Conversational Recommender Systems Based on Mobile Chatbot for Culinary

Ghazi Ahmad Fadhlullah, Z K Abdurrahman Baizal*, Nurul Ikhsan

School of Computing, Informatics Study Program, Telkom University, Bandung, Indonesia
Email: 1‘gojigoj@student.telkomuniversity.ac.id, 2*baizal@telkomuniversity.ac.id, likhsan@telkomuniversity.ac.id
Correspondence Author Email: baizal@telkomuniversity.ac.id

Abstract—Culinary places are one of the tourists' attractions in a place that makes many new culinary places appear. Various types of new foods and drinks are present along with the addition of culinary places. However, this can be a problem when tourists visit a new destination and look for a culinary place that suits their tastes. In the previous research on the recommendation system for culinary places, users only gave their preferences at the beginning of the recommendation process and ignored the operating hours of the recommended culinary places. Therefore, we developed a recommendation system for culinary places by utilizing the context of time from users. We use the Conversational Recommender System on the chatbot platform with the Personalized PageRank algorithm to generate recommendations. In addition, we also use the explanation facility to get an explanation of the recommended items. We use questionnaires and the accuracy of recommendation results to measure user satisfaction and system performance. The evaluation results with a questionnaire involving 81 respondents concluded that users are pretty satisfied with the system built. However, testing with accuracy yields a value of 40%, proving that the system performance is low.

Keywords: Culinary; Conversational Recommender Systems; Chatbot; PageRank Personalized; Explanation

1. INTRODUCTION

The human need for food continues to increase based on culinary places that always increase every year [1]. The increasing number of culinary places is also related to attracting tourists who visit a tourist destination because culinary places are now part of the local tourism industry [2]. Bandung is one of the choices of tourist destinations in Indonesia that is in demand by tourists. In addition, Bandung has its charm as a tourist place because of the increase in tourist visits every year. Therefore, it impacts the emergence of new culinary places to meet the needs of tourists for culinary tourism. The existence of new culinary places makes the growth of food and beverage variety increase, which can be a problem when tourists visit new destinations and choose a place to eat. Although there is currently a lot of information and media used to find a culinary place, this amount of information makes tourists confused in choosing a culinary place. They need references and recommendations regarding culinary places that suit their needs [3].

Mobile devices have an essential role in interacting and sharing information. The use of mobile devices has become mandatory today because it influences various sectors [4]. Therefore, everyone uses mobile devices to support their activities. People use mobile devices as directions to a place, get recommendations, or other information related to location [5].

Recommendation system is a system that recommends items to users using available data and user interests [6]. The provision of items relates to the decision-making process by the user. Recommendations given to users are made based on interactions by previous users or preferences given by users [7]. Many previous studies have used a recommendation system on a mobile application as a topic. One of those is a culinary place recommendation system [3], [8]–[12].

In general, someone will choose a culinary place based on the desired criteria. These criteria can be in location, price, menu served, available facilities, and others [9]. Therefore, the selection of a culinary place for each person will be subjective based on the tastes or preferences of that person. However, some previous studies still rely on ratings from users [3], [8], [10]–[12]. In addition, previous studies only require users to give their preferences at the beginning of the recommendation process and ignore the operating hours of the recommended culinary places [3], [8]–[12]. Therefore, the recommended culinary places are not relevant to the time when the user performs the recommendation process. The user also cannot modify the recommendations given by the recommender system.

In this study, we developed a recommendation system for culinary places using the chatbot platform on the Conversational Recommender System (CRS). We use CRS because it analyzes user preferences to provide recommendations according to user preferences [13], and CRS is a type of Knowledge-based Recommender Systems (KBRS) [14]. KBRS is an approach that solves the cold-start problem in Collaborative Filtering. We use chatbots to make the recommendation process more interactive, like human dialogue. This study develops previous research [15], [16], which used the CRS method with a chatbot platform in the movie and music domains. Previous studies have carried out the use of CRS on chatbots in the recommendation system for culinary places [17]. The research uses the Collaborative Filtering method and utilizes the location of the user to generate recommendations. We take advantage of user preferences and feedback during conversational interactions with bots and the time.
context of users to generate recommendations. We use the Telegram platform to implement the chatbot because smartphone users have widely used it.

