Journal of Information System Research (JOSH)
Volume 5, No. 2, Januari 2024, pp 631−638
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v5i2.4784

Ontology-based Conversational Recommender System for Smartwatches

Muh Thoriq Akhdan, Z K A Baizal*

School of Computing, Informatics, Telkom University, Bandung
Jl. Telekomunikasi. 1, Terusan Buahbatu - Bojongsoang, Telkom University, Sukapura, Kec. Dayeuhkolot, Kabupaten Bandung, Jawa Barat, Indonesia
Email: [1]akshazero@student.telkomuniversity.ac.id, [2,*]baizal@telkomuniversity.ac.id
Correspondence Author Email: baizal@telkomuniversity.ac.id
Submitted: **09/01/2024;** Accepted: **25/01/2024;** Published: **27/01/2024**

Abstract−In recent years, smartwatches have become popular in the mobile technology market. However, with various smartwatch models and brands available, prospective buyers often need help choosing the right product due to specifications that require technical understanding and expert opinions. Therefore, a recommender system is needed to assist prospective buyers in choosing the appropriate product. Several studies have been conducted on conversational recommender systems. However, the recommender systems used only provide recommendations based on technical specifications alone, so the recommendations given are less personalized. Therefore, we develop a conversational recommender system for smartwatches using ontology that considers the functional needs of users to produce customized recommendations. In this study, we have successfully built and evaluated this system using recommendation accuracy metrics and user satisfaction. The evaluation results show an accuracy of 86.67% and positive user feedback. This indicates that our system is accurate, easy to use, and well-accepted.

Keywords: Conversational Recommender System; Personalized Recommendation; Ontology; Knowledge-Based Recommender; Recommender System

1. INTRODUCTION

The popularity of smartwatches has significantly increased in recent years, mainly due to their advanced features that can be accessed directly from the wrist. However, with various smartwatch models and brands available, choosing the right product can take time and effort for prospective buyers. This is due to the specifications of smartwatches that require technical understanding and expert opinions to make the right choice. Hence, a recommender system is essential to help potential buyers in choosing the product they intend to purchase. A conversational recommender system can be a solution. The system will generate more personalized recommendations by providing users with several interactions related to the functionalities needed.

A conversational recommender system can ask questions about user preferences on different attributes, thus producing better recommendations [1]. In the recommendation process, ontology will depict the knowledge concerning items and users [2] to provide recommendation results that are more tailored to user preferences [3]. An Explanation Facility is also added to explain why the product is recommended to the user [4]. The system that has been created can help users make the right decision when choosing a smartwatch so that purchasing the recommended product becomes an indicator that the system has successfully provided the suitable recommendation [5].

Several studies have been conducted and developed on ontology-based conversational recommender systems. One of them is by Baizal et al. [6], built a conversational recommender system that provides recommendations by asking about the needs of the product (Functional Requirements) sought by users. From the conversational interaction conducted with users, a user profile will be generated that is used to get recommendations that meet the ontology that researchers define as a tuple in the user profile model. User studies show that systems with functional requirement-based interactions are more favored compared to interactions based on technical features. However, the ontology used in the system must cover all functional requirements that users may have. In another study conducted by Baizal et al. [7] that builts a system that provides tourist destination recommendations to users, it was found that the interaction of the CRS model can increase trust, perceived usefulness, and perceived enjoyment compared to the general model. The system navigates by asking questions, where users are given a question-and-answer interaction to get the needs of the destination desired by the user. Following numerous interactions, the system offers recommendations and describes its specifications. Subsequently, the user is requested to give feedback to enhance the efficiency of the interaction process. Overall, the system has provided more useful recommendations and used questions about the user's functional needs. However, with the many repeated interactions conducted with the user before providing recommendations, it takes quite a long time.

Christakopoulou et al. [8] developed a conversational recommender system using Bandit Strategies and Probabilistic Matrix Factorization (PMF) as latent factor learners and utilized them for interactive preference elicitation. By using PMF, the system can identify what kind of questions to ask new users to quickly understand their preferences. However, PMF requires a substantial amount of data to work effectively, leading to a cold-start problem. To address this issue, the authors [8] used online learning for all user and item parameters.

Obeid et al. [9] created a recommender system based on ontology to assist high school students in selecting their majors. They employed machine learning techniques, including the k-means algorithm for creating student profiles. The findings indicated that the user-based approach was superior to the item-based approach. Their developed system yielded satisfactory results and validated the generated rules with a minor prediction error deviation. However, the system was less effective when offline due to online recommendation learning.

