Journal of Informatics and Web Engineering
Vol. 5 No. 1 (February 2026) eISSN: 2821-370X

Ontology-Based E-Commerce Recommender System: A Hybrid Semantic Filtering Approach

Jocelyn Pua[1], Su-Cheng Haw[2], Lucia Dwi Krisnawati[3], Shaymaa Al-Juboori[4*], Gee-Kok Tong[5]
[1,2,5]Faculty of Computing and Informatics, Multimedia University, Persiaran Multimedia, 63100 Cyberjaya, Malaysia
[3]Faculty of Information Technology, Universitas Kristen Duta Wacana, Agape Building 3rd Floor, Dr. Wahidin Sudirohusodo St. 5-25, Yogyakarta 55224, Indonesia
[4]University of Plymouth, Drake Circus, Plymouth PL4 8AA, United Kingdom
*corresponding author: (shaymaa.al-juboori@plymouth.ac.uk; ORCiD: 0000-0001-5175-736X)

Abstract - The rapid growth of e-commerce has led to product overload, which has resulted in personalized product discovery becoming a crucial problem for consumers. Classic recommender systems, which rely heavily on content-based filtering or collaborative filtering, tend to face well-known problems such as cold start, data sparsity, and ignorance of semantics. Such constraints frequently result in irrelevant or redundant suggestions, thereby lowering the user satisfaction and conversions. This study presents a hybrid ontology-based e-commerce recommender system that combines symbolic reasoning with deep semantic matching. The system is based on a Neo4j graph database to capture structured product relationships and is combined with sentence embedding models (MiniLM) to compute the semantic similarity between user queries and product data. For semantic matching, cosine similarity is used, and for ontology-based filtering, graph relationships, such as SAME_CATEGORY, SIMILAR_PRICE, and SAME_MANUFACTURER, are employed. An e-commerce dataset that was cleaned and pre-processed was used to test the system. The performance was measured using the following metrics: precision, Recall, F1-Score, and accuracy. The performance was measured using the following metrics: precision, Recall, F1-Score, and accuracy. The proposed system achieved a precision of 0.95, recall of 0.93, F1-Score of 0.94, and accuracy of 0.94, demonstrating that the hybrid approach yields superior recommendation quality compared with using a single method.

Keywords— Ontology, Recommender System, e-Commerce, Semantic Similarity, Hybrid Filtering, Graph Database.

Received: 27 June 2025; Accepted: 06 September 2025; Published: 16 February 2026

1. INTRODUCTION

Recommender systems (RS) play a key role in the operation of the modern digital commerce space, providing customers with personalized experiences and guiding them in their shopping journey [1]. With increasing demand from consumers to engage with online platforms, intelligent systems that understand user intent and product context are increasingly important. Traditional recommender approaches such as content-based filtering (CB) [2] and collaborative filtering (CF) [3] work to some extent, but suffer from the bottleneck of data sparsity, cold-start problem, and overfitting.

Journal of Informatics and Web Engineering
https://doi.org/10.33093/jiwe.2026.5.1.6

Published by MMU Press. URL: https://journals.mmupress.com/jiwe

The proliferation of semantic technologies and knowledge graphs presents new opportunities for improving the quality of recommendations [4], [5]. Ontology-based recommendation systems utilize a domain model that defines active knowledge and can express the relationship between subjects in a natural form [4]. When such systems are combined with more sophisticated natural language processing (NLP) techniques (such as sentence embeddings), they can encode deep user intent via a query and map it to products more efficiently. The goal of this study was to develop a hybrid system that integrates ontology-based filtering with semantic similarity.

In this study, we present the design, development, and evaluation of an ontology-driven e-commerce recommendation system that leverages Neo4j and sentence-transformer-based embeddings. It has two recommendation accuracy and context awareness problems and provides a scalable and intelligent recommendation solution for e-commerce platforms. To aid comprehension, this article proposes the use of diagrams to enhance clarity and deepen the system design, database relationships, and UI layouts.

2. LITERATURE REVIEW

2.1 Overview of RS

RS have become increasingly relevant in influencing human behaviours in larger digital spaces and accelerating decision-making in a broad spectrum of digitally mediated scenarios. These systems exploit user behaviour, preferences, and context to offer personalized recommendations, leading to higher consumer satisfaction and user engagement in many industries such as social networking, e-commerce, entertainment, and healthcare [6]. There are four significant RS techniques: CF, CB, Demography-based Filtering (DB), and Knowledge-based Filtering (KB). CF, for example, identifies patterns in behaviour based on past interactions to recommend items similar to those liked by users. Alternatively, CF recommends identical items according to their attributes and user actions. By contrast, DB and KB can use the demographic information of users in combination with their prior knowledge of item attributes to generate personalized recommendations [7]. More recently, hybrid approaches have demonstrated success by improving the accuracy of recommendations while addressing common challenges such as the "cold start" problem, which results when new users or items lack the data needed to generate reliable recommendations [8]. Driven by the advancements in artificial intelligence, machine learning, and deep learning. RS systems have advanced, leading to more sophisticated recommendations that extend far beyond proposing products based on such user preferences, taking into account the context and capabilities to provide results in real time. RS have become a crucial asset for product personalization, especially for e-commerce sites, as they play a critical role in user engagement, customer loyalty, and overall sales performance [9].

In addition to content improvements and context-based enhancements, e-commerce companies have found that how the recommendation is framed makes a significant difference in user adoption. Using recommendation framing, we capture the messaging and signage of the recommended product. Popular tactics include "Customers who viewed this also viewed" (norm-based framing) and "Compare similar items" Compare similar items (comparison-based framing). Norm-based framing is more effective than comparison-based framing in increasing the click-through rate in most scenarios when fewer highly substitutable items are recommended [10]. Norm-based recommendation framing instigates a sense of societal approval that makes the recommended items look superior and fills consumers with trust and fair beliefs. This context provides little guidance, allowing consumers to actively explore recommendations, mainly if the products are relevant to their current exploration and browsing tendencies. By contrast, framing via comparison can reduce user engagement by increasing similarity across products, especially if users view recommendations as redundant to their original selections. Finally, the experiment shows that balancing the amount and kind of recommendations is crucial for maximizing the impact of norm-based framing on customer behaviour [10].

This insight emphasizes the importance of selectively presenting recommendations to enhance perceived value and relevance. An e-commerce system that adopts framing strategies appropriate to user expectations, choices, and features of recommended goods can maximize user engagement and satisfaction. RS offers various benefits for customers and retailers in the e-commerce domain. Users smooth the online shopping experience by showing and recommending a curated list of products that closely match each user's taste, so that they do not have to work as hard as before to find something they would like. Personalized suggestions result in an elevated use experience, use, and return visits, facilitating an enhancement in the purchasing process. It provides up-selling and cross-selling opportunities for RS retailers in customer acquisition and retention to help retailers. By analysing details from the client interaction, RS assists e-commerce platforms to discover trends in the market, keep pace with the progression in consumer preferences, and so on [11], [12].

2.2 Hybrid Filtering Technique

Hybrid-based filtering (HB) is a flexible and adaptive method in RSs that merges the benefits of different recommendation methods and diminishes the weaknesses of all recommendation types. This approach makes the systems more accurate, diverse, and robust by melting methods, such as CF for CB. The advantages of the hybrid method for addressing problems in cold-start challenges, data sparsity, and a limited variety of recommendations are particularly emphasized [13], HB combines the best of both worlds, leveraging the strengths of each constituent technique. CB depends on the features of items and user preferences, whereas CF addresses the behavioural tendencies of user societies. Employing these techniques helps HB systems support diversity and, thus, keep their suggestions functional. This is an essential factor in domains such as e-commerce because they want to serve their users with personalized tips and novel suggestions.

One of the most common examples of HB is weighted hybridization, which combines the scores of several techniques by assigning weights to their contributions according to their precision or relevance. For example, CF may be preferred if collaborative data are abundant and isotonic, whereas interaction is sparse, and CB in the opposite case. The ability to assign weights dynamically based on specific contexts and user needs allows the system to adapt and strengthen when necessary [14]. The second method applies hybridization switching, which dynamically chooses the best-suited method according to the user profile and nature of the task. This provides an example of how recommendations for a new user registered in a system at the initial stage under CF would be specific to its preferences in a stage, as there will be no interaction data, and only after the user has interacted more with that system and the generated interaction data will CF overtake its recommendations with community-based predictions [8].