2. RESEARCH METHODOLOGY

2.1 System Design

This study develops a Conversational Recommender System with a chatbot platform. At the initial stage, the system will collect preferences from users based on the properties of the culinary place or the culinary place itself, which is called the preference acquisition stage. Furthermore, if the user profile is complete, the user can ask the system for recommendations to provide recommendation results. For each recommendation item, users can provide feedback or skip the recommendation item. Users can also see details (properties) of the culinary place or explain why the culinary place is recommended using the explanation facility. We describe the system flow with a block diagram as shown in Figure 1 [16].

![System block diagram](image)

Figure 1. System block diagram

2.2 Dataset

We use the following criteria in the dataset used such as the name of the culinary place, location, address, operating hours, halal, as well as dishes, cuisine, and facilities available at the culinary place as shown in Table 1. 100 data were collected manually from Traveloka.com and PergiKuliner.com with coverage in the city of Bandung using the Indonesian Language.

| Name       | Location     | Address                                                                 | Operational Hours | Dishes          | Cuisine | Halal | Facility |
|------------|--------------|-------------------------------------------------------------------------|-------------------|-----------------|---------|-------|----------|
| Pepper     | Trans Studio | Trans Studio Mall, Lantai 2, Jl. Gatot Subroto No. 289, Cibangkong, Batununggal, Bandung | 10.00 - 22.00     | Yakiniku, Daging. | Asia    | ?     | Bawa Pulang |

Table 1. Examples of some culinary places in the dataset
2.3 Conversational Recommender Systems

Conversational Recommender Systems (CRS), a type of recommendation system from KBRS, uses user preferences from questions given explicitly and repeatedly or from user feedback. CRS performs the process of interacting with users until they select items that match the given preferences [18]. The interaction and criticism carried out in the recommendation process can be through voice input, text dialogue, or buttons. The output can be through voice, text, or multimedia content. The expected goal of this system is to increase the accuracy of the recommendations provided and facilitate interaction between users and the system.

In this study, we use text dialogs and buttons in interaction and criticism between the user and the system. A combined interaction between text dialogs and buttons (mixed interaction) has been carried out in previous studies [16]. The system requires components to extract the information provided by the user by analyzing natural language in the interaction process. There are four components used in this study, namely Dialog Manager, Intent Recognizer, Entity Recognizer, and Recommendation Services, as shown in Figure 2, which is the CRS architecture in this study.

![Figure 2. System Architecture](image)

**Dialog manager** is the main component of this system to maintain interaction between the system and the user. Dialog manager will receive messages from users, then call the components needed to process messages from users, and send responses back to users[16]. In this study, dialog manager uses a slot-filling model where the user must answer a series of questions from the system to achieve the user’s goals. The response sent to the user will use the template created and filled with the appropriate context.

**Intent recognizer** is a component that uses natural language to detect intent or intent from the user. From the detection results of the intent recognizer, the system will understand what goal is to be achieved and what is expressed in the message [16]. In this system, the intent recognizer detects user messages using Indonesian Language, as shown in Table 2. The intents used in this study are:

- **Preference** is where the user gives preference to the system. In addition, this intent has sub-intents: give preference type when the user provides the type of property that will be used as a preference to the system and give preference item when the user provides an item that they want to be a preference.
- **Recommendation** is where the user asks the system for recommendations. This intent also has sub-intents: feedback when the user provides feedback on the recommended item, details if the user wants the system to display the details (all properties) of the recommended items, and explanation when the user asks for an explanation of why the system recommends the item to the user.

| Name        | Location | Address                          | Operational Hours | Dishes | Cuisine | Halal | Facility |
|-------------|----------|---------------------------------|-------------------|--------|---------|-------|----------|
| Imah Seniman | Lembang  | Jl. Kolonel Masturi No. 8, Gudangkahuripan, Lembang, Bandung Barat | 07.00 - 21.00     | Ayam, Goreng, Nasi, Goreng, Ikan | Asia, Sunda | Ya Parkir, Wifi, Reservasi, Area | Merokok |

**Table 2. Example of multiple intents**

| Message          | Intent         |
|------------------|----------------|
| Aku mau kasih preferensi | preference     |
| Kasih aku rekomendasi    | recommendation |

**Entity Recognizer** is a component that determines the entity contained in the user's message, then connects it to the Knowledge Base (KB) [16]. KB uses utilizes culinary place data that has been collected manually. The type of entity to be determined is only related to the domain in this study, namely culinary places, as in Table 3. This component has two approaches in detecting entities, namely spotting and disambiguation. At the spotting stage, the system will analyze messages from users to determine candidate entities by detecting word sequences that have similarities with the entity. If there are no entities that have similarities at the spotting stage, then the next step is disambiguation. In the disambiguation stage, the system will analyze messages from users and use the
Jaro-Winkler algorithm to find a match between user messages and data from KB. The analysis results will then generate a score for each string of KB and send top-3 strings that are similar to the user's message. The user will later choose from the three strings that match the intent of the user.