Journal of Information System Research (JOSH)
Volume 5, No. 2, Januari 2024, pp 631−638
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v5i2.4784

In the hybrid recommender system created by Nilashi et al. [10] , the team employed a technique for reducing dimensionality, specifically Singular Value Decomposition (SVD), to tackle the problems of sparsity and scalability. This was substantiated by the evaluation outcomes derived from the Yahoo! Webscope R4 dataset and MovieLens. The results demonstrated that the use of ontology, enhanced by clustering and dimensionality reduction techniques, effectively boosts the performance of Collaborative Filtering (CF) recommender systems.

In this study, we propose an ontology-based conversational recommender system for smartwatches that can consider user preferences and other relevant information to produce recommendations tailored to the user's functional needs.

2. RESEARCH METHODOLOGY

2.1 Research Stages

In this study, we develop a conversational recommender system for smartwatches, utilizing a conversational recommender system framework that was previously established [6], [7]. A Conversational Recommender System (CRS) is a type of recommender system that leverages user interactions in conversational form to produce more tailored recommendations [11], utilizing Navigation by Asking (NBA) and Navigation by Proposing (NBP) strategies It operates on the notion of conversation, which are referred to as actions [12]. Unlike traditional recommender systems, Conversational Recommender Systems (CRS) emphasize interactive clarification and explicit feedback in natural language [13]. A conversational recommender system necessitates a sturdy database and efficient techniques for processing and constructing user and item models. Ontology can serve as a database for creating user and item models in a conversational recommender system [2]. The item in the database is popular because it is likely to be chosen by a new user [14] but the diversity of product also important [15].

The system starts by providing several interactions in the form of questions that will be filled out by users so that the system gets preferences that will be used to provide more relevant recommendations to users. The system asks questions related to the functional needs required by users. Whether for sports activities, all-day activities, entertainment, or other functional needs. Based on the information obtained from the interactions carried out, a user profile model is created that stores user preferences and will look for a class in the ontology that is relevant and suitable, which will then be given to the user. This system itself has three modules that can describe the entire designed system. These three modules can be seen in the following image.

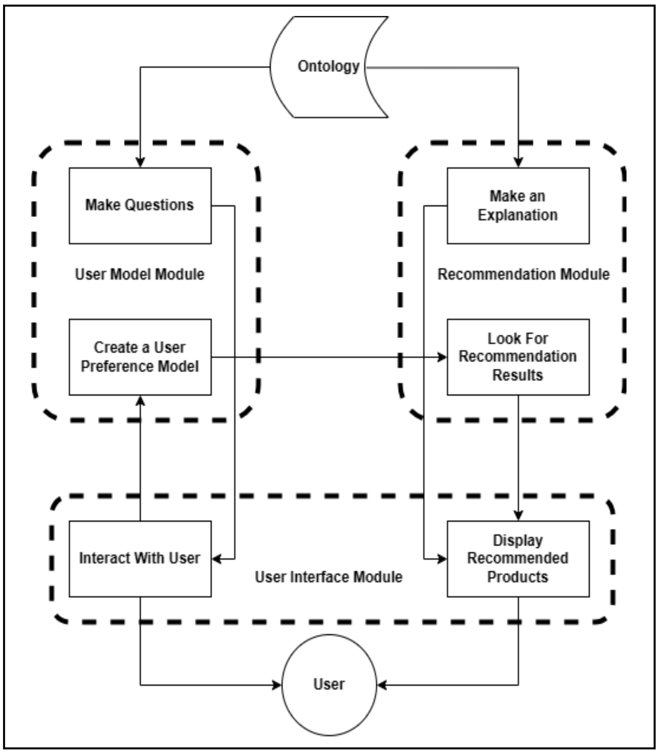

Figure 1. System Overview

Figure 1 shows a user model module that generates questions for users to build a user preference model. There is also a recommendation module that searches for products that suit the user’s profile and gives details about the product to make the features, benefits, and drawbacks of the recommended product clearer. Lastly, there is a user interface module that shows the questions from the user model module and the recommendation outcomes from the recommendation module.

