Term Paper: AI Customer Support Resolution, Discrete Simulation (Fall 2024)

Emily Ekdahl

MSDS 460: Decision Analytics

Dr. Thomas W. Miller

2024-12-05

Introduction

This simulation aims to determine optimal staffing given various deflection rates of frontline AI customer support agents and SLAs set by the business for wait times. The company in question has a massive end-of-year spike in customer service requests, as that is the time when most customers adopt their software in preparation for the upcoming year. Because a non-trivial amount of training is required to provide high-quality customer service, the business must know months in advance how well the AI customer support deflection feature is working to predict how many customer support staff they must hire and train.

Method

I started by building an event graph to describe how the process works for AI customer support deflection rates, depicted below in Figure 1. For simplicity and to keep this simulation manageable, I did not account for cases requiring multiple trips through the queue for full resolution. The flow is as follows: when requesting support, the customer is presented with a text box to describe their support request. The AI takes the first pass at the support request and attempts to resolve it. Though there would theoretically be an upper limit on API calls and tokens, since I chose to focus on human staffing, I will hand-wave over that limitation and say that Level One and Level Two Customer Support professionals are the two resource constraints. The event graph accounts for customers balking at the point where they would enter the queue for human support. I also accounted for customers' reneging as they waited for level-one support. There are two possible paths to resolution: resolution by level one support, escalation to level two, and resolution by level two support. See Figure 1 in the Appendix.

Next, I coded the simulation in the Python library Simpy using the customer support team members as my two resource constraints. Though, in reality, the support spike lasts for weeks or months, I am constraining the simulation to a single eight-hour workday and then running it multiple times. For the sake of simplicity, I also categorized AI resolution rates into two broad categories, simple and complex, which allows me to express a higher level of optimism for the ability of the AI customer support agent to resolve straightforward cases. The company has set an SLA for 2 minutes for Level 1 support and 5 minutes for Level 2 support. Given varying AI resolution rates, I set up the code to predict how many customer support professionals I needed for each level to stay within those SLAs. I did set a minimum staffing level of 2 for each level because, realistically, you need to allow for support reps to take time off and at least one person still being available to work.

Results and Management Recommendations

To stay within 2 minutes wait for SLA for level one support and 5 minutes wait for SLA for level two support, there were a variety of possible staffing combinations, especially since I ran the code many times in succession. For my first simulation, I ran an optimistic scenario with a simple AI case resolution rate of 0.7 and a complex resolution rate of 0.2, resulting in a total minimum viable staffing of 6 L1 agents and 2 L2 agents for eight agents. I also ran a less optimistic scenario where the AI customer support agent could only achieve a 0.3 resolution rate. I was surprised that I only increased my L2 agent, which was needed by one agent. See Figure 2 in the Appendix.

Further Research

If I were to extend this project to my current employer, I'd need to account for several complexities that this project doesn't cover. There would likely be reneging and balking at multiple points in the customer journey. Unfortunately, most cases are resolved in hours or days, not minutes, and I'd need to account for business hours and flex time during the spike season. I'd also need to deal with the percentage of customer support cases that cycle through the queue more than once before they are entirely resolved. The overall volume of customers and the number of agents would be much higher. I'd need to account for sick days, time off, and overtime to understand the dollar implications of various AI customer service resolution rates.

Conclusion

In conclusion, this project was more fun than I expected. At work, I was only responsible for building the AI solution to deflect customer support requests. I was not involved in calculating the staffing model implications and was only told that the project was a resounding success in reducing operational costs. It was interesting to play out some hypothetical scenarios to better understand the real impact of an AI customer support agent on customer experiences and wait times.

