UAM | ESCUELA POLITÉCNICA SUPERIOR

MASTER'S THESIS

Automated Exploration and Profiling of Conversational Agents

Master's in Data Science

Author:	Iván Sotillo del Horno
Supervisor:	Juan de Lara Jaramillo
Co-supervisor:	Esther Guerra Sánchez
Department:	Department of Computer Science
Submission Date:	July 10, 2025

Universidad Autónoma de Madrid
Escuela Politécnica Superior

I confirm that this master's is my own work and I have documented all sources and material used.

Madrid, Spain, July 10, 2025 Iván Sotillo del Horno

Acknowledgments

Abstract

Contents

Acknowledgments **iii**

Abstract **iv**

1 Introduction **1**
1.0.1 Subsection 3

2 Background and State of the Art **5**

3 TRACER: Automated Chatbot Exploration **6**

4 User Profile Structure and Generation **7**

5 Tool Support **8**

6 Evaluation **9**

7 Conclusions and Future Work **10**

Abbreviations **11**

List of Figures **12**

List of Tables **13**

Bibliography **14**

1 Introduction

The proliferation of conversational agents, commonly refered to as chatbots, has fundamentally transformed the landscape of human-computer interaction accross diverse domains. From general-purpose assistants such as OpenAI's ChatGPT [1] or Google's Gemini [2] to task-oriented agents that assist users in specific tasks like shopping, contracting services, or customer support. These intelligent systems allow for natural language interaction with services ranging from customer support and e-commerce platforms to municipal services and educational resources. The proliferation of these agents has been further accelerated by recent advances in generative Artificial Intelligence (AI), particularly the advance in Large Language Models (LLMs), which have significantly enhanced the capabilities of these chatbots, allowing them to both create and understand natural language without explicitly programmed rules.

The presence of these conversational agents in so many applications has elevated the concerns regarding their reliability, correctness, and overall quality assurance. As these systems appear in domains, such as healthcare or finances, which require high levels of trust, the need for rigorous testing and validation becomes paramount. However, the heterogeneous nature of chatbot development, with intent-based frameworks like Google's Dialogflow [3] or Rasa [4], multi-agent programming environments built upon LLMs such as LangGraph [5] and Microsoft's AutoGen [6], and Domain-Specific Languages (DSLs) like Taskyto [7], present significant challenges for finding a comprehensive methodology to test these systems.

Traditional software testing techniques are very limited when applied to chatbot systems. The complexity of Natural Language Processing (NLP), the non-deterministic nature of LLMs and the dynamic flow of a real conversation make traditional testing inadequate for conversational agents. While there have been some approaches for developing testing techniques for chatbots [8, 9], they often target specific chatbot technologies [10], require substantial manual effort including the provision of test conversations [10, 11] or synchronous human interaction [12], depend on existing conversation corpus [13], or need access to the cahtbot's source code [14–16], thereby limiting their applicability to deployed systems treated as black boxes.

The research presented in this thesis aims to solve these problems through the development of Task Recognition And Chatbot ExploreR (TRACER), a tool for extracting comprehensive models from deployed conversational agents, and then with this model,

create user profiles which will serve as test cases for a user simulator called Sensei [17]. TRACER employs an LLM agent to systematically explore the chatbot's capabilities via natural language interactions, eliminating the need for manual test case creation or access to the chatbot's source code. This black-box approach enables the automated generation of detailed chatbot models that encapsulate the system's supported languages, fallback mechanisms, functional capabilities, input parameters, admissible parameter values, output data structures, and conversational flow patterns.

The extracted chatbot model serves as the foundation for the automated synthesis of test cases. Specifically, TRACER generates user profiles that represent diverse users that interact with the chatbot using Sensei [17], but different implementations of TRACER could be made to generate different types of test cases based on the extracted model. The integration of TRACER with Sensei yields a testing methodology that only requires a connector for the chatbot's API.

To ensure the accessibility and reproducibility of this research, TRACER has been developed as a complete, open-source tool. It is publicly available as a Python Package Index (PyPI) package [18] and can be installed via `pip install chatbot-tracer`. The full source code is hosted on GitHub https://github.com/Chatbot-TRACER/TRACER, and a dedicated web application has been developed to provide a user-friendly experience for the entire testing pipeline, from model extraction and user profiles generation with TRACER to test execution with Sensei. This ecosystem provides a robust and practical testing methodology that requires only a connector for the chatbot's API.