Journal of Information System Research (JOSH)
Volume 5, No. 2, Januari 2024, pp 631−638
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v5i2.4784

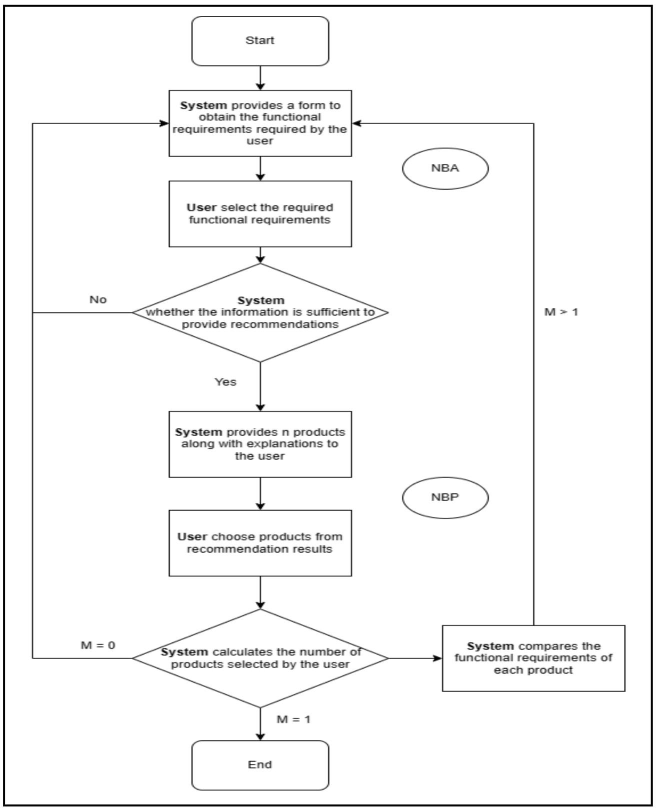

Figure 2. Flowchart of System and User Interactions Mechanism

Figure 2 illustrates the user and system interaction process. When the user first engages with the system, they will be prompted to fill out a form with questions about their functional needs and preferences. The system will ask for more details if the information provided is insufficient. Once the system has enough information, it will suggest several products that meet the user's needs and explain each suggestion. The user can then select the most suitable product from the options given. If the user fails to choose any product, they will be returned to the beginning of the interaction process. If the user selects more than one product, the system will compare the functional needs of each product and prompt the user to choose a single product. The interaction will end when the user selects a single product.

2.2 User Preference Model

The User Preference Model contains user preferences collected through NBA interactions provided by the system. This model is then used to predict the similarity of each product for the user. The categorization in the user preference model will follow the categories used in the system created by Baizal et al. [7] There are 5 cases as follows:

a. Empty user profile
   The situation described in this case is when a user uses the recommender system for the first time. Initially, the user's profile is not formed, and no information about the user's preferences and needs is available. As a solution, the strategy applied is to ask initial questions as a first step to interact with the user and gradually build the user's profile.
b. Multiple product selection
   When interacting with the system at the initial stage, users often need help selecting a product, so they consider many recommendation options. So, to help users narrow down product choices, the system can use a strategy by asking questions that consider differences between product groups.
c. No product selected
   When users have difficulty choosing a product that suits their functional needs, so they do not choose one of the recommended products, the strategy that can be done is to propose alternative functional needs at a certain level and return to the previous level that has not been asked. This way, the system can provide more specific choices and help users better understand their functional needs.
d. Inadequate specification of requirements to generate a recommendation
   This case shows that the initial requirement definition is insufficient to produce accurate recommendations. The requirements are still too general, making it difficult to identify products that match the user's needs. Hence, the approach is to request more detailed functional requirements at the subsequent level. In the next interaction, the system will ask questions related to the user's previous answers to clarify the functional requirements needed. This way, the system can produce more accurate recommendations matching the user's functional needs.
e. There is no product that aligns with the existing user profile model

Journal of Information System Research (JOSH)
Volume 5, No. 2, Januari 2024, pp 631−638
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v5i2.4784

This case shows that no products match the existing user profile model. The strategy that can be used is to mark irrelevant user profiles and ask for other functional needs at a higher hierarchy level. By marking irrelevant user profiles, the system can filter out products that do not match the user's needs.

