A Shopping Agent for Addressing Subjective Product Needs

Preetam Prabhu Srikar Dammu[*]
University of Washington
preetams@uw.edu

Omar Alonso
Amazon
omralon@amazon.com

Barbara Poblete
Amazon
bpoblete@amazon.com

Abstract

In e-commerce, customers often struggle to find relevant items when their needs involve subjective properties characterized by personal or collective perception, tastes, and opinions, which are typically not captured in catalog data. This challenge is particularly pronounced in event-based scenarios like gifting, where selecting the right product involves complex subjective reasoning. Customer reviews can be a valuable source of subjective information to bridge this gap. Consequently, customers often spend significant amount of time navigating multiple products and reading numerous reviews to find suitable gifts that meet their needs. In order to reduce the effort involved, we propose an agentic approach driven by large language models to streamline this process by autonomously executing various user actions. These include computational tasks like vagueness detection and subjective product needs extraction, conversational interactions to gather missing user information, and web browsing actions that search for product details, reviews, and review images. Additionally, the agent employs generative actions to synthesize gifting ideas and explanations, helping users discover suitable products more efficiently. The proposed approach not only reduces the cognitive burden on users but also facilitates the exploration of a wider range of products. Our solution highlights the potential of autonomous agents to handle subjective queries in e-commerce, enhancing personalization, product exploration, and selection in a user-centric manner.

CCS Concepts

• **Information systems → Web searching and information discovery**; **Information retrieval**.

Keywords

LLMs, Web Navigation Agents, Agents, Personalization

ACM Reference Format:
Preetam Prabhu Srikar Dammu, Omar Alonso, and Barbara Poblete. 2025. A Shopping Agent for Addressing Subjective Product Needs. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining (WSDM '25), March 10–14, 2025, Hannover, Germany.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3701551.3704124

1 Introduction

In e-commerce, customers often encounter challenges when searching for products with subjective needs. Search queries during such times tend to be vague or consisting subjective information not found in product catalog, leading to suboptimal results from traditional search engines or platforms. This increases customer effort and results in a less-than-ideal shopping experience. Current chatbots and search systems are not well-equipped to handle these exploratory and subjective e-commerce tasks efficiently, as they typically rely on catalog data.

To address this issue, we propose an agent framework that addresses subjective product needs (SPN) by exploring a wide range of products, reviews, and generating gifting ideas. The proposed agent is capable of executing live searches, interacting with users to gather missing information, and autonomously performing actions on behalf of the customer, reducing user effort and enhancing the overall shopping experience.

Autonomous software agents for executing complex multi-step IR tasks have been explored for decades [2]. However, these early efforts had limited success due to their inadequate natural language understanding and reasoning capabilities [11]. These shortcomings are now overcome by the advent of LLMs, leading to multiple recent studies exploring autonomous generative agents, which rely on LLMs as the driving component [3, 4, 6].

2 Related Work

Static generative agents replicate human activity patterns by automating tasks through an LLM with a predefined workflow [14]. This strategy has proven suitable for certain tasks such as question answering, perform multi-step web search, and generating programs [3, 4, 6]. In contrast, dynamic agents employ techniques like imitation learning [5] and reinforcement learning from human feedback (RLHF) [12] to predict the next action using special tokens [7].

Generative agents equipped with web interaction capabilities have demonstrated their suitability for carrying out tasks that involve multiple web surfing steps, such as collaborative search [3], question answering [1, 9], and shopping on e-commerce websites [1, 13]. The closest approach to the proposed method is the WebShop agent [13], which takes detailed instructions of customer requirements and executes actions guided by product catalog data. However, customers may not always have clear requirements or specific products in mind, especially in exploratory scenarios such as gifting. In cases where the intent is vague or subjective, interacting with the user to obtain missing details is crucial [8]. Furthermore, relying solely on catalog data may be inadequate for meeting subjective needs, and incorporating signals from reviews could be valuable, as they offer a rich source of subjective information.

3 Framework of SPN Shopping Agent

We present the details of the agent framework and other operational aspects. In §3.1, we discuss the different facets of SPN considered in the proposed solution. Next, in §3.2, we present the typical

[*]Work done during internship at Amazon.