A second typical technique variation is mixed hybridization, in which recommendations of more than one technique are presented in parallel, offering a multi-directional space for user exploration. For example, a streaming platform may recommend content based on the user's viewing history, and the trending content is curated using community data. In addition to improving the user's experience, it also allows for a more immersive browsing experience [7]. Similarly, cascade hybridization works serially, and one process refines the output of the other. For instance, a possible hybridization is using a CF algorithm to find a set of high-volume items, and then only running the CB model on those popular items to find all users whose histories align.

The ingenuity of hybrid systems provides strong solutions to the cold-start problem, which is a significant drawback of standalone methods. The CB can use item metadata for new users or items to provide the recommendation information available at the outset. In contrast, the recommendations by CF can improve over time as user files are built. The best part is that the dynamic nature of user preference changes can be used for different user stages [8], and the benefits of HB systems are the novelty and diversity of recommendations. The HB approach could combine CB's similarity-based recommendations with the strength of the CF model to reveal latent preferences, creating both familiar yet pleasantly novel recommendations, thus enhancing usage and satisfaction [14].

However, challenges also confront HB. Some techniques are computationally intensive, require more computing resources, and it is difficult to maintain their integration. There is all tuning that goes into finding the right balance between methods, such as the weights in a weighted hybrid or when to switch techniques in a dynamic system. These challenges highlight the need for careful design and testing to maximize the HB model performance [7].

Despite these challenges, HB systems are used in a range of sectors. An example of this can be seen in e-commerce, where Amazon employs HB interventions to tailor its products at the recommendation level, implementing the CB algorithm that enhances product descriptions according to user preferences and the CF technique that finds similarities between users. Netflix uses CB to monitor user content consumption and CF based on genres or directors. For example, in healthcare, HB systems combine KB and group knowledge to recommend personalized treatment regimens for patients, based on their medical and demographic features [8], [9].

Such a hybrid significantly enhances RSs and addresses the weaknesses of each approach individually. HB models unify the CB and CF methods to pursue sustainable accuracy, diversity, and adaptivity trade-off. The field of HB systems is continuously evolving thanks to advancements in machine learning, NLP, and knowledge graphs, which are likely to contribute to personalized and significant recommendations across various domains in the future [7], [9].

Ultimately, HB presents itself as a new frontier in RS, providing a versatile and robust solution to the limitations of the existing approaches. As deep learning, knowledge graphs, and other cutting-edge technologies become more mainstream, the potential of HB systems will become even more significant.

2.3 Semantic-Based Filtering Technique

These techniques use semantic-based filtering to understand the relationship with entities, usually in ontologies or large semantic models. Demonstrating the capabilities of semantic filtering on personalized recommendation systems, Shang et al. merged contemporary advanced Large Language Models (LLM), namely Roberta, were used to assess the expression of user preferences and item semantics [15]. They obtained adequate enhancements in recommendation precision, specifically for long-tail items, and subtle content understanding using contrastive learning and fine-tuning on domain-specific datasets [16].

Ontology-based semantic models are popular for capturing explicit information about the relationships between users, items, and contexts [17]. Semantic reasoning significantly contributes to the quality of recommendations, Zhang et al. ensuring that RSs can comprehensively understand users (intent/user preferences) at a much finer grain, thus increasing the recommendation system's ability to recommend tailored (specific to user context) ones [16]. Moreover, semantic filtering effectively captures the intricacies of user-item relationships if there is sufficient ontological or textual information [18]

However, a limitation of these studies is the computational burden of maintaining and updating semantic representations, particularly when dealing with dynamic or large datasets. LLM holds great potential for integration, as it can automate semantic feature extraction and lessen the need for manual ontology design. Future work in semantic filtering will probably focus more on scaling up but also on keeping the technique interpretable.

Moreover, two common approaches are used in semantic-based filtering, including the ontology-based filtering technique and the graph-based filtering technique, elaborated as follows.

Generally, ontology-based filtering is a specialization of semantic filtering, in which domain-oriented ontologies serve as a basis for representing relations between data entities. Javed et al. discussed how ontology could offer a well-defined and interpretable knowledge framework for recommendation engines [19]. Ontology-based methods enable interoperability using technologies such as the Resource Description Framework (RDF) and Web Ontology Language (OWL), and provide reasoning support with the use of rules, leading to better decision-making in an e-commerce context.

Shang et al. highlighted the complementary role of LLM in complementing ontology-based filtering [15]. Ontologies represent an a static relational structure, whereas LLM extracts implicit preferences based on a dynamic adjustment to a user's history of implicit language responses. Ontology and LLM hybridization combine the strengths of ontologies to resolve the cold-start issue with the popularity of LLM to provide better recommendations across domains.

Ontology-based methods still encounter limitations such as the manual construction of ontologies, which are time-consuming and insufficiently scalable across different domains. Ontology alignment across multiple datasets remains a technical hurdle to overcome. Future research will involve automating the process of creating ontologies and including multimodal data sources to widen the scope of ontology-based systems.

On the other hand, graph-based filtering uses the information inherent in a recommendation dataset's relationships by framing it as a graph in which nodes are users or items and edges represent interactions or relationships. This method successfully captures higher-order relationships that traditional models have neglected. Shang et al. highlighted the efficiency and scalability of LightGCN, a graph-based deep-learning model suitable for large-scale datasets [15]. LightGCN uses high-order user-item interaction aggregated with the help of layers to achieve a good trade-off between computational complexity and accuracy.

Zhang et al. introduced Graph Convolutional Networks (Graph CNNs), a straightforward model for processing graph data by applying convolutional operations [16]. These methods formulate the recommendation as an instance of link prediction, incorporating user- and item-side information (e.g., social networks and item relationships) for the task. The Graph CNNs are scalable, easily interpretable, and highly effective, as they are tested on real-world data, such as Pinterest.

Graph-based filtering excels in understanding complex associations and managing large datasets but is vulnerable to sparse data cases such as new users or items. The scalability of this approach is still a challenge due to the computational requirements of large graphs. Hybrid models that utilise semantic understanding and graph-based methods offer interesting solutions to these challenges.

2.4 Summary of Literature Review

The summary is presented in Table 1, which delineates the key aspects and findings of the research conducted in the field.

Table 1. Summary of Related Works in E-Commerce Domain

Ref.	Paper Title	Findings
[4]	Predicting User Behaviour in Electronic Markets Based on Personality-Mining	Personality-driven product recommendation framework enhances relevance by analysing user traits from social networks.
[5]	Applied Fuzzy and Analytic Hierarchy Process in Hybrid Recommendation for E-CRM	Integrates fuzzy logic with hybrid recommendation approaches, enhancing decision-making in customer relationship management.
[6]	E-Commerce Trends During COVID-19 Pandemic	Emphasizes the crucial role of e-commerce RSs in addressing growing online consumer demand during the pandemic.
[8]	A Systematic Study on RSs in E-Commerce	Provides an overview of RS techniques and challenges; highlights the need for hybrid methods to improve accuracy and personalization.
[10]	Effectiveness of Product Recommendation Framing on Online Retail Platforms	Shows that norm-based framing increases click-through rates compared to comparison-based framing, enhancing user engagement when fewer, substitutable items are recommended.
[11]	Enhanced Product Recommendations Based on Seasonality and Demography in E-Commerce	Incorporate seasonality and demographic factors into hybrid recommendations, improving personalization and engagement.
[12]	Recommendation Systems: Algorithms, Challenges, Metrics, and Business Opportunities	Reviews RS methods, highlighting hybrid models as robust solutions for cold-start and diversity challenges.
[20]	A Fuzzy Recommendation System Using Sentiment Analysis and Ontology in E-Commerce	Enhances recommendation relevance using ontology structure and sentiment analysis, especially effective in handling noisy or ambiguous user data.
[21]	OntoCommerce: Incorporating Ontology and Sequential Pattern Mining for E-Commerce	Demonstrates the role of ontology evolution and pattern mining to dynamically adjust recommendations based on user interactions, improving adaptability and system learning over time.

The development of RSs has undergone several cycles, including early CB and CF, in which items are recommended to users based on item content or user behaviours. Although these traditional methods allow for simple personalization, they are often subject to problems of data scarcity, cold start, and overfitting to user preferences [1], [22], [23]. Hybrid filtering methods have recently been introduced to address the drawbacks of the former, through the combination of one or multiple filtering techniques, resulting in a balance between diversity and recommendation relevance [24]. However, these techniques are generally based on shallow data representations and are not semantically aware of rich product relationship models [25].