2.3 Smartwatch Ontology

Ontology can be described as the characteristics of concepts, independent assertions, value limitations, and the definitions of logical relationships among entities. It has evolved into a tool for formally modeling the structure of systems based on relations that arise from observation [10] or to make conceptual modeling for a particular domain [16]. Ontology is used as a knowledge base to facilitate parsing, reasoning, sharing, and reusing knowledge to improve the quality and results of personalization [9]. The domain ontology manages domain knowledge that represents different levels of relationships among concepts, extracts conceptual knowledge from a specific domain from various sources, and provides structured storage facilities. It improves accuracy by refining the context, thereby enhancing the quality of recommendations [17]. The ontology structure has two types of links. One type is the link that belongs to a hierarchy. The other type is the link that joins individuals from different hierarchies [6].

The hierarchy of functional needs classes is used to map the functional needs of a smartwatch into a class. An example of a hierarchy for functional needs classes can be seen in Figure 3.

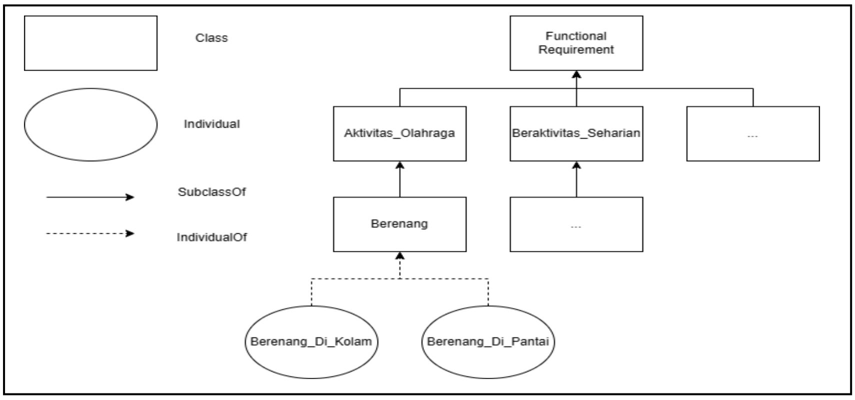

Figure 3. Hierarchy of Functional Requirements Classes for Smartwatch

The hierarchy of product classes is used to map smartwatch products into a class and subclass. An example of a hierarchy for product classes and subclasses can be seen in Figure 4.

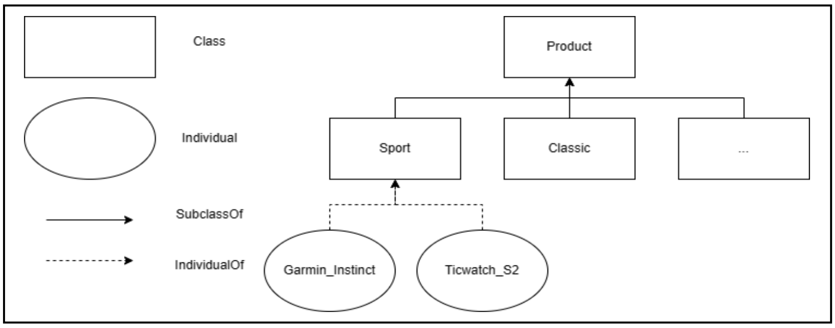

Figure 4. Hierarchy of Product Classes for Smartwatch

The hierarchy of functional needs classes is used to map the functional needs of a smartwatch into a class. An example of a hierarchy for functional needs classes can be seen in Figure 5.

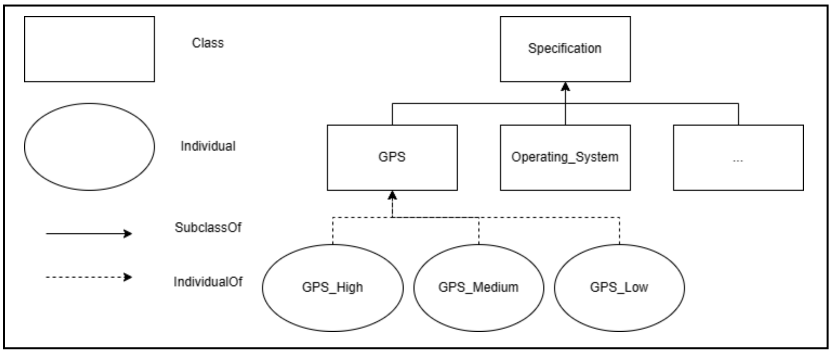

Figure 5. Hierarchy of Specification Classes for Smartwatch

Journal of Information System Research (JOSH)
Volume 5, No. 2, Januari 2024, pp 631−638
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v5i2.4784

3. RESULT AND DISCUSSION

3.1 Query Refinement

The query refining process is a critical first step in the search for the perfect product. This procedure, which is essential to the user-system interaction, is meant to reduce the wide range of items that are out there to a manageable selection that corresponds with the individual needs of the user. It is evidence of the system's intelligence and capacity to comprehend user preferences and make necessary adjustments.