WSDM '25, March 10–14, 2025, Hannover, Germany

ACM ISBN 979-8-4007-1329-3/25/03
https://doi.org/10.1145/3701551.3704124

user workflow and the corresponding agent workflow designed to streamline and reduce user effort. In §3.3, we describe the various actions executed by the agent to implement the proposed workflow. In §3.4, we explain the approach used to identify the most suitable reviews that potentially address the user's intent and subjective needs. Finally, we outline the strategy for presenting product suggestions in a way that minimizes the user's cognitive load.

3.1 Subjective Product Needs

We capture five facets of SPN, and present their definitions below:

- *Subjective property*: User requirement mentions a subjective attribute, property, or categorization as part of their need. Examples: *sturdy* table, *colorful* dress, *large* lunchbox.
- *Event*: User requirement indicates an event. Events can be either general public events or personal milestones and life events. Examples: *Easter* dress, *pregnancy* clothes, *Christmas* sweater.
- *Activity*: User requirement mentions an activity for which the product should be adequate or good. Examples: *gaming* chair, *travel* pillow, *running* shoes.
- *Goal purpose*: User requirement mentions an objective, goal, or purpose that the product is intended to fulfill. Examples: *weight-loss* gear, office *organizer*, *sleep* supplements.
- *Goal audience*: User requirement mentions an audience, or describes a type or group of people for whom the product is intended. Examples: *boy* hiking shoes, gifts for *boyfriend*, toys for *children with autism*.

We developed an LLM-based classifier to detect the presence of SPNs and extract their corresponding values. The SPN classifier was prompt-tuned to align with human annotator performance by using 1K queries across five SPN categories, resulting in a total of 5K labels.

3.2 Workflows

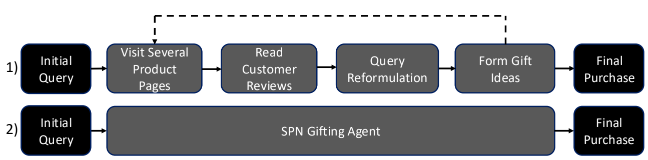

Figure 1: Flowcharts of (1) the typical user workflow and (2) the proposed agent workflow.

In contrast to shopping for personal needs, it is more common for users not to have a specific product in mind when shopping for others. While some cases are more exploratory than others, the gifting scenario is particularly suited to showcase the agent's ability to handle SPN. Therefore, for the purpose of this demo, we selected the gifting scenario to highlight how the agent can streamline product exploration and generate easy-to-process gifting ideas. Users might have vague concepts such as 'funny' or 'practical' and may seek the experiences of other customers who shopped for similar requirements and scenarios. As a result, they navigate through several product pages, read multiple reviews for each product, readjust their queries as they process new information, and form new ideas. They might repeat this process multiple times before arriving at a product that meets their needs. The workflow representative of this experience is shown in Figure 1 (1). A more desirable experience would involve the agent performing most of the heavy lifting of product exploration based on the user's SPN, thereby generating easy-to-process gifting ideas. The agent's workflow is shown in Figure 1 (2), where multiple time-consuming steps are replaced by the agent. This experience is achieved by equipping the agent with several actions, which are discussed in § 3.3.

3.3 Agent Actions

In order to effectively implement the proposed agent workflow, the agent performs several actions that can be grouped into four categories: computational actions, conversational actions, browsing actions, and generative actions (see Figure 2).

3.3.1 Computational Actions. To trigger the appropriate actions at the right time, we monitor the evolving task vagueness and requirements throughout the user session, which is handled by computational actions. The *Extract SPN* action calls the SPN classifier to obtain the five SPN values discussed in §3.1. The *Compute Vagueness* action triggers the computation of a vagueness score, which combines the upper funnel score and the weighted SPN score. The upper funnel score ranges between 0 and 1, where upper funnel queries like `electronics` get score closer to 1 and specific queries like `iPhone 16` get score closer to 0, and is computed using a BERT-based regression model. The vagueness score is computed as follows:

$$V = \alpha \cdot \left(1 - \sum_{i=1}^{5} w_i \cdot \text{SPN}_i\right) + \beta \cdot uf \tag{1}$$

where:

- V is the vagueness score that ranges from 0 to 1,
- α and β are weights that sum up to 1,
- w_i are individual weights for the SPN presence values SPN_i,
- uf is the upper funnel score.

