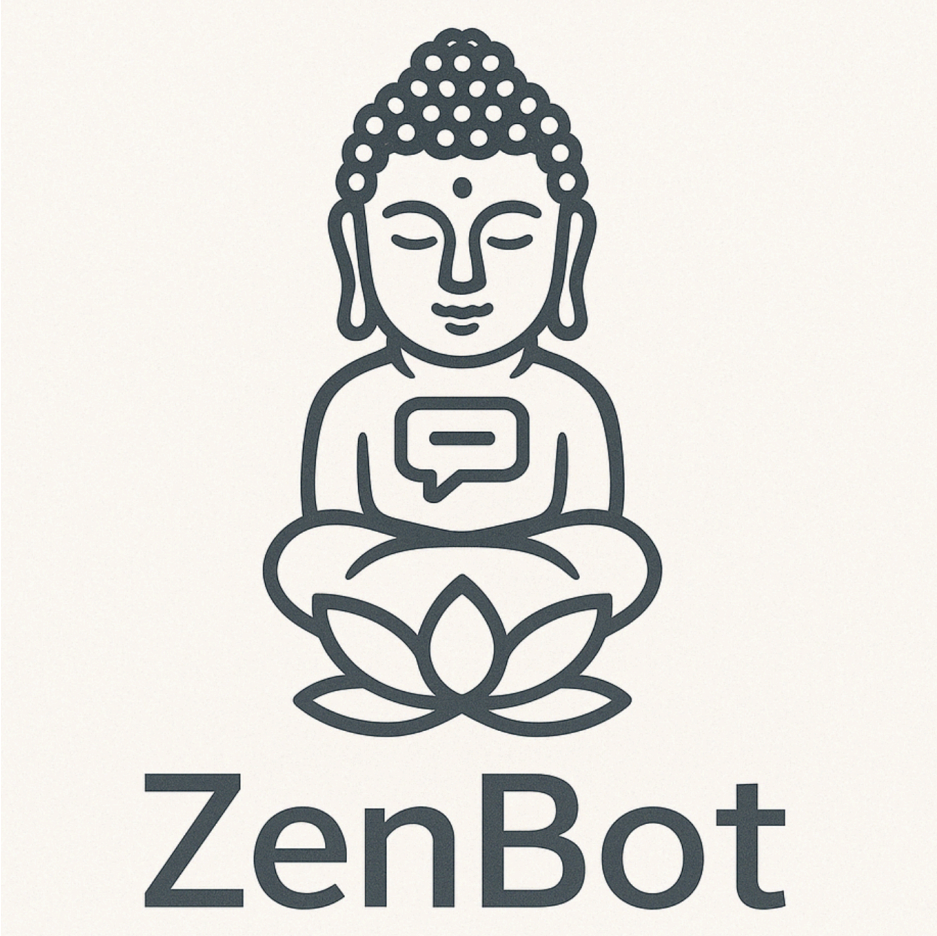
ZenBot: evaluation

Evaluation report for ZenBot, your helpful order assistant!



Joanna Równicka

24.04.2025

https://github.com/curly1/ZenBot

INTRODUCTION

In recent years, the rise of large language models (LLMs) has transformed the way conversational agents are developed, shifting from rigid rule-based systems to more flexible, context-aware generative models. This report evaluates ZenBot, a **fully generative, policy-aware chatbot** designed to assist users with **order tracking and cancellation** tasks. ZenBot integrates with mocked API endpoints and adheres to company-specific policies using an LLM-based decision framework. The evaluation aims to quantify ZenBot's capabilities in understanding user intent, enforcing business logic, and generating coherent, helpful responses. A rule-based agent serves as a baseline for comparison to highlight the benefits and trade-offs of LLM-based reasoning.