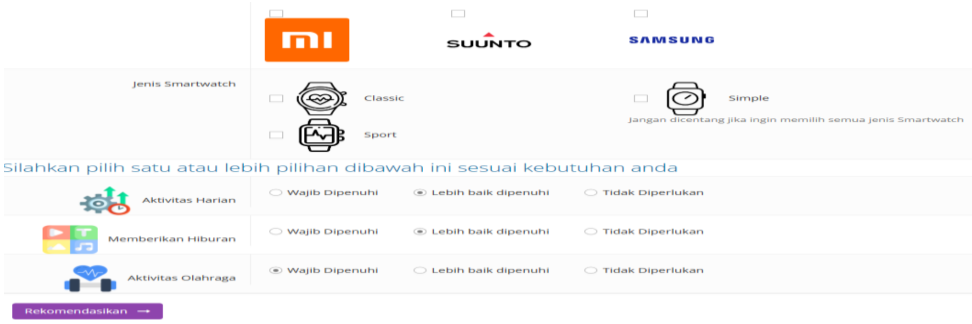

Figure 6. Initial Interaction

Figure 6 illustrates the first interaction between the system and the user, in which the user is presented with various functional requirements to choose from, including mandatory functions. Additionally, optional functions such as the type of smartwatch needed and the preferred brand are available for selection. The user selects the functional requirements that match their preferences, and the system narrows down the list of products that meet those requirements. If the list is still too long, the system uses the query refinement algorithm to ask more specific questions about the functional requirements. These questions are derived from the sub-nodes of the previously selected functional requirements in the ontology structure. By asking these questions, the system aims to increase the accuracy and relevance of the recommendations [6]. The query refinement algorithm is as shown below:

Table 1. Algorithm

Algorithm 1: Query Refinement	
	Input: userModel, maxFunct
	Output: nextFunct
	Start
1.	d ← userModel depth
2.	Prio ← nodes [mandFunc ∪ optiFunc] at level d in userModel
3.	subPrio ← children of Prio
4.	**If** subPrio ≠ null **then**
5.	candFunct ← subPrio
6.	**Else**
7.	candFunct ← Unvisited node that have potential preference from the user
8.	nextFunct ← select maxFunct from candFunct
	End

Algorithm 1 begins with finding the depth of the userModel, which is the level of detail that the user has specified so far. Then it finds the nodes at that level of the userModel that are either mandatory or optional for the user, and stores them in Prio. After that, it finds the children of those nodes, which are the sub-nodes that are more specific and detailed, and stores them in subPrio. The algorithm then selects the next set of functional requirements to ask the user from subPrio or from the unvisited nodes that have a potential preference from the user and display them to the user as shown in Figure 7 below.

Figure 7. Query refinement result

Journal of Information System Research (JOSH)
Volume 5, No. 2, Januari 2024, pp 631−638
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v5i2.4784

Figure 8 displays the product results that match all the user preferences that have been specified in the previous interactions. The system provides a list of products that are suitable for the user's needs and preferences, along with an explanation of why each product is recommended. The explanation includes the functionality requirements that are satisfied by the product, as well as the ratings and reviews from other users. The system also highlights the features and benefits of each product, and compares them with other products in the list. By providing this information, the system aims to help the user make an informed and confident decision.