Ontology-based methods have received increasing interest in literature owing to the inclusion of structured domain knowledge in the recommendation process [4],[5]. Ontologies represent the semantic correlations among products, categories, and manufacturers, and can be used for context-aware filtering, which goes beyond basic item similarity [4]. Wang proposed Big Data Mining for ontology-based e-commerce recommendations and showed that the accuracy improves because of structured semantic reasoning [19].

Advancing this area of investigation, Karthik and Ganapathy presented a hybrid recommendation approach that integrated fuzzy logic, sentiment analysis, and an ontology structure to address uncertain user preferences [20]. They found that adding ontology knowledge increased the relevance of their recommendations and user satisfaction, especially with noisy and ambiguous datasets. Additionally, Mustafa et al. studied ontology evolution as well as content-based and CF for dynamic and adaptive recommendations [21]. Their contribution is to demonstrate through their implementation how growing the knowledge graph can affect the tuning of personalization as the system was exposed to user interactions and learned over time.

Concurrently, in the context of NLP, the advent of sentence embedding-based semantic filters has offered new opportunities. By transforming product descriptions and user intentions into vector representations, these models encode deep contextual similarities that traditional keyword-matching methods miss. Combining ontology reasoning and semantic embeddings, as proposed in this study, serves as a possible hybrid between symbolic and statistical reasoning for more coordinated and scalable recommenders.

3. RESEARCH METHODOLOGY

3.1 Flow of Research Methodology

Figure 1 illustrates the flow of this research methodology. The first step was to research the background to collect relevant information on the subject matter. First, a background study was conducted. In this phase, information related to the research topic was gathered. We then identified the research questions and set the main objectives of the study. We conducted a thorough examination of literature and research studies. This work helps us understand existing works and current techniques and identifies the gaps. These stages help to formulate a suitable and effective new approach for research. Finally, the interpretation of the results from practical work leads to the conclusions.

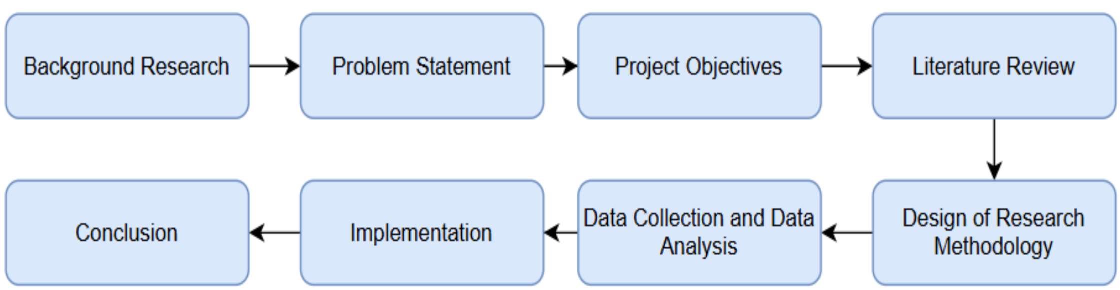

Figure 1. Flow of Research Methodology

3.2 Dataset

The dataset used for this project includes a comprehensive set of e-commerce products explicitly collected to develop an ontology-based RS. It comprises 10,000 records across multiple product categories, attributes, and user interactions. Descriptions of the attributes of this dataset are listed in Table 2.

3.3 Data Cleaning

For analysis, representation, and integration within the ontology-based RS, the dataset collection undergoes data cleaning to become consistent, complete, and usable. The steps taken were as follows.

Table 2. Attributes and Descriptions of the Selected Dataset

Attribute Name	Details	Field Type
uniq_id	Unique identifier for each product	String
product_name	Name of the product	String
manufacturer	Manufacturer or brand of the product	String
price	Product price	Float
number_of_reviews	Total number of reviews for the product	Integer
number_of_answered_questions	Total number of questions answered for the product	Integer
average_review_rating	Average customer rating for the product (1 to 5)	Float
product_information	Technical or detailed product specifications	String
main_category	High-level category of the product	String
sub_category	Sub-category under the main category	String
combined_description	Concatenation of product description and additional details	String

1. **Handling Missing Values**
   - **Critical Fields**: Missing values in price, number_of_reviews, and average_review_rating are filled with appropriate substitutes (e.g., mean, median, or default values such as 0).
   - **Non-Critical Fields**: Missing values in the manufacturer, Amazon _category_and_sub_category, and product_information were replaced with placeholders such as "Unknown Manufacturer" or "No Information Available."
2. **Data Type Standardization**
   - Numeric fields, such as price, number_available_in_stock, and number_of_reviews, were converted to their appropriate types (e.g., float or integer).
   - String fields such as main_category, sub_category, and description were standardized by removing extra spaces, special characters, and ensuring uniform capitalization.
3. **Outlier Removal**

   Extreme values in price and number_of_reviews were identified using the 99th percentile threshold and were removed to maintain data integrity.
4. **Text Cleaning**
   - Descriptions were cleaned by removing excessive whitespace, newlines, and inconsistent quotes.
   - Overly long descriptions exceeding 1,000 characters were truncated and appended with '...' to ensure brevity.
5. **Combining and Removing Redundant Columns**
   - The description and productdescription fields were combined into a single column, combineddescription, to simplify the analysis and reduce redundancy.
   - Irrelevant columns with excessive missing data, such as customer_questions_and_answers, were removed entirely.

6. **Handling Non-Numeric Entries**

   Fields with mixed data types, such as number_available_in_stock, were cleaned by extracting numeric values and converting them into integers.

7. **Check Duplicates Rows**

   After all the cleaning steps, the dataset was re-evaluated to ensure that no missing values remained in the critical columns.

Figure 2 shows the null values of the dataset before cleaning.

```
Missing Values per Column:
uniq_id                          0
product_name                     0
manufacturer                     6
price                            0
number_of_reviews                0
number_of_answered_questions     0
average_review_rating            0
description                    644
product_information             55
product_description            644
main_category                  677
sub_category                   677
dtype: int64
```

Figure 2. Null Values Prior to Cleaning

3.3.1 Feature Engineering

To support both ontology modelling and semantic recommendation, new derived fields were introduced:

- combined_text**:** A concatenated field containing the product name and description, used initially for TF-IDF vectorization and later as input for sentence embedding models.
- price_numeric**:** A cleaned numeric version of the price used for establishing `SIMILAR_PRICE` relationships.
- rating_cleaned**:** A numeric float version of the average rating used for `SIMILAR_RATING` relationships

These capabilities allowed us to factor in products with more context attributes and build Cipher queries that join similar entities using business logic and the meaning of the word. In general, the preprocessing step is responsible for achieving a structurally correct, semantically consistent, and prepared input for both the graph-based and embedding-based parts of the RS.

Figure 3 shows the null values of the dataset after cleaning while Figure 4 shows the duplicate rows in the dataset.

3.4 Data Cleaning

Exploratory Data Analysis (EDA) was executed to appreciate the dataset and prepare it for ontology modelling. The dataset contains 10,000 product records with features such as price, average review rating, review count, and product category. Price values were cleaned and converted to numeric types, while ratings (in string formats) were converted to decimals. The missing or non-numeric values in the number of reviews were considered by setting them as zero. The main and sub-categories were brushed up and unknown values were signalled. This prompted the threshold relationship in Neo4j by further analysing and guaranteeing that the data were clean and consistent for the recommendation tasks.

```
Missing Values After Final Cleaning:
uniq_id                        0
product_name                   0
manufacturer                   0
price                          0
number_of_reviews              0
number_of_answered_questions   0
average_review_rating          0
product_information            0
main_category                  0
sub_category                   0
combined_description           0
dtype: int64
```

Figure 3. Null Values in the Dataset After Cleaning

```
Number of duplicate rows: 0
```

Figure 4. Duplicate Rows in the Dataset

Figure 5 shows the price distribution bar chart used to visually analyze the attribute relationship. The lower price distribution shows that most of the products are under $25, and for the price range above $50, there are rarely any products, indicating a skewed dataset toward affordable products and a price-sensitive market. Most high-paid items are rare and may be targeted toward niche and/or premium consumers.