HYPOTHESIS

We hypothesize that ZenBot, powered by a local large language model, will **outperform the baseline rule-based agent** in all core dimensions of agent performance. Specifically, ZenBot is expected to achieve higher intent accuracy, better compliance with business policies, and superior response quality as judged by both automatic and human-in-the-loop evaluation methods. While response **latency is anticipated to be higher** due to the computational demands of generative reasoning, this trade-off is considered acceptable in exchange for **increased correctness and usefulness**.

PROCEDURE

The evaluation followed a structured, side-by-side **A/B testing** methodology. A **synthetic dataset** of 200 user queries was generated to represent common and edge-case scenarios in order-related customer support. Each query was **processed independently** by both the rule-based agent and ZenBot. Their outputs were **logged and analyzed** using a combination of **quantitative metrics** (e.g., intent accuracy, policy adherence, API correctness, latency) and **qualitative ratings** (e.g., naturalness, coherence, helpfulness, response quality).

ZenBot's outputs were assessed using an **LLM judge**. Classification metrics such as **precision, recall, and F1-score** were used to quantify the accuracy of policy and API error handling. **Descriptive statistics** and **correlation matrices** were also computed to

better understand patterns in the qualitative data.

DATA

The dataset used for evaluation comprised **200 synthetic user prompts** generated with **ChatGPT-4o**. It included:

- 50 **tracking** requests (some explicit, some implicit)
- 50 **cancellation** requests (eligible and ineligible)
- 50 **irrelevant** or out-of-scope prompts
- 50 **misleading** prompts **containing keywords** but no valid intent

Each prompt was paired with structured order metadata, expected tool choice, policy applicability, and ground-truth API response status. This setup enabled precise measurement of agent behavior under varied conditions, and ensured reproducibility. All evaluation runs were **logged**, and outputs were **saved in structured CSV format** for further analysis.

RESULTS

Quantitative metrics

Metric	Baseline Agent	ZenBot Agent
Intent Accuracy	28.00% (56/200)	93.50% (187/200)
Policy Adherence	0.00% (0/0)	100.00% (46/46)
API Status Accuracy	0.00% (0/0)	80.43% (37/46)
Response Time (mean, s)	0.000	6.529
Response Time (max, s)	0.001	12.616
Response Time (>1s proportion)	0.000	1.000
Intent Error Rate	0.720	0.065
Unknown Intent Ratio	0.000	0.000
Policy Error - Precision	⚠ N/A	1.000
Policy Error - Recall	⚠ N/A	1.000
Policy Error - F1 Score	⚠ N/A	1.000
API Error - Precision	⚠ N/A	1.000
API Error - Recall	⚠ N/A	0.804
API Error - F1 Score	⚠ N/A	0.892

Table 1. Summary of quantitative metrics for baseline vs. ZenBot agents.

Baseline Rule-Based Agent

The baseline agent demonstrates **limited capabilities** across all quantitative metrics. Its **intent classification accuracy** is only **28.00% (56/200)**, which indicates that it fails to correctly identify user intent in the vast majority of cases. This low level of understanding severely restricts the agent's usefulness in real-world interactions. There is **no recorded data** on **policy adherence** or **API status accuracy**, implying that the agent either lacks the functionality to handle such tasks or never encountered relevant scenarios during evaluation.

Where the agent excels is in **response speed**. With an average latency of **0.000 seconds** and a maximum of just **0.001 seconds**, its responses are nearly instantaneous. However, this **extreme speed comes at the cost of depth and accuracy**, as the agent relies on simple rule-based logic that does not support complex reasoning or dynamic interactions. This is further confirmed by its **zero performance** in both **policy error detection** and **API error detection**, due to a complete absence of valid data to compute these metrics.

Overall, while the baseline agent may appear efficient, its **lack of intelligence, adaptability, and task handling** severely limits its practical value.

ZenBot Agent

In stark contrast, ZenBot delivers **strong and reliable performance** across all quantitative metrics. It achieves an **intent classification accuracy of 93.50% (187/200)**, reflecting a deep understanding of user input. Furthermore, ZenBot demonstrates **perfect policy adherence**, correctly following rules in **100% (46/46)** of applicable cases, and performs robustly in managing API interactions, with an **API status accuracy of 80.43% (37/46)**.