Table 3. Example of multiple entity

| Message                  | Entity       |
|--------------------------|--------------|
| Aku mau tambah preferensi| makanan      |
| makanan                  | Wifi         |
| ada fasilitas Wifi       |              |

Recommendation Services in this system use the Personalized PageRank algorithm to generate recommendations for users. The Personalized PageRank algorithm will run on a graph consisting of items and attributes from the dataset. The nodes in the graph in the system will later contain entities from the dataset, and edges will contain links between these entities. This algorithm will generate a score for each node in the graph. The system will recommend to the user the top-5 score from the node that represents the culinary place. In addition, the system also uses an explanation facility. This facility utilizes the connection of recommended items and preferences from the user profile so that the system can explain why recommending the item to the user.

2.4 Jaro-Winkler Distance

Jaro-Winkler is a pattern matching technique to measure the match between two strings. This technique modifies the Jaro algorithm where there are errors at the beginning of the string, and the process is faster than the Levenshtein algorithm [19]. Jaro-Winkler uses the principle that the importance of similarity in the beginning of the string at the end of the string [20]. Here is the Jaro-Winkler equation [20]:

\[ \text{sim}_w = \text{sim}_j + lp(1 - \text{sim}_j) \]  

Where \( \text{sim}_j \) is the equation of \( s_1 \) and \( s_2 \):

\[ \text{sim}_j = \begin{cases} 0, & \text{if } m = 0 \\ \frac{1}{3} \left( \frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m - l}{|m|} \right), & \text{otherwise} \end{cases} \]  

While \( l \) is the prefix length of a string up to 4 characters. \( p \) is set with the value \( p = 0.1 \), which is the standard value of the constant. We use the Jaro-Winkler algorithm to measure the match of the string of user messages with the nodes of the knowledge graph in the disambiguation process.

2.5 Personalized PageRank

Personalized PageRank, commonly called PageRank with Priors, is an algorithm used to find popular nodes based on topics from user preferences. This algorithm is also used for the ranking process on items recommended to users by determining the best sample of each topic on a list of basic topics according to the domain. After that, modifications are made to the PageRank equation so that the teleportation process is only found in the previously determined sample as follows [7]:

\[ \tilde{\pi} = \frac{\alpha \tilde{e}_p}{n_p} + (1 - \alpha)P^T \tilde{\pi} \]  

Where \( \tilde{e}_p \) is an \( n \)-dimensional personalization vector (column) with each row containing the value 1 if nodes are included in the specified sample and otherwise is 0. Values that are not 0 in \( \tilde{e}_p \) are denoted by \( n_p \). The Personalized PageRank algorithm runs on a graph consisting of items and their properties from the dataset. The nodes in the graph are items from the dataset and also the properties of these items (e.g., Pago Restaurant, Wifi, Fried Chicken), and the edges in the graph are the relationships between nodes (e.g., food, facilities, operating hours) such as the model graph in Figure 3.

Figure 3. Graph Model
2.9 Explanation

Explanation is one of the facilities in the recommendation system where the system explains the recommendations given to users. This explanation facility explains why the product is recommended to users and provides details of the recommended product so that users can know the quality of the recommended product [21]. In this study, we use both types of explanations. The details of a product will display all the properties of a culinary place. Furthermore, explaining a recommended product to users will implement the previous research explanation facility [22]. The following is the score equation for the explanation facility [22]:

\[
\text{score}(c, \mathcal{I}_p, \mathcal{I}_r) = \left( \alpha \frac{n_c I_p}{|\mathcal{I}_p|} + \beta \frac{n_c I_r}{|\mathcal{I}_r|} \right) \times \text{IDF}_c
\]  

(4)

$I_p$ is a collection of items from the user profile, and $I_r$ is a collection of items recommended to users. In addition, $c$ is a property that relates the two items of the user profile and the recommendation results, which are called candidate properties. Based on these candidates, $n_c I_p$ is the number of nodes connected between $c$ and the items from the user profile, while $n_c I_r$ is the number of nodes connected between $c$ and the items from the recommendation. $\alpha$ and $\beta$ are weighting factors set with a value of 0.5 as in the study [22]. $\text{IDF}_c$ counts how many nodes of the graph are connected with that property. This formula will produce a score that is used to determine the ranking of the candidate properties. In this study, we use the top-5 of candidate properties to generate explanations. If $I_p = 0$, then a non-personalized explanation style will be used as an explanation facility.

