-Model Human Processor (QN-MHP), they modelled driver reaction times based on acoustic features such as pitch, repetition, and frequency, achieving 99.7%

5.4 Methods

The primary objective of this experiment was to examine the effect of a non-speech auditory cue, referred to as the "spoke-too-soon" (STS) tone, which was designed to sonify premature user speech during turn-taking. In simple words, we aim to investigate if a non-speech repetitive dissonant sound, called the "spoke-too-soon" (STS) tone, helps users during turn-taking by alerting them when they start speaking too early. The study employed a comparative design involving two groups: an experimental group and a control group. Participants in the experimental group interacted with a Voice User Interface (VUI) equipped with the STS tone, which played the tone whenever the user initiated speech earlier than the predefined threshold of 175 milliseconds. This tone served as immediate feedback, triggered by the user's premature response, marking a violation of the expected turn-taking interval. The 175 ms threshold corresponds to the ASR system's silence timer setting (see Table 4.1 in Chapter 4), representing the minimal duration required for the recognizer to switch from playback to input mode. The turn-taking mechanism was calibrated so that any speech initiated before this interval would not be processed by the system. The 175 ms duration was selected through a combination of empirical testing and insights from conversational theory, with pilot trials confirming that the ASR system needed approximately this much time to achieve reliable speech detection post-playback. In contrast, the control group interacted with a version of the VUI where the STS tone was deactivated—thus, participants received no feedback even if they began speaking prematurely.

5.4.1 Participants

A total of 34 participants were recruited and assigned to one of two experimental conditions: STS-present (n1 = 18) and STS-absent (n2 = 16). All participants were fluent English speakers, aged between 18 and 35 years, and reported normal hearing. Participants were either university students or working professionals, recruited through convenience sampling. None reported prior familiarity with the specific experimental system. Participants gave informed consent before the study began.

5.4.2 Experiment design

The study followed a between-subjects design with two groups, a control group and an experimental group:

- The experiment group (STS-present group), where participants received immediate feedback via a dissonant non-speech tone (Spoke-Too-Soon or STS tone) if they attempted to speak before the system was ready to listen.
- The control group (STS-absent group), where participants do not receive any feedback even if they speak before the system is ready to listen. Other protocols remain the same as the STS present group.

Participants were instructed to read aloud from a printed prompt script consisting of 12 number-string sequences, with each string composed of digits ranging from 0 to 9 and printed as English words (please refer to Appendix A). This task was performed across three separate passes by each participant, ensuring consistent repetition for both the control and experimental groups.

5.4.2.1 The task

The task required participants in both groups to read aloud a scripted sequence of numbers as instructed by the VUI (see Appendix A). The script contained 12 strings of digits from 0 to 9,

written in English and separated by hyphens. All participants completed the task in a controlled environment with consistent instructions.

User: “Six-Two-Eight-Three”
ASR: Two-Eight-Three

Figure 5.1: Dialogue 1: Deletion Error Caused by Speaking Too Soon

This experimental setup was intentionally chosen to simulate a prompt-response conversational context that involves rapid and frequent turn-taking. Each number in the string is typically one or two syllables and, when spoken in sequence, the words are co-articulated, producing a fast, fluent utterance with minimal pauses. This structure closely mimics high-tempo interactions where recognition timing is critical.

The key challenge in such scenarios is the risk of deletion errors at the very beginning of the user’s speech. If the recognizer’s listening window is not yet open when the user begins speaking, those initial digits are missed — and because sequences like “three-eight-nine” lack syntactic redundancy, the system cannot infer missing parts using language models or grammar. In many commercial VUIs, this issue is mitigated by encouraging users to place crucial information toward the end of the utterance — for example:

Voice Interaction
User: “Alexa, please add a timer for 3 seconds.”

Figure 5.2: A sample dialogue illustrating a user-issued timer command to a voice assistant

Here in the dialogue (please see figure 5.2), the important details ("timer" and "3 seconds") appear late, giving the system time to synchronize. However, this workaround does not suit applications that require precise, early-slot recognition — such as form-filling interfaces, telephone menu systems, or voice-controlled transactional systems. These domains often rely on quick, structured exchanges where the first few words are critical. By using a number-string task, our study deliberately forces early turn-taking errors in order to evaluate whether the STS tone improves users’ ability to time their input more accurately. This task thus serves as a controlled stress test for system readiness and user entrainment, allowing us to analyze

how non-speech repetitive dissonant tone used as feedback impacts timing and deletion error reduction.

For the current experiment, how a deletion error affects it is shown in the figure 5.1. The example in the figure shows that the word “Six” was spoken by the user at the very end of the system prompt, overlapping with the PromptTone. Due to its short duration and low acoustic energy—particularly the /s/ consonant, which VAD systems often misclassify as silence—the word was not captured by the speech recognizer. If such deletion errors go unnoticed, they can affect the continuity and accuracy of the interaction, highlighting the importance of well-timed turn-taking cues like the STS tone.

5.4.2.2 The experimental setup

The study occurred in a soundproof, well-lit room equipped with a chair and a table (please see figure 5.3). An iPhone, supported by a gorilla tripod, was positioned on the table. Participants sat in the chair to begin the experiment. Behind a glass pane, an observer outside the room monitored the participant. If the participant encountered any difficulties, the observer would step in to provide assistance with the task.

5.4.2.3 The application prototypes

The prototype used in this study was a Voice User Interface (VUI) application installed on an iPhone (see Figure 5.4). Participants interacted with the application by pressing a prominently displayed PLAY button on the screen.

It is important to note that this prototype is the same as the one developed and used in the previous study (Chapter 4), where the focus was on evaluating the overall effectiveness of a structured turn-taking protocol in supporting human-VUI interaction. In the current experiment, the prototype was adapted to include two versions (please refer figure 5.5)- (1) one with the dissonant, repetitive “Spoke-Too-Soon” (STS) tone enabled, and (2) another with the STS tone disabled. Apart from the presence or absence of the STS tone, all other interface features and functionalities remained identical across both versions.

This study specifically investigates the effect of non-speech auditory feedback—namely, the STS tone—on user timing awareness during interaction. By reusing the same application

Figure 5.3: Experimental setup

prototype, the study ensures consistency in interface design while isolating the auditory feedback cue as the variable of interest.

The VUI incorporates a custom-designed turn-taking protocol (see Algorithm 1), with built-in controls that allow the experimenter to toggle relevant parameters. Each interaction cycle begins with a spoken prompt instructing the user to say a number, followed by a PromptTone indicating that the system is ready to listen. After each user utterance, an AcknowledgementTone confirms receipt. The cycle repeats until the participant completes all twelve number strings from the prompt script, at which point the session ends automatically.

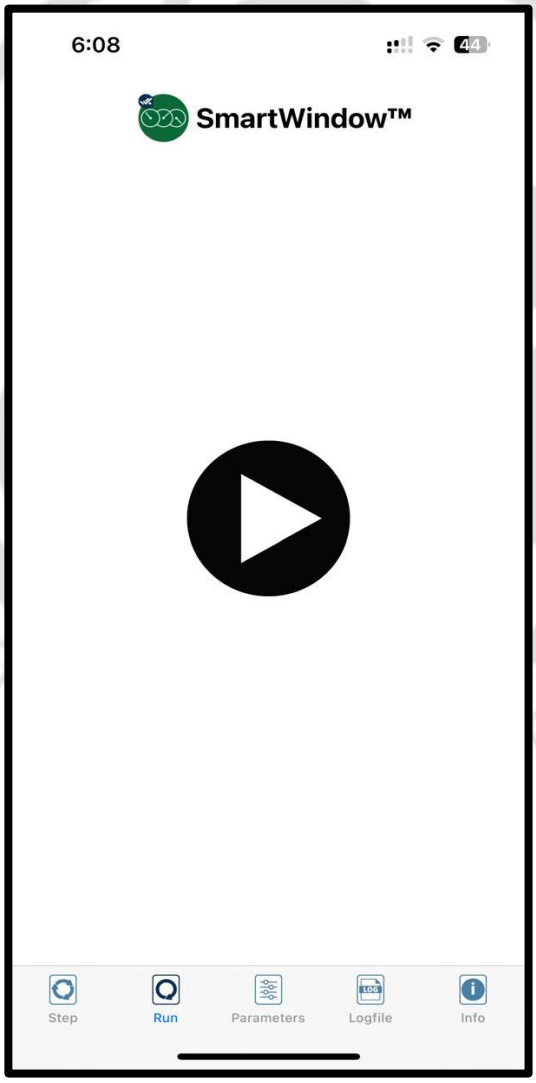


Figure 5.4: The application interface provided to the participants

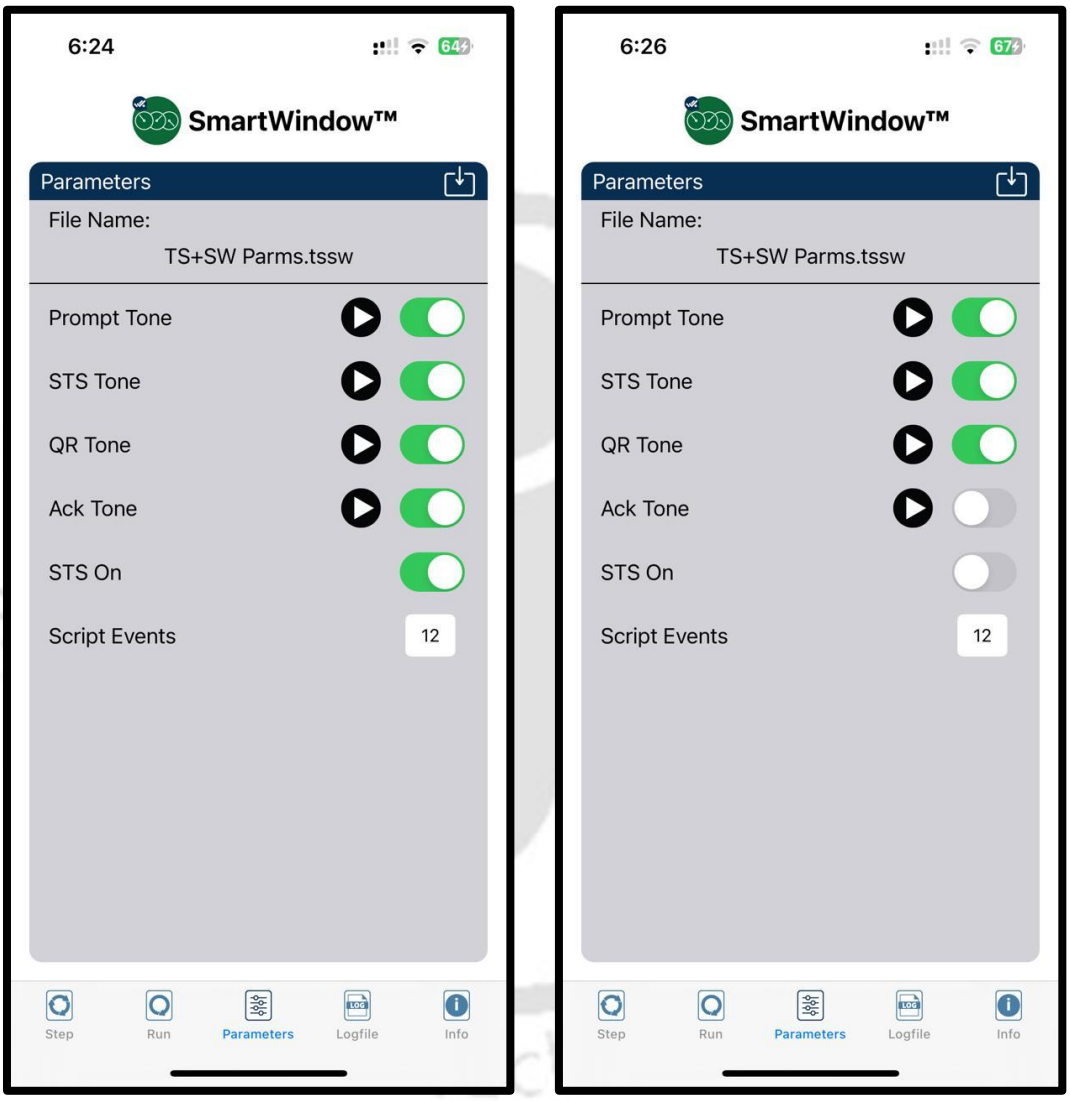


Figure 5.5: Interfaces used in the study: left—experimental group (STS-present); right—control group (STS-absent)

Figure 5.6: A participant performing experiment inside the lab

App: Thanks for doing this experiment. Please say the first number <prompt tone>

User: "Four-Three-Six-Zero-Three"

App: <ack tone> Say the next number <prompt tone>

User: "Eight-Two-Two"

. . .

App: <ack tone> Next number? <prompt tone>

User: "Zero-Four-Seven"

App: <ack tone> Next number? <prompt tone>

User: "Six-One"

Figure 5.7: Dialogue 2: Typical Pass Showing Variable-Length Digits & Tapering

5.4.2.4 The procedure for the experiment

The Participant would arrive at the pre-decided time and sit in the lab's common sitting area, where she is handed the printed details about the experiment description to read and the consent form to fill up. The experimenter would then assign the participant to either experiment/control group based on the random selection. A random number generation algorithm generates the random numbers to select the participants to be included in either the control or the experimental group. The experimenter takes the participant into the experiment room and seats her at the desk (please see figure 5.6. Then he describes the setup and the task, indicating availability to help in case of any problems or questions. After this short explanation, the experimenter leaves the room, observing the experiment through the glass pane of the recording room. The participant starts the application by touching a big PLAY button displayed on the screen. After starting, there is no touching or viewing of the display—the task is voice-only. The participant hears each spoken prompt and replies by speaking each number. The application responds with an acknowledgement tone after each turn. The process continues until the participant has uttered all twelve numbers in the script. The application automatically stops once the complete script is finished. See 5.7 for an example of how the sequence sounds. After each 12-turn pass, the experimenter re-enters the room to perform a brief interview. During the interview, an assistant stops and saves the video recording.

5.4.2.5 Repeated-measures design

Participants of both groups were asked to repeat the task of reading out the number script three times in 3 different passes (pass1 , pass 2, pass3) during the experiment. Reading out the number script for a single time is considered a single pass. It is to be noted that the numbers in the script were randomised during each pass.

The use of a repeated-measures design in this experiment was a deliberate methodological choice aimed at capturing the temporal dynamics of user behavior in response to the presence or absence of the STS tone. By requiring each participant to complete the task across three consecutive passes, we created a structure that allowed us to investigate not only moment-to-moment interactional outcomes but also whether and how participants' turn-taking behavior evolved over time. This design enabled within-subject comparisons, which are particularly valuable in

studies involving human-computer interaction, where individual differences in cognitive processing, familiarity with voice interfaces, and turn-taking strategies can introduce substantial variability. A single-pass interaction might reflect novelty effects, initial confusion, or random variance, whereas repeated measures increase the reliability and sensitivity of the analysis by observing trends and behavioral patterns across multiple trials. Additionally, by analyzing performance over time, we could examine whether exposure to the STS tone led to learning or adaptation effects in the experimental group, and whether the absence of such feedback in the control group resulted in stagnation or increased frustration. The repeated-measures approach thus allowed us to capture not just static performance differences between groups but also dynamic, time-sensitive changes in user response, which are critical to understanding the role of non-speech auditory cues in facilitating effective turn-taking in Voice User Interfaces. This temporal dimension adds depth to the comparative analysis and strengthens the ecological validity of our findings, as real-world VUI usage often involves repeated interactions rather than isolated tasks.

5.5 Data collection and data analysis

The data collection process collects mainly two types of data.

5.5.1 Video recording

Video and audio data were recorded for all participants across both experimental conditions (STS-present and STS-absent groups). The video recordings captured how participants engaged in turn-taking during the interaction, including their facial expressions and body language in response to the system's voice prompts and auditory cues. The audio recordings captured participants' spoken responses as they read aloud the scripted number sequences, along with any variations in vocal timing relative to the machine's turn-taking signals, such as the PromptTone and, where applicable, the STS tone. The recordings were obtained using a camera positioned in front of the participant, with an integrated microphone used to capture synchronised audio. Additionally, the post-experiment interviews, in which participants responded to a structured

questionnaire (please refer to questionnaire Appendix D), were also video recorded to support further qualitative analysis.

5.5.2 Machine's log file

For each participant, the experimenter saved one log file per pass, resulting in three log files per participant. These logs were used in conjunction with the video recordings to analyze two key aspects: (1) the occurrence of spoke-too-soon (STS) events, and (2) the presence of deletion errors at the beginning of the recognized number string. Apart from these deletion events, overall digit recognition accuracy was not the primary focus. The association between premature speech and deletion errors is already established (see Figure 5.1).

While the structure of the log files remained the same across both experiments, a key distinction in this study is that the log files were intentionally designed to register STS events for both experimental and control groups—regardless of whether the STS tone was actually played. In the control group, although the tone was disabled and not audible to participants, STS error events were still recorded based on the system's detection of early speech onset. This allowed for a consistent and comparable analysis of STS occurrences across conditions.

Additionally, the log files contained timestamped data, enabling further analysis of temporal features such as rhythm and tapering in user speech behaviour during the interaction process, allowing for detailed analysis and interpretation.

5.6 Results

The experiment collected a large amount of data. This includes videos, log files and interview responses. We divided the analysis into statistical analysis of the STS errors committed by the groups and the subjective user evaluation evaluated from the post-experiment interview.

5.6.1 Statistical analysis

This section presents the statistical analysis conducted to examine whether the presence of a non-speech auditory feedback tone (referred to as the Spoke-Too-Soon or STS tone) significantly influenced the number of STS errors made by users during task-based voice interaction. The dataset (please see Appendix G and H) included two groups: a control group without the STS tone and an experimental group where the tone was present. The sample size for the experiment group was 18*3=54 turns, and for the control group was 16*3=48 turns. Each participant completed three task passes, and the number of STS errors was logged for each. We began by conducting descriptive analyses to summarize central tendencies and distributions. Normality tests were performed to assess whether parametric tests were appropriate. As the assumption of normality was violated for both groups, a non-parametric Mann–Whitney U test was used to evaluate group differences in STS error frequency.

5.6.1.1 Descriptive Statistics

Descriptive analysis revealed a clear difference in the frequency of Spoke-Too-Soon (STS) errors between the control and experimental groups. The samples in the control group (N = 16*3=48) exhibited a higher mean number of STS errors ($M = 9.71$, $SD = 8.75$), whereas the samples in the experimental group (N = 18*3=54), which received non-speech auditory feedback in the form of the STS tone, showed a substantially lower mean error count ($M = 2.54$, $SD = 2.20$). The median values also differed, with the control group showing a median of 7.5 errors compared to a median of 2.0 in the experimental group.

5.6.1.2 Normality Assessment:

Tests of normality, including both the Kolmogorov–Smirnov and Shapiro–Wilk tests, indicated that the data for both groups violated the assumption of normality. For the control group, Shapiro–Wilk's $W = 0.893$, $p < .001$; for the experimental group, $W = 0.898$, $p < .001$. Consequently, a non-parametric test was deemed appropriate for comparing the groups.

Table 5.1: Descriptive statistics for STS errors in control and experimental groups. The control group shows higher mean, variance, and range of errors, whereas the experimental group demonstrates more consistent and reduced error behavior.

Statistic	**Control Group**	**Experimental Group**
Mean	9.71	2.54
95% CI for Mean	[7.17, 12.25]	[1.94, 3.14]
5% Trimmed Mean	9.25	2.37
Median	7.50	2.00
Variance	76.55	4.82
Standard Deviation	8.75	2.20
Range	28	8
Interquartile Range	15	3
Skewness	0.66	0.85
Kurtosis	-0.81	0.26

5.6.1.3 Inferential Statistics

A Mann–Whitney U test was conducted to compare the number of STS errors between the control and experimental groups (please refer to table 5.2). The result revealed a statistically significant difference in error counts between the two groups, $U = 664.00$, $Z = -4.264$, $p < .001$ (two-tailed). The mean rank for the control group was 64.67, while the experimental group had a lower mean rank of 39.80, indicating fewer errors when the STS tone was used.

Table 5.2: Mann–Whitney U test comparing STS errors between control and experimental groups.

