# NottReal: A tool for voice-based Wizard of Oz studies

Martin Porcheron
martin.porcheron@nottingham.ac.uk
Mixed Reality Lab
School of Computer Science
University of Nottingham, UK

Joel E. Fischer
joel.fischer@nottingham.ac.uk
Mixed Reality Lab
School of Computer Science
University of Nottingham, UK

Michel Valstar
michel.valstar@nottingham.ac.uk
Computer Vision/Mixed Reality Lab
School of Computer Science
University of Nottingham, UK

## ABSTRACT

We present NottReal, an application designed for simulating Voice User Interfaces (VUIs) in Wizard of Oz studies. We briefly discuss the premise and advantages of the Wizard of Oz method before moving onto introducing the design of the application, which we have iteratively developed and refined through a number of studies.

## CCS CONCEPTS

• **Human-centered computing → Systems and tools for interaction design**; *Natural language interfaces.*

## KEYWORDS

conversational interfaces, voice interfaces, vuis, woz

## 1 THE WIZARD OF OZ METHOD

The recent growth in popularity of Voice User Interfaces (VUIs), from smartphone assistants (e.g. Siri) through to smart speakers (e.g. Amazon Echo) have led to a recent resurgence of research in the HCI (e.g. [10, 16]) and CSCW (e.g. [12, 15, 17]) communities that examines the design and use of novel technologies such as 'natural language' interfaces. Implementing these sorts of technologies can be a complex, lengthy, and costly endeavour, involving a host of computational techniques including lookup [13], gestural/spatial recognition [7], robot control [18], mixed reality techniques [4], machine learning [3], or natural language processing [8]. Thus, when it comes to prototyping ideas or conducting research with these interfaces, the Wizard of Oz method or 'experiment' (often abbreviated to simply *WOz* or *WoZ*) is often used as part of the development process [13]. The method prescribes that rather than actually implementing all the elements of a digital system, the 'intelligence' of a machine can be performed by a human operator concealed from the participant, who is led to believe that the system or machine itself is 'intelligent' [5].

The method was originally referred to as "experimenter in the loop" [6, pp. 1–2] or given the epithet of "The Perfect System" [11, p. 843]. The theatrical name of *Wizard of Oz*, perhaps of no surprise, stems from the fictional novel *The Wonderful Wizard of Oz* [2]. The story evolves around the characters' journey to meet a supposedly wonderful wizard that is later revealed to be a sham. The Wizard is, in fact, an 'ordinary' human behind a curtain controlling a machine. In other words, the wonderful wizard is an orchestrated illusion and this is what inspired the method's name.

The Wizard of Oz method proffers designers and researchers the ability to develop medium-fidelity prototypes [13] that can be used for the exploration and testing of ideas as part of iterative design processes and research, and feeds into a number of different forms of analyses, from qualitative Conversation Analysis [19] through to quantitative task performance analysis [1, 9], as well as product development. However, these studies require a flexible tool to enable live performance of the VUI simulation to effectively respond to participants.

## 2 NOTTREAL

NottReal is a cross-platform Python-based desktop application for Wizard-controlled voice interface studies, where the intent detection and slot filling of typical natural language interfaces [14] is completed by a human operator.

The primary window (see Figure 1a) consists of a number of controls for simulating a VUI. Through a number of internal research studies and our experience of the design of VUIs, we have progressively refined the application for quick operation during a study. The controls include: 1) tabbed lists of pre-scripted messages, 2) entry for custom messages, 3) currently queued messages, 4) previously sent messages, 5) previously filled slots, and 6) options to log events and send messages with a loading animation. Additional options are also shown—these mostly originate from features which are enabled through command-line arguments. We iterate through this list in more detail below, explaining how each feature works in practice.

### 2.1 Main features and interaction design

We now work through these features and the basis for their design.

*2.1.1 Message queue and delivery.* NottReal, by default, queues messages to send to the participant and blocks queue processing while a message is being 'delivered' to participants, e.g. through a text-to-speech (TTS) engine. This allows for multiple messages to be queued up with the intent being that the messages be delivered sequentially. This has been useful for us in situations when a large dialogue may be delivered in chunks to the participant.

For delivering these messages, NottReal supports various TTS engines including CereVoice[1] (including support for CereVoice's non-verbal features and spurts), macOS's speech library[2], engines