Appendix

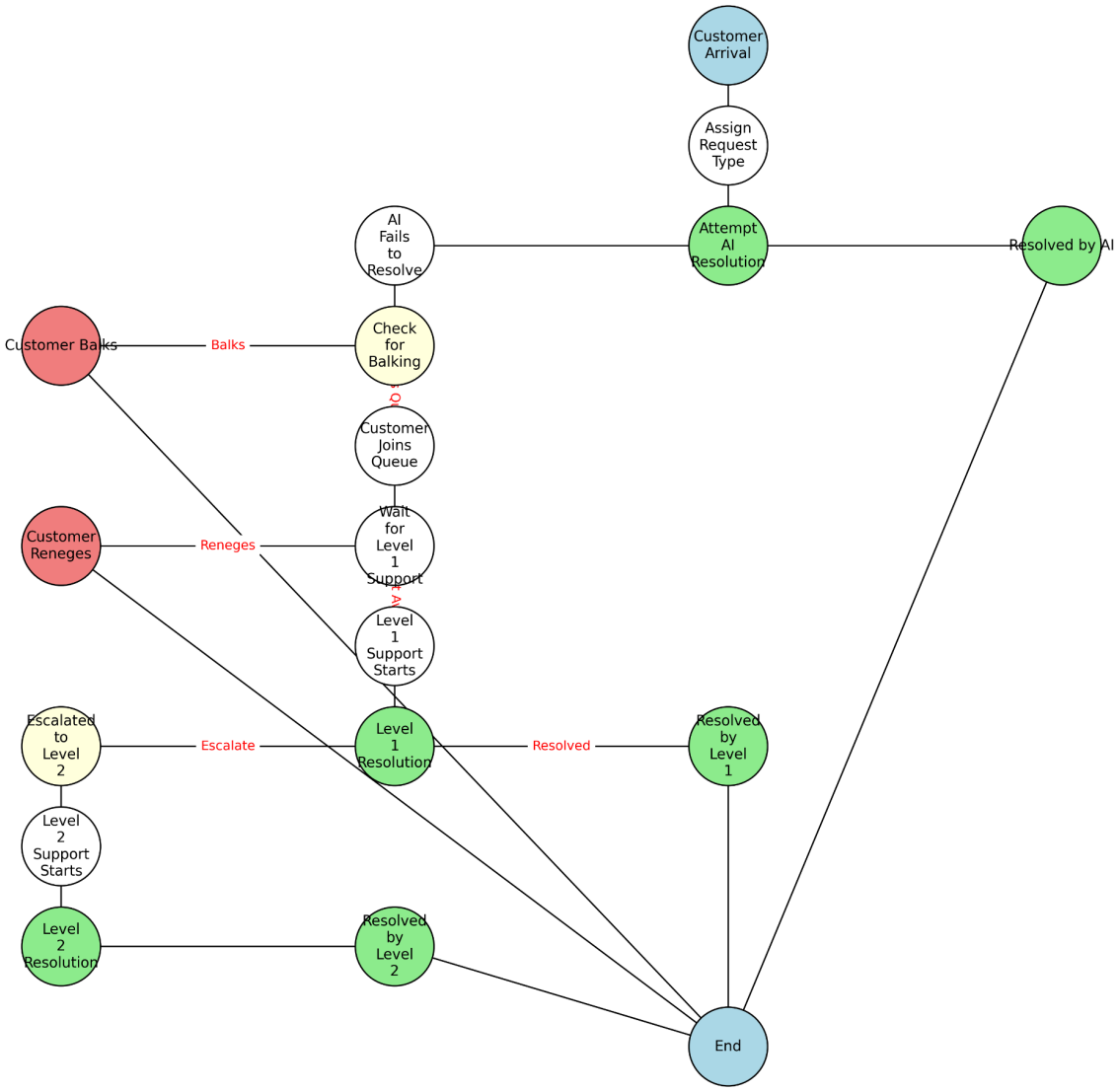

Figure 1

Simple AI Resolution Rate	Complex AI Resolution Rate	L1 Agents	L2 Agents	Avg Wait L1 (min)	Avg Wait L2 (min)	Total Agents
0.7	0.2	6	2	1.91	2.76	8
0.3	0	6	3	1.99	2.13	9

Figure 2

References

Beazley, David M. 2022. **Python Distilled.** Boston: Addison-Wesley/Pearson Education. [ISBN-13: 978-0-13-417327-6] Chapter 6, Generators, pages 139–152. Available on Course Reserves.

SimPy Developers, "**SimPy Documentation**," Version 4.0.1, 2019, accessed November 25, 2024, https://simpy.readthedocs.io/en/latest/.