Depending on the requirements of the use case, these weights can be adjusted accordingly. For the gifting use case demonstrated in this work, we use a higher α value of 0.8 and a β value of 0.2, as the gifting scenario is highly subjective. For the individual SPN weights, we assign 0.35 to both *event* and *goal audience* because they are the most relevant to the gifting use case, with the remaining weight distributed equally among the other SPN values.

3.3.2 Conversational Actions. When necessary, the agent needs to interact with the user to procure the required information, which is achieved using *Inquire Missing Info* action. As the goal is to reduce user effort, we only request additional inputs from the user when the vagueness score is higher than a threshold, which is 0.4 in this particular setting. We limit the questions to straightforward *wh* questions corresponding to the SPN values, as the answers to these questions are expected to reduce vagueness. Once the product suggestions are presented to the user, the agent facilitates further interaction using follow-up questions about the products.

3.3.3 Browsing Actions. Exploring products, reading corresponding reviews, and fetching customer images are accomplished through browsing actions. The agent calls functions to crawl product pages and extract the required information, such as product details and

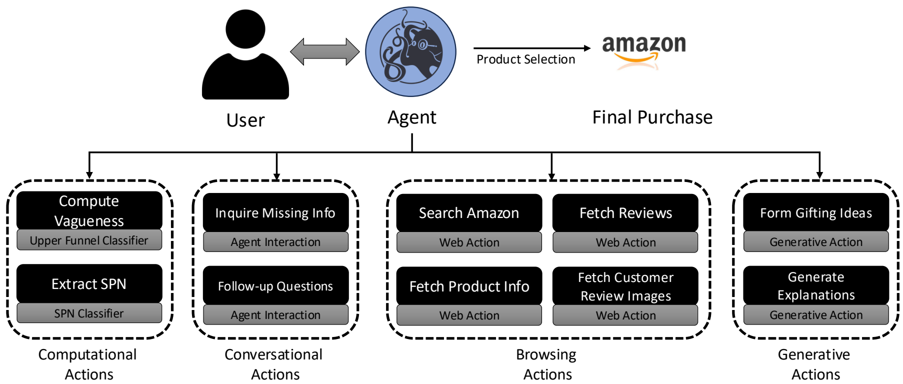

Figure 2: Architecture of the SPN Shopping Agent.

customer reviews, by processing the HTML code of the webpage, similar to the WebShop agent [13]. The different actions that fall under this category are presented in Figure. 2.

3.3.4 Generative Actions. The agent procures catalog and review information for numerous products, and this information needs to be presented to the user in a clear and concise manner that reduces cognitive load. To achieve this, we use two generative actions. The first action, *Form Gifting Ideas*, helps generate gifting ideas at a higher level of abstraction, such as product types. The second action, *Generate Explanations*, describes why a particular product is suitable and how it addresses the user's SPN.

3.4 Reviews Ranking

Presenting the most helpful review to the user is desirable, as customers rely on the experiences of others for subjective information, such as personal opinions about the product, which are typically not captured in catalog data. However, finding the most relevant review is non-trivial, as different users may find different reviews more helpful depending on their requirements. It is common for users to browse through multiple reviews to find the most helpful one. We address this challenge by ranking the reviews using a score that accounts for the SPN extracted from the user and agent conversation. Review scores are computed as follows:

$$R = \sigma\left(\alpha \cdot \left(\sum_{i=1}^{5} w_i \cdot \mathrm{SPN}_i\right) + \beta \cdot \mathrm{sim}(D, R)\right) \tag{2}$$

where,

- R is the review score that ranges from 0 to 1,
- α and β are weights that sum up to 1,
- w_i are individual weights for the SPN presence values SPN_i,
- $\mathrm{sim}(D, R)$ is the semantic similarity between the user description D and the review R,
- $\sigma(x)$ is the sigmoid function defined as $\sigma(x) = \frac{1}{1+e^{-x}}$.

Similar to the vagueness score (Eq. 1), the weights can be adjusted according to the use case. For this demo, we use the same weights as the vagueness score and S-BERT [10] to compute semantic similarity.