Group	**N**	**Mean Rank**
Control	48	64.67
Experimental	54	39.80
Test Statistic	**Value**	
Mann–Whitney U	664.000	
Z	-4.264	
Asymp. Sig. (2-tailed)	<.001	

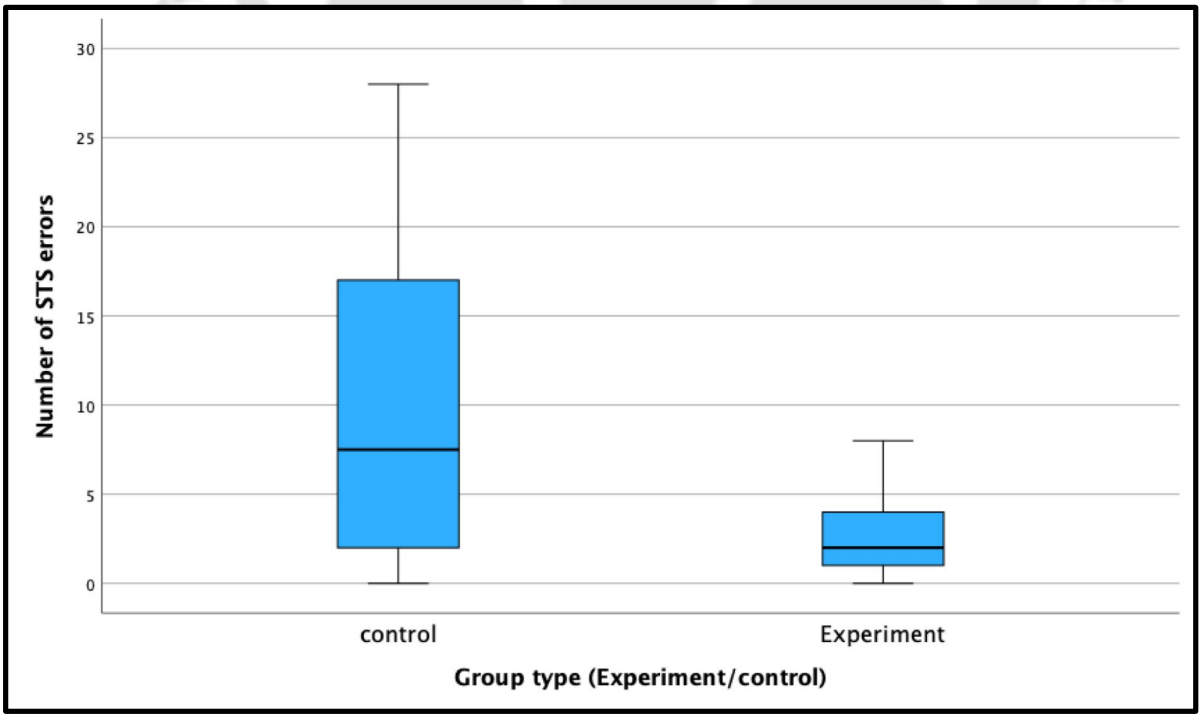


Figure 5.8: Distribution of Spoke-Too-Soon (STS) errors across control and experimental groups.

5.6.2 Subjective User Evaluation

The post-experiment interviews revealed a striking contrast between the STS-present and STS-absent groups in terms of cue awareness and learning behavior. In the control group (STS-absent), the majority of participants reported that they relied solely on the spoken voice prompt to determine when to speak. Notably, only two participants reported noticing the PromptTone or feedback tone, and even they stated that they became aware of these cues only during the second or third pass of the experiment. This suggests that, in the absence of an explicit corrective signal like the STS tone, more subtle auditory cues failed to capture user attention or guide behavior effectively.

In contrast, participants in the STS-present group not only consistently recognized the STS tone but also frequently reported that the PromptTone became more salient as the interaction progressed. Many described the STS tone as a dissonant alert that helped them infer timing violations and, in turn, become more attuned to the system's overall feedback structure. Interestingly, several participants credited the PromptTone for their adaptation—yet this awareness appears to have been triggered or reinforced by the presence of the STS tone itself. These observations suggest that the STS tone served a dual function: not only as direct feedback for mistimed responses but also as a meta-attentional cue, sharpening users' sensitivity to the system's broader turn-taking protocol. In short, the STS tone amplified user awareness, allowing other cues—like the PromptTone and feedback tone—to be perceived and internalized more effectively, ultimately supporting more reliable entrainment.

5.7 Discussion

This study builds directly on our prior experiment (chapter 4) by employing the same structured turn-taking protocol that was previously shown to support user entrainment in voice-based interactions. In this extension, we introduced an experimental comparison across two groups: one in which the dissonant Spoke-Too-Soon (STS) tone was present (STS-present condition), and one in which the STS tone was removed (STS-absent condition). Crucially, all other elements of the turn-taking protocol remained identical across both groups, including the voice prompt,

PromptTone, start tone, task type, and half-duplex constraint. This controlled design allowed us to isolate and evaluate the specific contribution of the STS tone to user adaptation and temporal alignment.

The goal was to examine how the presence or absence of this corrective auditory cue influenced the emergence of entrainment over repeated prompt-response exchanges. By comparing error trends, user timing behavior, and post-task feedback, we were able to assess not only how users aligned with the system's temporal expectations but also whether the absence of explicit feedback inhibited or delayed learning. The following sections synthesize behavioral and perceptual findings from both conditions to evaluate the role of corrective feedback in guiding turn-taking behavior.

The statistical analysis showed that the range was broader in the control group (0–28) than in the experimental group (0–8) (Please refer to table 5.1), suggesting greater variability in turn-taking errors when auditory feedback was absent. The figure 5.8 also suggests that participants in the control group not only made more errors on average but also showed greater variability in performance. Such high dispersion may reflect individual differences in users' ability to grab user attention for turn-taking without dissonant STS tone, pointing to a lack of consistent timing cues. In contrast, the lower standard deviations in the experimental group indicate that the STS tone may have provided a sense of temporal aspects of turn-taking, helping users align their interaction patterns more predictably and uniformly. Overall, these results imply that the presence of a non-speech auditory cue like the STS tone not only reduces error frequency but also improves the consistency and reliability of user behavior during voice interactions.

The non-normal distribution of STS errors, observed across both experimental (STS-present) and control (STS-absent) groups, can be attributed to enduring user-behavior patterns previously described in speech technology research through the metaphor of "sheep" and "goats" Doddington et al. (1998). This classification captures a well-documented bifurcation in ASR user populations, where "sheep" adapt readily to system cues and timing expectations, while "goats" persist in maladaptive behavior, often resisting or misinterpreting feedback.

In the STS-present group, sheep rapidly entrained to the feedback tone and reduced errors over time, while goats continued to step on the invisible seam despite explicit cues—resulting in a skewed error distribution. In the STS-absent group, a similar pattern emerged: in the absence of feedback, sheep inferred system timing and adjusted behavior effectively, whereas goats

struggled to form an accurate mental model, producing higher and more consistent error rates. This behaviorally driven variance accounts for the non-normality in both groups' STS data and highlights that such distributional patterns are not merely statistical artefacts but reflections of deeper user cognition and adaptation processes.

In the STS-present condition, participants displayed both natural and unnatural entrainment over time. Natural entrainment was seen in the form of progressive synchronisation between user speech timing and the VUI's listening state. Users gradually adapted to the timing of system and began responding at appropriate intervals without direct instruction. Additionally, unnatural entrainment was evident among those who initially committed timing errors but later corrected their behavior based on feedback from the STS tone. Notably, users with high STS errors in Pass 1 showed significant improvement by Pass 3 (see Appendix), suggesting that they formed a better mental model of the system's timing. While multiple turn-taking cues were available—including Prompt tone, feedback tone, and voice prompts—these were consistent across both groups. However, only participants in the STS-present condition exhibited marked reductions in error and began referencing the Prompt tone in post-experiment interviews. This contrast implies that the STS tone may have served as an attentional primer, enhancing users' sensitivity to other auditory cues and helping them internalize the system's temporal structure.

In contrast, participants in the STS-absent condition struggled to achieve the same level of alignment. Despite completing an identical prompt-response task structure, they exhibited higher and more persistent STS errors throughout all three passes. Unlike the STS-present group, these users received no dissonant signal when speaking too early, depriving them of an immediate behavioral correction mechanism.

The video analysis revealed a clear behavioral divergence between the STS-present and STS-absent groups, highlighting the role of the Spoke-Too-Soon (STS) tone in shaping user turn-taking behavior. In the STS-present condition, sixteen of the eighteen participants exhibited immediate and noticeable reactions—such as gaze shifts, widened eyes, or expressions of confusion—upon hearing the STS tone, with the remaining two acknowledging the tone during post-task interviews. Initially, participants in this group tended to speak immediately after the voice prompt, often overlooking the subsequent PromptTone. However, over successive sequences, many began to delay their responses until after the PromptTone, gradually aligning their speech with the system's intended rhythm.

This behavioral adjustment was further supported by post-experiment interviews. Most participants in the STS-present group reported that during the first one or two sequences of Pass 1, they would begin speaking as soon as the spoken prompt ended. However, after encountering repeated feedback in the initial sequences, they began to notice the presence of the PromptTone and started waiting for it before initiating speech. Interestingly, participants did not explicitly mention noticing the STS tone itself, suggesting that while the tone did not register consciously, its disruptive presence may have played a pivotal role in drawing attention to the timing structure of the interaction. In this way, the STS tone likely functioned as an implicit corrective signal—strong enough to interrupt habitual responses and orient users toward other cues like the PromptTone, thereby facilitating a shift toward the desired turn-taking behavior.

In contrast, participants in the STS-absent group exhibited little such adaptation. Most continued to initiate speech immediately after the spoken prompt throughout the interaction, with minimal indication of learning the required delay. Although they often redirected their attention to the screen upon hearing the PromptTone, this did not consistently translate into appropriate timing adjustments. Interview data confirmed this pattern: nearly all participants in the control group stated that their speech timing was primarily guided by the StartPrompt, with only two indicating that they had become aware of and responded to the PromptTone.

Together, these findings underscore the importance of explicit and strategically designed auditory feedback in shaping user behavior. The STS tone was not hidden; rather, it was triggered whenever a participant began speaking within the first 175 ms of the system's turn—an interval so brief that first-time users could not consciously interpret its meaning. Despite this, the tone effectively interrupted premature responses and redirected users' attention toward the system's timing. With repeated exposure across passes, participants increasingly recognised the STS alert and began adjusting their speech patterns to avoid triggering it. This contrast highlights how mechanomorphic interaction strategies, even when initially difficult to consciously perceive, can still foster more accurate and entrained turn-taking behavior in voice user interfaces.

Interestingly, post-interview data revealed that while Prompt tone was present in both conditions, participants in the STS-absent group did not report noticing it. Instead, they cited the spoken voice prompt as their primary cue for initiating speech. Conversely, participants in the STS-present group frequently mentioned both the STS tone and PromptTone, with many

stating that the STS tone helped them notice other cues more attentively. This suggests that the STS tone acted as a salience enhancer, not only correcting missteps but also drawing user attention to subtle, system-generated indicators like the PromptTone.

Crucially, none of the participants in the STS-present group described the STS tone as interruptive during interviews. In fact, most found it helpful for guiding behavior, and only two individuals noted interruption—but it was due to the system's voice prompt, not the tone itself. This observation supports earlier findings from Lei et al. (2022) who highlighted that dissonant, perceptually salient audio cues can be effective attention-capturing mechanisms without being disruptive.

Moreover, participant descriptions—such as "beep," "tin-tin sound," "chime," or "tuntun sound"—demonstrate that the tone was clearly recognized and associated with actionable feedback. Its consistent perception and positive reception indicate that it was well-integrated into users' mental models, reinforcing its value in VUI interaction design.

The STS tone served not only as a dissonant feedback signal for premature speech but also as a catalyst for broader cue awareness. Once users noticed the STS tone, they became more sensitive to other auditory cues, such as the Prompt tone and brief pauses in system speech. The STS tone provided a machine-like, transparent signal that made the turn-taking boundary both perceivable and learnable. This dual role—as both error signal and attention trainer—makes the STS tone uniquely valuable in half-duplex environments, where users overlape conversational cues. While subtle timing cues may suffice in human-human interaction, our findings suggest that such cues alone are often inadequate for users interacting with constrained voice interfaces, at least for prompt-response-type interactive conversations. Without a dissonant signal like the STS tone, users may fail to perceive system readiness or misjudge the appropriate moment to speak—leading to premature input and task disruption. In contrast, the STS tone offered a direct and immediate signal of misalignment, enabling users to calibrate their timing with the system's expected turn structure. Over time, this resulted in a more consistent and accurate interaction rhythm, with participants entraining to the system's temporal pacing more effectively than in the absence of such feedback.

The findings resonate with prior literature on the importance of clear turn-taking cues in spoken systems. Skantze (2021) and Ekstedt and Skantze (2022) emphasized that conversational systems must transparently signal when it is the user's turn to speak. Similarly, Borrie

et al. (2015) and Levitan et al. (2015)) linked prosodic and lexical entrainment with increased communicative success. Our study extends this body of work by demonstrating that temporal entrainment—driven by explicit feedback—can lead to behavioral alignment and task success, even in simple, structured tasks like form-filling.

These insights reinforce the critical role of dissonant auditory feedback in VUI design. Designers often hesitate to include interruptive or non-speech tones, fearing they may break natural flow. However, our findings suggest that when deployed judiciously, such tones can improve user performance, reduce errors, and enhance alignment, especially in time-constrained or feedback-scarce settings. This is particularly relevant for voice-only applications like driving, assistive technologies, or home automation, where visual cues are absent, and timing errors have greater cost.

Moreover, the distinction between STS-present and STS-absent users' cue awareness indicates that subtle cues like Prompt tones may require priming through feedback mechanisms. Without reinforcement from a dissonant signal, such cues risk going unnoticed—resulting in user confusion or poor timing performance. Incorporating structured, scaffolded learning via tones like STS may be especially valuable in onboarding new users or designing systems for high-efficiency applications.

Taken together, these findings also foreground a broader design tension that runs through this thesis- whether we should train users to adapt to machine-like behaviour, or continue to make constrained systems appear more human-like. The half-duplex protocol and tone-based feedback used in this study explicitly ask users to align their speech timing with the temporal pattern of the machine, rather than inviting them to behave "as if" they were talking to a human interlocutor. By contrast, most commercial voice interfaces lean heavily on anthropomorphic presentation—natural voices, small talk, and social cues—even when their underlying capabilities cannot sustain genuinely human-like conversation. This strategy can make interactions feel more familiar in the short term, but it also risks widening the gap between what the system seems to be and what it can actually do. The mechanomorphic stance adopted here takes the opposite view- that making the system's constraints and states more visible offers a more honest and ultimately more sustainable basis for user adaptation. I return to this tension in Chapter 6, where I develop the mechanomorphic versus anthropomorphic distinction in greater detail and consider its implications for the future design of VUIs.

5.8 Conclusion

This chapter presented a controlled comparative study investigating the role of a dissonant, non-speech auditory cue—termed the Spoke-Too-Soon (STS) tone—as corrective feedback within a structured turn-taking protocol designed for half-duplex Voice User Interfaces (VUIs). Building on earlier findings that established the efficacy of the base protocol in promoting user entrainment, this experiment specifically isolated the impact of the STS tone in enhancing system-user coordination.

Participants were divided into two groups—one exposed to the full protocol including the STS cue, and the other experiencing the same interaction structure without it. Both groups completed identical tasks, allowing for a direct comparison of their behavior and performance. Quantitative analysis revealed that the presence of the STS cue significantly reduced timing errors and facilitated faster and more consistent adaptation to the system's prompt-response rhythm. Qualitative reflections reinforced these findings, with participants in the STS-present group reporting a clearer understanding of the interaction structure and greater ease in managing turn-taking.

These results highlight the STS tone's dual role as both an error signal and a perceptual anchor. While repetition and structural regularity in the interaction protocol did contribute to gradual adaptation, the STS tone offered salient, immediate corrective feedback that enhanced users' ability to align their responses with the system's temporal expectations. In its absence, participants often struggled to recognize timing mismatches or adjust their behavior accordingly, leading to more frequent errors and weaker entrainment.

Together, these findings suggest that robust conversational alignment in voice-based systems—particularly in half-duplex, prompt-response scenarios—depends not only on structured interaction patterns but also on the inclusion of salient non-speech feedback. The strategic use of dissonant auditory cues like the STS tone offers a compelling design solution, transforming abstract system rules into tangible, perceivable signals that support user learning and interaction fluency.

These insights contribute to the broader design implications of mechanomorphic VUI systems by emphasizing that machine-like transparency and explicit feedback—not anthropomor-

phic mimicry—can effectively support user adaptation. As such, this study sets the stage for further exploration of how auditory cues can be leveraged to scaffold turn-taking behavior, particularly in contexts where timing precision and conversational reliability are critical.

Chapter 6

Discussion

6.1 Overview

This chapter discusses the key findings of the thesis in relation to existing literature and the identified research gaps mentioned in the literature review chapter. It begins by addressing the underexplored issue of turn-taking in half-duplex Voice User Interfaces (VUIs), especially the lack of robust design strategies to manage conversation flow in such constrained environments. Building on this, the discussion highlights how temporal entrainment where users align their turn-taking behavior with system cues can be deliberately integrated into VUI design to support smoother conversational exchanges. The chapter then examines the role of non-speech audio cues, such as the Spoke-Too-Soon (STS) tone, in guiding user timing and preventing interruptions. Finally, it contrasts mechanomorphic and anthropomorphic design philosophies, emphasizing how a mechanomorphic approach—rooted in transparent system signaling—can enhance user understanding and improve conversation outcomes in half-duplex systems.

6.2 Achieving quick task success in conversation using half-duplex turn-taking protocol

Despite the promise of full-duplex conversation, current systems built on audio foundation models like Moshi, Whisper+GPT-4o, Qwen-Audio-Chat, and SALMONN still struggle in real-time interaction. These models have not been rigorously tested for live turn-taking behavior, and often fail to manage interruptions, backchannels, and floor transitions naturally. As highlighted by Arora et al. (2025), researchers are only beginning to develop systematic frameworks to evaluate and improve these capabilities. Moreover, achieving low-latency interaction in full-duplex dialogue systems remains a significant challenge Zhang et al. (2024), particularly in time-sensitive, task-oriented, or high-load environments. Full-duplex systems theoretically allow for fluid, overlapping speech, but in practice, they often struggle with latency, barge-in misfires, and confusion around when the system is listening or speaking—issues that can disrupt task completion Veluri et al. (2024).

In contrast, our findings in Chapter 4 which examine effect of a half-duplex turn-taking protocol in human turn-taking behaviour, demonstrate that a structured half-duplex turn-taking protocol, with clearly defined speaking and listening states, enables users to complete interactive prompt-response conversations more efficiently. This protocol is particularly effective in contexts where the interaction goal is narrowly scoped—such as voice-assisted form filling, smart appliance control, or number-based data entry—where conversational realism is less critical than temporal clarity.

Previous research has shown that users often adapt to technological constraints when those constraints are predictable and consistent. For example, Woodruff and Aoki (2003) observed that users of push-to-talk systems quickly learn to manage turn exchanges despite the lack of simultaneous speech. Our results build on these findings by demonstrating that users not only tolerate half-duplex turn-taking in well-designed systems but also perform more reliably and with fewer errors when the system clearly regulates turn flow.

Post-experiment interviews further in our study reported that the predictability of the system's timing cues made the participants feel more confident about when to speak. They expressed a preference for this structured coordination over ambiguous "natural" timing systems,

which often led to overlaps or premature speech in real-world VUI use. This highlights a critical design insight—half-duplex systems, when thoughtfully implemented, are not merely fallback options but can serve as robust solutions for achieving quick, reliable task success.

Foundational theories of turn-taking Sacks et al. (1974) emphasize rapid, overlapping transitions as a feature of natural conversation. Yet, in voice-based interactions—especially those driven by tasks—the priority shifts from natural to functional clarity. In such contexts, the absence of overlap is not a limitation but a design affordance, offering users a clean and dependable conversational rhythm. Our study repositions half-duplex protocols not as outdated or inferior but as purpose-fit strategies for constrained, efficiency-driven voice interaction

6.3 Integrating temporal entrainment principles in VUI design

A key insight emerging from our studies is the presence of temporal entrainment, where users adjusted their speaking behavior in alignment with the system's turn-taking rhythm. Unlike human-human interaction—where entrainment is often spontaneous and subconscious—we observed both natural (implicit) and strategic (conscious) entrainment in our participants. Some users adapted organically over repeated interactions, while others reported making deliberate efforts to wait for cues or avoid speaking too soon, especially after encountering errors. This duality reflects a more complex form of human-machine coordination than typically assumed.

Entrainment has been well-documented in human-human conversation across various levels: acoustic-prosodic entrainment (e.g., speech rate, pause duration, pitch), lexical entrainment (e.g., repeating words or phrases), and syntactic entrainment (e.g., mirroring sentence structure) Natale (1975); Heldner et al. (2010); Levitan and Hirschberg (2011); Brennan and Clark (1996); Pickering and Garrod (2004). These forms help establish conversational alignment and reduce friction in turn exchanges. However, while these types of entrainment are widely studied in interpersonal dialogue, the concept remains underexplored in human-machine interactions, particularly in terms of temporal alignment during turn-taking.

Our study addresses this gap by demonstrating that users engaging with half-duplex Voice

User Interfaces (VUIs) exhibit clear temporal entrainment, guided by the system's consistent timing cues. This represents a novel contribution to the field of human-machine conversation. Unlike full-duplex systems that support overlapping speech, half-duplex systems benefit from rigid boundaries—making them ideal testbeds for studying entrainment as a design mechanism.