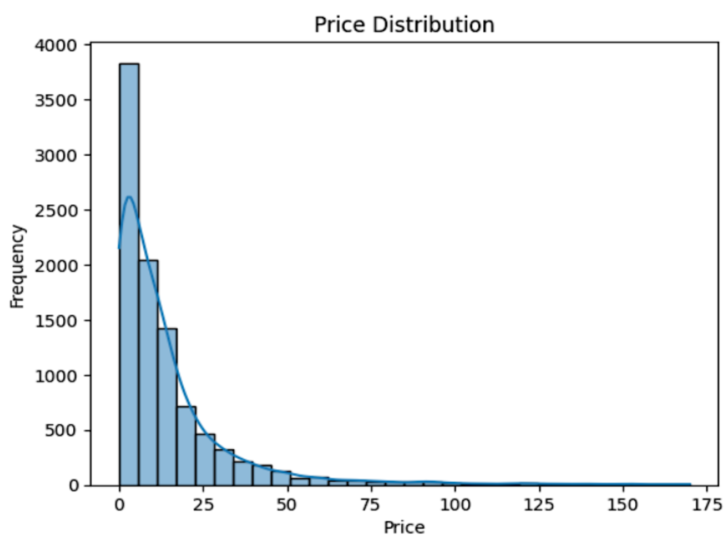

Figure 5. Histogram of Price Distribution

Figure 6 shows the price vs. average review rating scatter plot to visually analyze the attribute relationship. Analysis of price versus average review ratings revealed consistently high ratings (4.0 to 5.0) in most price ranges. The ratings were low across all price points, although there were clusters below $20 and a few expensive outliers. This finding indicates that price has little effect on mean ratings, whereas the disparity between expectations for high-cost products and the perception of quality in low-cost products may drive low ratings.

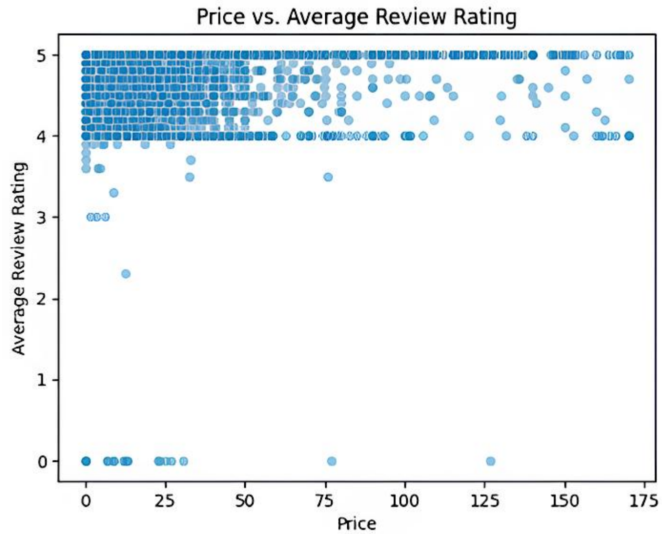

Figure 6. Price versus Average Review Rating Scatter Plot

Figure 7 shows the total reviews by category bar chart to visually analyze the attribute relationship. Another visualization provided a categorization of the total reviews by category, which told us that Games, Characters, and Brands had the highest total reviews, suggesting how much user engagement, which have categories with average reviews that can be potential drivers for providing general recommendations, while low-review categories can be improved through targeted promotions.

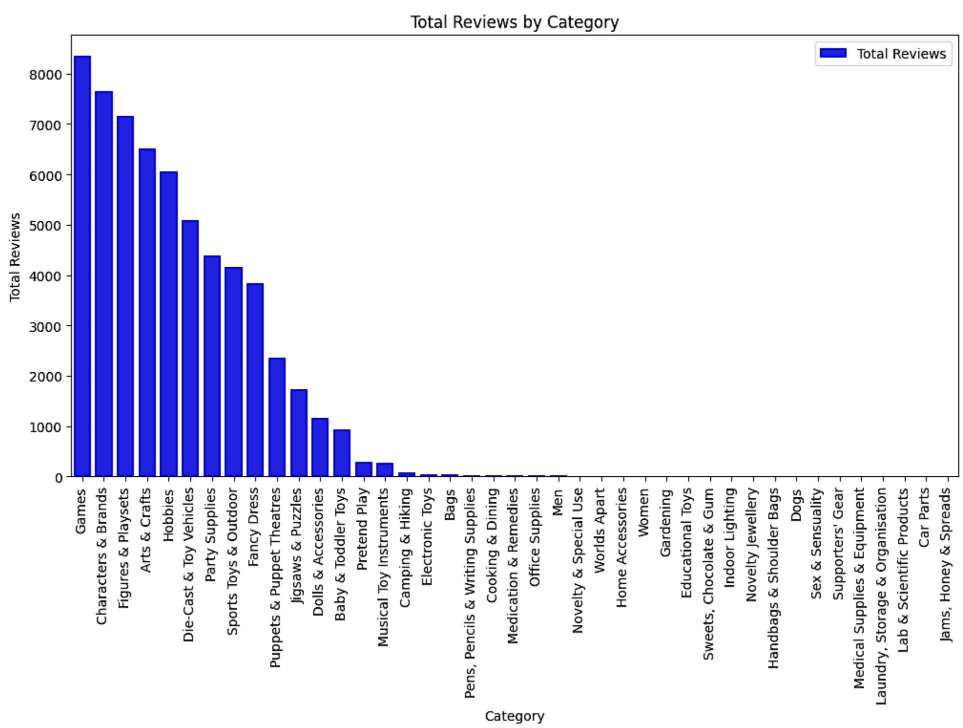

Figure 7. Bar Graph of Total Reviews by Category

Figure 8 shows the average ratings by category bar chart to visually analyze the attribute relationship.

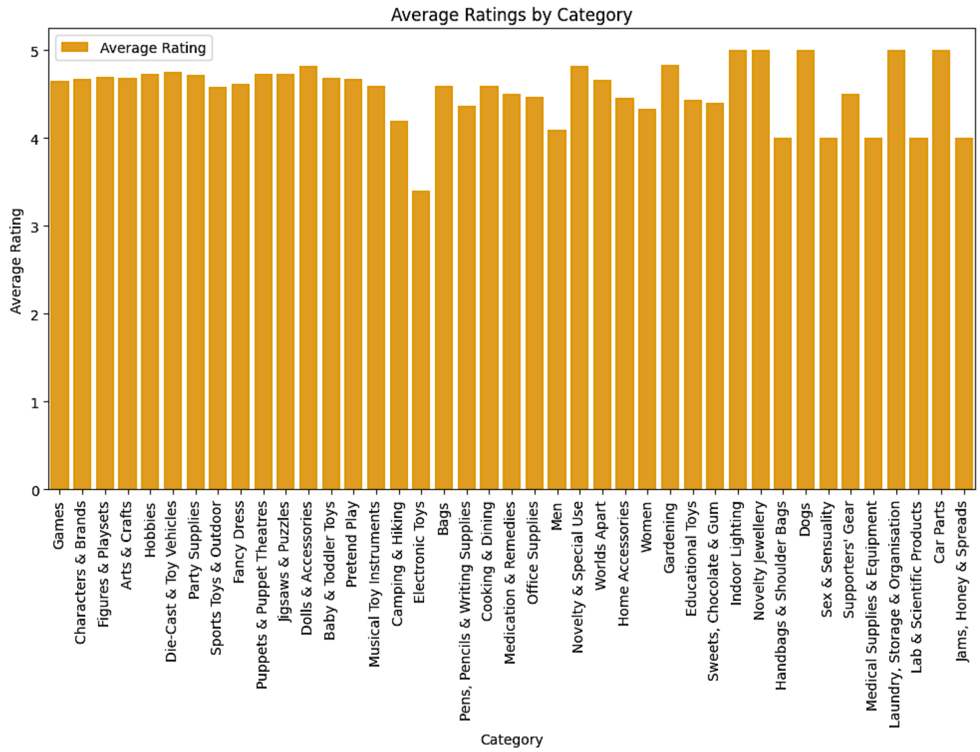

Figure 8. Bar Graph of Average Ratings by Category

With most categories averaging 4.5, the customer satisfaction remained high. The clear high ratings indicate that the dataset likely contains mostly favourably reviewed products; however, further exploration into slightly lower-rated categories is advised.

Figure 9 shows the low-rated products by the category bar chart to visually analyze the attribute relationship better. The bar chart above illustrates the distribution of low-rated products across the different toy categories. The category "Figures & Playsets' had the highest number of low-rated products, with more than four items receiving poor ratings. This is followed by "Games" and "Characters and Brands," each with three low-rated products. "Die-Cast & Toy Vehicles" has 2 low-rated products, while "Electronic Toys" has the fewest, with only 1 low-rated product. The chart highlights that "Figures & Playsets" stands out as the category with the most significant quality concerns based on customer ratings, suggesting a potential area for product improvement or quality review.