3. RESULT AND DISCUSSION

3.1 System Implementation

The system built aims to provide recommendations for culinary places according to user preferences. We developed a Conversational Recommender System where the system will ask for preferences from users, generate recommendations, and provide feedback regarding the results of the recommendations.

3.1.1 Preference Acquisition

The first step the system does is to explain instructions regarding the use of bots. After that, the system will ask at least three preferences to the user to form a user profile, as shown in Figure 4. This system uses mixed-mode interaction [23], where all interactions use natural language except in the disambiguation process, where there are several candidate entities using Buttons. Before providing the desired preference, the system will ask the user to select the type of preference in question, such as food, cuisine, facilities, or restaurants. After that, users can give preference items on the system. Users can also see some lists of available nodes on the Knowledge Base. When the user gives an item preference, the system will use the Jaro-Winkler algorithm to measure the match of the user-supplied item with the nodes in the graph. The system will implement the disambiguation process by asking the user to select the top-3 string from the results of the Jaro-Winkler algorithm if the system does not find a match from the item provided by the user, as shown in Figure 4.

Figure 4. Preference acquisition (left), disambiguation process (center) and multiple node list from KB (right)
3.1.2 Recommendation process

Users can request recommendations to the system when the preferences in the user profile are ready. The algorithm used in this system is Personalized PageRank with nodes in the graph as entities from the dataset. The Personalized PageRank algorithm has been used in previous studies [15], [16]. Before carrying out the recommendation process, the system will delete nodes (restaurant) on the graph that do not match the time context of the user. Next, the system will use the preferences of the user profile in the Personalized PageRank algorithm. After the recommendation process is complete, the system will display the top-5 recommendation results to the user. Each item will be asked for feedback from the user, as shown in Figure 5.

3.1.3 Item Explanation

At this stage, the user can ask the system why an item is recommended to the user. The method exploits the connection between user preferences and recommended items. The explanation process uses the algorithm used in research [22], to rank the connected properties between user preferences and recommended items. In addition, users can also request details of the recommended items, as shown in figure 6.
Figure 7. The user likes the item (left) and dislikes the item (right)

3.1.4 User Feedback

At the last stage, users can provide feedback on the results of the recommendations given. This process will later filter the recommendation results based on the feedback provided by the user. Users can provide three feedback choices to the system: 1) the user likes the recommended item, 2) the user does not like the recommended item, and 3) the user wants to skip the recommended item. For options when the user does not like the recommended item, the user can choose not to like the item or dislike one of the item's properties, as shown in Figure 7. The feedback provided will provide updates to the user profile and graph.

3.2 Evaluation

In this study, we use two types of tests to evaluate the system, namely qualitative and quantitative. For qualitative testing, we used a questionnaire. Furthermore, for quantitative testing, we use the accuracy of the recommendations. 81 respondents carried out both tests.

3.2.1 Questionnaire

The questionnaire in this study covers eight factors and uses 15 questions that have been used in previous research [16]. We use a Likert scale which is represented by numbers 1 to 5. With number 1 equal to “Strongly disagree” up to number 5 equal to “Strongly agree”. The value of each question in table 4 is presented in the form of average and median.

| Question ID | Median | Average |
|-------------|--------|---------|
| Q7          | 4      | 4,20    |
| Q8          | 4      | 4,19    |
| Q9          | 4      | 4,24    |
| Q10         | 4      | 4,10    |
| Q11         | 4      | 4,08    |
| Q12         | 4      | 4,23    |
| Q13         | 4      | 4,20    |
| Q14         | 4      | 4,24    |
| Q15         | 4      | 4,16    |
| Q16         | 4      | 4,20    |
| Q17         | 4      | 4,25    |
| Q18         | 4      | 4,26    |
| Q20         | 4      | 3,99    |
| Q21         | 4      | 4,16    |
| Q22         | 4      | 4,21    |