Moreover, we found that entrainment was learned and reinforced over time. Participants became more aligned with the system's rhythm, with a notable decline in premature speech and turn-taking hesitation across repeated measures. This behavioral adaptation—whether conscious or unconscious—demonstrates users' capacity to internalize structured interaction patterns, particularly when system signals are predictable.

These findings align with design arguments made by Kirschthaler et al. (2020), who advocate for transparent, discoverable VUI systems that allow users to model system behavior. In our case, entrainment was not a side effect but a deliberate outcome of system design—supported by consistent timing, silence intervals, and auditory cues.

In summary, our work (experiments in Chapter 4 and chapter 5) provides one of the first focused accounts of temporal entrainment in human-machine dialogue, revealing it as both a natural and trainable behavior. By designing for entrainment—not merely observing it—voice systems can facilitate smoother, more efficient interactions, particularly in constrained, half-duplex contexts.

6.4 Exploring non-speech audio cues for turn-taking

In addition to structured timing and entrainment, our research highlights the effectiveness of non-speech audio cues—specifically the Spoke-Too-Soon (STS) tone—in shaping user turn-taking behavior. Unlike verbal prompts or visual indicators, non-speech cues such as tones and beeps offer low cognitive load, immediate recognizability, and modality independence, making them ideal for guiding interaction in audio-only or hands-busy contexts.

The STS tone, introduced in Chapter 5 as a short dissonant feedback signal triggered when a user spoke too early (i.e., before the system was ready to listen), served as a temporal boundary marker. Our experimental results show that participants in the STS-present group committed

significantly fewer early-start errors over time compared to the control group. This suggests that even brief, non-verbal feedback can effectively train users to pace their responses, aligning their behavior with system readiness.

These findings are in line with earlier work suggesting that non-speech auditory signals enhance interactional clarity. For example Nees and Walker (2011) found that non-speech auditory cues, such as earcons and spearcons, have been shown to effectively capture user attention and enhance responsiveness in human-machine interfaces, making them valuable for feedback in voice-based interactions where visual or cognitive load is high. Again, Lei et al. (2022) found that earcons, though more cognitively demanding, have been shown to strongly disrupt ongoing working memory tasks, highlighting their capacity to capture user attention even in multitasking environments

Our study extends this conversation by showing that non-speech cues are not merely informative—they are behavior-shaping. Participants often described the STS tone as a kind of "alert" or "checkpoint," which helped them recalibrate their timing in subsequent turns. Some reported becoming more cautious or reflective about when to speak, while others described the tone as helping them "learn the rhythm" of the system. Notably, this behavior was more than reactive—it contributed to temporal entrainment, reinforcing users' internal sense of when the system was ready.

Importantly, the STS tone's success also reveals a limitation in many commercial VUIs: the lack of timely, perceivable feedback when turn-taking errors occur. In typical systems, early speech is either ignored or poorly handled, leading users to repeat themselves or grow frustrated. By contrast, the STS tone provides immediate, context-sensitive feedback, enabling users to self-correct without needing to guess system behavior.

In sum, non-speech audio cues—particularly when designed as explicit feedback mechanisms—offer a lightweight yet powerful tool for managing turn transitions in half-duplex systems. Rather than relying solely on anthropomorphic mimicry or verbose prompts, these cues support discoverability, efficiency, and learnability, making them highly suitable for constrained conversational environments.

6.5 Mechanomorphic design strategies for VUIs

Contemporary Voice User Interfaces (VUIs) are often modeled on anthropomorphic metaphors, attempting to replicate the dynamics of human-human conversation through natural language processing, backchanneling, and full-duplex communication. This approach aims to create interfaces that feel familiar and socially engaging. However, our research raises critical questions about the effectiveness and appropriateness of such metaphors—especially in constrained environments like half-duplex VUIs, where technical and contextual limitations make naturalistic conversation difficult to sustain. It also raises the question of why anthropomorphic design has become the default, despite these limitations. Part of the answer lies in commercial and cultural expectations: anthropomorphic interfaces are easy to market and align with the widely accepted narrative of "talking to an assistant", even when the interaction is in practice governed by rigid timing, narrow task structures, and partial understanding.

In contrast, a mechanomorphic design that emphasises the system's machine-like qualities rather than masking them can foster clearer communication and user alignment. This approach prioritises discoverability Furqan et al. (2017), predictability, and transparency over mimicry. In our studies, users responded positively to explicit, machine-like signals such as non-speech tones and structured timing protocols, which offered clarity over conversational fluidity. Yet such designs are rarely seen in deployed systems, in part because they risk breaking the illusion of "natural" conversation and may initially feel less magical or personable, even if they ultimately support more reliable interaction.

Previous literature has often positioned anthropomorphic design as the default goal for spoken interaction systems (e.g., virtual assistants attempting casual, free-flowing dialogue). However, studies by Kirschthaler et al. (2020) and Luger and Sellen (2016) suggest that creating an illusion of humanness can be counterproductive, as it may lead users to expect social skills or adaptability that the system is ultimately incapable of providing. In contrast, when systems behave consistently and transparently as machines, users tend to adapt their behaviour more successfully and with less frustration. From this perspective, anthropomorphism can be seen as a kind of interface fiction: it smooths over system constraints at the cost of miscalibrated expectations, whereas mechanomorphism treats those constraints as shared, negotiable

structure.

Our research supports this position. The mechanomorphic design choices—including clearly marked turn boundaries, silence thresholds, and the STS tone—helped participants form accurate mental models of system timing and expectations. These design elements supported both temporal entrainment and task success, especially in constrained, interactive dialogue settings such as number entry, form filling, or quick prompt-response systems. The findings suggest that users are capable of adapting to machine-timed behaviour when the system makes its logic legible, which challenges the assumption that anthropomorphic presentation is necessary for good user experience.

This does not imply that anthropomorphism has no place in VUI design, but rather, that context matters. For exploratory, social, or affective interactions, anthropomorphic cues may enhance engagement. But in structured, task-oriented voice interactions, especially those occurring via half-duplex channels, a mechanomorphic approach offers greater functional value. The relative absence of such designs in commercial systems therefore reflects not a lack of viability, but a set of design and business choices that prioritise familiarity and brand "personality" over transparency about system constraints.

In sum, our findings contribute to the growing body of work advocating for design honesty in voice interfaces. By acknowledging the system's constraints and leveraging them through mechanomorphic design strategies, developers can create VUIs that are not only usable but also intuitively learnable, entrainable, and robust. More broadly, the work invites a reconsideration of what "natural" should mean in human–VUI interaction: not an imitation of human conversation at all costs, but an alignment between what the system promises, what it can do, and how users are supported in adapting to it.

6.6 Conclusion

In summary, this chapter discussed the empirical and theoretical insights generated throughout the thesis to reflect on the broader implications for VUI design. It reaffirmed the need for specialized turn-taking protocols tailored to the constraints of half-duplex systems, showing that structured feedback mechanisms can facilitate temporal alignment between users and systems.

The discussion also established that non-speech auditory cues, when carefully designed, significantly enhance user awareness and reduce conversational breakdowns. Moreover, the analysis underscored the value of adopting a mechanomorphic stance in VUI design—one that prioritizes clarity and machine-like signaling over human mimicry. These reflections collectively reinforce the thesis's contributions to the ongoing discourse on designing effective, entrainment-sensitive VUIs.

Chapter 7

Conclusion

7.1 Overview

This chapter concludes the thesis by summarizing the research journey, highlighting its key contributions, and discussing implications for the design of Voice User Interfaces (VUIs). It opens with a synthesis of the major findings across all studies and articulates how these findings respond to the research questions and objectives. The chapter then outlines the novel contributions made to both theoretical discourse and practical design of conversational systems. It further explores how these insights can inform future VUI design, particularly in terms of turn-taking coordination, feedback mechanisms, and the role of mechanomorphic cues. The chapter also critically reflects on the limitations of the current work and concludes by suggesting directions for future research to build upon the findings presented.

7.2 Revisiting the research problem

The central research problem addressed in this thesis is how to support effective turn-taking in half-duplex Voice User Interfaces (VUIs), where speaking and listening cannot occur simultaneously. The findings demonstrate that achieving temporal entrainment—the alignment of user speech timing with system cues—is essential for overcoming this constraint, as it enables users to coordinate their turns with the system more accurately, thereby improving interaction flow and task success. Crucially, when users were guided by a structured turn-taking protocol reinforced by perceptually salient non-speech cues—particularly the dissonant STS tone—they adapted their speaking behavior to match the system's timing expectations. This behavioral alignment reduced premature speech, minimized interruptions, and led to more successful task completion. These outcomes demonstrate that entrainment can be fostered through clear and consistent system signaling, rather than through attempts to mimic human-like conversational norms. By adopting a mechanomorphic design approach—where the VUI explicitly communicates its internal state and interaction boundaries—the system enables users to learn and synchronize with its temporal rhythm. In this light, temporal entrainment emerges not as a passive consequence of dialogue, but as an active outcome of well-designed interaction protocols—critical for achieving fluency and reliability in constrained, turn-based VUI environments.

7.3 Summary of research

This thesis investigated the challenges and opportunities associated with turn-taking in Voice User Interfaces (VUIs), with a particular focus on half-duplex systems where simultaneous speaking and listening are not possible. Through a series of three studies, the research generated empirical insights and design strategies aimed at improving conversational coordination between human users and machine agents.

The first study involved a qualitative content analysis of real-world interactions with commercially available VUIs, Microsoft's Cortana, Apple's Siri, Amazon's Alexa, and Google Assistant. The findings revealed a systematic pattern of turn-taking failures that disrupted task completion. These included premature user responses, delayed or ambiguous system prompts,

absence of feedback indicating system readiness, and inadequate handling of repair. Such breakdowns were often rooted in the lack of clear, structured turn-taking signals from the system, leading to misaligned expectations and increased user effort.

The second phase of the research introduced a mechanomorphic turn-taking protocol specifically designed for constrained, half-duplex environments. Unlike anthropomorphic models that attempt to replicate human-human timing dynamics, the proposed protocol prioritized functional clarity and system-led pacing. It incorporated explicit feedback elements—such as structured auditory cues and prompt-response sequences—to guide user behavior and mitigate timing-related errors.

The third and final study employed a controlled experimental design to assess the efficacy of a non-speech auditory feedback tone, referred to as the Spoke-Too-Soon (STS) tone. Results indicated that participants in the experimental group, who received immediate non-speech feedback when speaking prematurely, committed significantly fewer timing violations compared to those in the control condition. Moreover, the study uncovered evidence of both natural and deliberate entrainment to system pacing, demonstrating that turn-taking adaptation can emerge through repeated interaction, even in the absence of anthropomorphic signals.

Collectively, these findings underscore the need for rethinking turn-taking in VUIs not as an emulation of human dialogue, but as an interaction design challenge grounded in the constraints and affordances of machine-mediated communication. The research provides empirical validation for structured, feedback-driven approaches that enhance user-system coordination and improve the overall robustness of voice interactions in constrained settings.

7.4 Contribution to knowledge

This thesis makes the following contributions to the fields of Human–Computer Interaction (HCI), Conversational Interfaces, and Voice User Interface (VUI) design, with a particular emphasis on constrained, half-duplex systems.

1. Theoretical Contributions: This research contributes to theory by establishing the presence of temporal entrainment in human–machine interaction, a phenomenon previously

studied mainly in human–human conversation. The findings demonstrate that users entrain/adapt to system-imposed timing cues—both unconsciously and deliberately—when interacting with VUI equipped with a half-duplex turn-taking protocol. This extends entrainment theory into the human–VUI domain, highlighting that conversational alignment can emerge from consistent, mechanomorphic system behavior. The work also advances a mechanomorphic design perspective, positioning VUIs as systems that guide interaction through transparent machine-led pacing rather than human-like mimicry. This reframing offers a theoretical basis for designing predictable, rhythm-aware voice interfaces under system constraints.

2. Methodological contributions: This research offers a practical methodological contribution by demonstrating how human users' turn-taking behaviour with Voice User Interfaces (VUIs) can be systematically evaluated in low-resource environments. A key aspect of this approach involved the use of a lightweight, standalone interaction logging tool capable of capturing timestamped user-system exchanges, including premature responses, timing delays, and error patterns. Rather than relying on cloud-based ASR services or complex conversational analytics platforms, the experimental setup operated entirely offline—making it feasible for constrained academic or field research contexts. Crucially, the interaction task was carefully designed to align with the capabilities of the logging tool: participants engaged in structured, repeatable sequences that allowed for fine-grained analysis of turn-taking coordination across multiple passes. This integration of a transparent, diagnostic tool with a theory-informed task design enabled a detailed examination of how users entrain to system feedback under controlled timing constraints. The result is a replicable, scalable, and accessible framework for evaluating conversational timing in half-duplex VUIs—providing a methodological pathway for researchers to investigate turn-taking dynamics without requiring high-end infrastructure. By lowering the technical barrier to entry, this approach broadens the scope of empirical research in voice interface design and supports more inclusive experimentation across diverse settings.

In sum, the thesis contributes a re-conceptualization of turn-taking in VUIs as a design problem suited to mechanomorphic solutions, supported by empirical analysis and experimental validation. It challenges prevailing assumptions about naturalness in voice interaction, offering

instead a model grounded in clarity, discoverability, and adaptive feedback.

7.5 Implications for design

This research provides practical guidelines for designing Voice User Interfaces, especially those operating under constrained conditions like half-duplex systems. Rather than striving to replicate natural human conversation styles, designers may benefit from exploring systematic, machine-led interaction flows that align more closely with the capabilities and constraints of Voice User Interfaces. Such an approach can support more predictable turn-taking and reduce user confusion in interactive environments. In systems where speaking and listening cannot happen simultaneously, it is essential to implement clear indicators of turn boundaries—such as distinct auditory cues or timed pauses—so that users know exactly when to speak and when to wait. Incorporating distinct non-speech audio cues—such as those indicating when the system is ready to listen or when the user speaks prematurely—can facilitate smoother and more fluent turn-taking in voice interactions. Furthermore, designers should consider introducing rhythmic patterns or timed delays that encourage users to adapt their timing in line with the system's feedback—this promotes entrainment and reduces turn-taking errors. Design efforts might be better directed toward enhancing the transparency and predictability of the system's internal logic and state transitions, rather than relying solely on human-like conversational mimicry to signal intelligence. This approach helps users feel more in control, reduces errors, and enhances trust, especially in safety-critical or task-focused scenarios.

7.6 Limitations and future work

This research has a few notable limitations. First, the participant pool for experiments 2 and 3 was small and recruited through snowball sampling within a specific setting. Future studies could explore the impact of providing explicit instructions to novice users—particularly for adopting behaviour #6 (see Appendix F) in response to the STS tone. Informal observations suggest that users who understand how to pause and restart after encountering the STS tone find

the process intuitive and use it effectively to recover from early speech errors. This significantly reduces the penalty for Spoke-Too-Soon incidents and encourages users to approach the invisible seam more confidently, potentially increasing throughput. If validated, this could lead to integrating real-time instructional cues or "tips" into the system, triggered by behaviours #3–#5 (see Appendix F) transforming SmartWindow (see chapter 4, section 4.7.1) into a self-teaching protocol.

Second, there remains an open question about the role of tones used in the study. While the musical syntax of the current tone set may aid comprehension and adaptation, it's worth investigating whether alternative tone types—such as short ticks, mechanical clicks, tonal chimes, or percussive bongs, or mechanical sounds—could be equally or more effective. These variations might also serve branding or personalization purposes, making them selectable like modern ringtones.

Third, the task design was limited to a digit-based vocabulary of variable length. Future studies could broaden this by introducing tasks with more semantic or cognitive complexity—such as healthcare or political surveys, translation flashcards, or word-meaning drills. This would allow researchers to examine how meaningfulness and cognitive load influence user behaviour under consistent prompt-response turn-taking conditions.

List of Publications

List of Published Papers

International Conference

1. **Phukon, Mridumoni**, Abhishek Shrivastava, and Bruce Balentine. "Can VUI turn-taking entrain user behaviours? voice user interfaces that disallow overlapping speech present turn-taking challenges." Proceedings of the 13th Indian conference on human-computer interaction. 2022.
   Candidate's contribution:
   The candidate concieved the core idea for the paper. The candidate led the overall study design concept, developed the initial interaction concept, created the experimental materials, and implemented the prototype in collaboration with co-authors. The candidate was primarily responsible for data collection, data cleaning, and statistical analysis. The candidate drafted the first version of the manuscript and integrated co-authors' feedback into the final submitted version.

2. **Phukon, Mridumoni**, and Abhishek Shrivastava. "Effect of Speech Entrainment in Human-Computer Conversation: A Review." International Conference on Intelligent Human Computer Interaction. Cham: Springer Nature Switzerland, 2023.

Candidate's contribution:

I conceived the core idea for this review, carried out the literature survey, and was the primary author of the manuscript.

3. Deka, C., Sah, S., Shrivastava, A., **Phukon, M.**, & Routray, L. (2021, December). Assessing a Voice-Based Conversational AI prototype for Banking Application. In 2021 8th NAFOSTED Conference on Information and Computer Science (NICS) (pp. 211-216). IEEE.

   contributed to the study design, drafted substantial parts of the literature review, and assisted in writing and refining the overall manuscript.

List of Submitted paper

International Journal

1. **Phukon, Mridumoni**, Abhishek Shrivastava. "Non-Speech Auditory Cues as Corrective Signals for Turn-Taking in Half-Duplex Voice User Interfaces" Interacting with Computers. 2025.

Appendix A

The number scripts to be read by the participants during the task

Table A.1: Script 1

Line No.	Script 1
1	Six-One
2	Nine-Eight-Seven
3	Six-Three-Nine
4	Zero-Three-Nine-Zero-Five
5	Zero-Four-Seven
6	Three-Zero
7	Eight-Two-Two
8	Zero-One-Eight-Zero
9	One-Two-Five-Seven
10	Two-Eight-Four-One
11	Seven-Zero-Four-Five-Nine-Five
12	Four-Three-Six-Zero-Three

Table A.2: Script 2

Line No.	Script 2
1	Four-Three-Six-Zero-Three
2	Seven-Zero-Four-Five-Nine-Five
3	Zero-Three-Nine-Zero-Five
4	Zero-One-Eight-Zero
5	Nine-Eight-Seven
6	Zero-Four-Seven
7	One-Two-Five-Seven
8	Six-One
9	Zero-One-Eight-Zero
10	Eight-Two-Two
11	Six-Three-Nine
12	Two-Eight-Four-One

Table A.3: Script 3

Line No.	Script 3
1	Four-Three-Six-Zero-Three
2	Seven-Zero-Four-Five-Nine-Five
3	Eight-Two-Two
4	Zero-One-Eight-Zero
5	Six-Three-Nine
6	One-Two-Five-Seven
7	Nine-Eight-Seven
8	Zero-Three-Nine-Zero-Five
9	Two-Eight-Four-One
10	Three-Zero
11	Zero-Four-Seven
12	Six-One

This page was intentionally left blank.

Appendix B

The transcription of the conversation analysed between VUI and human user in real-life scenario

Table B.1: Sample Conversation Analysis Entry

Transcription (Verbatim)	**User's Facial Expression**	**Condensed Meaning(s) Unit**	**Sub Categories**	**Categories**	**Theme**
P: Hey Cortana A: *beeps* P: Play the song "Perfect" in YouTube.. A: Here is what I found. [Displays search results on the screen] [Participant manually selects a video to play on the browser.]	Confident Content	Greeting, Trigger Listening Issues a command to play a song Presents options for self-service	Issues a command Acceptable but undesired response	Acceptable but undesired response	Acceptable but undesired response by the Assistant

Table B.2: Sample Conversation Analysis Entry – Shopping List Task

Transcription (Verbatim)	**User's Facial Expression**	**Condensed Meaning(s) Unit**	**Sub Categories**	**Categories**	**Theme**
P: Hey Cortana. A: *beeps* P: Add milk to my shopping list. A: Alright. I added that to your shopping list. [Displays list with updated items]	Confident Content	Greeting, Trigger Listening Issues a command to add milk to shopping list Updates list and displays	Issues a command Desired Response	Desired Response	Desired Response by the Assistant

Table B.3: Transcript Analysis: Reminder Setting Breakdown

Transcription (Verbatim)	**User's Facial Expression**	**Condensed Meaning(s) Unit**	**Sub Categories**	**Categories**	**Theme**
P: Hey Cortana.	Confident	Greeting, Trigger	Issues a command	Acceptable but undesired response	Acceptable but undesired response by the Assistant
P: Hey Cortana.	Doubtful	Greeting, Trigger	Issues a command	Acceptable but undesired response	Acceptable but undesired response by the Assistant
A: *beep*	–	Listening	–	–	–
P: Set me a reminder, today at 4:30 PM	Confident	Issues a command to set a reminder	Issues a command	Acceptable but undesired response	Acceptable but undesired response by the Assistant
A: Sure thing. What do you want to be reminded about?	Content	Accepts. Enquires about the title.	Requests for input	Acceptable but undesired response	Acceptable but undesired response by the Assistant
P: A meeting [detected as: Tell meeting]	Confident	Inputs the title of the meeting	Inputs data	Acceptable but undesired response	Acceptable but undesired response by the Assistant
A: Okay.	–	Processing	–	–	–
A: Great [Displays reminder with "Tell Meeting" as the title]	Confused	Sets reminder and displays	Acceptable but undesired response	Acceptable but undesired response	Acceptable but undesired response by the Assistant

This page was intentionally left blank.