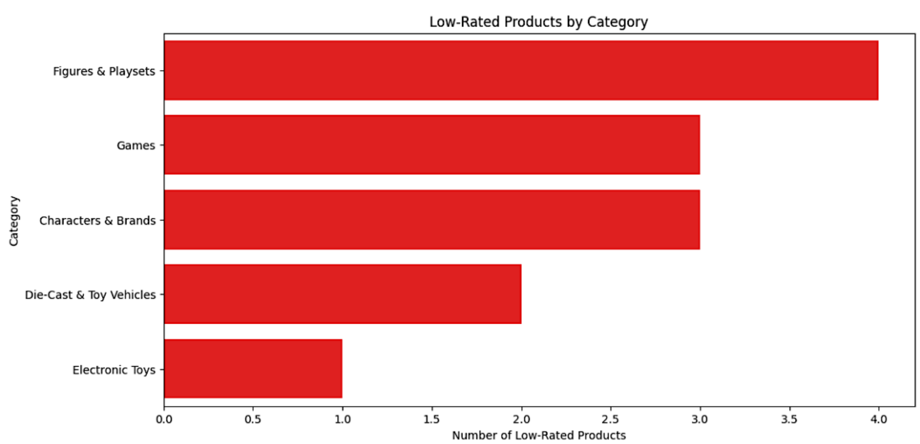

Figure 9. Bar Graph of Low-Rated Products by Category

Figure 10 shows the price distribution of the low-rated product bar chart to visually analyze the attribute relationship. Top-selling products suffered from low ratings in their categories, such as Figures & Playsets and Games. Most products are priced below $20, suggesting that popular categories may contain a mixture of high- and low-quality offerings. Improving quality issues for lower-priced items may reduce customer dissatisfaction.

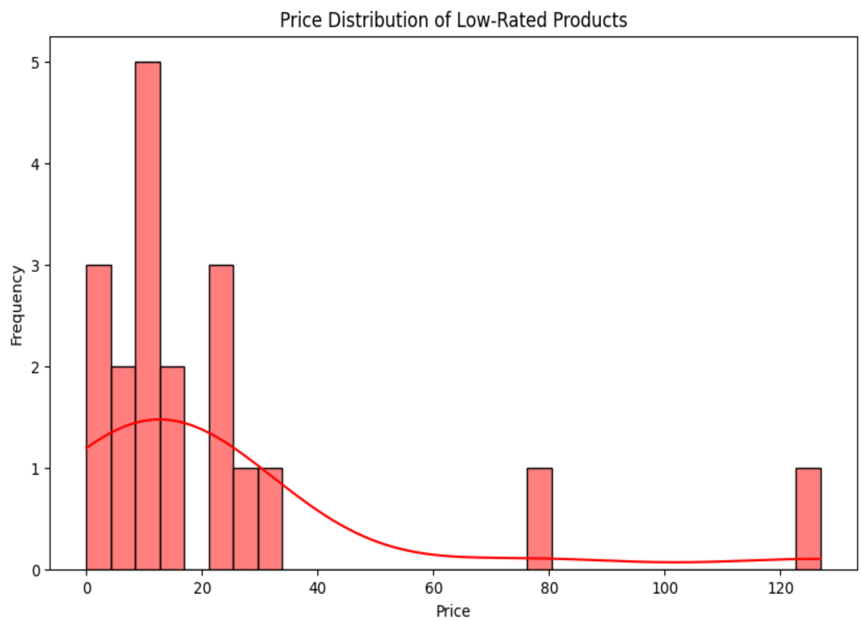

Figure 10. Price Distribution of Low-related Products

Figure 11 shows a bar chart of the top 10 Most Reviewed Products to visually analyze the attribute relationship. Other popular categories reflect these interests, with the top 10 most reviewed products including Toys, Sports Toys & Outdoor, Party Supplies, and Figures & Playsets. Highly reviewed products reflect significant customer engagement and are ideal review candidates.

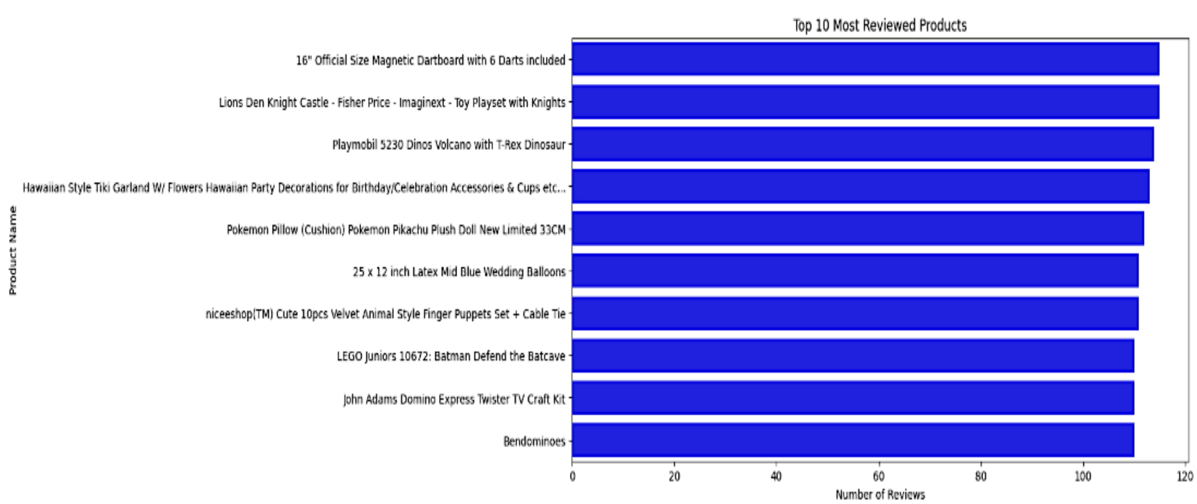

Figure 11. Bar Graph of Top 10 Most Reviewed Products

Figure 12 shows a correlation heatmap to visually analyze the attribute relationship. Correlation analysis showed that price had no significant correlation with ratings or reviews and a slightly negative correlation between reviews and ratings. One implication is that price is not a key driver of user engagement or satisfaction and that popular products can incur positive and critical user feedback.

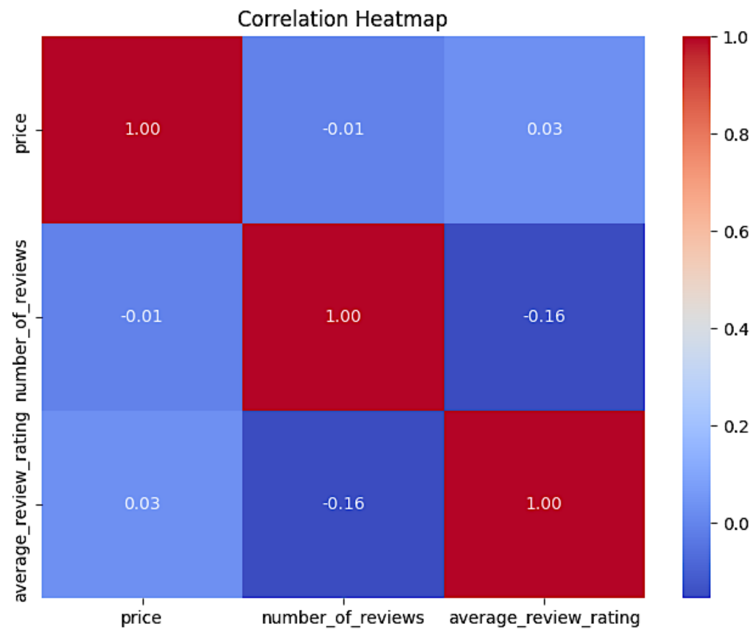

Figure 12. Correlation Heatmap

By strengthening high-engagement categories, such as Games, Characters & Brands, the recommendation system can be enhanced to improve the overall user experience. Looking into why the product is rated low, maybe the quality is poor, or the description of the product is not clear. Insights into pricing can also be useful; one can enable users to filter product listings to suit their price range and should also highlight value for items with high costs. This allows greater engagement and satisfaction on the part of the user to promote highly reviewed products or create a tailored strategy for relatively underrepresented categories. EDA allowed us to gain insights into user behaviour and product performance. These insights will inform the design of robust and user-centric recommendation systems. The future direction would include feeding this structured dataset into Neo4j to build an ontological-based recommendation model that utilizes hierarchical categories and user behaviour.