Based on the median of the questionnaire results, it was found that the score obtained on all questions was 4, which is the value of "Agree". Therefore, it can be concluded that generally, users are satisfied with the system. There is an interesting thing in Q20 where only the question produces an average of < 4. This can happen because
users can only correct misunderstandings in the feedback process, which makes the user unable to correct misunderstandings in other processes, such as the preference acquisition process. However, on another question, the resulting average value > 4, which proves the results of the qualitative system evaluation received a positive response. We did not use Q19 on the questionnaire because the NLU sample on the system is made manually and is limited, so the possibility of the system not understanding user messages is high.

3.2.2 Recommendation accuracy

In quantitative testing, we use the accuracy of the system used by users. According to research [24], the equations used to measure accuracy are as follows:

\[
\text{accuracy} = \frac{\text{number of successful recommendations}}{\text{number of recommendations}} \tag{5}
\]

For each recommendation result, the system generates five recommended items for the user. Therefore, based on equation 4, it is found that the resulting accuracy is 40%. However, this accuracy shows that the performance of the system is not good. This can happen because the data from KB is still limited. In addition, successful items are recommended to users only when users tell the system that they like the recommended item, which results in users leaving the recommendation process before providing feedback on all items.

4. CONCLUSION

In this study, we developed a recommendation system using the Conversational Recommender System with a chatbot platform in the culinary place domain. We use mixed-mode in the interaction process and time information context in the recommendation process. We also use a critiquing strategy to get feedback from users. We evaluated this study using two methods, qualitative and quantitative. Qualitative results conducted using questionnaires show that users are satisfied with using the recommendation system that was built. However, users are still unable to correct misunderstandings in giving messages to the system. For quantitative results, we use accuracy from the recommendation results and get a value of 40%. The low performance of the system occurs because of the limited Knowledge Base available. In addition, users do not provide feedback on all recommendation items because users can leave the recommendation process before providing feedback on all items. Suggestions for further research can increase the data on culinary places in KB to display more culinary places. In addition, properties used in culinary places can be added (e.g., food prices) to add preference options and recommendation results.

REFERENCES

[1] D. T. Untari, “The development strategy of betawi eco-culinary tourism as a potential business in DKI Jakarta, Indonesia,” African Journal of Hospitality, Tourism and Leisure, vol. 2019, no. Special Issue, pp. 1–9, 2019.
[2] R. Komaladewi, “The Representation of Culinary Experience as the Future of Indonesian Tourism Cases in Bandung City, West Java,” International Journal of Business and Economic Affairs, vol. 2, no. 5, pp. 268–275, 2017, doi: 10.24088/ijbea-2017-25001.
[3] E. Riswanto, B. Robi’In, and Suparyanto, “Mobile Recommendation System for Culinary Tourism Destination using KNN (K-nearest neighbor),” in Journal of Physics: Conference Series, 2019, vol. 1201, no. 1. doi: 10.1088/1742-6596/1201/1/012039.
[4] S. Malathy and P. Kama, “Application of mobile technologies to libraries,” DESIDOC Journal of Library and Information Technology, vol. 33, no. 5, pp. 361–366, 2013, doi: 10.14429/djilt.33.5098.
[5] Maeve. Duggan, “Cell Phone Activities 2013.” https://www.pewresearch.org/internet/2013/09/19/cell-phone-activities-2013/ (accessed Aug. 11, 2021).
[6] F. Ricci, L. Rokach, and B. Shapira, “Introduction to Recommender Systems Handbook,” in Recommender Systems Handbook, Springer, Boston, MA, 2011, pp. 1–35. doi: 10.1007/978-0-387-85820-3_1.
[7] C. C. Aggarwal, Recommender Systems, vol. 1. Cham: Springer International Publishing, 2016. doi: 10.1007/978-3-319-29659-3.
[8] R. K. Dewi, M. T. Ananta, and L. Fanani, “The Development of Mobile Culinary Recommendation System Based on Group Decision Support System,” Int. J. Interact. Mob. Technol., vol. 12, no. 3, pp. 209–216, 2018.
[9] F. Effendy, B. Nuqoba, and Taufik, “Culinary recommendation application based on user preferences using fuzzy topsis,” IJUM Engineering Journal, vol. 20, no. 2, pp. 163–175, 2019, doi: 10.31436/ijuem.v20i2.1023.
[10] A. Pinandito, M. T. Ananta, K. C. Brata, and L. Fanani, “Alternatives weighting in analytic hierarchy process of mobile culinary recommendation system using fuzzy,” ARPN Journal of Engineering and Applied Science, vol. 10, no. 19, pp. 8791–8798, 2015.
[11] A. A. Fakhri, Z. K. A. Baizal, and E. B. Setiawan, “Restaurant Recommender System Using User-Based Collaborative Filtering Approach: A Case Study at Bandung Raya Region,” Journal of Physics: Conference Series, vol. 1192, no. 1, 2019, doi: 10.1088/1742-6596/1192/1/012023.
A. Pinandito, C. Putri, and R. Kartika, “Culinary Recommendation Systems Using Analytical Hierarchy Process on Google Android Platform,” in *International Conference on Engineering and Information Technology (ICEIT)*, 2015, pp. 138–147.