To guide this investigation, we have defined the following research questions:

- **RQ1: How effective is TRACER in modeling chatbot functionality?** This question assesses the ability of our model exploration technique to achieve high functional coverage in a controlled setting where the ground truth is known.
- **RQ2: How effective are the synthesized profiles at detecting faults in controlled environments?** This question evaluates the precision of our approach by using mutation testing [15] to measure the ability of the generated profiles to identify specific, injected faults.
- **RQ3: How effective is the approach at identifying real-world bugs and ensuring task completion in deployed chatbots?** This question addresses the practical, real-world applicability of our framework by measuring the Bug Detection Rate (BDR) and Task Completion Rate (TCR) of the generated profiles against real-world chatbots.

Thesis structure. Chapter 2 establishes the background and state of the art in chatbot test. Chapter 3 presents the core methodoly of how TRACER extracts models from chatbots. Chapter 4 describes the structure of the user profiles, and how TRACER generates them.

Chapter 5 shows TRACER Command Line Interface (CLI) and the web application to use both TRACER and Sensei. Chapter 6 presents the evaluation of TRACER against the research questions. Chapter 7 concludes the thesis and discusses future work.

1.0.1 Subsection

Table 1.1: An example for a simple table.

A	B	C	D
1	2	1	2
2	3	2	3

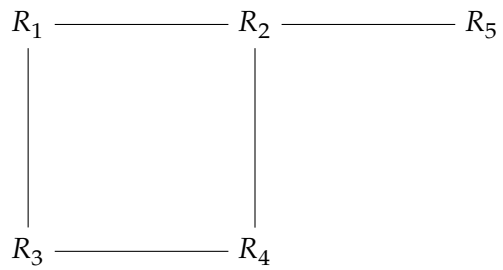

Figure 1.1: An example for a simple drawing.

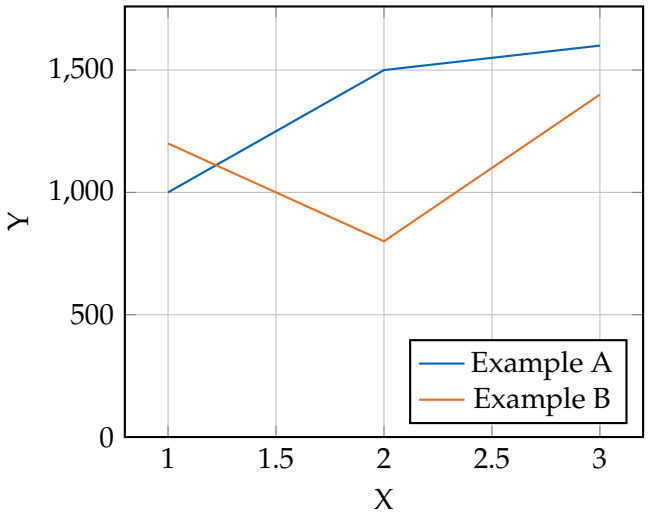

Figure 1.2: An example for a simple plot.

```
SELECT * FROM tbl WHERE tbl.str = "str"
```

Figure 1.3: An example for a source code listing.

2 Background and State of the Art

3 TRACER: Automated Chatbot Exploration

4 User Profile Structure and Generation

5 Tool Support

6 Evaluation

7 Conclusions and Future Work

Abbreviations

AI Artificial Intelligence

NLP Natural Language Processing

LLM Large Language Model

DSL Domain-Specific Language

TRACER Task Recognition And Chatbot ExploreR

BDR Bug Detection Rate

TCR Task Completion Rate

CLI Command Line Interface

PyPI Python Package Index

List of Figures

1.1 Example drawing 3
1.2 Example plot 4
1.3 Example listing 4

List of Tables

1.1 Example table 3

Bibliography

[1] "ChatGPT," Accessed: Jul. 10, 2025. [Online]. Available: https://chatgpt.com.

[2] "Google Gemini," Gemini, Accessed: Jul. 10, 2025. [Online]. Available: https://gemini.google.com.

[3] "Dialogflow," Google Cloud, Accessed: Jul. 10, 2025. [Online]. Available: https://cloud.google.com/products/conversational-agents.

[4] "Rasa," Rasa, Accessed: Jul. 10, 2025. [Online]. Available: https://rasa.com/.

[5] "LangGraph," Accessed: Jul. 10, 2025. [Online]. Available: https://www.langchain.com/langgraph.

[6] "AutoGen," Accessed: Jul. 10, 2025. [Online]. Available: https://microsoft.github.io/autogen/stable/.

[7] J. Sánchez Cuadrado, S. Pérez-Soler, E. Guerra, and J. De Lara, "Automating the Development of Task-oriented LLM-based Chatbots," in *Proceedings of the 6th ACM Conference on Conversational User Interfaces*, ser. CUI '24, New York, NY, USA: Association for Computing Machinery, Jul. 8, 2024, pp. 1–10, ISBN: 979-8-4007-0511-3. DOI: 10.1145/3640794.3665538. Accessed: Mar. 19, 2025. [Online]. Available: https://doi.org/10.1145/3640794.3665538.

