

Comparative Analysis: AI Chatbot vs. Rules-Based Chatbot

Group 5

Contents

Introduction	2
Notes for running the models:	3
Comparative Analysis:	4
1. Complexity.....	4
2. Accuracy.....	4
3. Response Time.....	5
4. Usability.....	5
5. User Experience	6
6. Other Dimensions	6

Introduction

In this project, we developed and analyzed two chatbot systems designed to answer frequently asked questions (FAQs): an **AI-based chatbot** powered by a pre-trained transformer model and a **rules-based chatbot** utilizing manually defined responses. The primary objective was to explore and compare the effectiveness of these two distinct approaches in addressing user queries, focusing on key factors such as **complexity**, **accuracy**, **response time**, and **usability**.

- **AI-Based Chatbot:** This chatbot leverages advanced machine learning techniques, including retrieval-augmented generation, to understand user intent and provide accurate, context-aware responses. It is trained on a dataset of 226 question-answer pairs scraped from the SMU website, allowing it to generate responses beyond simple string matching.
- **Rules-Based Chatbot:** This chatbot relies on a set of predefined rules and patterns to match user inputs with corresponding responses. While straightforward and quick, this approach lacks flexibility and is dependent on precise input matching.

By comparing these two systems, we aim to evaluate the trade-offs between **modern AI-driven methods** and **traditional rule-based systems**, providing insights into their practicality for different use cases.

All code and models for this project are hosted on GitHub and can be accessed here: [Models On GitHub](#)

This report details our methodology, implementation, and findings, offering a comprehensive understanding of the strengths and limitations of each chatbot approach.

Notes for running the models:

1. For the AI model:

- **Locate and Open:** Download the aibot.ipynb file from GitHub and open it in Google Colab.
- **Replace API Key:** Replace the placeholder API key in the notebook with the following:
- OpenAI API Key: [Get yours](#)
- **Run the Notebook:** Execute all the code cells sequentially to set up the model.
- **Access via Anvil:** After the setup, visit the Anvil link to interact with the AI model: <https://bountiful-qualified-vehicle.anvil.app/>.
- If the Anvil interface fails to load or work, you can interact with the AI model directly within the notebook itself.
- **Anvil Key:** The Anvil server key is as follows: [Get yours](#)

2. For the Rules Based one:

- **Download Files:** Download the server.py and SMULibrarybot.html files from the repository.
- **Run the Server:**
 1. Open a terminal and navigate to the directory containing the downloaded files.
 2. Run the following command to start the server: `python server.py`
- **Access the Chatbot:** Once the server is running, open your browser and navigate to <http://localhost:8000/>.

Comparative Analysis:

1. Complexity

The AI chatbot clearly outperforms the rules-based chatbot in complexity due to its ability to encode and understand language contextually. It was trained on a dataset with over 200 question-answer pairs, allowing it to handle nuanced and varied questions.

On the other hand, the rules-based chatbot relies heavily on string matching. Each possible input must be manually engineered. For example, the query "What is the academic calendar?" will only work if the input precisely matches the tokenized string "academic calendar." Any deviation, such as "academic year schedule" or a typo like "acadmic calndar," will fail. This limitation highlights the difficulty in scaling the rules-based approach, especially when dealing with a larger or more diverse dataset.

Conclusion: The AI chatbot is far superior in terms of complexity, offering more flexibility and scalability.

2. Accuracy

In terms of accuracy, both models have their pros and cons.

- The **rules-based chatbot** delivers precise answers if the user inputs match predefined keywords exactly. However, it struggles with synonyms, typos, and phrasing variations. For example, "academic calendar" works, but "What document has info about the school year?" does not.
- The **AI chatbot**, powered by embeddings and generative AI, can retrieve relevant answers even with varied phrasing or minor typos. It can also infer related concepts, e.g., "How do I find deadlines for academic activities?" retrieves the answer to "academic calendar." However, its reliance on generative AI introduces the risk of "hallucinations" where the bot might generate incorrect responses for unrelated or out-of-scope queries. **Error Handling:** Default response is a generic apology when it cannot match the input to any rule.

Conclusion: While the rules-based bot ensures perfect accuracy within its predefined scope, the AI chatbot excels in adaptability and context recognition, though it may occasionally generate inaccurate responses for queries outside the dataset.

3. Response Time

We measured the average response time for both models using a Python script.

- **Rules-Based Chatbot:** Responses are almost instantaneous (<0.1 seconds) since it directly matches input strings with predefined answers. The simplicity of its logic ensures high speed.
- **AI Chatbot:** Due to the need for embedding generation, context retrieval, and response generation, the AI chatbot takes longer. The average response time ranges from **1.5 to 3 seconds** depending on the question length and the number of similar answers in the dataset.

Chatbot	Average Time to Respond (ms)
Rules- Based	~50
AI (RAG)	~2000

Conclusion: The rules-based bot is faster due to its simplicity, while the AI chatbot trades speed for adaptability and contextual understanding.

4. Usability

The AI chatbot is significantly more user-friendly. It can handle typos, variations in phrasing, and even answer related queries. For instance:

- Input: "What's the school schedule doc called?" → AI chatbot retrieves the answer for "academic calendar."
- Input: "acadmic calndar" → AI chatbot still responds correctly, whereas the rules-based bot fails.

The rules-based chatbot's dependency on exact string matches limits its usability, requiring users to input precise terms that may not always be intuitive.

Takeaway: The AI chatbot delivers a better user experience due to its flexibility and robustness.

5. User Experience

We evaluated the user experience provided by both chatbots. The AI Chatbot delivers detailed and contextually appropriate answers, effectively handling typos and variations in phrasing. Its responses feel more conversational and natural, enhancing user engagement. However, the slightly longer response times may affect user satisfaction, and it might occasionally provide answers to questions outside its intended scope.

In contrast, the Rules-Based Chatbot offers instantaneous responses—often under 0.1 seconds—since it directly matches input strings with predefined answers. The simplicity of its logic ensures high speed and predictable behavior, improving perceived responsiveness. Nonetheless, it can be frustrating when it fails to understand valid queries due to strict keyword matching, and the conversations may feel mechanical and limited.

Conclusion: The AI chatbot offers a superior user experience by understanding and responding to a broader range of queries in a more natural manner. While its response time is longer, the richness and relevance of its answers outweigh this drawback. The rules-based chatbot, despite its speed, may lead to user frustration due to its limitations in understanding and flexibility.

6. Other Dimensions

Beyond the discussed factors, here are additional comparison points:

- **Maintenance:** The AI chatbot requires retraining for dataset updates, while the rules-based bot requires manual updates to its predefined rules. As the dataset grows, maintaining the rules-based bot becomes increasingly impractical.
- **Scalability:** The AI chatbot scales seamlessly with larger datasets, while the rules-based bot's complexity grows linearly with the number of rules added.
- **Cost:** The rules-based chatbot is resource-light and runs on minimal infrastructure. The AI chatbot, depending on its model size and external API usage (e.g., OpenAI GPT), incurs computational and monetary costs