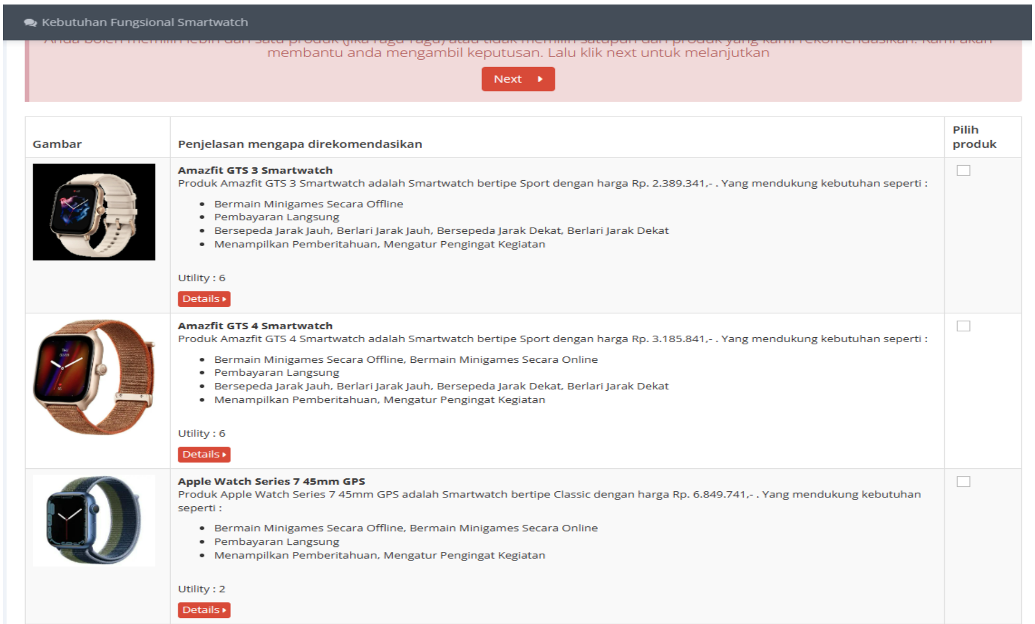

Figure 8. Product recommended by the system

3.2 System Performance

We want to evaluate how well our system performs in providing recommendations to the users. To do this, we need to measure how accurate the recommendations are using one of the three commonly used measures for conversational system that is recommendation accuracy [18]. We ask the users to rate the recommendations that they get from the system on a scale of 1 to 5, where 1 means very dissatisfied and 5 means very satisfied. Based on these ratings, we categorize the recommendations into two types: successful and unsuccessful. A successful recommendation is one that has a rating of 4 or 5, which shows that the user is happy with the recommendation and finds it helpful. An unsuccessful recommendation is one that has a rating of 1, 2, or 3, which shows that the user is unhappy with the recommendation and finds it unhelpful. After we have sorted the recommendations, we calculate the success rate of the recommendation by using the following formula [19]:

$$Recommendation\ Accuracy = \frac{Successful\ Recommendations}{Number\ of\ Recommendations} \quad (1)$$

This formula gives us the percentage of recommendations that the system provides that are successful and satisfactory for the users.

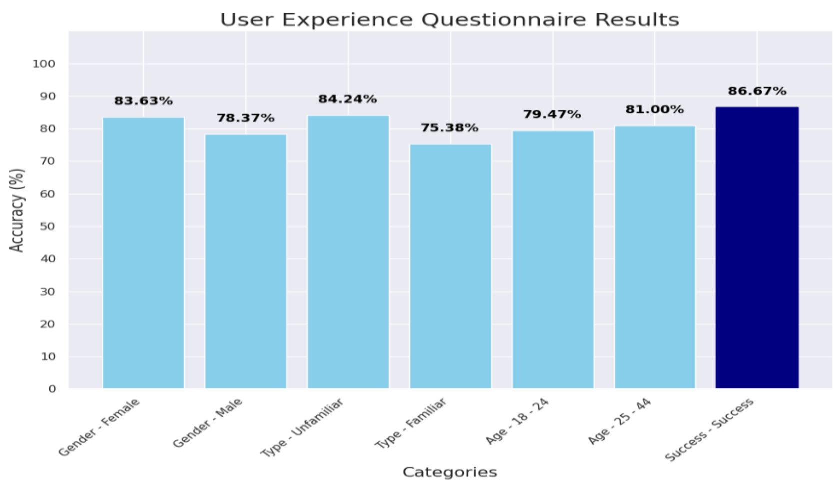

Figure 9. Accuracy results of the recommendations provided

Journal of Information System Research (JOSH)
Volume 5, No. 2, Januari 2024, pp 631−638
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v5i2.4784

Figure 9 shows the evaluation results of the system that has been developed. The system achieved an overall accuracy of 86.67%, meaning that it provided accurate recommendations for most of the users. The accuracy varied slightly depending on the user's gender, familiarity, and age group. The system performed better for female users (83.63%) than for male users 78.37%), and for unfamiliar users (84.24%) than for familiar users (75.38%). The system also had a higher accuracy for users in the 25-44 age group (81.00%) than for users in the 18-24 age group (79.47%). These results indicate that the system can adapt to different user characteristics and preferences.