Appendix C

Summary of observed-behaviour of the participants taking part in the experiment to analyse human turn-taking behaviour

Table C.1: Summary of Observed Behaviours

#	Count	— Normal Expected Behaviours – no STS —	Expected?
1	352	normal capture	many more than expected
2a	0	silence timeout (no speech)	none, as expected
2b	9	silence timeout due to mistimed speech	slightly more than expected
		—– Expected Reactions to STS –(in order of frequency) —–	
3	55	“plough-through”	slightly less than expected, limited to three participants
4	2	abruptly stop speaking and wait for next step	as expected
5	3	stop speaking briefly, but then continue with utterance	as expected
6	8	abruptly stop speaking and then start over from beginning	as expected
		— Reactions to Tapering —	
7	-	STS on first event followed by no STS on second and third	not enough data to draw a conclusion
8	-	increase in STS on events 4 through 12	not enough data to draw a conclusion
9	1	1st-try/2nd-try STS pattern on quick retry	as expected
10	11	general synchronization on prompt-response rhythm and/or rate of speech from Pass 1 to Pass 3.	as expected

Appendix D

Log Files generated during the experiment

The log files for participant with PID=23

13:30 – 01 April 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number. — SmartWindow Parameters —- WinTimer : 6000.0 SilenceTimer : 175.0

——————— RUN MODE ——————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.004 - Suspend ASR 1.924 - Prompt+Tone End 1.925 - Window Open (Listening) 1.926 - Resume ASR 3.663 - Onset Detected (1.736) 5.068 - Speecru Ended (1.405) 5.097 - SIX ONE 5.098 - Accept Pattern: 1 5.160 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.889 - Prompt+Tone

End 1.889 - Window Open (Listening) 1.891 - Resume ASR 2.001 - Onset Detected (0.110) 2.006 - SPOKE-TOO-SOON 2.054 - Suspend ASR 2.573 - Resume ASR (0.519) 2.789 - Onset Detected (0.216) 4.684 - Speech Ended (1.894) 4.724 - NINE TWO 4.725 - Accept Pattern: 2 4.784 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.541 - Prompt+Tone End 1.542 - Window Open (Listening) 1.544 - Resume ASR 1.738 - Onset Detected (0.194) 2.514 - Speech Ended (0.775) 2.530 - OH 2.532 - Accept Pattern: 1 2.557 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.533 - Prompt+Tone End 1.533 - Window Open (Listening) 1.535 - Resume ASR 1.651 - Onset Detected (0.115) 1.656 - SPOKE-TOO-SOON 1.698 - Suspend ASR 2.219 - Resume ASR (0.521) 2.421 - Onset Detected (0.201) 5.002 - Speech Ended (2.581) 5.050 - TWO NINE TWO 5.052 - Accept Pattern: 2 5.108 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.534 - Prompt+Tone End 1.534 - Window Open (Listening) 1.536 - Resume ASR 2.204 - Onset Detected (0.667) 3.519 - Speech Ended (1.315) 3.549 - TWO 3.551 - Accept Pattern: 1 3.574 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.533 - Prompt+Tone End 1.534 - Window Open (Listening) 1.535 - Resume ASR 1.738 - Onset Detected (0.202) 3.133 - Speech Ended (1.395) 3.165 - THREE 3.167 - Accept Pattern: 1 3.189 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.532 - Prompt+Tone End 1.532 - Window Open (Listening) 1.534 - Resume ASR 2.172 - Onset Detected (0.637) 3.315 - Speech Ended (1.143) 3.342 - NULL 6.522 - Onset Detected (4.987) 7.543 - Pattern: 5b 7.545 - Full Retry 7.546 - Prompt+Tone Start 7.549 - Suspend ASR 9.637 - Prompt+Tone End 9.638 - Window Open (Listening) 9.640 - Resume ASR 10.375 - Onset Detected (0.735) 12.016 - Speech Ended (1.641) 12.049 - TWO ONE 12.050 - Accept Pattern: 1 12.107 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.543 - Prompt+Tone End 1.544 - Window Open (Listening) 1.546 - Resume ASR 2.251 - Onset Detected (0.705) 4.300 - Speech Ended (2.048) 4.342 - ZERO ONE 4.343 - Accept Pattern: 1 4.395 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.536 - Prompt+Tone End 1.536 - Window Open (Listening) 1.538 - Resume ASR 2.390 - Onset Detected (0.852) 4.082 - Speech Ended (1.691) 4.118 - ONE TWO FIVE SEVEN 4.120 - Accept Pattern: 1 4.175 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.549 - Prompt+Tone End 1.549 - Window Open (Listening) 1.551 - Resume ASR 2.606 - Onset Detected (1.054) 4.034 - Speech Ended (1.428) 4.064 - TWO 4.065 - Accept Pattern: 1 4.102 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.543 - Prompt+Tone End 1.544 - Window Open (Listening) 1.546 - Resume ASR 2.255 - Onset Detected (0.709) 4.600 - Speech Ended (2.344) 4.649 - TWO NINE 4.651 - Accept Pattern: 1 4.693 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.551 - Prompt+Tone End 1.552 - Window Open (Listening) 1.555 - Resume ASR 2.487 - Onset Detected (0.931) 4.534 - Speech Ended (2.047) 4.575 - FOUR TWO SIX TWO 4.577 - Accept Pattern: 1 4.617 - Suspend ASR 13:32 – 01 April 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number.

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 -

Prompt+Tone Start 0.005 - Suspend ASR 1.923 - Prompt+Tone End 1.924 - Window Open (Listening) 1.926 - Resume ASR 3.022 - Onset Detected (1.096) 4.549 - Speech Ended (1.527) 4.589 - TWO 4.590 - Accept Pattern: 1 4.648 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.896 - Prompt+Tone End 1.896 - Window Open (Listening) 1.898 - Resume ASR 2.521 - Onset Detected (0.622) 4.319 - Speech Ended (1.797) 4.363 - ZERO OH ONE 4.365 - Accept Pattern: 1 4.407 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.547 - Prompt+Tone End 1.548 - Window Open (Listening) 1.550 - Resume ASR 2.006 - Onset Detected (0.456) 3.665 - Speech Ended (1.658) 3.700 - TWO 3.702 - Accept Pattern: 1 3.760 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.546 - Prompt+Tone End 1.546 - Window Open (Listening) 1.548 - Resume ASR 2.383 - Onset Detected (0.835) 3.912 - Speech Ended (1.529) 3.949 - ZERO OH 3.951 - Accept Pattern: 1 3.969 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.550 - Prompt+Tone End 1.550 - Window Open (Listening) 1.552 - Resume ASR 2.437 - Onset Detected (0.884) 3.965 - Speech Ended (1.528) 3.995 - SIX 3.997 - Accept Pattern: 1 4.016 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.543 - Prompt+Tone End 1.544 - Window Open (Listening) 1.546 - Resume ASR 2.468 - Onset Detected (0.922) 3.965 - Speech Ended (1.496) 3.997 - ONE TWO FIVE 3.998 - Accept Pattern: 1 4.054 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.544 - Prompt+Tone

End 1.544 - Window Open (Listening) 1.546 - Resume ASR 2.522 - Onset Detected (0.975) 3.667 - Speech Ended (1.144) 3.691 - NINE 3.693 - Accept Pattern: 1 3.711 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.537 - Prompt+Tone End 1.538 - Window Open (Listening) 1.540 - Resume ASR 2.423 - Onset Detected (0.883) 3.949 - Speech Ended (1.526) 3.987 - TWO TWO 3.989 - Accept Pattern: 1 4.049 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.540 - Prompt+Tone End 1.541 - Window Open (Listening) 1.543 - Resume ASR 2.437 - Onset Detected (0.894) 3.649 - Speech Ended (1.211) 3.675 - TWO 3.676 - Accept Pattern: 1 3.709 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.550 - Prompt+Tone End 1.550 - Window Open (Listening) 1.552 - Resume ASR 3.073 - Onset Detected (1.520) 3.966 - Speech Ended (0.893) 3.992 - TWO 3.994 - Accept Pattern: 1 4.016 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.551 - Prompt+Tone End 1.552 - Window Open (Listening) 1.553 - Resume ASR 2.389 - Onset Detected (0.835) 3.551 - Speech Ended (1.161) 3.580 - ZERO 3.581 - Accept Pattern: 1 3.636 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.543 - Prompt+Tone End 1.544 - Window Open (Listening) 1.547 - Resume ASR 3.790 - Onset Detected (2.243) 4.816 - Speech Ended (1.026) 4.839 - SIX ONE 4.841 - Accept Pattern: 1 4.862 - Suspend ASR 13:34 – 01 April 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number.

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 -

Prompt+Tone Start 0.005 - Suspend ASR 1.924 - Prompt+Tone End 1.924 - Window Open (Listening) 1.926 - Resume ASR 3.040 - Onset Detected (1.114) 5.582 - Speech Ended (2.541) 5.633 - TWO THREE 5.634 - Accept Pattern: 1 5.671 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.887 - Prompt+Tone End 1.888 - Window Open (Listening) 1.889 - Resume ASR 2.657 - Onset Detected (0.767) 4.450 - Speech Ended (1.792) 4.495 - TWO TWO 4.496 - Accept Pattern: 1 4.526 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.550 - Prompt+Tone End 1.551 - Window Open (Listening) 1.553 - Resume ASR 2.271 - Onset Detected (0.718) 3.683 - Speech Ended (1.412) 3.719 - TWO 3.721 - Accept Pattern: 1 3.762 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.537 - Prompt+Tone End 1.538 - Window Open (Listening) 1.540 - Resume ASR 2.388 - Onset Detected (0.847) 3.798 - Speech Ended (1.410) 3.833 - ZERO ONE 3.835 - Accept Pattern: 1 3.876 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.534 - Prompt+Tone End 1.535 - Window Open (Listening) 1.538 - Resume ASR 2.520 - Onset Detected (0.982) 3.832 - Speech Ended (1.311) 3.859 - NINE SEVEN 3.861 - Accept Pattern: 1 3.915 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.542 - Prompt+Tone End 1.543 - Window Open (Listening) 1.545 - Resume ASR 2.088 - Onset Detected (0.542) 3.748 - Speech Ended (1.659) 3.786 - TWO 3.787 - Accept Pattern: 1 3.838 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.546 - Prompt+Tone

End 1.546 - Window Open (Listening) 1.549 - Resume ASR 2.518 - Onset Detected (0.969) 4.318 - Speech Ended (1.799) 4.352 - ONE TWO 4.354 - Accept Pattern: 1 4.398 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.539 - Prompt+Tone End 1.540 - Window Open (Listening) 1.542 - Resume ASR 1.922 - Onset Detected (0.380) 2.764 - Speech Ended (0.841) 2.787 - SIX 2.789 - Accept Pattern: 1 2.810 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.536 - Prompt+Tone End 1.537 - Window Open (Listening) 1.539 - Resume ASR 2.170 - Onset Detected (0.630) 3.583 - Speech Ended (1.412) 3.615 - TWO TWO 3.617 - Accept Pattern: 1 3.663 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.537 - Prompt+Tone End 1.538 - Window Open (Listening) 1.540 - Resume ASR 2.140 - Onset Detected (0.600) 3.332 - Speech Ended (1.192) 3.358 - TWO 3.360 - Accept Pattern: 1 3.409 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.550 - Prompt+Tone End 1.550 - Window Open (Listening) 1.552 - Resume ASR 2.486 - Onset Detected (0.934) 3.933 - Speech Ended (1.447) 3.961 - SIX 3.962 - Accept Pattern: 1 4.018 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.541 - Prompt+Tone End 1.542 - Window Open (Listening) 1.543 - Resume ASR 2.357 - Onset Detected (0.813) 3.500 - Speech Ended (1.143) 3.524 - TWO 3.526 - Accept Pattern: 1 3.583 - Suspend ASR

The log files for participant with PID=24

13:55 – 01 April 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true

ReturnScores : false MaxSts : 8 Prompt : Say the first number. — SmartWindow Parameters —- WinTimer : 6000.0 SilenceTimer : 175.0

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.003 - Suspend ASR 1.963 - Prompt+Tone End 1.963 - Window Open (Listening) 1.964 - Resume ASR 2.141 - Onset Detected (0.177) 5.333 - Speech Ended (3.191) 5.385 - FOUR SIX ZERO THREE 5.385 - Accept Pattern: 1 5.411 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.895 - Prompt+Tone End 1.895 - Window Open (Listening) 1.897 - Resume ASR 2.287 - Onset Detected (0.390) 5.530 - Speech Ended (3.243) 5.578 - SEVEN ZERO FOUR FIVE NINE 5.578 - Accept Pattern: 1 5.599 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.540 - Prompt+Tone End 1.541 - Window Open (Listening) 1.541 - Resume ASR 1.721 - Onset Detected (0.179) 4.650 - Speech Ended (2.929) 4.697 - ZERO THREE NINE ZERO FIVE 4.698 - Accept Pattern: 1 4.735 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.541 - Prompt+Tone End 1.541 - Window Open (Listening) 1.542 - Resume ASR 1.633 - Onset Detected (0.091) 1.636 - SPOKE-TOO-SOON 1.707 - Suspend ASR 2.225 - Resume ASR (0.517) 2.903 - Onset Detected (0.678) 5.086 - Speech Ended (2.183) 5.121 - ZERO ONE EIGHT ZERO 5.122 - Accept Pattern: 2 5.162 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.536 - Prompt+Tone End 1.536 - Window Open (Listening) 1.537 - Resume ASR 2.656 - Onset Detected (1.119) 4.186 - Speech Ended (1.529) 4.214 - NINE EIGHT SEVEN 4.214 - Accept Pattern: 1 4.261 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.560 - Prompt+Tone End 1.561 - Window Open (Listening) 1.561 - Resume ASR 1.736 - Onset Detected (0.174) 4.416 - Speech Ended (2.680) 4.455 - ZERO FOUR SEVEN 4.456 - Accept Pattern: 1 4.498 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.539 - Prompt+Tone End 1.539 - Window Open (Listening) 1.540 - Resume ASR 2.324 - Onset Detected (0.784) 4.186 - Speech Ended (1.862) 4.217 - ONE TWO FIVE SEVEN 4.218 - Accept Pattern: 1 4.264 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.554 - Prompt+Tone End 1.554 - Window Open (Listening) 1.556 - Resume ASR 1.685 - Onset Detected (0.128) 1.687 - SPOKE-TOO-SOON 1.719 - Suspend ASR 2.238 - Resume ASR (0.518) 2.462 - Onset Detected (0.224) 3.684 - Speech Ended (1.221) 3.711 - SIX 3.712 - Accept Pattern: 2 3.767 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.540 - Prompt+Tone End 1.541 - Window Open (Listening) 1.542 - Resume ASR 1.756 - Onset Detected (0.214) 4.369 - Speech Ended (2.612) 4.411 - ZERO ONE EIGHT ZERO 4.412 - Accept Pattern: 1 4.433 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.558 - Prompt+Tone End 1.558 - Window Open (Listening) 1.560 - Resume ASR 1.771 - Onset Detected (0.211) 3.517 - Speech Ended (1.746) 3.551 - EIGHT TWO TWO 3.551 - Accept Pattern: 1 3.601 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.541 - Prompt+Tone End 1.541 - Window Open (Listening) 1.542 - Resume ASR 2.274 - Onset Detected (0.732) 3.719 - Speech Ended (1.444) 3.748 - SIX THREE NINE 3.749 - Accept Pattern: 1 3.797 -

Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.001 - Suspend ASR 1.554 - Prompt+Tone End 1.555 - Window Open (Listening) 1.556 - Resume ASR 1.652 - Onset Detected (0.095) 1.654 - SPOKE-TOO-SOON 1.675 - Suspend ASR 2.195 - Resume ASR (0.520) 2.369 - Onset Detected (0.173) 2.372 - SPOKE-TOO-SOON 2.375 - Suspend ASR 2.375 - Resume ASR (0.000) 2.773 - Onset Detected (0.398) 4.450 - Speech Ended (1.676) 4.481 - TWO EIGHT FOUR 4.482 - Accept Pattern: 2 4.534 - Suspend ASR 13:57 – 01 April 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number.

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.004 - Suspend ASR 1.918 - Prompt+Tone End 1.918 - Window Open (Listening) 1.919 - Resume ASR 2.078 - Onset Detected (0.158) 2.079 - SPOKE-TOO-SOON 2.126 - Suspend ASR 2.645 - Resume ASR (0.518) 2.803 - Onset Detected (0.158) 2.806 - SPOKE-TOO-SOON 2.808 - Suspend ASR 2.808 - Resume ASR (0.000) 2.923 - Onset Detected (0.114) 2.925 - SPOKE-TOO-SOON 2.927 - Suspend ASR 2.927 - Resume ASR (0.000) 3.072 - Onset Detected (0.145) 3.074 - SPOKE-TOO-SOON 3.076 - Suspend ASR 3.076 - Resume ASR (0.000) 7.272 - Onset Detected (4.195) 7.926 - Speech timeout 7.927 - Pattern: 5c 7.928 - Full Retry 7.930 - Prompt+Tone Start 7.932 - Suspend ASR 10.024 - Prompt+Tone End 10.025 - Window Open (Listening) 10.026 - Resume ASR 10.656 - Onset Detected (0.629) 12.001 - Speech Ended (1.345) 12.027 - SIX ONE 12.028 - Accept Pattern: 1 12.067 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.868 - Prompt+Tone End 1.868 - Window Open (Listening) 1.869 - Resume ASR 2.839 - Onset Detected (0.970) 4.401 - Speech Ended (1.561) 4.430 - NINE EIGHT SEVEN 4.431 - Accept Pattern: 1 4.465 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.538 - Prompt+Tone End 1.539 - Window Open (Listening) 1.540 - Resume ASR 2.288 - Onset Detected (0.747) 3.901 - Speech Ended (1.612) 3.930 - SIX NINE 3.931 - Accept Pattern: 1 3.965 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.564 - Prompt+Tone End 1.565 - Window Open (Listening) 1.566 - Resume ASR 1.755 - Onset Detected (0.188) 4.567 - Speech Ended (2.812) 4.612 - ZERO THREE NINE ZERO FIVE 4.612 - Accept Pattern: 1 4.673 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.546 - Prompt+Tone End 1.546 - Window Open (Listening) 1.547 - Resume ASR 2.120 - Onset Detected (0.573) 3.783 - Speech Ended (1.663) 3.814 - ZERO FOUR SEVEN 3.815 - Accept Pattern: 1 3.844 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.534 - Prompt+Tone End 1.535 - Window Open (Listening) 1.536 - Resume ASR 1.634 - Onset Detected (0.098) 1.637 - SPOKE-TOO-SOON 1.700 - Suspend ASR 2.218 - Resume ASR (0.518) 2.421 - Onset Detected (0.202) 4.003 - Speech Ended (1.581) 4.033 - TWO 4.034 - Accept Pattern: 2 4.089 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.545 - Prompt+Tone End 1.545 - Window Open (Listening) 1.546 - Resume ASR 2.204 - Onset Detected (0.657) 3.619 - Speech Ended (1.414) 3.645 - EIGHT TWO TWO 3.646 - Accept Pattern: 1 3.673 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.546 - Prompt+Tone End 1.546 - Window Open (Listening) 1.547 - Resume ASR 1.651 - Onset Detected (0.103) 1.654 - SPOKE-TOO-SOON 1.712 - Suspend ASR 2.231 - Resume ASR (0.519) 2.421 - On-

set Detected (0.190) 5.750 - Speech Ended (3.328) 5.796 - ZERO ONE ZERO 5.796 - Accept Pattern: 2 5.850 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.540 - Prompt+Tone End 1.541 - Resume ASR PID 2.372 - Onset Detected (0.830) 4.169 - Speech Ended (1.797) 4.199 - ONE TWO FIVE SEVEN 4.200 - Accept Pattern: 1 4.220 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.545 - Prompt+Tone End 1.546 - Window Open (Listening) 1.548 - Resume ASR 1.651 - Onset Detected (0.103) 1.654 - SPOKE-TOO-SOON 1.711 - Suspend ASR 2.228 - Resume ASR (0.517) 2.419 - Onset Detected (0.190) 4.734 - Speech Ended (2.315) 4.775 - TWO FOUR 4.775 - Accept Pattern: 2 4.825 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.548 - Prompt+Tone End 1.549 - Window Open (Listening) 1.550 - Resume ASR 2.387 - Onset Detected (0.837) 5.200 - Speech Ended (2.812) 5.247 - SEVEN ZERO FOUR FIVE NINE 5.247 - Accept Pattern: 1 5.297 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.562 - Prompt+Tone End 1.563 - Window Open (Listening) 1.564 - Resume ASR 2.352 - Onset Detected (0.788) 4.653 - Speech Ended (2.300) 4.698 - FOUR THREE 4.699 - Accept Pattern: 1 4.757 - Suspend ASR 13:59 – 01 April 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number.