---

[1]https://www.cereproc.com/en/products/sdk
[2]https://ss64.com/osx/say.html

**(a) Wizard window**
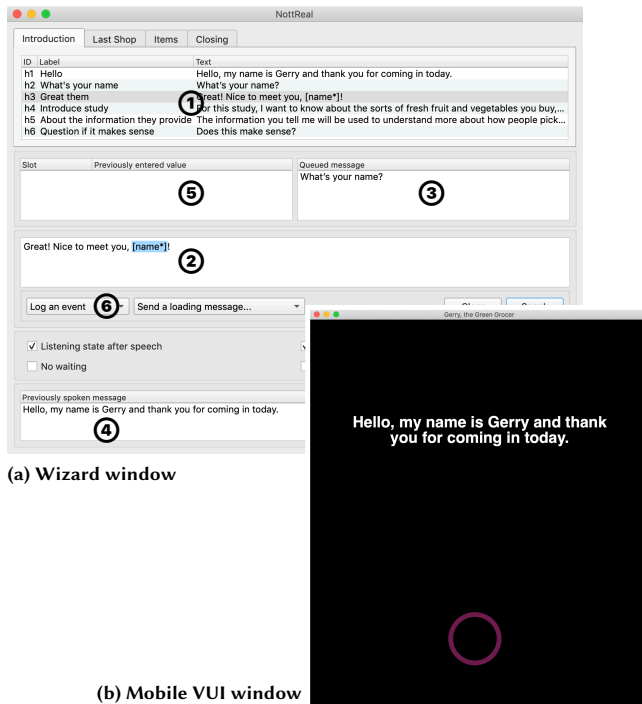


**(b) Mobile VUI window**

**Figure 1: Screenshots from NottReal**

listening to an ActiveMQ channel[3] (we have used this to connect to a virtual avatar, for example), and no TTS output (for visual-only output). The choice of these is configured at run-time with a command-line argument and/or static configuration files.

NottReal also has a simulated 'Mobile VUI' window (show in Figure 1b). We have used this on a second screen which faces participants, which has proved useful when we have wanted participants to face a camera throughout a study. The interface includes an 'orb' that displays the VUI's 'state'. The orb is a pink ring that fades in and out while the VUI is 'talking', a static blue ring while the VUI is 'resting', a white ring with a rotating 'slice' missing during 'computation', and a purple circle that flutters in brightness based on the microphone input sound level while 'listening'. In our experience, this provides a semblance of realism to the system.

We have also added the ability to interrupt the TTS engine and clear the queue by pressing Ctrl+C. This was useful in cases where a participant might try to interrupt the system by 'overtalking', as occurs if a user interrupts a consumer-grade VUI such as Siri or Alexa by uttering the 'wake word' during a response.

*2.1.2 Pre-scripted messages.* NottReal is primarily designed to allow the Wizard to deliver a message 'quickly' to a participant (i.e. within what may be considered a reasonable time for a VUI's latency). The primary configuration of the application consists of pre-scripted messages, separated into categories (displayed as tabs). In our studies, this categorisation has been used to delineate messages used in different experimental conditions or at different stages

of a study. Each message consists of a label for the Wizard to identify it, as well as the message to be delivered. Double-clicking on a message, or pressing the Enter key while one of these messages is highlighted, automatically queues a message to send to the participant, whereas single-clicking copies the message to the text box to enable customisation.

*2.1.3 Slots and slot-tracking.* To allow for responses that incorporate participant/context-specific words (e.g. the words used by a participant in a request), NottReal allows messages to include 'slots', denoted in the pre-scripted messages as text [within square brackets]. If a pre-scripted message with a slot is selected, this message is automatically copied to the text box and the slot is highlighted (if there are multiple slots, the system moves between these on the press of the Tab or Enter keys). The Wizard can type the value of the slot and send the message with Ctrl/Cmd+Enter without moving the mouse pointer to select the slot. Previously used slot names and values are displayed in the window in a list.

Pre-scripted messages can also contain *tracked slots*, which are automatically substituted based on their previously substituted value if the pre-scripted message is double-clicked. This has been especially handy if multiple messages refer to the same thing (e.g. a participant's name). This can be enabled by appending an asterisk to the slot's name, e.g. [name*]. A pre-scripted message can clear the tracking for the slot by appending a dollar to the name, e.g. [name$]. We found this beneficial for responding to state changes in the study (e.g. a subsequent stage, task, or condition).

*2.1.4 Logging.* NottReal records all activities including message delivery with timestamps. We have used this in conjunction with logging and recording from other equipment to index and segment participant data and parse interaction data semi-automatically.

## 2.2 Summary

We have presented NottReal, an interactive tool to support Wizard of Oz studies by simulating a Voice User Interface, including both optional audio and visual output. Its main features include: 1) tabbed lists of pre-scripted messages, 2) entry for custom messages, 3) currently queued messages, 4) previously sent messages, 5) previously filled slots, and 6) options to log events and send messages with a loading animation.

## 2.3 Source code and continuing development

NottReal is open source and licensed under the MIT Licence, and is available from https://github.com/MixedRealityLab/nottreal/. We welcome support from the community in terms of bug reports and contributions and hope to expand the codebase to include support for additional TTS engines, offline automatic speech-to-text support, and features to make the software more capable in supporting the broadest possible range of voice-based Wizard of Oz studies.

---

[3]https://activemq.apache.org/

# REFERENCES

[1] Roldano Cattoni Mauro Cettolo Antoniol, Giuliano and Marcello Federico. 1993. Robust Speech Understanding for Robot Telecontrol. In *Proceedings of the 6th International Conference on Advanced Robotics*. IEEE, Piscataway, NJ, USA, 205–209.

[2] L Frank Baum. 1900. *The Wonderful Wizard of Oz*. George M Hill Company, Chicago, IL, USA.