ZenBot also excels in **error detection**. Its handling of **policy errors** is flawless, with **precision, recall, and F1-score all at 1.000**, indicating it detected every issue without false alarms or missed cases. In terms of **API errors**, ZenBot shows strong performance as well, with a **precision of 1.000**, **recall of 0.804**, and an **F1-score of 0.892**. This is an impressive result considering the complexity of real-time API handling.

The primary trade-off is **latency**. ZenBot's average response time is **6.529 seconds**, with a maximum of **12.616 seconds**, and **every response takes longer than one second**. While

this is significantly slower than the baseline agent, it reflects the computational demands of its more sophisticated reasoning and language understanding.

Summary

In summary, ZenBot provides **highly accurate, policy-compliant, and API-aware behavior**, making it a reliable choice for intelligent agent applications. Despite its slower response time, its **substantial improvements in correctness and robustness** mark a significant leap forward in capability compared to the baseline.

Qualitative metrics

Metric	Baseline Agent	ZenBot Agent
Naturalness (1-5 avg)	2.98	2.95
Coherence (1-5 avg)	3.77	3.67
Helpfulness (1-5 avg)	1.96	2.62
Response Quality Pass Rate (0-1 avg)	0.07	0.47
Naturalness-Helpfulness Correlation	0.56	0.88
Coherence-Helpfulness Correlation	0.76	0.86
Helpfulness-Pass Correlation	0.80	0.96
Naturalness-Pass Correlation	0.36	0.86

Table 2. Summary of quantitative metrics for baseline vs. ZenBot agents.

Baseline Rule-Based Agent

The baseline agent received mixed ratings for its conversational quality. On a 1-5 scale, an LLM judge rated its **naturalness** at an average of **2.98**, suggesting that while it occasionally mimicked human-like dialogue, it often felt mechanical or rigid. Its **coherence** fared better, with an average score of **3.77**, indicating that its responses were generally logical and on-topic, though not always deeply connected to the user's input.

However, the agent struggled significantly in terms of **helpfulness**, scoring just **1.96** on average. This low score indicates that most of its responses failed to provide useful or actionable information. The overall **response quality pass rate**, which measures whether the agent's replies were good enough to be considered acceptable, was a mere **7%** (0.07), underscoring its limited utility in real interactions.

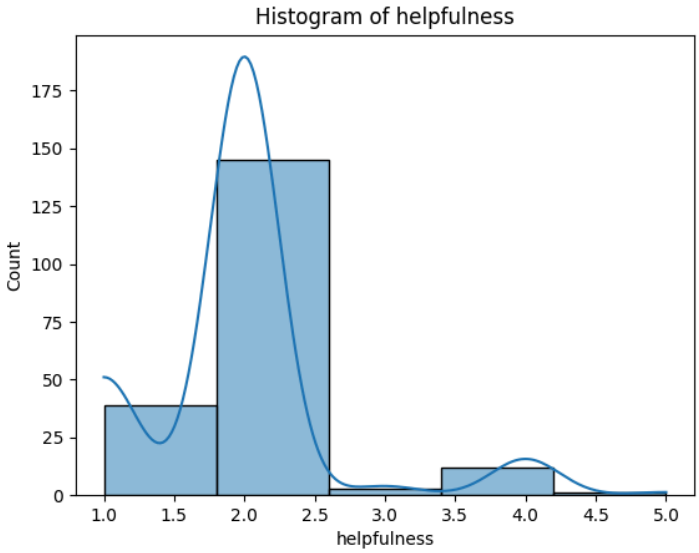


Fig 1. Histogram of helpfulness scores for baseline rule-based agent.

Statistical analysis supports these findings: the distributions for naturalness and coherence were relatively narrow, with most values clustered around the mid-range. The **correlation analysis** reveals that helpfulness was strongly associated with overall response quality (**correlation of 0.80** with the binary pass rate), and also moderately correlated with naturalness and coherence.