3.5 Effort Saved

On average, assuming the agent processes around 400 reviews, it would save around 2.67 hours of time for the user (note: users typically do not read 400 reviews). Apart from time, the agent also helps in reducing decision fatigue and promotes the exploration of a wider range of products.

$$T_{\text{saved}} = \frac{N_r \times W_r}{R_s} \tag{3}$$

where,

- T_{saved} = Time saved by the agent in minutes
- N_r = Number of reviews the agent reads (e.g., 400)
- W_r = Average words per review (e.g., 100)
- R_s = User's reading speed in words per minute (e.g., 250)

4 Demonstration

In the demonstration of the SPN Shopping Agent, we showcase how the agent simplifies the gifting process by automatically reducing user effort and converting vague requests into specific, actionable product recommendations. The user starts with a general idea about buying gifts, providing minimal information. The agent identifies the need for more specificity and engages in a brief conversation to gather additional details, such as the occasion (Halloween) and the intended audience (grand kids).

With this clearer context, the agent autonomously searches through multiple product pages, reads reviews, and compiles a curated list of Halloween-themed gift ideas suitable for the target audience. For example, it suggests products like "Spooktacular Halloween Treats for Grandkids", which is described as a pre-filled Halloween candy bucket with healthy snacks and sweet treats, or "Glow-in-the-dark Halloween tattoos for kids," which offers a fun and safe way to enhance the trick-or-treating experience. Each product is accompanied by relevant customer testimonials, which are selected based on their alignment with the user's subjective needs, such as festive appeal and suitability for young children.

Beyond product recommendations, the agent supports the user's decision-making process by responding to follow-up requests. For instance, when the user asks to see customer images for a specific product, the agent retrieves these visuals, providing real-world usage examples that enhance the product's appeal. In this case,

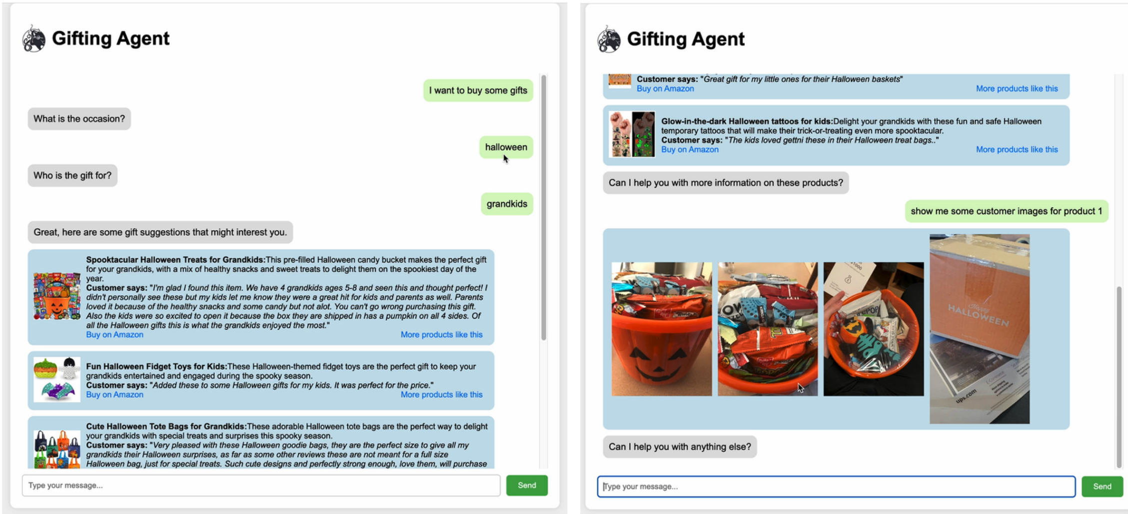

Figure 3: Agent gathers missing information before generating recommendations (left) and displays product recommendations while retrieving customer images upon request (right).

images of the pre-filled Halloween candy bucket were displayed, showing actual customer experiences, which further validated the product's relevance to the user's needs.