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.004 - Suspend ASR 1.912 - Prompt+Tone End 1.912 - Window Open (Listening) 1.914 - Resume ASR 2.805 - Onset Detected (0.890) 5.119 - Speech Ended (2.313) 5.158 - FOUR EXIT ZERO THREE 5.159 - Accept Pattern: 1 5.191 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.889 - Prompt+Tone End 1.889 - Window Open (Listening) 1.891 - Resume ASR 2.523 - Onset Detected (0.632) 5.469 - Speech Ended (2.946) 5.514 - SEVEN ZERO FOUR FIVE NINE FIVE 5.514 - Accept Pattern: 1 5.553 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.543 - Prompt+Tone End 1.543 - Window Open (Listening) 1.545 - Resume ASR 2.271 - Onset Detected (0.725) 3.669 - Speech Ended (1.398) 3.696 - EIGHT TWO TWO 3.697 - Accept Pattern: 1 3.756 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.545 - Prompt+Tone End 1.546 - Window Open (Listening) 1.547 - Resume ASR 1.872 - Onset Detected (0.325) 3.970 - Speech Ended (2.097) 4.003 - ZERO ONE EIGHT ZERO 4.004 - Accept Pattern: 1 4.058 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.530 - Prompt+Tone End 1.531 - Window Open (Listening) 1.532 - Resume ASR 2.204 - Onset Detected (0.672) 3.867 - Speech Ended (1.662) 3.897 - SIX NINE 3.898 - Accept Pattern: 1 3.957 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.547 - Prompt+Tone End 1.547 - Window Open (Listening) 1.549 - Resume ASR 2.005 - Onset Detected (0.455) 4.185 - Speech Ended (2.179) 4.221 - ONE TWO FIVE SEVEN 4.222 - Accept Pattern: 1 4.272 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.545 - Prompt+Tone End 1.546 - Window Open (Listening) 1.547 - Resume ASR 2.304 - Onset Detected (0.757)

4.451 - Speech Ended (2.146) 4.482 - NINE EIGHT SEVEN 4.483 - Accept Pattern: 1 4.527 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.550 - Prompt+Tone End 1.551 - Window Open (Listening) 1.552 - Resume ASR 2.268 - Onset Detected (0.715) 4.878 - Speech Ended (2.610) 4.920 - ZERO THREE NINE ZERO FIVE 4.920 - Accept Pattern: 1 4.957 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.548 - Prompt+Tone End 1.548 - Window Open (Listening) 1.549 - Resume ASR 2.522 - Onset Detected (0.972) 4.268 - Speech Ended (1.745) 4.301 - TWO FOUR ONE 4.302 - Accept Pattern: 1 4.358 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.547 - Prompt+Tone End 1.547 - Window Open (Listening) 1.549 - Resume ASR 2.188 - Onset Detected (0.639) 3.589 - Speech Ended (1.400) 3.612 - THREE ZERO 3.613 - Accept Pattern: 1 3.630 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.537 - Prompt+Tone End 1.537 - Window Open (Listening) 1.538 - Resume ASR 2.005 - Onset Detected (0.467) 3.667 - Speech Ended (1.661) 3.694 - ZERO FOUR SEVEN 3.695 - Accept Pattern: 1 3.753 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.542 - Prompt+Tone End 1.543 - Window Open (Listening) 1.544 - Resume ASR 2.255 - Onset Detected (0.710) 3.585 - Speech Ended (1.330) 3.609 - SIX 3.610 - Accept Pattern: 1 3.668 - Suspend ASR

The log files for participant with PID=26

11:27 – 10 May 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number. — SmartWindow Parameters —- WinTimer : 6000.0 SilenceTimer : 175.0

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.004 - Suspend ASR 1.934 - Prompt+Tone End 1.934 - Window Open (Listening) 1.935 - Resume ASR 2.531 - Onset Detected (0.596) 6.101 - Speech Ended (3.569) 6.163 - OH THREE TWO THREE 6.164 - Accept Pattern: 1 6.194 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.893 - Prompt+Tone End 1.893 - Window Open (Listening) 1.894 - Resume ASR 2.020 - Onset Detected (0.125) 2.022 - SPOKE-TOO-SOON 2.058 - Suspend ASR 2.577 - Resume ASR (0.519) 3.405 - Onset Detected (0.827) 7.668 - Speech Ended (4.263) 7.734 - TWO OH FIVE NINE FIVE 7.735 - Accept Pattern: 2 7.775 - Suspend ASR 7.901 - Speech timeout 7.901 - Pattern: 5c 7.902 - Full Retry 7.904 - Prompt+Tone Start 7.906 - Suspend ASR 10.000 - Prompt+Tone End 10.001 - Window Open (Listening) 10.002 - Resume ASR 10.704 - Onset Detected (0.701) 15.569 - Speech Ended (4.865) 15.640 - SEVEN TWO OH FIVE NINE FIVE 15.641 - Accept Pattern: 1 15.668 - Suspend ASR 16.008 - Pattern: 5b 16.008 - Give Up 16.010 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.533 - Prompt+Tone End 1.534 - Window Open (Listening) 1.535 - Resume ASR 2.454 - Onset Detected (0.918) 4.751 - Speech Ended (2.297) 4.790 - EIGHT TWO TWO 4.791 - Accept Pattern: 1 4.811 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.554 - Prompt+Tone End 1.554 - Window Open (Listening) 1.556 - Resume ASR 2.440 - Onset Detected (0.884) 6.535 - Speech Ended (4.095) 6.599 - TWO ONE EIGHT TWO 6.600 - Accept Pattern: 1 6.625 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.549 - Prompt+Tone End 1.549 - Window Open (Listening) 1.550 - Resume ASR 2.254 - Onset Detected (0.703) 4.850 - Speech Ended (2.596) 4.893 - SIX THREE NINE 4.893 - Accept Pattern: 1 4.911 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.553 - Prompt+Tone End 1.553 - Window Open (Listening) 1.554 - Resume ASR 2.574 - Onset Detected (1.019) 5.733 - Speech Ended (3.159) 5.785 - ONE TWO FIVE SEVEN 5.786 - Accept Pattern: 1 5.814 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.537 - Prompt+Tone End 1.538 - Window Open (Listening) 1.539 - Resume ASR 2.254 - Onset Detected (0.715) 4.551 - Speech Ended (2.297) 4.592 - NINE EIGHT SEVEN 4.592 - Accept Pattern: 1 4.646 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.553 - Prompt+Tone End 1.553 - Window Open (Listening) 1.554 - Resume ASR 2.555 - Onset Detected (1.000) 6.484 - Speech Ended (3.929) 6.547 - TWO TWO NINE TWO FIVE 6.547 - Accept Pattern: 1 6.582 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.538 - Prompt+Tone End 1.538 - Window Open (Listening) 1.539 - Resume ASR 2.373 - Onset Detected (0.833) 5.318 - Speech Ended (2.944) 5.365 - TWO EIGHT FOUR ONE 5.366 - Accept Pattern: 1 5.415 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.555 - Prompt+Tone End 1.555 - Window Open (Listening) 1.556 - Resume ASR 2.389 - Onset Detected (0.833) 4.182 - Speech Ended (1.793) 4.220 - TWO 4.221 - Accept Pattern: 1 4.281 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.554 - Prompt+Tone End 1.555 - Window Open (Listening) 1.556 - Resume ASR 2.271 - Onset Detected (0.715) 4.699 - Speech Ended (2.428) 4.744 - TWO FOUR SEVEN 4.745 - Accept Pattern: 1 4.790 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.547 - Prompt+Tone End 1.547 - Window Open (Listening) 1.548 - Resume ASR 2.389 - Onset Detected (0.840) 4.301 - Speech Ended (1.911) 4.334 - SIX ONE 4.335 - Accept Pattern: 1 4.355 - Suspend ASR 11:31 – 10 May 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number.

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.004 - Suspend ASR 1.924 - Prompt+Tone End 1.925 - Window Open (Listening) 1.926 - Resume ASR 2.672 - Onset Detected (0.746) 6.368 - Speech Ended (3.696) 6.424 - OH THREE SIX TWO THREE 6.425 - Accept Pattern: 1 6.484 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.881 - Prompt+Tone End 1.881 - Window Open (Listening) 1.883 - Resume ASR 2.355 - Onset Detected (0.472) 6.834 - Speech Ended (4.478) 6.895 - SEVEN TWO FOUR FIVE NINE FIVE 6.896 - Accept Pattern: 1 6.953 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.543 - Prompt+Tone End 1.543 - Window Open (Listening) 1.545 - Resume ASR 2.413 - Onset Detected (0.868) 6.119 - Speech Ended (3.705) 6.177 - TWO TWO NINE TWO FIVE 6.178 - Accept Pattern: 1 6.230 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.554 - Prompt+Tone End 1.554 - Window Open (Listening) 1.556 - Resume ASR 1.936 - Onset Detected (0.379) 5.133 - Speech Ended (3.197) 5.184 - TWO EIGHT TWO 5.184 - Accept Pattern: 1 5.217 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.531 - Prompt+Tone End 1.531 - Window Open (Listening) 1.532 - Resume ASR 2.258 - Onset Detected (0.726) 4.690 - Speech Ended (2.431) 4.730 - NINE EIGHT SEVEN 4.730 - Accept Pattern: 1 4.768 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.567 - Prompt+Tone End 1.568 - Window Open (Listening) 1.569 - Resume ASR 2.379 - Onset Detected (0.809) 5.859 - Speech Ended (3.479) 5.919 - TWO FOUR SEVEN EIGHT OH 5.919 - Accept Pattern: 1 5.956 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.548 - Prompt+Tone End 1.549 - Window Open (Listening) 1.550 - Resume ASR 1.686 - Onset Detected (0.136) 1.689 - SPOKE-TOO-SOON 1.712 - Suspend ASR 2.232 - Resume ASR (0.520) 2.322 - Onset Detected (0.089) 2.324 - SPOKE-TOO-SOON 2.327 - Suspend ASR 2.327 - Resume ASR (0.000) 7.058 - Onset Detected (4.731) 7.557 - Speech timeout 7.558 - Pattern: 5c 7.559 - Full Retry 7.560 - Prompt+Tone Start 7.562 - Suspend ASR 9.655 - Prompt+Tone End 9.655 - Window Open (Listening) 9.656 - Resume ASR 11.025 - Onset Detected (1.368) 13.755 - Speech Ended (2.730) 13.805 - ONE TWO OH FIVE SEVEN 13.806 - Accept Pattern: 1 13.830 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.566 - Prompt+Tone End 1.566 - Window Open (Listening) 1.568 - Resume ASR 2.431 - Onset Detected (0.862) 4.171 - Speech Ended (1.740) 4.199 - SIX ONE 4.200 - Accept Pattern: 1 4.249 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.555 - Prompt+Tone End 1.555 - Window Open (Listening) 1.556 - Resume ASR 2.082 - Onset Detected (0.525) 5.350 - Speech Ended (3.268) 5.401 - TWO ONE EIGHT TWO 5.401 - Accept Pattern: 1 5.432 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.547 - Prompt+Tone End 1.547 - Window Open (Listening) 1.549 - Resume ASR 2.104 - Onset Detected (0.555) 4.400 - Speech Ended (2.295) 4.440 - EIGHT TWO TWO 4.441 - Accept Pattern: 1 4.486 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.543 - Prompt+Tone End 1.544 - Window Open (Listening) 1.545 - Resume ASR 2.272 - Onset Detected (0.726) 4.702 - Speech Ended (2.430) 4.741 - SIX THREE NINE 4.742 - Accept Pattern: 1 4.780 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.537 - Prompt+Tone End 1.538 - Window Open (Listening) 1.539 - Resume ASR 2.385 - Onset Detected (0.845) 5.254 - Speech Ended (2.869) 5.298 - TWO OH EIGHT FOUR ONE 5.298 - Accept Pattern: 1 5.328 - Suspend ASR 11:33 – 10 May 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number.

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.004 - Suspend ASR 1.923 - Prompt+Tone End 1.923 - Window Open (Listening) 1.924 - Resume ASR 2.739 - Onset Detected (0.814) 4.468 - Speech Ended (1.729) 4.498 - SIX ONE 4.500 - Accept Pattern: 1 4.517 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.901 - Prompt+Tone End 1.901 - Window Open (Listening) 1.903 - Resume ASR 2.052 - Onset Detected (0.148)

2.055 - SPOKE-TOO-SOON 2.109 - Suspend ASR 2.628 - Resume ASR (0.518) 7.055 - Onset Detected (4.426) 7.911 - Speech timeout 7.911 - Pattern: 5c 7.913 - Full Retry 7.914 - Prompt+Tone Start 7.916 - Suspend ASR 10.008 - Prompt+Tone End 10.008 - Window Open (Listening) 10.009 - Resume ASR 10.773 - Onset Detected (0.763) 13.835 - Speech Ended (3.061) 13.880 - NINE EIGHT SEVEN 13.880 - Accept Pattern: 1 13.927 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.562 - Prompt+Tone End 1.562 - Window Open (Listening) 1.564 - Resume ASR 2.154 - Onset Detected (0.589) 5.483 - Speech Ended (3.329) 5.530 - SIX THREE NINE 5.530 - Accept Pattern: 1 5.567 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.558 - Prompt+Tone End 1.558 - Window Open (Listening) 1.560 - Resume ASR 2.407 - Onset Detected (0.846) 7.001 - Speech Ended (4.594) 7.074 - TWO THREE NINE ZERO FIVE 7.075 - Accept Pattern: 1 7.096 - Suspend ASR 7.567 - Pattern: 5b 7.569 - Full Retry 7.571 - Prompt+Tone Start 7.573 - Suspend ASR 9.664 - Prompt+Tone End 9.665 - Window Open (Listening) 9.666 - Resume ASR 10.207 - Onset Detected (0.540) 13.950 - Speech Ended (3.743) 14.008 - TWO THREE NINE TWO FIVE 14.009 - Accept Pattern: 1 14.053 - Suspend ASR 15.672 - Pattern: 5b 15.672 - Give Up 15.674 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.567 - Prompt+Tone End 1.567 - Window Open (Listening) 1.569 - Resume ASR 2.321 - Onset Detected (0.752) 4.753 - Speech Ended (2.431) 4.794 - TWO FOUR SEVEN 4.795 - Accept Pattern: 1 4.846 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.550 - Prompt+Tone End 1.551 - Window Open (Listening) 1.552 - Resume ASR 2.139 - Onset Detected (0.587) 3.801 - Speech Ended (1.662) 3.832 - THREE TWO 3.833 - Accept Pattern: 1 3.892 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.549 - Prompt+Tone End 1.549 - Window Open (Listening) 1.551 - Resume ASR 2.137 - Onset Detected (0.585) 5.199 - Speech Ended (3.062) 5.254 - EIGHT TWO TWO 5.254 - Accept Pattern: 1 5.297 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.537 - Prompt+Tone End 1.537 - Window Open (Listening) 1.539 - Resume ASR 2.120 - Onset Detected (0.581) 5.252 - Speech Ended (3.131) 5.305 - TWO EIGHT TWO 5.306 - Accept Pattern: 1 5.326 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.535 - Prompt+Tone End 1.536 - Window Open (Listening) 1.537 - Resume ASR 2.206 - Onset Detected (0.669) 5.018 - Speech Ended (2.812) 5.069 - TWO FIVE ZERO 5.070 - Accept Pattern: 1 5.113 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.537 - Prompt+Tone End 1.538 - Window Open (Listening) 1.539 - Resume ASR 2.254 - Onset Detected (0.715) 4.934 - Speech Ended (2.680) 4.978 - TWO EIGHT FOUR ONE 4.978 - Accept Pattern: 1 5.030 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.553 - Prompt+Tone End 1.554 - Window Open (Listening) 1.555 - Resume ASR 1.888 - Onset Detected (0.332) 6.749 - Speech Ended (4.861) 6.819 - SEVEN TWO OH FIVE NINE FIVE 6.819 - Accept Pattern: 1 6.836 - Suspend ASR 7.562 - Pattern: 5b 7.563 - Full Retry 7.565 - Prompt+Tone Start 7.567 - Suspend ASR 9.702 - Prompt+Tone End 9.703 - Window Open (Listening) 9.704 - Resume ASR 10.474 - Onset Detected (0.770) 15.069 - Speech Ended (4.594) 15.134 - SEVEN TWO OH FIVE NINE FIVE 15.135 - Accept Pattern: 1 15.156 - Suspend ASR 15.711 - Pattern: 5b 15.711 - Give Up 15.714 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.537 - Prompt+Tone End 1.538 - Window Open (Listening) 1.539 - Resume ASR 1.868 - Onset Detected (0.329) 6.351 - Speech Ended (4.482) 6.415 - OH EIGHT SIX TWO TWO 6.416 - Accept Pattern: 1 6.436 - Suspend ASR

The log files for participant with PID=30

12:06 – 28 June 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number.

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.003 - Suspend ASR 1.921 - Prompt+Tone End 1.922 - Window Open (Listening) 1.923 - Resume ASR 2.194 - Onset Detected (0.271) 3.803 - Speech Ended (1.608) 3.829 - SIX ONE 3.830 - Accept Pattern: 1 3.878 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.892 - Prompt+Tone End 1.893 - Window Open (Listening) 1.894 - Resume ASR 2.404 - Onset Detected (0.510) 3.984 - Speech Ended (1.580) 4.015 - SEVEN 4.016 - Accept Pattern: 1 4.063 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.536 - Prompt+Tone End 1.536 - Window Open (Listening) 1.537 - Resume ASR 1.761 - Onset Detected (0.223) 3.549 - Speech Ended (1.788) 3.583 - SIX THREE 3.584 - Accept Pattern: 1 3.621 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.554 - Prompt+Tone End 1.554 - Window Open (Listening) 1.555 - Resume ASR 1.986 - Onset Detected (0.430) 4.285 - Speech Ended (2.298) 4.332 - ZERO THREE 4.332 - Accept Pattern: 1 4.364 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.547 - Prompt+Tone End 1.547 - Window Open (Listening) 1.548 - Resume ASR 1.853 - Onset Detected (0.304) 3.517 - Speech Ended (1.664) 3.545 - ZERO FOUR SEVEN 3.546 - Accept Pattern: 1 3.589 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.555 - Prompt+Tone End 1.555 - Window Open (Listening) 1.556 - Resume ASR 1.986 - Onset Detected (0.430) 3.266 - Speech Ended (1.279) 3.292 - THREE TWO 3.293 - Accept Pattern: 1 3.342 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.557 - Prompt+Tone End 1.558 - Window Open (Listening) 1.559 - Resume ASR 2.074 - Onset Detected (0.514) 3.737 - Speech Ended (1.663) 3.768 - EIGHT TWO TWO 3.769 - Accept Pattern: 1 3.813 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.553 - Prompt+Tone End 1.554 - Window Open (Listening) 1.555 - Resume ASR 2.020 - Onset Detected (0.465) 4.452 - Speech Ended (2.431) 4.497 - ZERO ONE 4.498 - Accept Pattern: 1 4.534 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.557 - Prompt+Tone End 1.557 - Window Open (Listening) 1.558 - Resume ASR 1.955 - Onset Detected (0.397) 4.333 - Speech Ended (2.377) 4.376 - ONE TWO FIVE SEVEN 4.376 - Accept Pattern: 1 4.410 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.550 - Prompt+Tone End 1.550 - Window Open (Listening) 1.551 - Resume ASR 2.191 - Onset Detected (0.639)

4.067 - Speech Ended (1.876) 4.102 - TWO EIGHT ONE 4.103 - Accept Pattern: 1 4.148 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.554 - Prompt+Tone End 1.555 - Window Open (Listening) 1.556 - Resume ASR 1.686 - Onset Detected (0.130) 1.689 - SPOKE-TOO-SOON 1.719 - Suspend ASR 2.238 - Resume ASR (0.519) 3.304 - Onset Detected (1.065) 7.020 - Speech Ended (3.716) 7.075 - SEVEN EXIT ZERO OH FOUR 7.076 - Accept Pattern: 2 7.092 - Suspend ASR 7.563 - Speech timeout 7.563 - Pattern: 5c 7.564 - Full Retry 7.566 - Prompt+Tone Start 7.568 - Suspend ASR 9.661 - Prompt+Tone End 9.662 - Window Open (Listening) 9.663 - Resume ASR 10.105 - Onset Detected (0.441) 13.169 - Speech Ended (3.064) 13.225 - SEVEN ZERO FOUR 13.226 - Accept Pattern: 1 13.281 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.567 - Prompt+Tone End 1.568 - Window Open (Listening) 1.569 - Resume ASR 2.005 - Onset Detected (0.435) 4.583 - Speech Ended (2.578) 4.627 - FOUR THREE SIX TWO THREE 4.628 - Accept Pattern: 1 4.676 - Suspend ASR 12:09 – 28 June 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number.