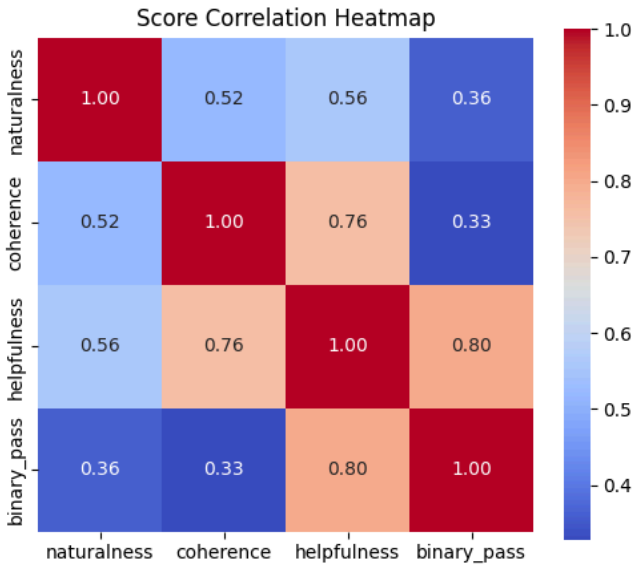


Fig 2. Score correlation heatmap for baseline rule-based agent.

ZenBot Agent

ZenBot also received average scores for **naturalness** (**2.95**) and **coherence** (**3.67**), which are comparable to the baseline agent. This suggests that despite its advanced capabilities, an LLM judge still perceived its tone and flow as only moderately human-like and occasionally disconnected. However, where ZenBot truly outshines the baseline agent is in **helpfulness**, where it earned a significantly higher average score of **2.62**.

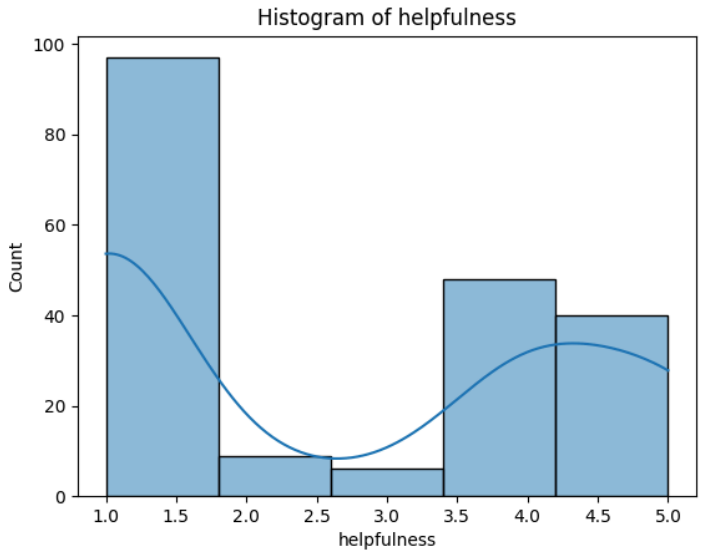


Fig 3. Histogram of helpfulness scores for ZenBot agent.

Most notably, ZenBot's **response quality pass rate** is **47%**, a major improvement over the baseline's 7%. This reflects a much greater proportion of responses deemed acceptable or valuable by users. The broader spread in its descriptive statistics, particularly the higher standard deviations, suggests a wider range of user experiences (from poor to excellent) though with a clear trend toward more positive assessments.

The **correlation analysis** for ZenBot is striking. All qualitative metrics (naturalness, coherence, and helpfulness) are very strongly interrelated (correlations above 0.85) and show a near-perfect correlation with the overall pass rate. Specifically, helpfulness has a **correlation of 0.96** with the binary pass rate, suggesting that the perceived utility of a response is the strongest determinant of whether an LLM judge found it acceptable.