Throughout the entire workflow, the agent automates several key actions – gathering essential user information, exploring relevant products, summarizing customer reviews, and offering curated gifting suggestions. By reading through hundreds of reviews and ranking them based on how well they align with the user's needs, the agent not only saves the user significant time but also reduces decision fatigue by presenting a concise, easy-to-process list of products. This demonstration illustrates the agent's ability to streamline shopping for gifts, transforming vague inquiries into well-matched product suggestions with minimal effort from the user.

5 Conclusion and Future Work

While our current solution focuses on static workflows, which effectively mimic repetitive human tasks through predefined steps and conditions, we recognize the potential for further development. Future work could explore the integration of dynamic workflows that mimic more complex human decision-making, leveraging techniques such as reinforcement learning and action prediction. This would enable the agent to adapt more fluidly to evolving user needs and contexts, offering even greater personalization and efficiency in handling gift shopping queries. Identifying and implementing such dynamic features represents a promising direction for enhancing the flexibility and effectiveness of intelligent agents in this domain.

We present an agent that addresses SPN and demonstrate its capabilities in the context of gifting, a use case typically characterized by extensive exploration and subjectivity. The proposed framework showcases the ability of autonomous agents to handle subjective queries in e-commerce, enhancing personalization, product discovery, and decision-making with a user-centric approach. Finally, we highlight the potential of such an agent in reducing user effort and identify possible future directions.

References

[1] Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Sam Stevens, Boshi Wang, Huan Sun, and Yu Su. 2024. Mind2web: Towards a generalist agent for the web. *Advances in Neural Information Processing Systems* 36 (2024).
[2] Tim Finin and Anupam Joshi. 2002. Agents, trust, and information access on the semantic web. *ACM Sigmod Record* 31, 4 (2002), 30–35.
[3] Peiyuan Gong, Jiamian Li, and Jiaxin Mao. 2024. CoSearchAgent: A Lightweight Collaborative Search Agent with Large Language Models. In *Proc. of SIGIR*. 2729–2733.
[4] Izzeddin Gur, Hiroki Furuta, Austin Huang, Mustafa Safdari, Yutaka Matsuo, Douglas Eck, and Aleksandra Faust. 2023. A real-world webagent with planning, long context understanding, and program synthesis. *arXiv preprint arXiv:2307.12856* (2023).
[5] Ahmed Hussein, Mohamed Medhat Gaber, Eyad Elyan, and Chrisina Jayne. 2017. Imitation learning: A survey of learning methods. *Comput. Surveys* 50, 2 (2017), 1–35.
[6] Xiao Liu, Hanyu Lai, Hao Yu, Yifan Xu, Aohan Zeng, Zhengxiao Du, Peng Zhang, Yuxiao Dong, and Jie Tang. 2023. WebGLM: Towards an efficient web-enhanced question answering system with human preferences. In *Proc. of SIGKDD*. 4549–4560.
[7] Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. 2021. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332* (2021).
[8] Cheng Qian, Bingxiang He, Zhong Zhuang, Jia Deng, Yujia Qin, Xin Cong, Yankai Lin, Zhong Zhang, Zhiyuan Liu, and Maosong Sun. 2024. Tell me more! towards implicit user intention understanding of language model driven agents. *arXiv preprint arXiv:2402.09205* (2024).
[9] Yujia Qin, Zihan Cai, Dian Jin, Lan Yan, Shihao Liang, Kunlun Zhu, Yankai Lin, Xu Han, Ning Ding, Huadong Wang, et al. 2023. Webcpm: Interactive web search for chinese long-form question answering. *arXiv preprint arXiv:2305.06849* (2023).
[10] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proc. of EMNLP*.
[11] Urvi Shah, Tim Finin, Anupam Joshi, R Scott Cost, and James Matfield. 2002. Information retrieval on the semantic web. In *Proc. of CIKM*. 461–468.
[12] Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems* 33 (2020), 3008–3021.
[13] Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. 2022. Webshop: Towards scalable real-world web interaction with grounded language agents. *Advances in Neural Information Processing Systems* 35 (2022), 20744–20757.
[14] Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Haonan Chen, Zhicheng Dou, and Ji-Rong Wen. 2023. Large language models for information retrieval: A survey. *arXiv preprint arXiv:2308.07107* (2023).