3.3 User Satisfaction

User satisfaction evaluation will be conducted by providing interaction in the form of questions [19] to find out user satisfaction related to 6 main factors, i.e., User-Friendliness (ETU), Quality of Information (INF), Assumed Efficiency (PE), Simplicity of Understanding (EOU), Assumed Quality of Recommendation (PRQ), and Reliability (TR). These factors will help us determine if the product is compatible with the user, which is essential for people's views of usefulness and ease of use, as well as their attitude, which ultimately influences their adoption of a smartwatch [20]. The details of the questions given can be seen in Table 1.

Table 2. Statement of Questionnaire

ID	Factor	Information
SQ1	INF	I have no difficulty in searching for information about smartwatches.
SQ2	PE	I have no trouble in locating the product I desire.
SQ3	TR	I am determined to buy the product that I get from the system
SQ4	TR	Should I choose to buy a smartwatch in the future, I would utilize this system.
SQ5	ETU	I had trouble searching for the products I needed.
SQ6	ETU	I had a smooth and effortless experience with this system.
SQ7	EOU	The choices given are simple to understand.
SQ8	EOU	All the questions and choices I got are clear to me.
SQ9	PRQ	This system's interaction does not meet my expectations.

The user satisfaction evaluation results are shown in Figure 10. From the results, the majority of users gave positive feedback. Only on SQ5 and SQ9 did users give negative feedback. This shows that users have well received the system we develop.

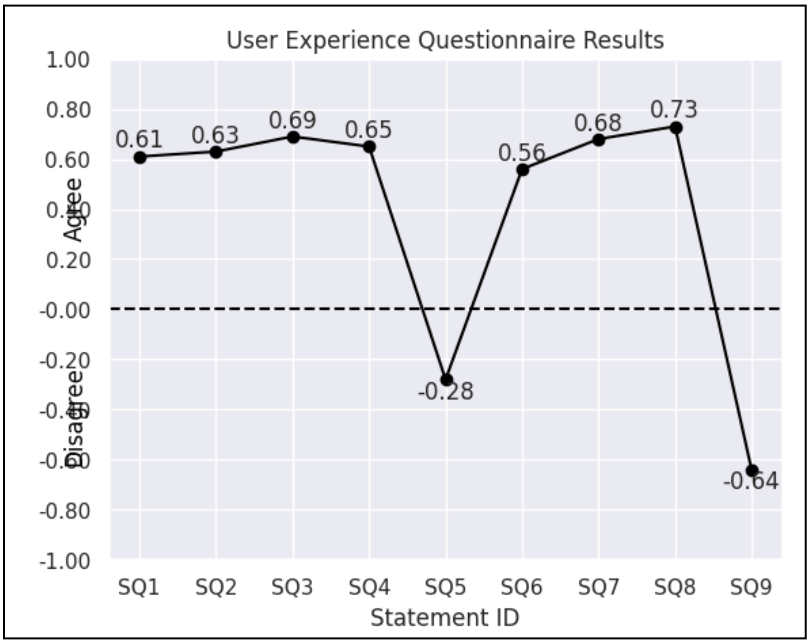

Figure 10. User Satisfaction Evaluation Results

4. CONCLUSION

We have developed an ontology-based conversational recommender system for smartwatches that can provide personalized recommendations based on user profiles and preferences. Our experimental results show that our system has a high accuracy of 86.67%, meaning that it can deliver relevant and suitable recommendations for most of the users. Moreover, our user satisfaction evaluation reveals that our system has positive ratings in various aspects, such as User-Friendliness (ETU), Quality of Information (INF), Assumed Efficiency (PE), Simplicity of Understanding

Journal of Information System Research (JOSH)
Volume 5, No. 2, Januari 2024, pp 631−638
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v5i2.4784

(EOU), Assumed Quality of Recommendation (PRQ), and Reliability (TR). These results demonstrate that our system is not only effective but also user-friendly and reliable. For future work, we plan to extend our system by incorporating more features and data sources to enhance the user experience and the recommendation performance. We also aim to conduct a more comprehensive evaluation with a larger and more diverse user group to validate the generalizability and scalability of our system. Furthermore, we intend to explore the use of natural language processing and dialogue management techniques to improve the conversational interaction and the recommendation explanation.