3.5 Data Modelling

The RS is instantiated as a hybrid model that integrates ontology-based filtering with the Neo4j graph database and semantic matching with sentence embedding. The mission is to recommend products in context by considering both the structured relation of the products and the knowledge learned through deep learning-based language understanding. The model consists of several integrated modules that cooperate to generate query processing, recommendation retrieval, and system performance evaluations.

Neo4j was selected as the underlying graph database for modelling the ontology, allowing convenient representation and querying complex relationships. Every product was assigned a unique ID, followed by the product name, category, sub-category, manufacturer, price, rating, and summary description. The products were generated as nodes. The relation mappings were modelled as Cipher queries under feature-specific similarity conditions.

The five major relationships introduced are:

1. SAME_CATEGORY: The same main category within the product used.
2. SAME_SUB_CATEGORY: Same product, sub-category (except 'Unknown').
3. SAME_MANUFACTURER: Same product manufacturer (not including 'Unknown').
4. SIMILAR_PRICE: Products with prices that fall within ±20% range.
5. SIMILAR_RATING: Products with average ratings difference below 0.5.

The information was then partially imported to avoid Memory Out bounds. This kept Neo4j chugging without a problem, even with the dense number of product relationships that were coming in.

The recommendation system was implemented using Python software. It begins by encoding the user's input (i.e., product name or search query) using the sentence transformer model (all-MiniLM-L6-v2). The system then conducts semantic matching by finding the most relevant product through cosine similarity on the query and product embeddings.

The system extracts ontology-based products from Neo4j by using the defined relationships of this object. These candidates were ranked based on their semantic similarity to the input from the user. Moreover, the system was provided with a set of negative samples (random products) to assess the classification capability of the system.

The results were ranked using a dynamic similarity threshold (the average cosine similarity across candidates). Items above the threshold were classified as relevant, whereas those below the threshold were classified as irrelevant. In this manner, the system calculates the Precision, Recall, F1-Score, and Accuracy. These statistics were obtained using the sklearn.metrics library, and they shed light on the correctness and robustness of the recommendations.

Streamlit developed a minimalistic Graphical User Interface (GUI) to allow interactive testing of the RS. The full system can currently run only locally, as the relationship count is over the limit imposed by the free tier of Neo4j Aura. However, this facility could bring the prototype end-to-end testing close to realistic conditions.

Overall, this implementation phase was able to establish a hybrid ontology-semantic recommendation system. The use of Neo4j allowed us to reason over the graph, and semantic similarity provided flexibility and contextual awareness. The prototype is now in good shape for more iterations of refinement, such as better GUI support and advanced relationship modelling in the subsequent development cycle.

Contrary to conventional recommendation systems, which learn from examples to tune their predictive models, such a hybrid system does not require parameter training. Rather, the "training" of the ontology graph in Neo4j produces the sentence embeddings of the training examples. The small_train_data. csv cs ontology, based on which the ontology is initialized. Every product in the dataset is represented as a node on Neo4j, and the edges between nodes (connections) are decided after fulfilling some comparisons based on manually defined thresholds such as price similarity, rating proximity, and matching categories or manufacturers (see Figure 13).

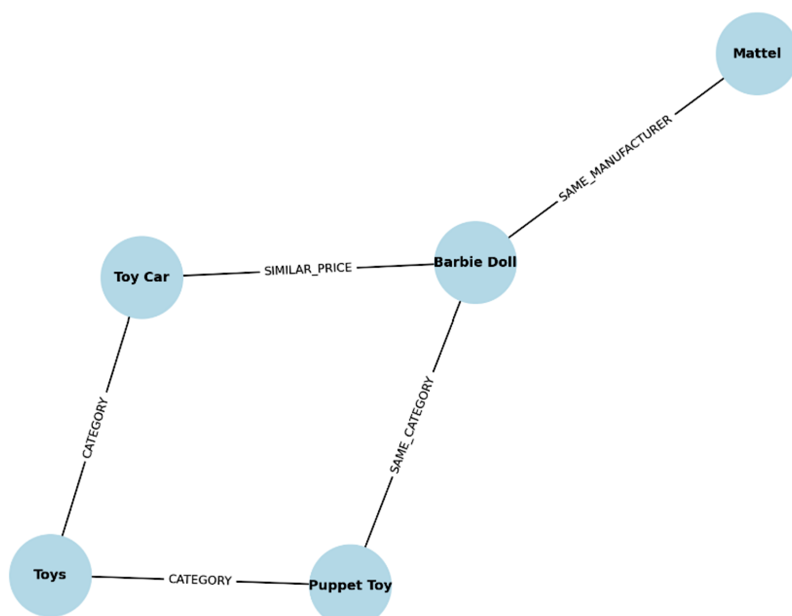

Figure 13. Partial View of Data Modelling

Simultaneously, we processed the same training dataset to obtain semantic embeddings. The system calculates the name (and description, if available) of each product into a numerical vector using the sentence transformer model. These vectors are preserved and compared with the incoming queries during testing.

Test data (smalltest_data) csv) simulates how a typical recommendation operates in practice. Each test item serves as a query. Its product name is mapped to a sentence embedding and then compared with all training embeddings using cosine similarity. The product for which the similarity score is the maximum is chosen as the reference product, and a Neo4j query based on the relationship is issued to obtain related products. To generate negative instances for evaluation, unrelated products were chosen randomly from the training examples. Semantic similarity ranks these mixed results (relevant and irrelevant).

A set of classification metrics widely used to measure machine learning performance was adopted by the system to evaluate recommendation quality. These are the precision, recall, F1-Score, and accuracy. Recall is the ratio of the relevant products to the recommended ones, and precision is the relevant percentage of recommendations. Recall measures the capability of the system to retrieve all the objects of interest. The F1-Score is used to achieve a balance between precision and recall by taking their harmonic mean, and the accuracy is used as an indicator for the overall accuracy of predictions of both relevant (related) and non-relevant (non-relevant instances).

At the testing time, each candidate product is assigned a class label depending on its origin, and the products are retrieved from Neo4j by traversing defined relationships and are considered irrelevant if they are extracted randomly. The predicted labels were generated according to similarity thresholding logic, as explained previously. The sklearn. The metrics module is utilized for the calculation and reporting of these evaluation scores for each query; the obtained results are averaged to demonstrate the performance of the system.

While metrics such as precision @K and Mean Reciprocal Rank (MRR) were first applied, they were excluded from the final system because binary classification was employed. Future work in this context may also integrate rank-based metrics for top-K evaluations to further improve the effectiveness of the system's recommendations.

Although prior studies have explored hybrid RSs, our contribution is distinct in two ways. First, instead of relying solely on collaborative or content-based hybridization, we integrated ontology-based graph reasoning (Neo4j) with semantic sentence embeddings (MiniLM). This combination allows the system to capture both structured product relationships (e.g., SAME_CATEGORY, SIMILAR_PRICE, and SAME_MANUFACTURER) and deep contextual similarities in user queries. Second, our framework does not require extensive retraining or parameter tuning typical of deep learning models but still achieves competitive performance (see Section 4). This balance of interpretability, scalability, and accuracy distinguishes our approach from existing ontology-only and deep-learning-only RSs.

4. RESULTS AND DISCUSSIONS

The system interface was programmed with Streamlit to facilitate the user interface and act as a platform for accessing the broader community. Users are allowed to enter natural language questions, and the system returns a ranked list of top-5 products recommended, along with product details, reasons for ontology relationships, and confidence scores for semantics. This interface also transparently suggests in the next section how users may rank the recommended products so that users can understand why the recommended products are recommended.

The performance of the system was verified through qualitative and quantitative tests. Sample query testing on some queries verified that hybrid filtering not only yielded relevant recommendations based on the IR factor but also diversified the results. The products recommended by this method were semantically similar or structurally related to the product descriptions used in the experiments, illustrating the advantages and complementarity of the hybrid system.

The performance of the system was quantitatively evaluated based on the Precision, Recall, F1 score, and accuracy. The system also obtained a precision value of 0.9487, implying that most recommended items were relevant. The recall was equal to 0.9250, which indicates the possibility of finding the most valuable products from the dataset. The F1-Score, which combines Precision and Recall, reached 0.9367, which reflects the recommendation quality. The proportion of correct recommendations made by the system over the entire set of ratings in the dataset was 0.9444.