Z. K. A. Baizal, D. H. Widyanotoro, and N. U. Maulidevi, “Computational model for generating interactions in conversational recommender system based on product functional requirements,” *Data & Knowledge Engineering*, vol. 128, p. 101813, Jul. 2020, doi: 10.1016/J.DATARK.2020.101813.

Z. K. A. Baizal, D. Tarwidi, and B. Wijaya, “Tourism Destination Recommendation Using Ontology-based Conversational Recommender System,” *International Journal of Computing and Digital System*, vol. 1, no. 1, 2021.

F. Narducci, M. de Gemmis, P. Lops, and G. Semeraro, “Improving the User Experience with a Conversational Recommender System,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11298 LNAl, pp. 528–538, Nov. 2018, doi: 10.1007/978-3-030-03840-3_39.

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,” *User Modeling and User-Adapted Interaction 2019 30:2*, vol. 30, no. 2, pp. 251–284, Nov. 2019, doi: 10.1007/S11257-019-09250-7.

N. Sardella, C. Biancalana, A. Micarelli, and G. Sansonetti, “An Approach to Conversational Recommendation of Restaurants,” *Communications in Computer and Information Science*, vol. 1034, pp. 123–130, Jul. 2019, doi: 10.1007/978-3-030-23525-3_16.

Z. K. Abdurahman Baizal, Y. R. Murti, and Adiwijaya, “Evaluating functional requirements-based compound critiquing on conversational recommender system,” *2017 5th International Conference on Information and Communication Technology, ICOICT 2017*, pp. 1–6, Oct. 2017, doi: 10.1109/ICOICT.2017.8074656.

P. Christen, “A comparison of personal name matching: Techniques and practical issues,” *Proceedings - IEEE International Conference on Data Mining, ICDM*, pp. 290–294, 2006, doi: 10.1109/ICDMW.2006.2.

B. M. Changmai, D. Nagaraju, D. P. Mohanty, K. Singh, K. Bansal, and S. Moharana, “On-Device User Intent Prediction for Context and Sequence Aware Recommendation,” Sep. 2019, Accessed: Aug. 12, 2021. [Online]. Available: https://arxiv.org/abs/1909.12756v1

N. Tintarev and J. Mastroff, “Evaluating the effectiveness of explanations for recommender systems,” *User Modeling and User-Adapted Interaction 2012 22:4*, vol. 22, no. 4, pp. 399–439, Feb. 2012, doi: 10.1007/S11257-011-9117-5.

C. Musto, F. Narducci, P. Lops, M. de Gemmis, and G. Semeraro, “ExpLOD: A framework for explaining recommendations based on the linked open data cloud,” in *RecSys 2016 - Proceedings of the 10th ACM Conference on Recommender Systems*, Sep. 2016, pp. 151–154, doi: 10.1145/2959100.2959173.

A. Iovine, F. Narducci, and G. Semeraro, “Conversational Recommender Systems and natural language: A study through the ConveRSE framework,” *Decision Support Systems*, vol. 131, no. January, 2020, doi: 10.1016/j.dss.2020.113250.

F. Hernández del Olmo and E. Gaudioso, “Evaluation of recommender systems: A new approach,” *Expert Systems with Applications*, vol. 35, no. 3, pp. 790–804, Oct. 2008, doi: 10.1016/J.ESWA.2007.07.047.