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.004 - Suspend ASR 1.918 - Prompt+Tone End 1.918 - Window Open (Listening) 1.920 - Resume ASR 2.590 - Onset Detected (0.670) 5.517 - Speech Ended (2.926) 5.570 - FOUR TWO SIX TWO THREE 5.571 - Accept Pattern: 1 5.624 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.888 - Prompt+Tone End 1.888 - Window Open (Listening) 1.890 - Resume ASR 2.106 - Onset Detected (0.216) 4.783 - Speech Ended (2.677) 4.838 - TWO FOUR 4.839 - Accept Pattern: 1 4.869 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.542 - Prompt+Tone End 1.543 - Window Open (Listening) 1.544 - Resume ASR 2.276 - Onset Detected (0.731) 4.569 - Speech Ended (2.293) 4.612 - ZERO THREE NINE ZERO 4.612 - Accept Pattern: 1 4.651 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.541 - Prompt+Tone End 1.542 - Window Open (Listening) 1.543 - Resume ASR 2.189 - Onset Detected (0.645) 4.233 - Speech Ended (2.044) 4.276 - ZERO ONE ZERO 4.276 - Accept Pattern: 1 4.309 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.558 - Prompt+Tone End 1.559 - Window Open (Listening) 1.560 - Resume ASR 1.986 - Onset Detected (0.426) 3.849 - Speech Ended (1.863) 3.882 - NINE EIGHT SEVEN 3.882 - Accept Pattern: 1 3.943 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.549 - Prompt+Tone End 1.549 - Window Open (Listening) 1.551 - Resume ASR 2.021 - Onset Detected (0.470) 3.554 - Speech Ended (1.533) 3.583 - ZERO FOUR SEVEN 3.584 - Accept Pattern: 1 3.634 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.541 - Prompt+Tone End 1.541 - Window Open (Listening) 1.542 - Resume ASR 2.270 - Onset Detected (0.727) 4.682 - Speech Ended (2.412) 4.726 - ONE TWO FIVE SEVEN 4.726 - Accept Pattern: 1 4.777 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.550 - Prompt+Tone End 1.551 - Window Open (Listening) 1.552 - Resume ASR 2.019 - Onset Detected (0.466)

3.303 - Speech Ended (1.284) 3.327 - SIX ONE 3.328 - Accept Pattern: 1 3.380 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.553 - Prompt+Tone End 1.554 - Window Open (Listening) 1.555 - Resume ASR 2.158 - Onset Detected (0.602) 4.200 - Speech Ended (2.041) 4.244 - ZERO ONE TWO OH 4.244 - Accept Pattern: 1 4.278 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.552 - Prompt+Tone End 1.552 - Window Open (Listening) 1.553 - Resume ASR 2.273 - Onset Detected (0.719) 3.720 - Speech Ended (1.447) 3.748 - EIGHT TWO TWO 3.749 - Accept Pattern: 1 3.808 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.531 - Prompt+Tone End 1.531 - Window Open (Listening) 1.532 - Resume ASR 2.007 - Onset Detected (0.474) 3.368 - Speech Ended (1.361) 3.396 - SIX 3.397 - Accept Pattern: 1 3.445 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.551 - Prompt+Tone End 1.552 - Window Open (Listening) 1.553 - Resume ASR 2.407 - Onset Detected (0.854) 4.069 - Speech Ended (1.661) 4.100 - TWO EIGHT ONE 4.101 - Accept Pattern: 1 4.149 - Suspend ASR 12:11 – 28 June 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number.

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.003 - Suspend ASR 1.947 - Prompt+Tone End 1.947 - Window Open (Listening) 1.949 - Resume ASR 2.840 - Onset Detected (0.891) 5.142 - Speech Ended (2.302) 5.192 - FOUR TWO THREE 5.192 - Accept Pattern: 1 5.226 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.892 - Prompt+Tone End 1.893 - Window Open (Listening) 1.894 - Resume ASR 2.490 - Onset Detected (0.596) 5.424 - Speech Ended (2.933) 5.481 - TWO FOUR 5.482 - Accept Pattern: 1 5.513 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.554 - Prompt+Tone End 1.554 - Window Open (Listening) 1.556 - Resume ASR 2.105 - Onset Detected (0.549) 3.636 - Speech Ended (1.530) 3.664 - EIGHT TWO 3.665 - Accept Pattern: 1 3.725 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.549 - Prompt+Tone End 1.549 - Window Open (Listening) 1.551 - Resume ASR 2.228 - Onset Detected (0.676) 4.274 - Speech Ended (2.045) 4.312 - ZERO ONE TWO 4.313 - Accept Pattern: 1 4.359 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.550 - Prompt+Tone End 1.551 - Window Open (Listening) 1.552 - Resume ASR 2.355 - Onset Detected (0.803) 3.760 - Speech Ended (1.404) 3.790 - SIX 3.791 - Accept Pattern: 1 3.849 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.549 - Prompt+Tone End 1.550 - Window Open (Listening) 1.551 - Resume ASR 2.354 - Onset Detected (0.802) 4.476 - Speech Ended (2.122) 4.517 - ONE FIVE SEVEN 4.518 - Accept Pattern: 1 4.573 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.544 - Prompt+Tone End 1.545 - Window Open (Listening) 1.546 - Resume ASR 2.259 - Onset Detected (0.713) 3.927 - Speech Ended (1.667) 3.960 - SEVEN 3.961 - Accept Pattern: 1 4.013 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.545 - Prompt+Tone End 1.546 - Window Open (Listening) 1.547 - Resume ASR 2.257 - Onset Detected (0.709) 4.551 - Speech Ended (2.294) 4.596 - ZERO THREE OH 4.597 - Accept Pattern: 1 4.654 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.544 - Prompt+Tone End 1.544 - Window Open (Listening) 1.545 - Resume ASR 2.257 - Onset Detected (0.711) 3.917 - Speech Ended (1.660) 3.949 - TWO EIGHT FOUR ONE 3.950 - Accept Pattern: 1 3.968 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.544 - Prompt+Tone End 1.544 - Window Open (Listening) 1.546 - Resume ASR 2.321 - Onset Detected (0.775) 3.603 - Speech Ended (1.281) 3.629 - THREE ZERO 3.630 - Accept Pattern: 1 3.671 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.546 - Prompt+Tone End 1.546 - Window Open (Listening) 1.548 - Resume ASR 2.156 - Onset Detected (0.608) 3.799 - Speech Ended (1.643) 3.832 - ZERO FOUR SEVEN 3.833 - Accept Pattern: 1 3.886 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.543 - Prompt+Tone End 1.544 - Window Open (Listening) 1.546 - Resume ASR 2.274 - Onset Detected (0.728) 3.433 - Speech Ended (1.158) 3.457 - SIX ONE 3.459 - Accept Pattern: 1 3.500 - Suspend ASR

The log files for participant with PID=31

12:27 – 28 June 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number.

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.004 - Suspend ASR 1.921 - Prompt+Tone End 1.921 - Window Open (Listening) 1.923 - Resume ASR 3.225 - Onset Detected (1.302) 6.002 - Speech Ended (2.777) 6.051 - FOUR SIX ZERO THREE 6.051 - Accept Pattern: 1 6.096 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.878 - Prompt+Tone End 1.878 - Window Open (Listening) 1.880 - Resume ASR 2.637 - Onset Detected (0.757) 6.139 - Speech Ended (3.502) 6.198 - SEVEN OH FOUR FIVE NINE 6.199 - Accept Pattern: 1 6.224 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.547 - Prompt+Tone End 1.547 - Window Open (Listening) 1.549 - Resume ASR 2.218 - Onset Detected (0.669) 4.001 - Speech Ended (1.783) 4.032 - EIGHT OH TWO 4.033 - Accept Pattern: 1 4.057 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.558 - Prompt+Tone End 1.558 - Window Open (Listening) 1.560 - Resume ASR 2.138 - Onset Detected (0.578) 4.567 - Speech Ended (2.429) 4.612 - ZERO ONE 4.613 - Accept Pattern: 1 4.667 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.540 - Prompt+Tone End 1.540 - Window Open (Listening) 1.542 - Resume ASR 2.505 - Onset Detected (0.963) 4.301 - Speech Ended (1.795) 4.337 - SIX 4.338 - Accept Pattern: 1 4.394 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.550 - Prompt+Tone End 1.550 - Window Open (Listening) 1.552 - Resume ASR 2.439 - Onset Detected (0.887) 4.699 - Speech Ended (2.259) 4.744 - ONE TWO SEVEN 4.744 - Accept Pattern: 1 4.787 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.544 - Prompt+Tone End 1.545 - Window Open (Listening) 1.547 - Resume ASR 2.387 - Onset Detected (0.840) 4.165 - Speech Ended (1.777) 4.197 - NINE EIGHT SEVEN 4.198 - Accept Pattern: 1 4.226 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.552 - Prompt+Tone End 1.552 - Window Open (Listening) 1.554 - Resume ASR 2.165 - Onset Detected (0.610) 4.945 - Speech Ended (2.780) 4.998 - ZERO NINE OH 4.998 - Accept Pattern: 1 5.044 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.551 - Prompt+Tone End 1.551 - Window Open (Listening) 1.553 - Resume ASR 2.356 - Onset Detected (0.803) 4.400 - Speech Ended (2.043) 4.441 - TWO FOUR ONE 4.442 - Accept Pattern: 1 4.489 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.546 - Prompt+Tone End 1.546 - Window Open (Listening) 1.548 - Resume ASR 2.272 - Onset Detected (0.724) 3.668 - Speech Ended (1.396) 3.698 - ZERO 3.699 - Accept Pattern: 1 3.759 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.549 - Prompt+Tone End 1.550 - Window Open (Listening) 1.551 - Resume ASR 2.254 - Onset Detected (0.702) 4.182 - Speech Ended (1.928) 4.218 - ZERO ONE SEVEN 4.219 - Accept Pattern: 1 4.274 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.547 - Prompt+Tone End 1.548 - Window Open (Listening) 1.549 - Resume ASR 2.138 - Onset Detected (0.588) 3.534 - Speech Ended (1.396) 3.561 - SIX ONE 3.562 - Accept Pattern: 1 3.589 - Suspend ASR

12:29 – 28 June 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number.

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.004 - Suspend ASR 1.946 - Prompt+Tone End 1.947 - Window Open (Listening) 1.948 - Resume ASR 2.703 - Onset Detected (0.754) 3.975 - Speech Ended (1.272) 4.002 - SIX ONE 4.003 - Accept Pattern: 1 4.031 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.895 - Prompt+Tone End 1.896 - Window Open (Listening) 1.897 - Resume ASR 2.697 - Onset Detected (0.799) 4.356 - Speech Ended (1.659) 4.386 - NINE EIGHT SEVEN 4.387 - Accept Pattern: 1 4.405 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.550 - Prompt+Tone End 1.550 - Window Open (Listening) 1.552 - Resume ASR 2.265 - Onset Detected (0.712) 3.794 - Speech Ended (1.528) 3.825 - SIX EIGHT NINE 3.826 - Accept Pattern: 1 3.845 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.554 - Prompt+Tone End 1.554 - Window Open (Listening) 1.556 - Resume ASR 2.181 - Onset Detected (0.624) 4.612 - Speech Ended (2.431) 4.658 - ZERO THREE NINE NINE 4.659 - Accept Pattern: 1 4.706 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.545 - Prompt+Tone End 1.546 - Window Open (Listening) 1.547 - Resume ASR 2.211 - Onset Detected (0.664) 3.751 - Speech Ended (1.539) 3.779 - ZERO ONE SEVEN 3.780 - Accept Pattern: 1 3.799 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.552 - Prompt+Tone End 1.552 - Window Open (Listening) 1.553 - Resume ASR 2.302 - Onset Detected (0.748) 3.659 - Speech Ended (1.357) 3.686 - THREE ZERO 3.687 - Accept Pattern: 1 3.722 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.551 - Prompt+Tone End 1.552 - Window Open (Listening) 1.553 - Resume ASR 2.170 - Onset Detected (0.616) 3.585 - Speech Ended (1.414) 3.610 - EIGHT TWO 3.612 - Accept Pattern: 1 3.637 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.544 - Prompt+Tone End 1.544 - Window Open (Listening) 1.546 - Resume ASR 2.418 - Onset Detected (0.872) 4.466 - Speech Ended (2.048) 4.505 - ZERO NINE 4.506 - Accept Pattern: 1 4.522 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.548 - Prompt+Tone End 1.548 - Window Open (Listening) 1.550 - Resume ASR 2.557 - Onset Detected (1.007) 4.335 - Speech Ended (1.778) 4.375 - ONE SEVEN 4.376 - Accept Pattern: 1 4.400 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.540 - Prompt+Tone End 1.541 - Window Open (Listening) 1.542 - Resume ASR 2.537 - Onset Detected (0.995) 4.333 - Speech Ended (1.795) 4.369 - OH EIGHT FOUR ONE 4.370 - Accept Pattern: 1 4.391 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.550 - Prompt+Tone End 1.550 - Window Open (Listening) 1.552 - Resume ASR 2.172 - Onset Detected (0.619)

4.982 - Speech Ended (2.810) 5.034 - SEVEN ZERO ONE THREE 5.035 - Accept Pattern: 1 5.086 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.543 - Prompt+Tone End 1.544 - Window Open (Listening) 1.545 - Resume ASR 2.371 - Onset Detected (0.825) 4.416 - Speech Ended (2.045) 4.463 - FOUR 4.464 - Accept Pattern: 1 4.523 - Suspend ASR 12:32 – 28 June 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number.

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.004 - Suspend ASR 1.929 - Prompt+Tone End 1.929 - Window Open (Listening) 1.931 - Resume ASR 2.805 - Onset Detected (0.874) 4.849 - Speech Ended (2.043) 4.898 - ONE EIGHT EXIT ONE 4.899 - Accept Pattern: 1 4.952 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.883 - Prompt+Tone End 1.884 - Window Open (Listening) 1.885 - Resume ASR 2.560 - Onset Detected (0.675) 5.200 - Speech Ended (2.639) 5.252 - SEVEN ZERO ONE NINE 5.252 - Accept Pattern: 1 5.291 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.549 - Prompt+Tone End 1.549 - Window Open (Listening) 1.551 - Resume ASR 2.640 - Onset Detected (1.088) 4.618 - Speech Ended (1.977) 4.658 - THREE 4.659 - Accept Pattern: 1 4.699 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.540 - Prompt+Tone End 1.541 - Window Open (Listening) 1.543 - Resume ASR 2.472 - Onset Detected (0.929) 4.285 - Speech Ended (1.812) 4.322 - ZERO ONE EXIT OH 4.323 - Accept Pattern: 1 4.350 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.002 - Suspend ASR 1.541 - Prompt+Tone End 1.541 - Window Open (Listening) 1.543 - Resume ASR 2.508 - Onset Detected (0.964) 3.950 - Speech Ended (1.442) 3.980 - NINE EIGHT SEVEN 3.981 - Accept Pattern: 1 4.009 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.549 - Prompt+Tone End 1.550 - Window Open (Listening) 1.551 - Resume ASR 2.390 - Onset Detected (0.839) 3.916 - Speech Ended (1.526) 3.946 - ZERO ONE SEVEN 3.948 - Accept Pattern: 1 3.975 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.549 - Prompt+Tone End 1.550 - Window Open (Listening) 1.552 - Resume ASR 2.614 - Onset Detected (1.062) 4.251 - Speech Ended (1.636) 4.288 - NINE EXIT 4.289 - Accept Pattern: 1 4.316 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.540 - Prompt+Tone End 1.541 - Window Open (Listening) 1.543 - Resume ASR 2.172 - Onset Detected (0.629) 3.268 - Speech Ended (1.096) 3.293 - SIX ONE 3.295 - Accept Pattern: 1 3.326 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.550 - Prompt+Tone End 1.550 - Window Open (Listening) 1.552 - Resume ASR 2.306 - Onset Detected (0.754) 4.085 - Speech Ended (1.779) 4.121 - ZERO ONE EIGHT 4.122 - Accept Pattern: 1 4.144 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.554 - Prompt+Tone End 1.554 - Window Open (Listening) 1.556 - Resume ASR 2.363 - Onset Detected (0.806) 3.635 - Speech Ended (1.272) 3.662 - EIGHT 3.664 - Accept Pattern: 1 3.723 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.547 - Prompt+Tone End 1.547 - Window Open (Listening) 1.549 - Resume ASR 2.223 - Onset Detected (0.674) 3.615 - Speech Ended (1.392) 3.645 - SIX NINE 3.646 - Accept Pattern: 1 3.674 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.548 - Prompt+Tone End 1.548 - Window Open (Listening) 1.550 - Resume ASR 2.441 - Onset Detected (0.891) 3.967 - Speech Ended (1.526) 4.005 - EIGHT 4.006 - Accept Pattern: 1 4.060 - Suspend ASR

The log files for participant with PID=33

15:35 – 28 June 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number. — SmartWindow Parameters —- WinTimer : 6000.0 SilenceTimer : 175.0

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.004 - Suspend ASR 1.953 - Prompt+Tone End 1.954 - Window Open (Listening) 1.956 - Resume ASR 2.062 - Onset Detected (0.106) 2.065 - SPOKE-TOO-SOON 2.118 - Suspend ASR 2.638 - Resume ASR (0.519) 4.242 - Onset Detected (1.604) 5.636 - Speech Ended (1.394) 5.673 - NINE EIGHT 5.674 - Accept Pattern: 2 5.701 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.900 - Prompt+Tone End 1.901 - Window Open (Listening) 1.902 - Resume ASR 2.438 - Onset Detected (0.535) 4.353 - Speech Ended (1.915) 4.395 - OH 4.396 - Accept Pattern: 1 4.453 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.551 - Prompt+Tone End 1.551 - Window Open (Listening) 1.553 - Resume ASR 1.823 - Onset Detected (0.269) 3.232 - Speech Ended (1.409) 3.266 - TWO SEVEN 3.268 - Accept Pattern: 1 3.294 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.544 - Prompt+Tone End 1.544 - Window Open (Listening) 1.546 - Resume ASR 1.737 - Onset Detected (0.191) 2.749 - Speech Ended (1.011) 2.774 - NULL 6.723 - Onset Detected (5.176) 7.554 - Pattern: 5b 7.556 - Full Retry 7.558 - Prompt+Tone Start 7.561 - Suspend ASR 9.650 - Prompt+Tone End 9.651 - Window Open (Listening) 9.653 - Resume ASR 10.773 - Onset Detected (1.119) 11.950 - Speech Ended (1.177) 11.974 - THREE TWO 11.976 - Accept Pattern: 1 12.034 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.528 - Prompt+Tone End 1.528 - Window Open (Listening) 1.531 - Resume ASR 2.324 - Onset Detected (0.793) 3.334 - Speech Ended (1.009) 3.358 - EIGHT TWO 3.360 - Accept Pattern: 1 3.398 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.555 - Prompt+Tone End 1.556 - Window Open (Listening) 1.557 - Resume ASR 1.873 - Onset Detected (0.316) 3.399 - Speech Ended (1.526) 3.439 - TWO 3.440 - Accept Pattern: 1 3.469 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.542 - Prompt+Tone End 1.543 - Window Open (Listening) 1.545 - Resume ASR 1.772 - Onset Detected (0.227) 3.565 - Speech Ended (1.793) 3.611 - ONE 3.612 - Accept Pattern: 1 3.669 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.543 - Prompt+Tone End 1.544 - Window Open (Listening) 1.545 - Resume ASR 1.738 - Onset Detected (0.192) 3.001 - Speech Ended (1.262) 3.026 - ONE 3.027 - Accept Pattern: 1 3.074 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.548 - Prompt+Tone End 1.548 - Window Open (Listening) 1.550 - Resume ASR 1.651 - Onset Detected (0.100)

1.655 - SPOKE-TOO-SOON 1.713 - Suspend ASR 2.233 - Resume ASR (0.520) 2.422 - Onset Detected (0.188) 3.684 - Speech Ended (1.262) 3.713 - NINE 3.715 - Accept Pattern: 2 3.760 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.559 - Prompt+Tone End 1.560 - Window Open (Listening) 1.562 - Resume ASR 2.004 - Onset Detected (0.441) 3.780 - Speech Ended (1.776) 3.819 - FOUR 3.820 - Accept Pattern: 1 3.856 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.548 - Prompt+Tone End 1.549 - Window Open (Listening) 1.551 - Resume ASR 7.557 - Pattern: 4 7.560 - Full Retry 7.562 - Prompt+Tone Start 7.565 - Suspend ASR 9.654 - Prompt+Tone End 9.655 - Window Open (Listening) 9.657 - Resume ASR 15.664 - Pattern: 4 15.664 - Give Up 15.667 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.566 - Prompt+Tone End 1.566 - Window Open (Listening) 1.568 - Resume ASR 7.039 - Onset Detected (5.471) 7.576 - Pattern: 5b 7.579 - Full Retry 7.581 - Prompt+Tone Start 7.583 - Suspend ASR 9.672 - Prompt+Tone End 9.672 - Window Open (Listening) 9.674 - Resume ASR 11.390 - Onset Detected (1.716) 12.150 - Speech Ended (0.759) 12.167 - NULL 12.422 - Onset Detected (2.747) 13.182 - Speech Ended (0.760) 13.199 - NULL 13.319 - Onset Detected (3.645) 15:37 – 28 June 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number.