[8] J. S. Cuadrado, D. Ávila, S. Pérez-Soler, P. C. Cañizares, E. Guerra, and J. De Lara, "Integrating Static Quality Assurance in CI Chatbot Development Workflows," *IEEE Software*, vol. 41, no. 5, pp. 60–69, Sep. 2024, ISSN: 0740-7459, 1937-4194. DOI: 10.1109/ms.2024.3401551. Accessed: Jul. 10, 2025. [Online]. Available: https://ieeexplore.ieee.org/document/10533225/.

[9] P. C. Cañizares, J. M. López-Morales, S. Pérez-Soler, E. Guerra, and J. De Lara, "Measuring and Clustering Heterogeneous Chatbot Designs," *ACM Transactions on Software Engineering and Methodology*, vol. 33, no. 4, pp. 1–43, May 31, 2024, ISSN: 1049-331X, 1557-7392. DOI: 10.1145/3637228. Accessed: Jul. 10, 2025. [Online]. Available: https://dl.acm.org/doi/10.1145/3637228.

[10] "Rasa Test," Accessed: Jul. 10, 2025. [Online]. Available: https://rasa.com/docs/pro/testing/evaluating-assistant/.

[11] "Cyara Botium," Cyara, Accessed: Jul. 10, 2025. [Online]. Available: https://cyara.com/products/botium/.

[12] R. Ren, J. W. Castro, S. T. Acuña, and J. De Lara, "Evaluation Techniques for Chatbot Usability: A Systematic Mapping Study," *International Journal of Software Engineering and Knowledge Engineering*, vol. 29, pp. 1673–1702, 11n12 Nov. 2019, ISSN: 0218-1940, 1793-6403. DOI: 10.1142/s0218194019400163. Accessed: Jul. 10, 2025. [Online]. Available: https://www.worldscientific.com/doi/abs/10.1142/S0218194019400163.

[13] M. Vasconcelos, H. Candello, C. Pinhanez, and T. Dos Santos, "Bottester: Testing Conversational Systems with Simulated Users," in *Proceedings of the XVI Brazilian Symposium on Human Factors in Computing Systems*, Joinville Brazil: ACM, Oct. 23, 2017, pp. 1–4. DOI: 10.1145/3160504.3160584. Accessed: Jul. 10, 2025. [Online]. Available: https://dl.acm.org/doi/10.1145/3160504.3160584.

[14] P. C. Canizares, D. Ávila, S. Perez-Soler, E. Guerra, and J. de Lara, "Coverage-based Strategies for the Automated Synthesis of Test Scenarios for Conversational Agents," in *Proceedings of the 5th ACM/IEEE International Conference on Automation of Software Test (AST 2024)*, ser. AST '24, New York, NY, USA: Association for Computing Machinery, Jun. 10, 2024, pp. 23–33, ISBN: 979-8-4007-0588-5. DOI: 10.1145/3644032.3644456. Accessed: Mar. 19, 2025. [Online]. Available: https://doi.org/10.1145/3644032.3644456.

[15] P. Gómez-Abajo, S. Pérez-Soler, P. C. Cañizares, E. Guerra, and J. de Lara, "Mutation Testing for Task-Oriented Chatbots," in *Proceedings of the 28th International Conference on Evaluation and Assessment in Software Engineering*, ser. EASE '24, New York, NY, USA: Association for Computing Machinery, Jun. 18, 2024, pp. 232–241, ISBN: 979-8-4007-1701-7. DOI: 10.1145/3661167.3661220. Accessed: Mar. 19, 2025. [Online]. Available: https://doi.org/10.1145/3661167.3661220.

[16] M. F. Urrico, D. Clerissi, and L. Mariani, "MutaBot: A Mutation Testing Approach for Chatbots," in *Proceedings of the 2024 IEEE/ACM 46th International Conference on Software Engineering: Companion Proceedings*, Lisbon Portugal: ACM, Apr. 14, 2024, pp. 79–83. DOI: 10.1145/3639478.3640032. Accessed: Jul. 10, 2025. [Online]. Available: https://dl.acm.org/doi/10.1145/3639478.3640032.

[17] J. De Lara, E. Guerra, A. del Pozzo, and J. Sanchez Cuadrado. "Sensei," GitHub, Accessed: Jul. 10, 2025. [Online]. Available: https://github.com/satori-chatbots/user-simulator.

[18] S. del Horno, *Chatbot-tracer: A tool to model chatbots and create profiles to test them.* Version 0.2.10. Accessed: Jul. 10, 2025. [Online]. Available: https://github.com/Chatbot-TRACER/TRACER.