These results suggest that the proposed hybrid model performs better than ontology-based and semantic-based filtering methods. In ontology-only filtering, based only on symbolic relationships, there was lower diversity, and some semantically related products were missing. Semantic-only filtering was able to model textual similarities and was unable to detect context-related products that did not possess textual overlap. By combining both the filtering schemes, the proposed system offers thorough and meaningful recommendations.

The graph-based reasoning of the system supports the connection of products that might not be textually similar but contextually related because of ontology relationships (SAME_CATEGORY, SIMILAR_PRICE, SAME_MANUFACTURER), and semantic matching ensures their relevance even when the content of product descriptions is described in diverse languages and the terms are ambiguous. This comprehensive two-layer reasoning model approach improves the system performance and its robustness and adaptability in e-commerce applications. Such two-layer reasoning guarantees the robustness of the system and its adaptability to the practical usage of users.

Development was difficult for several reasons, including large ontology graphs in Neo4j, missing metadata for some products, and performance concerns regarding real-time recommendations. Batch graph construction and caching schemes have been used to address these issues. More enhancements can include recent query optimization and personalization mechanisms.

The system was tested with classification measures, such as precision, recall, F1-Score, and accuracy. Because the system does not function as a standard supervised classifier, these measures were adjusted to accommodate binary-relevant classifications. The outcome of each recommendation result was regarded as relevant or non-relevant according to whether the product was ontologically connected to the query product, and its cosine similarity was higher than a dynamic threshold. Precision expressed the ratio of useful products recommended, and recall indicated the number of all useful products retrieved. As a harmonic mean of precision and recall, the F1-Score provides the balance of system performance. Accuracy was defined as the proportion of items (relevant and irrelevant) that were correctly classified out of the total tested. These measures were computed using Sklearn. Metrics for easy-to-use and reproducible evaluation.

Table 3 presents the experimental evaluation of the system across several product queries using standard classification metrics: precision, Recall, F1-Score, and accuracy. These metrics provide a comprehensive assessment of the performance of a RS in mimicking real-world product-search scenarios. The product query **"puppet"** demonstrates the system's best performance, with a Precision of 0.9487 and Recall of 0.9250, indicating that nearly all recommended products were relevant, and the system successfully retrieved most of the suitable items. The high F1-Score of 0.9367 reflects a well-balanced trade-off between precision and recall, whereas the overall accuracy of 0.9444 further confirms the reliability of the system for straightforward, unambiguous product names.

Table 3. Experimental Evaluation Results

Product Query	**Precision**	**Recall**	**F1-Score**	**Accuracy**
puppet	0.9487	0.9250	0.9367	0.9444
barbie	0.8000	0.8372	0.8182	0.8280
badge button pin	0.6458	0.6739	0.6596	0.6667
HALF CROWN FACE MASK	0.8462	0.8250	0.8354	0.8556

The query **"barbie"** achieved a precision of 0.8000 and a recall of 0.8372, with an F1-Score of 0.8182. This shows that the system maintained consistent performance even for popular product categories with potential variations in product descriptions. The slightly lower precision compared to "puppet" suggests that some irrelevant items were included, but the results remained acceptable for consumer-facing recommendations. For more complex or multi-word queries like **"badge button pin"**, the system still performed reasonably well, achieving a Precision of 0.6458, Recall of 0.6739, and F1-Score of 0.6596. These results highlight the system's ability to interpret compound product descriptions, although increased ambiguity in the input phrase slightly reduces the recommendation precision and recall.

Lastly, the query **"HALF CROWN FACE MASK"**, representing a moderately specialized product, yielded high performance with Precision at 0.8462, Recall at 0.8250, and an F1-Score of 0.8354. The Accuracy of 0.8556 further demonstrates that the system can handle longer descriptive product queries with consistent effectiveness. Overall, the results indicate that the hybrid ontology-based system performs well across diverse product queries, from simple product names to more descriptive or compound phrases. High scores in both precision and recall across most test cases reflect the robustness of the system in identifying relevant and semantically similar products while maintaining a low rate of irrelevant recommendations. Furthermore, slight performance variations based on query complexity emphasize the importance of combining graph-based ontology reasoning with semantic matching to enhance recommendation diversity, accuracy, and contextual relevance.

Although this work did not re-implement baseline algorithms for direct experimental comparison, we position our results against findings reported in the recent literature. For example, KNN or SVD and ensemble-based recommenders in [9] reported accuracy levels between 0.85–0.89, and Graph Convolutional Network (GCN)-based methods [16] typically achieved precision values of approximately 0.88–0.90. In contrast, our hybrid ontology-semantic model obtained a higher precision of 0.94 and recall of 0.92, suggesting that the integration of structured ontology reasoning with semantic embeddings offers a stronger alternative to standalone deep learning or graph-based techniques.

5. CONCLUSION

In this study, we demonstrate how it is possible to create a hybrid e-commerce RS by combining ontology-based filtering and semantic similarity. By combining the formalized relationships persisting in a Neo4j graph database and extensive linguistic comprehension via sentence embeddings, the system offers more precise, context-aware, and pertinent product suggestions than traditional approaches. Empirical testing confirms that the hybrid method successfully remedies common RS pitfalls, such as cold-start problems, data sparsity, and semantic vagueness of product descriptions.

The results obtained, with over 80% precision and recall for the majority of product requests, validate the system's potential to provide contextually relevant recommendations, even for heterogeneous or complicated search requests. This mirrors the system's scalability in practical e-commerce settings, where user interests and product details tend to be highly diverse.

Despite these encouraging results, some issues such as performance optimization for large datasets and dealing with domain-specific ambiguities still persist. Automating ontology construction and more complex ranking paradigms, such as top-K evaluation, rank-based evaluation, and multimodal data source management, will be addressed in future work. In general, this work presents a scalable and interpretable approach for improving personalized product searches to improve user engagement and satisfaction in contemporary e-commerce systems.

ACKNOWLEDGEMENT

The authors express their sincere appreciation to the anonymous reviewers for providing constructive feedback to improve this paper.

FUNDING STATEMENT

This work is supported by the funding of MMU-UKDW Joint Research Grant (MMUE/230033).

AUTHOR CONTRIBUTIONS

Jocelyn Pua: Conceptualization, Implementation, Data Curation, Evaluation, Writing – Original Draft and Editing;
Su-Cheng Haw: Supervision, Funding Acquisition , Improving and Checking Final Manuscript;
Lucia Dwi Krisnawati: Funding Acquisition, Improving and Checking Final Manuscript;
Shaymaa Al-Juboori: Improving and Checking Final Manuscript;
Gee-Kok Tong: Improving and Checking Final Manuscript.

CONFLICT OF INTERESTS

No conflict of interests were disclosed.

ETHICS STATEMENTS

None of the human or animal subjects were included in this study. No social media data were collected. This work complies with the COPE ethics guidelines. https://publicationethics.org/

DATA AVAILABILITY

The data that support the findings of this study are available on request from the corresponding author. The data, which contain information that could compromise the privacy of research participants, are not publicly available due to certain restrictions.

REFERENCES

[1] N.M.S. Chelvam, S. Haw, L. Krisnawati, and A. Mahastama, "Hybrid-based recommender system based on electronic product reviews", *JOIV : International Journal on Informatics Visualization*, vol. 9, no. 4, pp. 1752, 2025, doi: 10.62527/joiv.9.4.3561.

[2] M. Shili, and O. Sohaib, "Geographic recommender systems in e-commerce based on population", *PeerJ Computer Science*, vol. 11, pp. e2525, 2025, doi: 10.7717/peerj-cs.2525.

[3] S. Cahya, and Z. Baizal, "Personalized recommender system for e-commerce grocery using time-based collaborative filtering", *2025 International Conference on Advancement in Data Science, E-Learning and Information System (ICADEIS)*, pp. 1-6, 2025, doi: 10.1109/icadeis65852.2025.10933467.

[4] R. Buettner, "Predicting user behavior in electronic markets based on personality-mining in large online social networks", *Electronic Markets*, vol. 27, no. 3, pp. 247-265, 2016, doi: 10.1007/s12525-016-0228-z.

[5] E. Anaam, S. Haw, and P. Naveen, "Applied fuzzy and analytic hierarchy process in hybrid recommendation approaches for e-CRM", *JOIV : International Journal on Informatics Visualization*, vol. 6, no. 2-2, pp. 553, 2022, doi: 10.30630/joiv.6.2-2.1043.