[3] Jacob T Browne. 2019. Wizard of Oz Prototyping for Machine Learning Experiences. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems (CHI EA '19)*. ACM, New York, NY, USA, Article LBW2621, 6 pages. https://doi.org/10.1145/3290607.3312877

[4] Steven P Dow, Jaemin Lee, Christopher Oezbek, Blair MacIntyre, Jay David Bolter, and Maribeth Gandy. 2005. Wizard of Oz Interfaces for Mixed Reality Applications. In *CHI '05 extended abstracts on Human factors in computing systems - (CHI '05)*. ACM, New York, NY, USA, 1339–1142. https://doi.org/10.1145/1056808.1056911

[5] Norman M Fraser and Nigel Gilbert. 1991. Simulating speech systems. *Computer Speech & Language* 5, 1 (1 1991), 81–99. https://doi.org/10.1016/0885-2308(91)90019-M

[6] Paul Green and Lisa Wei-Haas. 1985. *The Wizard of Oz: a tool for rapid development of user interfaces. Final report.* Technical Report. University of Michigan.

[7] Johanna Höysniemi, Perttu Hämäläinen, and Laura Turkki. 2004. Wizard of Oz Prototyping of Computer Vision Based Action Games for Children. In *Proceeding of the 2004 conference on Interaction design and children building a community (IDC '04)*. ACM, New York, NY, USA, 27–34. https://doi.org/10.1145/1017833.1017837

[8] John F Kelley. 1984. An Iterative Design Methodology for User-friendly Natural Language Office Information Applications. *ACM Transactions on Information Systems* 2, 1 (Jan. 1984), 26–41. https://doi.org/10.1145/357417.357420

[9] Philipp Kirschthaler, Martin Porcheron, and Joel E Fischer. 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* (Bilbao, Spain) *(CUI '20)*. ACM, New York, NY, USA, 9. https://doi.org/10.1145/3405755.3406119

[10] Ewa Luger and Abigail Sellen. 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 (CHI '16)*. ACM, New York, NY, USA, 5286–5297. https://doi.org/10.1145/2858036.2858288

[11] Ashok Malhotra. 1975. Knowledge-Based English Language Systems for Management Support: An Analysis of Requirements. In *Proceedings of the 4th International Joint Conference on Artificial Intelligence - Volume 1* (Tblisi, USSR)

*(IJCAI'75)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 842–847.

[12] Nikolas Martelaro and Wendy Ju. 2017. WoZ Way: Enabling Real-Time Remote Interaction Prototyping & Observation in On-Road Vehicles. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW '17)*. ACM, New York, NY, USA, 169–182. https://doi.org/10.1145/2998181.2998293

[13] David Maulsby, Saul Greenberg, and Richard Mander. 1993. Prototyping an Intelligent Agent through Wizard of Oz. In *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems (CHI '93)*. ACM, New York, NY, USA, 277–284. https://doi.org/10.1145/169059.169215

[14] Pin Ni, Yuming Li, Gangmin Li, and Victor Chang. 2020. Natural language understanding approaches based on joint task of intent detection and slot filling for IoT voice interaction. *Neural Computing and Applications* Online (3 2020), 18. https://doi.org/10.1007/s00521-020-04805-x

[15] Martin Porcheron, Joel E Fischer, Moira McGregor, Barry Brown, Ewa Luger, Heloisa Candello, and Kenton O'Hara. 2017. Talking with Conversational Agents in Collaborative Action. In *Companion of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing* (Portland, Oregon, USA) *(CSCW '17 Companion)*. Association for Computing Machinery, New York, NY, USA, 431–436. https://doi.org/10.1145/3022198.3022666

[16] Martin Porcheron, Joel E Fischer, Stuart Reeves, and Sarah Sharples. 2018. Voice Interfaces in Everyday Life. In *Proceedings of the 2018 ACM Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 640, 12 pages. https://doi.org/10.1145/3173574.3174214

[17] Martin Porcheron, Joel E Fischer, and Sarah Sharples. 2017. "Do Animals Have Accents?": Talking with Agents in Multi-Party Conversation. In *Proceedings of the 20th ACM Conference on Computer-Supported Cooperative Work & Social Computing (CSCW '17)*. ACM, New York, NY, USA, 207–219. https://doi.org/10.1145/2998181.2998298

[18] Ben Robins, Kerstin Dautenhahn, Rene te Boekhorst, and Chrystopher L Nehaniv. 2008. Behaviour Delay and Robot Expressiveness in Child-robot Interactions: A User Study on Interaction Kinesics. In *Proceedings of the 3rd ACM/IEEE International Conference on Human Robot Interaction (HRI '08)*. ACM, New York, NY, USA, 17–24. https://doi.org/10.1145/1349822.1349826

[19] Robin Wooffitt. 1994. Applying Sociology: Conversation Analysis in the Study of Human-(Simulated) Computer Interaction. *Bulletin of Sociological Methodology/Bulletin de Méthodologie Sociologique* 43, 1 (1994), 7–33. https://doi.org/10.1177/075910639404300103