REFERENCES

[1] W. Lei, X. He, M. De Rijke, and T. S. Chua, “Conversational Recommendation: Formulation, Methods, and Evaluation,” SIGIR 2020 - Proc. 43rd Int. ACM SIGIR Conf. Res. Dev. Inf. Retr., pp. 2425–2428, 2020.
[2] J. K. Tarus, Z. Niu, and A. Yousif, “A hybrid knowledge-based recommender system for e-learning based on ontology and sequential pattern mining,” Futur. Gener. Comput. Syst., vol. 72, pp. 37–48, 2017.
[3] J. K. Tarus, Z. Niu, and G. Mustafa, “Knowledge-based recommendation: a review of ontology-based recommender systems for e-learning,” Artif. Intell. Rev., vol. 50, no. 1, pp. 21–48, 2018.
[4] A. Felfernig, E. Teppan, and B. Gula, “Knowledge-Based Recommender Technologies for Marketing and Sales,” Int. J. Pattern Recognit. Artif. Intell., vol. 21, no. 2, pp. 333–354, 2007.
[5] S. Milano, M. Taddeo, and L. Floridi, “Recommender systems and their ethical challenges,” AI Soc., vol. 35, no. 4, pp. 957–967, 2020.
[6] Z. K. A. Baizal, D. H. Widyantoro, and N. U. Maulidevi, “Computational model for generating interactions in conversational recommender system based on product functional requirements,” Data Knowl. Eng., vol. 128, no. October 2018, p. 101813, 2020.
[7] Z. K. A. Baizal, D. Tarwidi, Adiwijaya, and B. Wijaya, “Tourism Destination Recommendation Using Ontology-based Conversational Recommender System,” Int. J. Comput. Digit. Syst., vol. 10, no. 1, pp. 829–838, 2021.
[8] K. Christakopoulou, F. Radlinski, and K. Hofmann, “Towards conversational recommender systems,” Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., vol. 13-17-Augu, no. 3, pp. 815–824, 2016.
[9] C. Obeid, I. Lahoud, H. El Khoury, and P. A. Champin, “Ontology-based Recommender System in Higher Education,” Web Conf. 2018 - Companion World Wide Web Conf. WWW 2018, vol. 2, pp. 1031–1034, 2018.
[10] M. Nilashi, O. Ibrahim, and K. Bagherifard, “A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques,” Expert Syst. Appl., vol. 92, pp. 507–520, 2018.
[11] F. Narducci, P. Basile, M. De Gemmis, P. Lops, and G. Semeraro, An investigation on the user interaction modes of conversational recommender systems for the music domain, vol. 30, no. 2. Springer Netherlands, 2020.
[12] S. Zhang and K. Balog, “Evaluating Conversational Recommender Systems via User Simulation,” Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., pp. 1512–1520, 2020.
[13] K. Zhou, Y. Zhou, W. X. Zhao, X. Wang, and J. R. Wen, “Towards Topic-Guided Conversational Recommender System,” COLING 2020 - 28th Int. Conf. Comput. Linguist. Proc. Conf., pp. 4128–4139, 2020.
[14] K. Zhou, W. X. Zhao, S. Bian, Y. Zhou, J. R. Wen, and J. Yu, “Improving Conversational Recommender Systems via Knowledge Graph based Semantic Fusion,” Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., no. 2007, pp. 1006–1014, 2020.
[15] M. Chen and P. Liu, “Performance Evaluation of Recommender Systems,” vol. 13, no. 8, pp. 1246–1256, 2017.
[16] K. B. Fard, M. Rahmani, M. Nilashi, and V. Rafe, “Performance Improvement for Recommender Systems Using Ontology,” Telemat. Informatics, 2017.
[17] A. Razia Sulthana and S. Ramasamy, “Ontology and context based recommendation system using Neuro-Fuzzy Classification,” Comput. Electr. Eng., vol. 74, pp. 498–510, 2019.
[18] Y. Sun and Y. Zhang, “Conversational Recommender System,” pp. 235–244, 2018.
[19] Z. K. Abdurahman Baizal, Y. R. Murti, and Adiwijaya, “Evaluating functional requirements-based compound critiquing on conversational recommender system,” 2017 5th Int. Conf. Inf. Commun. Technol. ICoIC7 2017, vol. 0, no. c, 2017.
[20] J. Choi and S. Kim, “Computers in Human Behavior Is the smartwatch an IT product or a fashion product ? A study on factors affecting the intention to use smartwatches,” Comput. Human Behav., vol. 63, pp. 777–786, 2016.