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.004 - Suspend ASR 1.921 - Prompt+Tone End 1.921 - Window Open (Listening) 1.924 - Resume ASR 2.086 - Onset Detected (0.162) 2.091 - SPOKE-TOO-SOON 2.127 - Suspend ASR 2.647 - Resume ASR (0.519) 2.856 - Onset Detected (0.209) 4.133 - Speech Ended (1.277) 4.159 - TWO 4.161 - Accept Pattern: 2 4.218 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.892 - Prompt+Tone End 1.892 - Window Open (Listening) 1.894 - Resume ASR 2.105 - Onset Detected (0.210) 4.150 - Speech Ended (2.044) 4.194 - ZERO FOUR EIGHT 4.195 - Accept Pattern: 1 4.233 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.550 - Prompt+Tone End 1.550 - Window Open (Listening) 1.553 - Resume ASR 1.722 - Onset Detected (0.168) 1.726 - SPOKE-TOO-SOON 1.756 - Suspend ASR 2.278 - Resume ASR (0.521) 2.621 - Onset Detected (0.343) 3.384 - Speech Ended (0.762) 3.402 - NINE 3.404 - Accept Pattern: 2 3.463 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.543 - Prompt+Tone End 1.544 - Window Open (Listening) 1.546 - Resume ASR 1.721 - Onset Detected (0.175) 3.115 - Speech Ended (1.393) 3.146 - TWO 3.148 - Accept Pattern: 1 3.204 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.553 - Prompt+Tone End 1.554 - Window Open (Listening) 1.556 - Resume ASR 1.719 - Onset Detected (0.163) 1.724 - SPOKE-TOO-SOON 1.761 - Suspend ASR 2.281 - Resume ASR (0.519) 7.566 - No-Speech Timeout 7.566 - Pattern: 3 7.568 - Quick Retry 7.569 - Suspend ASR 9.277 - Window Open (Listening) 9.279 - Resume ASR 9.622 - Onset Detected (0.343) 11.017 - Speech Ended (1.394) 11.048 - NINE EXIT 11.050 - Accept Pattern: 1 11.104 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.557 - Prompt+Tone End 1.558 - Window Open (Listening) 1.560 - Resume ASR 1.668 - Onset Detected (0.108) 1.673 - SPOKE-TOO-SOON 1.722 - Suspend ASR 2.242 - Resume ASR (0.520) 7.571 - No-Speech Timeout 7.571 - Pattern: 3 7.573 - Quick Retry 7.574 - Suspend ASR 9.195 - Window Open (Listening) 9.197 - Resume ASR 9.454 - Onset Detected (0.257) 10.982 - Speech Ended (1.528) 11.016 - ZERO 11.017 - Accept Pattern: 1 11.066 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.568 - Prompt+Tone End 1.568 - Window Open (Listening) 1.571 - Resume ASR 2.039 - Onset Detected (0.468) 3.432 - Speech Ended (1.393) 3.469 - ONE 3.471 - Accept Pattern: 1 3.525 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.548 - Prompt+Tone End 1.549 - Window Open (Listening) 1.551 - Resume ASR 2.058 - Onset Detected (0.507) 3.182 - Speech Ended (1.124) 3.210 - ONE 3.212 - Accept Pattern: 1 3.249 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.540 - Prompt+Tone End 1.540 - Window Open (Listening) 1.542 - Resume ASR 2.005 - Onset Detected (0.462) 3.665 - Speech Ended (1.660) 3.704 - ZERO ONE 3.706 - Accept Pattern: 1 3.752 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.543 - Prompt+Tone End 1.543 - Window Open (Listening) 1.545 - Resume ASR 2.003 - Onset Detected (0.458) 3.281 - Speech Ended (1.278) 3.311 - EIGHT TWO 3.312 - Accept Pattern: 1 3.371 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.529 - Prompt+Tone End 1.530 - Window Open (Listening) 1.532 - Resume ASR 2.005 - Onset Detected (0.472) 3.266 - Speech Ended (1.261) 3.298 - SIX THREE 3.300 - Accept Pattern: 1 3.357 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.547 - Prompt+Tone End 1.548 - Window Open (Listening) 1.550 - Resume ASR 2.189 - Onset Detected (0.639) 3.418 - Speech Ended (1.228) 3.445 - TWO EIGHT FOUR 3.446 - Accept Pattern: 1 3.504 - Suspend ASR 15:40 – 28 June 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number.

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.004 - Suspend ASR 1.954 - Prompt+Tone End 1.954 - Window Open (Listening) 1.957 - Resume ASR 2.462 - Onset Detected (0.505) 4.372 - Speech Ended (1.909) 4.410 - FOUR TWO TWO 4.411 - Accept Pattern: 1 4.464 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.889 - Prompt+Tone End 1.889 - Window Open (Listening) 1.891 - Resume ASR 2.357 - Onset Detected (0.466) 4.524 - Speech Ended (2.167) 4.571 - SEVEN ZERO FOUR 4.572 - Accept Pattern: 1 4.613 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.547 - Prompt+Tone End 1.548 - Window Open (Listening) 1.551 - Resume ASR 2.222 - Onset Detected (0.671) 3.498 - Speech Ended (1.276) 3.527 - EIGHT TWO TWO 3.529 - Accept Pattern: 1 3.588 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.549 - Prompt+Tone End 1.549 - Window Open (Listening) 1.551 - Resume ASR 1.835 - Onset Detected (0.283) 3.625 - Speech Ended (1.790) 3.664 - ZERO 3.666 - Accept Pattern: 1 3.718 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.551 - Prompt+Tone End 1.552 - Window Open (Listening) 1.554 - Resume ASR 1.971 - Onset Detected (0.417) 3.377 - Speech Ended (1.406) 3.405 - SIX 3.407 - Accept Pattern: 1 3.466 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.549 - Prompt+Tone End 1.550 - Window Open (Listening) 1.552 - Resume ASR 2.050 - Onset Detected (0.497)

3.444 - Speech Ended (1.394) 3.481 - ONE 3.482 - Accept Pattern: 1 3.503 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.535 - Prompt+Tone End 1.535 - Window Open (Listening) 1.537 - Resume ASR 2.037 - Onset Detected (0.499) 3.483 - Speech Ended (1.446) 3.518 - SEVEN 3.520 - Accept Pattern: 1 3.577 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.550 - Prompt+Tone End 1.551 - Window Open (Listening) 1.552 - Resume ASR 2.178 - Onset Detected (0.625) 3.991 - Speech Ended (1.812) 4.035 - TWO 4.036 - Accept Pattern: 1 4.060 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.545 - Prompt+Tone End 1.546 - Window Open (Listening) 1.548 - Resume ASR 2.004 - Onset Detected (0.455) 3.400 - Speech Ended (1.395) 3.431 - TWO EIGHT ONE 3.432 - Accept Pattern: 1 3.459 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.550 - Prompt+Tone End 1.550 - Window Open (Listening) 1.552 - Resume ASR 1.921 - Onset Detected (0.368) 3.065 - Speech Ended (1.144) 3.094 - TWO 3.096 - Accept Pattern: 1 3.122 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.546 - Prompt+Tone End 1.547 - Window Open (Listening) 1.549 - Resume ASR 1.651 - Onset Detected (0.102) 1.656 - SPOKE-TOO-SOON 1.711 - Suspend ASR 2.231 - Resume ASR (0.520) 7.560 - No-Speech Timeout 7.560 - Pattern: 3 7.562 - Quick Retry 7.563 - Suspend ASR 9.268 - Window Open (Listening) 9.271 - Resume ASR 9.854 - Onset Detected (0.583) 11.249 - Speech Ended (1.395) 11.284 - FOUR 11.286 - Accept Pattern: 1 11.308 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.563 - Prompt+Tone End 1.563 - Window Open (Listening) 1.565 - Resume ASR 1.938 - Onset Detected (0.373)

3.083 - Speech Ended (1.144) 3.110 - SIX 3.111 - Accept Pattern: 1 3.133 - Suspend ASR

The log files for participant with PID=08

13:30 – 30 March 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number. — SmartWindow Parameters —- WinTimer : 6000.0 SilenceTimer : 175.0

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.004 - Suspend ASR 1.935 - Prompt+Tone End 1.936 - Window Open (Listening) 1.938 - Resume ASR 2.302 - Onset Detected (0.363) 5.615 - Speech Ended (3.313) 5.671 - FOUR SIX TWO THREE 5.672 - Accept Pattern: 1 5.725 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.890 - Prompt+Tone End 1.891 - Window Open (Listening) 1.893 - Resume ASR 2.001 - Onset Detected (0.107) 2.006 - SPOKE-TOO-SOON 2.056 - Suspend ASR 2.575 - Resume ASR (0.519) 3.538 - Onset Detected (0.963) 6.735 - Speech Ended (3.196) 6.785 - SEVEN ZERO FOUR FIVE NINE 6.786 - Accept Pattern: 2 6.833 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.541 - Prompt+Tone End 1.541 - Window Open (Listening) 1.543 - Resume ASR 2.255 - Onset Detected (0.711) 3.915 - Speech Ended (1.659) 3.946 - EIGHT TWO 3.947 - Accept Pattern: 1 4.009 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.549 - Prompt+Tone End 1.549 - Window Open (Listening) 1.551 - Resume ASR 1.754 - Onset Detected (0.202) 4.050 - Speech Ended (2.295) 4.091 - OH ONE EIGHT ZERO 4.093 - Accept Pattern: 1 4.146 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.553 - Prompt+Tone

End 1.554 - Window Open (Listening) 1.556 - Resume ASR 1.754 - Onset Detected (0.197) 3.533 - Speech Ended (1.779) 3.566 - SIX THREE NINE 3.567 - Accept Pattern: 1 3.595 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.536 - Prompt+Tone End 1.536 - Window Open (Listening) 1.538 - Resume ASR 1.699 - Onset Detected (0.160) 1.703 - SPOKE-TOO-SOON 1.743 - Suspend ASR 2.263 - Resume ASR (0.520) 3.057 - Onset Detected (0.793) 5.098 - Speech Ended (2.040) 5.135 - ONE OH FIVE SEVEN 5.137 - Accept Pattern: 2 5.153 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.547 - Prompt+Tone End 1.548 - Window Open (Listening) 1.551 - Resume ASR 1.651 - Onset Detected (0.100) 1.655 - SPOKE-TOO-SOON 1.712 - Suspend ASR 2.233 - Resume ASR (0.521) 3.074 - Onset Detected (0.840) 4.849 - Speech Ended (1.775) 4.880 - NINE EIGHT EIGHT SEVEN 4.882 - Accept Pattern: 2 4.911 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.550 - Prompt+Tone End 1.551 - Window Open (Listening) 1.553 - Resume ASR 2.072 - Onset Detected (0.519) 4.733 - Speech Ended (2.661) 4.781 - ZERO NINE TWO OH 4.782 - Accept Pattern: 1 4.830 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.538 - Prompt+Tone End 1.538 - Window Open (Listening) 1.540 - Resume ASR 2.120 - Onset Detected (0.580) 4.050 - Speech Ended (1.929) 4.083 - FOUR FOUR ONE 4.085 - Accept Pattern: 1 4.134 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.541 - Prompt+Tone End 1.542 - Window Open (Listening) 1.543 - Resume ASR 2.256 - Onset Detected (0.712) 3.452 - Speech Ended (1.196) 3.473 - ZERO 3.475 - Accept Pattern: 1 3.495 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.551 - Prompt+Tone End 1.552 - Window Open (Listening) 1.553 - Resume ASR 1.750 - Onset Detected (0.196) 3.145 - Speech Ended (1.395) 3.170 - OH FOUR SEVEN 3.172 - Accept Pattern: 1 3.209 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.562 - Prompt+Tone End 1.563 - Window Open (Listening) 1.565 - Resume ASR 1.886 - Onset Detected (0.321) 3.165 - Speech Ended (1.278) 3.190 - SIX ONE 3.192 - Accept Pattern: 1 3.220 - Suspend ASR 13:32 – 30 March 2022 – TS+SW Parms.tssw

— ASR Parameters —- PostSpeech : 750.0 PreSpeech : 100.0 StartSpeech : 50.0 VadThreshold : 3.0 RemoveNoise : false RemoveSilence : true nBest : 0 ReturnNullHypothesis : true ReturnScores : false MaxSts : 8 Prompt : Say the first number.

———————— RUN MODE ———————— 0.000 - Event 1 START 0.001 - Prompt+Tone Start 0.005 - Suspend ASR 1.917 - Prompt+Tone End 1.918 - Window Open (Listening) 1.920 - Resume ASR 2.462 - Onset Detected (0.542) 5.017 - Speech Ended (2.554) 5.065 - FOUR TWO THREE 5.066 - Accept Pattern: 1 5.111 - Suspend ASR

— ASR Parameters —- Prompt : Say the next number.

0.000 - Event 2 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.902 - Prompt+Tone End 1.903 - Window Open (Listening) 1.905 - Resume ASR 2.235 - Onset Detected (0.329) 5.301 - Speech Ended (3.066) 5.349 - SEVEN TWO FOUR FIVE NINE 5.351 - Accept Pattern: 1 5.394 - Suspend ASR

— ASR Parameters —- Prompt : Next number?

0.000 - Event 3 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.548 - Prompt+Tone End 1.549 - Window Open (Listening) 1.551 - Resume ASR 2.088 - Onset Detected (0.537) 4.520 - Speech Ended (2.432) 4.563 - TWO THREE TWO 4.565 - Accept Pattern: 1 4.613 - Suspend ASR

— ASR Parameters —-

0.000 - Event 4 START 0.000 - Prompt+Tone Start 0.003 - Suspend ASR 1.543 - Prompt+Tone End 1.544 - Window Open (Listening) 1.546 - Resume ASR 2.216 - Onset Detected (0.670) 4.261 - Speech Ended (2.044) 4.303 - ZERO ONE TWO 4.305 - Accept Pattern: 1 4.353 - Suspend ASR

— ASR Parameters —-

0.000 - Event 5 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.554 - Prompt+Tone End 1.554 - Window Open (Listening) 1.556 - Resume ASR 2.101 - Onset Detected (0.544) 3.635 - Speech Ended (1.534) 3.664 - EIGHT SEVEN 3.665 - Accept Pattern: 1 3.723 - Suspend ASR

— ASR Parameters —-

0.000 - Event 6 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.544 - Prompt+Tone End 1.545 - Window Open (Listening) 1.547 - Resume ASR 2.093 - Onset Detected (0.545) 3.747 - Speech Ended (1.654) 3.778 - ZERO FOUR SEVEN 3.780 - Accept Pattern: 1 3.842 - Suspend ASR

— ASR Parameters —-

0.000 - Event 7 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.554 - Prompt+Tone End 1.555 - Window Open (Listening) 1.556 - Resume ASR 2.356 - Onset Detected (0.799) 4.266 - Speech Ended (1.910) 4.306 - ONE OH FIVE 4.307 - Accept Pattern: 1 4.363 - Suspend ASR

— ASR Parameters —-

0.000 - Event 8 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.547 - Prompt+Tone End 1.547 - Window Open (Listening) 1.550 - Resume ASR 2.092 - Onset Detected (0.542) 3.237 - Speech Ended (1.145) 3.264 - SIX 3.266 - Accept Pattern: 1 3.287 - Suspend ASR

— ASR Parameters —-

0.000 - Event 9 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.546 - Prompt+Tone End 1.546 - Window Open (Listening) 1.549 - Resume ASR 1.651 - Onset Detected (0.102) 1.655 - SPOKE-TOO-SOON 1.711 - Suspend ASR 2.230 - Resume ASR (0.519) 2.941 - Onset Detected (0.710) 4.980 - Speech Ended (2.039) 5.017 - ZERO ONE EIGHT TWO 5.019 - Accept Pattern: 2 5.080 - Suspend ASR

— ASR Parameters —-

0.000 - Event 10 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.548 - Prompt+Tone End 1.549 - Window Open (Listening) 1.551 - Resume ASR 2.089 - Onset Detected (0.537) 3.499 - Speech Ended (1.410) 3.527 - EIGHT TWO 3.529 - Accept Pattern: 1 3.591 - Suspend ASR

— ASR Parameters —-

0.000 - Event 11 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.547 - Prompt+Tone End 1.548 - Window Open (Listening) 1.550 - Resume ASR 2.094 - Onset Detected (0.544) 3.624 - Speech Ended (1.530) 3.652 - SIX THREE 3.654 - Accept Pattern: 1 3.717 - Suspend ASR

— ASR Parameters —-

0.000 - Event 12 START 0.000 - Prompt+Tone Start 0.004 - Suspend ASR 1.543 - Prompt+Tone End 1.544 - Window Open (Listening) 1.546 - Resume ASR 2.090 - Onset Detected (0.544) 3.871 - Speech Ended (1.781) 3.902 - FOUR FOUR 3.904 - Accept Pattern: 1 3.924 - Suspend ASR 13:34 – 30 March 2022 – TS+SW Parms.tssw

Appendix E

Post experiment questionnaire

1. Was it comfortable to speak to the machine?

2. How did you know when to speak?

3. Did you ever have to repeat your response?

4. (If yes on above question # 3) How did you know to do it?

5. Do you think that the machine recognised your voice correctly?

6. Was there ever a problem that the machine interrupted you at some point of time?

7. (If yes on above question) How did you know the machine is interrupting you and how you manage the interruption.

A question to know the general overview to know what kind of mental model the participant might have developed after the experiment.

Question-Ok, now let me ask you a general question, Do you have any idea about what rules or law have been implemented into the machine that in such a simple conversation like this, the machine is behaving in this particular manner?

This page was intentionally left blank.

Appendix F

Summary of turns taken by the participants during conversation with VUI equipped with half-duplex turn-taking protocol

Table F.1: Summary of Turns

Participant ID	Total Turns 1st-try/2nd-try	Total STS Pass: 1/2/3	Recovered STS Self-/Algorithm	Failed Event
p31	36/0	0/0/0	0/0	0
p15	36/2	1/1*/0	2/0	0
p16	36/2	2/0/0	2/0	0
P08	36/4	3/1/0	4/0	0
P30	36/1	1/0/0	0/1	0
P26	36/3	1/1/1	0/3	0
P23	36/4	2*/0/0	2/0	0
P24	36/8	3*/3*/0	5/1	0
P33	36/4	2/4/1	0/4	4
P14	36/4	6/0/2	1/4	4
P10	36/10	5/7/10	2/12	8
P12	36/5	4/7/1	1/4	1
P17	36/2	7/3/3	0/2	11
TOTALS	517	82	19/31	28

Appendix G

Likely Behaviours of the participants for Various turn-taking Conditions

Table G.1: Likely Behaviours for various conditions

#	Behaviour	conditions
1	Normal-expected-No STS	normal capture (second-most-desired behaviour)
2a		silence timeout (no speech) – very unlikely, predict zero occurrences
2b		silence timeout due to mistimed speech (too early, or embedded silence)
3	Expected reactions to STS (in order of frequency)	"Plough-through" (ignore or exhibit confusion with no effect on speaking behaviour)
4		abruptly stop speaking and wait for next step (relinquish the turn)
5		stop speaking briefly, and continue with utterance (keep the turn after consideration)
6		abruptly stop speaking and then start over from beginning (keep the turn after learning) – very likely after some period of learning (most-desired behaviour)
7	Expected Reactions to Tapering (can't predict frequency)	STS on first event followed by no STS on second and third (initial prompt-response entrainment) – more likely on Pass 1, diminishing over time increase in STS on events 4 through 12 (steady-state prompt-response rhythm)
8		increase in STS on events 4 through 12 (steady-state prompt-response rhythm) 1st-try/2nd-try STS pattern on quick retry
9		1st-try/2nd-try STS pattern on quick retry
10		general synchronization on prompt-response rhythm and/or rate of speech from Pass 1 to Pass 3.
11	Anomalous ASR events (not caused by user, can't predict frequency)	false STS from environment (no user speech) (sometimes followed by user speech)
12		false accept from environment (no user speech)
13		false accept due to embedded silence (user ploughing through)
14		speech timeout (user speech or noise or both)
15		concurrent timeout (known technology flaw)
16		undetected STS (known technology flaw)

Appendix H

Number of turns taken by the participants of the experiment group during comparative study

Table H.1: Number of errors per participant across the three passes (Experimental Group)

Participant ID	Errors - Pass 1	Errors - Pass 2	Errors - Pass 3
P23	4	0	0
P24	4	8	0
P26	2	3	4
P30	2	0	0
P31	0	0	0
P33	6	4	1
P08	3	1	0
P12	3	2	2
P14	2	1	4
P15	2	2	2
P16	2	2	2
P17	8	8	3
P28	6	5	5
P36	6	3	2
P37	4	2	0
P39	5	1	0
P34	3	1	0
P40	4	2	1

Appendix I

Number of turns taken by the participants of the control group during comparative study

Table I.1: Number of errors per participant across the three passes (Control Group)

Participant ID	Errors - Pass 1	Errors - Pass 2	Errors - Pass 3
P20	1	3	9
P21	8	17	1
P22	0	0	1
P25	6	3	0
P29	15	23	20
P32	9	8	14
P35	17	26	28
P11	2	0	0
P13	6	5	4
P18	14	4	0
P19	13	19	2
P05	17	28	26
P07	20	8	1
P09	6	4	7
P36	1	3	9
P37	15	23	20

References

Arora, S., Lu, Z., Chiu, C.-C., Pang, R., Watanabe, S., 2025. Talking turns: Benchmarking audio foundation models on turn-taking dynamics. arXiv preprint arXiv:2503.01174.