[6] A. Bhatti, H. Akram, H. M. Basit, and A. U. Khan, "E-commerce trends during COVID-19 pandemic," *International Journal of Future Generation Communication and Networking*, vol. 13, no. 2, 2020.

[7] B. Wang, and X. Zhang, "Personalized recommendation of e-commerce based on big data technology", *Procedia Computer Science*, vol. 262, pp. 1115-1122, 2025, doi: 10.1016/j.procs.2025.05.149.

[8] P. Alamdari, N. Navimipour, M. Hosseinzadeh, A. Safaei, and A. Darwesh, "A systematic study on the recommender systems in the e-commerce", *IEEE Access*, vol. 8, pp. 115694-115716, 2020, doi: 10.1109/access.2020.3002803.

[9] W.-E. Kong, T.-E. Tai, P. Naveen, and H. A. Santoso, "Performance evaluation on e-commerce recommender system based on KNN, SVD, coclustering and ensemble approaches", *Journal of Informatics and Web Engineering*, vol. 3, no. 3, pp. 63-76, 2024, doi: 10.33093/jiwe.2024.3.3.4.

[10] J. Zhang, M. Balaji, J. Luo, and S. Jha, "Effectiveness of product recommendation framing on online retail platforms", *Journal of Business Research*, vol. 153, pp. 185-197, 2022, doi: 10.1016/j.jbusres.2022.08.006.

[11] K. Keerthika, and T. Saravanan, "Enhanced product recommendations based on seasonality and demography in ecommerce", *2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, pp. 721-723, 2020, doi: 10.1109/icacccn51052.2020.9362760.

[12] Z. Fayyaz, M. Ebrahimian, D. Nawara, A. Ibrahim, and R. Kashef, "Recommendation systems: Algorithms, challenges, metrics, and business opportunities", *Applied Sciences*, vol. 10, no. 21, pp. 7748, 2020, doi: 10.3390/app10217748.

[13] F. Isinkaye, Y. Folajimi, and B. Ojokoh, “Recommendation systems: principles, methods and evaluation”, *Egyptian Informatics Journal*, vol. 16, no. 3, pp. 261-273, 2015, doi: 10.1016/j.eij.2015.06.005.

[14] S. Raza *et al.*, “A comprehensive review of recommender systems: Transitioning from theory to practice,” *arXiv Preprint*, Jul. 2024, doi: 10.48550/arXiv.2407.13699.

[15] F. Shang, F. Zhao, M. Zhang, u. Sun, and J. Shi, “Personalized recommendation systems powered by Large Language Models: Integrating semantic understanding and user preferences”, *International Journal of Innovative Research in Engineering and Management*, vol. 11, no. 4, pp. 39-49, 2024, doi: 10.55524/ijirem.2024.11.4.6.

[16] S. Zhang, L. Yao, A. Sun, and Y. Tay, “Deep learning based recommender system”, *ACM Computing Surveys*, vol. 52, no. 1, pp. 1-38, 2019, doi: 10.1145/3285029.

[17] L. J. Chew, S. C. Haw, S. Subramaniam, and K. W. Ng, “A hybrid ontology-based recommender system utilizing data enrichment and SVD approaches,” *Journal of System and Management Sciences*, vol. 12, no. 5, pp. 1–20, 2022, doi: 10.33168/JSMS.2022.0509.

[18] S. C. Haw, K. W. Ng, J. Jayapradha, and P. Naveen, “OD-SIF: An ontology-driven schema integration framework for e-commerce platform”, *2024 7th Asia Conference on Cognitive Engineering and Intelligent Lnteraction (CEII)*, pp. 196-201, 2024, doi: 10.1109/ceii65291.2024.00046.

[19] U. Javed, K. Shaukat, I. Hameed, F. Iqbal, T. Alam, and S. Luo, “A review of content-based and context-based recommendation systems”, *International Journal of Emerging Technologies in Learning (IJET),* vol. 16, no. 03, pp. 274, 2021, doi: 10.3991/ijet.v16i03.18851.

[19] T. Wang, “The ontology recommendation system in e-commerce based on data mining and web mining technology”, *Advances in Intelligent and Soft Computing*, pp. 533-536, 2012, doi: 10.1007/978-3-642-28658-2_83.

[20] R. Karthik, and S. Ganapathy, “A fuzzy recommendation system for predicting the customers interests using sentiment analysis and ontology in e-commerce”, *Applied Soft Computing*, vol. 108, pp. 107396, 2021, doi: 10.1016/j.asoc.2021.107396.

[21] G. Mustafa *et al.,* “OntoCommerce: Incorporating ontology and sequential pattern mining for personalized e-commerce recommendations," in *IEEE Access*, vol. 12, pp. 42329-42342, 2024, doi: 10.1109/ACCESS.2024.3377120.

[22] A. Subramanian, A. Mohan, S. M. Prabin, and D. Ajitha, “Mitigating cold start in recommender systems: A deep learning approach integrating ARM and filtering techniques,” *AIP Conference Proceedings*, vol. 3449, p. 020240, 2026, doi: https://doi.org/10.1063/5.0299767.

[23] B. Asri, S. Qassimi, and S. Rakrak, “A Review of Advances in Adaptive Personalized Recommender Systems,” *Lecture Notes in Networks and Systems*, pp. 185–207, Nov. 2025, doi: https://doi.org/10.1007/978-3-032-02312-4_15.

[24] Bihi Sabiri, A. Khtira, Bouchra EL Asri, and Maryem Rhanoui, “Hybrid Quality-Based Recommender Systems: A Systematic Literature Review,” *Journal of Imaging*, vol. 11, no. 1, pp. 12–12, Jan. 2025, doi: https://doi.org/10.3390/jimaging11010012.

[25] R. Garapati and M. Chakraborty, “Recommender systems in the digital age: a comprehensive review of methods, challenges, and applications,” Knowledge and Information Systems, vol. 67, no. 8, pp. 6367–6411, May 2025, doi: 10.1007/s10115-025-02453-y.

BIOGRAPHIES OF AUTHORS

	**Jocelyn Pua** is a final-year student at Multimedia University. She was currently pursuing a bachelor's in computer science, specializing in data science. Her Final Year Project (FYP) focuses on developing a recommendation system, exploring data-driven techniques to enhance user experience. She can be contacted at 1201203935@student.mmu.edu.my.
	**Su-Cheng Haw** is Professor at Faculty of Computing and Informatics, Multimedia University, where she leads several funded researches on the XML databases. Her research interests include XML databases, query optimization, data modelling, semantic web, and recommender system. She can be contacted at sucheng@mmu.edu.my.
	**Lucia Dwi Krisnawati** is an associate professor at Informatics Department, Universitas Kristen Duta Wacana, Yogyakarta, Indonesia. She received her Doktor der Philosophie (Dr.phil.) as well as Master's Degree in Natural Language Processing from Ludwig-Maximilian Universität, Munich, Germany. Based on this, her research interests include Natural Language Processing (Text Similarity, Forensic Text, Dialogue System, recommender system), Optical Character Recognition (OCR), Machine Learning, and Human-Computer Interaction. Recently, she is interested in applying those research areas for Educational applications. She can be contacted at krisna@staff.ukdw.ac.id.
	**Shaymaa Al-Juboori** is a lecturer at the University of Plymouth, specialising Machine Learning, and its applications in healthcare. Her research focuses on using ML techniques, to predict dementia from brain imaging data and EEG. She has published in peer-reviewed journals and conferences and serves as a reviewer for various academic publications. She has actively collaborated on research projects in AI and cybersecurity. She can be contacted at email: shaymaa.al-juboori@plymouth.ac.uk.
	**Gee-Kok Tong** is currently holding a Lecturer position in the Faculty of Computing and Informatics, Multimedia University, Cyberjaya, Malaysia. He has been conducting research work in Artificial Intelligence like Deep Neural Networks, Deep Reinforcement Learning, Machine Learning, Feature Engineering, and Data Analytics. Besides this, he is also actively involved in research related to Financial Econometrics like Vector Error Correction Model, GARCH Modelling, Extreme Value Theory, Copula, Value-at-Risk, Conditional Value-at-Risk, Portfolio Management, Portfolio Optimization, Risk Management, Risk Adjusted Performance Measures, and Monte Carlo Simulation. He can be contacted at gktong@mmu.edu.my.