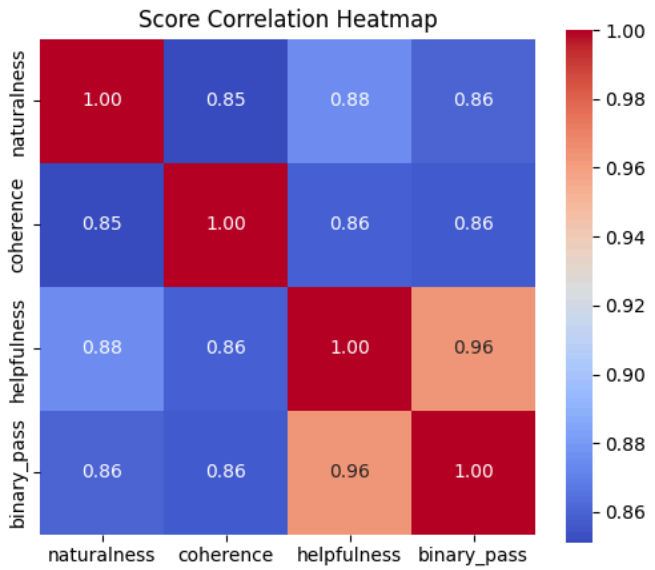


Fig 4. Score correlation heatmap for ZenBot agent.

Summary

While both agents score similarly in naturalness and coherence, **ZenBot delivers significantly more helpful responses**, leading to a much higher overall pass rate. This makes ZenBot more likely to meet user expectations in real-world scenarios. The correlation patterns further suggest that **helpfulness is the key driver of perceived response quality**, and ZenBot's superior performance in this area is a clear differentiator.

CONCLUSION

The evaluation demonstrates that ZenBot significantly outperforms the baseline rule-based agent across nearly all measured dimensions. ZenBot achieves an **intent classification accuracy** of **93.5%, compared to just 28%** for the baseline. It also exhibits **perfect policy adherence** (100%) and **strong API error handling capabilities** (80.43% accuracy). In contrast, the rule-based agent failed to generate valid outputs for policy or API-related metrics.

On the qualitative side, ZenBot also shows improvements, **particularly in helpfulness** (2.62 vs. 1.96) and **overall response quality** (47% pass rate vs. 7%). Although both agents

scored similarly on naturalness and coherence, ZenBot's responses were more varied and more frequently rated as useful. Correlation analysis confirmed that **helpfulness is the strongest predictor of overall response quality**, and ZenBot's superior helpfulness score directly contributed to its higher pass rate.

While ZenBot's **latency is higher** (average of 6.53 seconds vs. 0.001 seconds), this is expected due to its reliance on LLM inference. The trade-off is justified by the dramatic gains in correctness, policy compliance, and user-perceived value. These results validate the effectiveness of generative, policy-aware agents for real-world customer support scenarios and provide a solid foundation for future enhancements.

FUTURE ENHANCEMENTS

Based on current findings and limitations observed during the evaluation, several areas have been identified for future improvement:

1. **Model Upgrades**: Explore larger or more advanced models such as Mistral-24B or Llama 3 to improve reasoning, reduce hallucinations, and increase user satisfaction.
2. **Ambiguity Handling**: Introduce confirmation prompts when user intent is unclear to reduce erroneous tool calls.
3. **Real Data Evaluation**: Supplement synthetic evaluations with real user data to improve validity.
4. **Latency Optimization**: Investigate caching, model distillation, and hardware acceleration to reduce response times.
5. **Safety and Ethics**: Extend evaluations to include safety, bias, and hallucination metrics, and implement safeguards for sensitive scenarios.
6. **Inter-Rater Agreement**: Incorporate human raters and compute agreement metrics (e.g., Cohen's Kappa) to validate LLM-based judgments.

These enhancements aim to make ZenBot more capable, trustworthy, and adaptable for enterprise use.