Balentine, B., 2007a. It's Better to Be a Good Machine Than a Bad Person: Speech Recognition and Other Exotic User Interfaces in the Twilight of the Jetsonian Age. ICMI Press.

Balentine, B., 2007b. It's Better to Be a Good Machine Than a Bad Person: Speech Recognition and Other Exotic User Interfaces in the Twilight of the Jetsonian Age. ICMI Press.

Baughan, A., Wang, X., Liu, A., Mercurio, A., Chen, J., Ma, X., 2023. A mixed-methods approach to understanding user trust after voice assistant failures. In: Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems. pp. 1–16.

Bell, L., Gustafson, J., Heldner, M., 2003. Prosodic adaptation in human-computer interaction. In: Proceedings of ICPHS. Vol. 3. Citeseer, pp. 833–836.

Beňuš, Š., Trnka, M., Kuric, E., Marták, L., Gravano, A., Hirschberg, J., Levitan, R., 2018. Prosodic entrainment and trust in human-computer interaction. In: Proceedings of the 9th International Conference on Speech Prosody. International Speech Communication Association Baixas, France, pp. 220–224.

Benus, s., et al., 2018. Prosodic entrainment and trust in human-computer interaction. In: Proceedings of the 9th International Conference on Speech Prosody. Baixas, France: International Speech Communication Association.

Beňuš, , 2014. Social aspects of entrainment in spoken interaction. Cognitive Computation 6, 802–813.

Blattner, M. M., Sumikawa, D. A., Greenberg, R. M., 1989. Earcons and icons: Their structure and common design principles. Human–Computer Interaction 4 (1), 11–44.

Borrie, S. A., Lubold, N., Pon-Barry, H., 2015. Disordered speech disrupts conversational entrainment: a study of acoustic-prosodic entrainment and communicative success in populations with communication challenges. Frontiers in psychology 6, 1187.

Braun, V., Clarke, V., 2006. Using thematic analysis in psychology. Qualitative research in psychology 3 (2), 77–101.

Brennan, E., 1996. Lexical choice and conceptual pacts in conversation. Journal of Experimental Psychology: Learning, Memory, and Cognition 22, 1482–1493.

Brennan, S. E., Clark, H. H., 1996. Conceptual pacts and lexical choice in conversation. Journal of experimental psychology: Learning, memory, and cognition 22 (6), 1482.

Bretan, M., Hoffman, G., Weinberg, G., 2015. Emotionally expressive dynamic physical behaviors in robots. International Journal of Human-Computer Studies 78, 1–16.

Brewster, S. A., Wright, P. C., Edwards, A. D., 1993. An evaluation of earcons for use in auditory human-computer interfaces. In: Proceedings of the INTERACT'93 and CHI'93 conference on Human factors in computing systems. pp. 222–227.

Chang, S.-y., Li, B., Sainath, T. N., Zhang, C., Strohman, T., Liang, Q., He, Y., 2022. Turn-taking prediction for natural conversational speech. arXiv preprint arXiv:2208.13321.

Chuklin, A., Severyn, A., Trippas, J., Alfonseca, E., Silen, H., Spina, D., 2018. Prosody modifications for question-answering in voice-only settings. arXiv preprint arXiv:1806.03957.

Chuklin, A., Severyn, A., Trippas, J. R., Alfonseca, E., Silen, H., Spina, D., 2019. Using audio transformations to improve comprehension in voice question answering. In: International Conference of the Cross-Language Evaluation Forum for European Languages. Springer, pp. 164–170.

Cohn, M., Liang, K.-H., Sarian, M., Zellou, G., Yu, Z., 2021. Speech rate adjustments in conversations with an amazon alexa socialbot. Frontiers in Communication 6, 671429.

Desai, S., Dubiel, M., Leiva, L. A., 2024. Examining humanness as a metaphor to design voice user interfaces. In: Proceedings of the 6th ACM Conference on Conversational User Interfaces. pp. 1–15.

Dingler, T., Lindsay, J., Walker, B. N., et al., 2008. Learnability of sound cues for environmental features: Auditory icons, earcons, spearcons, and speech. In: Proceedings of the 14th International Conference on Auditory Display, Paris, France. pp. 1–6.

Doddington, G., Liggett, W., Martin, A., Przybocki, M., Reynolds, D., 1998. Sheep, goats, lambs and wolves: A statistical analysis of speaker performance in the nist 1998 speaker recognition evaluation.

Ekstedt, E., Skantze, G., 2022. How much does prosody help turn-taking? investigations using voice activity projection models. arXiv preprint arXiv:2209.05161.

Furqan, A., Myers, C., Zhu, J., 2017. Learnability through adaptive discovery tools in voice user interfaces. In: Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems. pp. 1617–1623.

Gálvez, R. H., et al., 2020. An empirical study of the effect of acoustic-prosodic entrainment on the perceived trustworthiness of conversational avatars. Speech Communication 124, 46–67.

Gaver, W. W., 1986. Auditory icons: Using sound in computer interfaces. Human-Computer Interaction 2 (2), 167–177.
URL https://doi.org/10.1207/s15327051hci0202_3

Gaver, W. W., 1993. Synthesizing auditory icons. In: Proceedings of the INTERACT'93 and CHI'93 conference on Human factors in computing systems. pp. 228–235.

Gessinger, I., Möbius, B., Fakhar, N., Raveh, E., Steiner, I., 2019. A wizard-of-oz experiment to study phonetic accommodation in human-computer interaction. In: International Congress of Phonetic Sciences (ICPhS), Melbourne. pp. 1475–1479.

Giles, H., 1975. Speech style and social evaluation.

Giles, H., Coupland, J., Coupland, N., 1991. Contexts of accommodation: Developments in applied sociolinguistics. Vol. 10. Cambridge University Press.

Goetsu, S., Sakai, T., 2020. Different types of voice user interface failures may cause different degrees of frustration. arXiv preprint arXiv:2002.03582.

Gustafson, J., Larsson, A., Carlson, R., Hellman, K., 1997. How do system questions influence lexical choices in user answers? In: Eurospeech'97, 5th European Conference on Speech Communication and Technology: Rhodes, Greece, 22-25 September 1997. European Speech Communication Association (ESCA), pp. 2275–2278.

Heldner, M., Edlund, J., 2010. Pauses, gaps and overlaps in conversations. Journal of Phonetics 38 (4), 555–568.

Heldner, M., Edlund, J., Hirschberg, J. B., 2010. Pitch similarity in the vicinity of backchannels.

Hermann, T., Hunt, A., Neuhoff, J. G., et al., 2011. The sonification handbook. Vol. 1. Logos Verlag Berlin.

Huggins-Daines, D., Kumar, M., Chan, A., Black, A. W., Ravishankar, M., Rudnicky, A. I., 2006. Pocketsphinx: A free, real-time continuous speech recognition system for hand-held devices. In: 2006 IEEE international conference on acoustics speech and signal processing proceedings. Vol. 1. IEEE, pp. I–I.

Huiyang, S., Min, W., 2022. Improving interaction experience through lexical convergence: the prosocial effect of lexical alignment in human-human and human-computer interactions. International Journal of Human–Computer Interaction 38 (1), 28–41.

Jamshed, H., Nurain, N., Brewer, R. N., 2025. Designing accessible audio nudges for voice interfaces. In: Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems. pp. 1–16.

Jiang, J., Jeng, W., He, D., 2013. How do users respond to voice input errors? lexical and phonetic query reformulation in voice search. In: Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval. pp. 143–152.

Kennedy, A., Wilkes, A., Elder, L., Murray, W. S., 1988. Dialogue with machines. Cognition 30 (1), 37–72.

Kirschthaler, P., Porcheron, M., Fischer, J. E., 2020. What can i say? effects of discoverability in vuis on task performance and user experience. In: Proceedings of the 2nd Conference on Conversational User Interfaces. pp. 1–9.

Klein, A. M., Deutschländer, J., Kölln, K., Rauschenberger, M., Escalona, M. J., 2024. Exploring the context of use for voice user interfaces: Toward context-dependent user experience quality testing. Journal of Software: Evolution and Process 36 (7), e2618.

Ko, S., Kutchek, K., Zhang, Y., Jeon, M., 2022. Effects of non-speech auditory cues on control transition behaviors in semi-automated vehicles: Empirical study, modeling, and validation. International Journal of Human–Computer Interaction 38 (2), 185–200.

Kutchek, K., Jeon, M., 2019. Takeover and handover requests using non-speech auditory displays in semi-automated vehicles. In: Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems. pp. 1–6.

Lakin, J. L., Jefferis, V. E., Cheng, C. M., Chartrand, T. L., 2003. The chameleon effect as social glue: Evidence for the evolutionary significance of nonconscious mimicry. Journal of nonverbal behavior 27, 145–162.

Lei, Z., Ma, S., Li, H., Yang, Z., 2022. The impact of different types of auditory warnings on working memory. Frontiers in psychology 13, 780657.

Leviathan, Y., Matias, Y., 2018. Google duplex: An ai system for accomplishing real-world tasks over the phone. Google AI blog 8.

Levitan, R., 2014. Acoustic-prosodic entrainment in human-human and human-computer dialogue. Columbia University.

Levitan, R., 2020. Developing an integrated model of speech entrainment. In: Proceedings of the twenty-ninth international joint conference on artificial intelligence.

Levitan, R., Benus, S., Gálvez, R. H., Gravano, A., Savoretti, F., Trnka, M., Weise, A., Hirschberg, J., 2016. Implementing acoustic-prosodic entrainment in a conversational avatar. In: Interspeech. Vol. 16. San Francisco, CA, pp. 1166–1170.

Levitan, R., Gravano, A., Willson, L., Beňuš, Š., Hirschberg, J., Nenkova, A., 2012. Acoustic-prosodic entrainment and social behavior. In: Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human language technologies. pp. 11–19.

Levitan, R., Hirschberg, J. B., 2011. Measuring acoustic-prosodic entrainment with respect to multiple levels and dimensions.

Levitan, R., et al., 2015. Entrainment and turn-taking in human-human dialogue. In: 2015 AAAI spring symposium series.

Liesenfeld, A., Lopez, A., Dingemanse, M., 2023. The timing bottleneck: Why timing and overlap are mission-critical for conversational user interfaces, speech recognition and dialogue systems. arXiv preprint arXiv:2307.15493.

Lin, T.-E., Wu, Y., Huang, F., Si, L., Sun, J., Li, Y., 2022. Duplex conversation: Towards human-like interaction in spoken dialogue systems. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 3299–3308.

Lohse, M., Rohlfing, K. J., Wrede, B., Sagerer, G., 2008. "try something else!"—when users change their discursive behavior in human-robot interaction. In: 2008 IEEE International Conference on Robotics and Automation. IEEE, pp. 3481–3486.

Lubold, N., Pon-Barry, H., 2014. Acoustic-prosodic entrainment and rapport in collaborative learning dialogues. In: Proceedings of the 2014 ACM workshop on Multimodal Learning Analytics Workshop and Grand Challenge. pp. 5–12.

Lubold, N., Pon-Barry, H., Walker, E., 2015. Naturalness and rapport in a pitch adaptive learning companion. In: 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU). IEEE, pp. 103–110.

Luger, E., Sellen, A., 2016. " like having a really bad pa" the gulf between user expectation and experience of conversational agents. In: Proceedings of the 2016 CHI conference on human factors in computing systems. pp. 5286–5297.

McWilliams, T., Reimer, B., Mehler, B., Dobres, J., McAnulty, H., 2015. A secondary assessment of the impact of voice interface turn delays on driver attention and arousal in field conditions. In: Driving Assessment Conference. Vol. 8. University of Iowa.

Moore, R. K., 2017a. Appropriate voices for artefacts: some key insights. In: 1st International workshop on vocal interactivity in-and-between humans, animals and robots.

Moore, R. K., 2017b. Is spoken language all-or-nothing? implications for future speech-based human-machine interaction. Dialogues with Social Robots: Enablements, Analyses, and Evaluation, 281–291.

Mori, M., MacDorman, K. F., Kageki, N., 2012. The uncanny valley [from the field]. IEEE Robotics & automation magazine 19 (2), 98–100.

Myers, C., Furqan, A., Nebolsky, J., Caro, K., Zhu, J., 2018. Patterns for how users overcome obstacles in voice user interfaces. In: Proceedings of the 2018 CHI conference on human factors in computing systems. pp. 1–7.

Natale, M., 1975. Convergence of mean vocal intensity in dyadic communication as a function of social desirability. Journal of Personality and Social Psychology 32 (5), 790.

Nees, M. A., Liebman, E., 2023. Auditory icons, earcons, spearcons, and speech: A systematic review and meta-analysis of brief audio alerts in human-machine interfaces. Auditory Perception & Cognition 6 (3-4), 300–329.

Nees, M. A., Walker, B. N., 2011. Auditory displays for in-vehicle technologies. Reviews of human factors and ergonomics 7 (1), 58–99.

Ni, J., Young, T., Pandelea, V., Xue, F., Cambria, E., 2023. Recent advances in deep learning based dialogue systems: A systematic survey. Artificial intelligence review 56 (4), 3055–3155.

Niederhoffer, K. G., Pennebaker, J. W., 2002. Linguistic style matching in social interaction. Journal of Language and Social Psychology 21 (4), 337–360.

Oviatt, S., Darves, C., Coulston, R., 2004. Toward adaptive conversational interfaces: Modeling speech convergence with animated personas. ACM Transactions on Computer-Human Interaction (TOCHI) 11 (3), 300–328.

Paletz, S. B., Litman, D., Karuzis, V., Jones, K. M., Rahimi, Z., 2023. Speaking similarly: team personality composition and acoustic-prosodic entrainment. Small Group Research 54 (6), 860–898.

Pardo, J. S., 2006. On phonetic convergence during conversational interaction. The Journal of the Acoustical Society of America 119 (4), 2382–2393.

Pelikan, H. R., Broth, M., 2016. Why that nao? how humans adapt to a conventional humanoid robot in taking turns-at-talk. In: Proceedings of the 2016 CHI conference on human factors in computing systems. pp. 4921–4932.

Phukon, M., Shrivastava, A., Balentine, B., 2022. Can vui turn-taking entrain user behaviours? voice user interfaces that disallow overlapping speech present turn-taking challenges. In: Proceedings of the 13th Indian Conference on Human-Computer Interaction. pp. 1–16.

Pickering, M. J., Garrod, S., 2004. Toward a mechanistic psychology of dialogue. Behavioral and brain sciences 27 (2), 169–190.

Porcheron, M., Fischer, J. E., Reeves, S., Sharples, S., 2018. Voice interfaces in everyday life. In: proceedings of the 2018 CHI conference on human factors in computing systems. pp. 1–12.

Pyae, A., Joelsson, T. N., 2018. Investigating the usability and user experiences of voice user interface: a case of google home smart speaker. In: Proceedings of the 20th international conference on human-computer interaction with mobile devices and services adjunct. pp. 127–131.

Raux, A., 2008. Flexible turn-taking for spoken dialog systems. Language Technologies Institute, CMU Dec 12.

Raux, A., Eskenazi, M., 2009. A finite-state turn-taking model for spoken dialog systems. In: Proceedings of human language technologies: The 2009 annual conference of the North American chapter of the association for computational linguistics. pp. 629–637.

Raux, A., Eskenazi, M., 2012. Optimizing the turn-taking behavior of task-oriented spoken dialog systems. ACM Transactions on Speech and Language Processing (TSLP) 9 (1), 1–23.

Razavi, S. Z., Kane, B., Schubert, L. K., 2019. Investigating linguistic and semantic features for turn-taking prediction in open-domain human-computer conversation. In: INTERSPEECH. pp. 4140–4144.

Read, R., Belpaeme, T., 2012. How to use non-linguistic utterances to convey emotion in child-robot interaction. In: Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction. pp. 219–220.

Sacks, H., Schegloff, E. A., Jefferson, G., 1974. A simplest systematics for the organization of turn-taking for conversation. language 50 (4), 696–735.

Schlangen, D., Skantze, G., 2011. A general, abstract model of incremental dialogue processing. Dialogue & Discourse 2 (1), 83–111.

Sciuto, A., Saini, A., Forlizzi, J., Hong, J. I., 2018. " hey alexa, what's up?" a mixed-methods studies of in-home conversational agent usage. In: Proceedings of the 2018 designing interactive systems conference. pp. 857–868.

Sigtia, S., Bridle, J., Richards, H., Clark, P., Marchi, E., Garg, V., 2021. Progressive voice trigger detection: Accuracy vs latency. In: ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp. 6843–6847.

Skantze, G., 2007. Error handling in spoken dialogue systems-managing uncertainty, grounding and miscommunication. Gabriel Skantze.

Skantze, G., 2021. Turn-taking in conversational systems and human-robot interaction: a review. Computer Speech & Language 67, 101178.

Skantze, G., Irfan, B., 2025. Applying general turn-taking models to conversational human-robot interaction. In: 2025 20th ACM/IEEE International Conference on Human-Robot Interaction (HRI). IEEE, pp. 859–868.

Strategy Analytics, 2021. Global smart home market roaring back in 2021. https://www.businesswire.com/news/home/20210706005692/en/Strategy-Analytics-Global-Smart-Home-Market-Roaring-Back-in-2021, accessed: 2025-04-07.
URL https://www.businesswire.com/news/home/20210706005692/en/Strategy-Analytics-Global-Smart-Home-Market-Roaring-Back-in-2021

Takayama, L., Dooley, D., Ju, W., 2011. Expressing thought: improving robot readability with animation principles. In: Proceedings of the 6th international conference on Human-robot interaction. pp. 69–76.

Tannen, D., 2007. Talking voices: Repetition, dialogue, and imagery in conversational discourse. Vol. 26. Cambridge University Press.

Thomason, J., Nguyen, H. V., Litman, D., 2013. Prosodic entrainment and tutoring dialogue success. In: International conference on artificial intelligence in education. Springer, pp. 750–753.

Veluri, B., Peloquin, B. N., Yu, B., Gong, H., Gollakota, S., 2024. Beyond turn-based interfaces: Synchronous llms as full-duplex dialogue agents. arXiv preprint arXiv:2409.15594.

Wang, J., Li, Y., Yang, S., Dong, S., Li, J., 2023. Waiting experience: Optimization of feedback mechanism of voice user interfaces based on time perception. IEEE Access 11, 21241–21251.

Woodruff, A., Aoki, P. M., 2003. How push-to-talk makes talk less pushy. In: Proceedings of the 2003 ACM International Conference on Supporting Group Work. pp. 170–179.

Wu, C.-S., Hoi, S., Socher, R., Xiong, C., 2020. Tod-bert: Pre-trained natural language understanding for task-oriented dialogue. arXiv preprint arXiv:2004.06871.

Wynn, C. J., Barrett, T. S., Borrie, S. A., 2022. Rhythm perception, speaking rate entrainment, and conversational quality: A mediated model. Journal of Speech, Language, and Hearing Research 65 (6), 2187–2203.

Wynn, C. J., et al., 2023. Speech entrainment in adolescent conversations: A developmental perspective. Journal of Speech, Language, and Hearing Research, 1–19.

Zhang, B. J., Fitter, N. T., 2023. Nonverbal sound in human-robot interaction: A systematic review. ACM Transactions on Human-Robot Interaction 12 (4), 1–46.

Zhang, Q., Cheng, L., Deng, C., Chen, Q., Wang, W., Zheng, S., Liu, J., Yu, H., Tan, C., Du, Z., et al., 2024. Omniflatten: An end-to-end gpt model for seamless voice conversation. arXiv preprint arXiv:2410.17799.

Zhao, T., Black, A. W., Eskenazi, M., 2015. An incremental turn-taking model with active system barge-in for spoken dialog systems. In: Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue. pp. 42–50.