Context-Aware Recommendation Via Interactive Conversational Agents: A Case in Business Analytics

A Technical Report submitted to the Department of Engineering Systems and Environment

Presented to the Faculty of the School of Engineering and Applied Science

University of Virginia • Charlottesville, Virginia

In Partial Fulfillment of the Requirements for the Degree

Bachelor of Science, School of Engineering

Harish Satya Karumuri

Spring, 2022

Technical Project Team Members

Livia Kimche

Omer Toker

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Afsaneh Doryab, Department of Systems and Information Engineering, Computer Science

Context-Aware Recommendation Via Interactive Conversational Agents: A Case in Business Analytics

Harish Karumuri, Livia Kimche, Omer Toker, Afsaneh Doryab
University of Virginia, Charlottesville VA
{ot3ae, lrk2pb, hsk5pa, ad4ks}@virginia.edu

Abstract— **In the era of information overload, the ability to access key information instantaneously is extremely important. While technological advances such as keyword search, dashboards, customizable data reports, and notifications have made information access more flexible, the underlying assumption is that the user knows what to look for. However, this assumption may not hold in many situations. For example, identifying needed information and key metrics affecting a business in Human Resource Management Systems (HRMS) can prove to be difficult. Voice assistance and recommendation systems can help improve these issues by allowing users to efficiently reach key insights which are relevant to their needs and their context. This research presents the design and evaluation of a conversational context-aware information recommendation system for business analytics where a conversational voice assistant helps the user specify the information needed for different analytics by suggesting reports and metrics often used by similar users and companies in their industry. Our prototype evaluation results show the potential of such a system to improve the user experience of searching for efficient and meaningful information in an organization using the data available within their HRMS.**

I. Introduction

Searching for information can be frustrating due to information overload and an initial lack of knowledge. Complex scenarios such as creating analytic reports for businesses can further contribute to frustration and cause delays in retrieving necessary and useful insights. For example, human resource management systems have large quantities of data that can be used to provide valuable insights for a company. However, there is a lack of accessible and digestible analytics from current human resource management systems. Despite the efforts to ameliorate such problems through offering features such as flexible dashboards, customizable data reports, PDF exportation, and notification systems. Context-aware recommenders allow systems to analyze user context to provide personalized services. Personalized recommendations can be generated from past user activity by analyzing their preferences. Combining these capabilities with an interactive conversational agent has the potential to help users efficiently access needed information, especially for novice users and those with visual disabilities. An added recommendation system can deliver helpful insights before the user even knows they are looking for them. Overall, this system can improve the user experience of searching for relevant and meaningful information from an organization's available data.

We designed a conversational context-aware recommendation system to provide customized insights based on the user's interaction with data features. We developed and evaluated a stand-alone prototype to act as a technical proof-of-concept for a human resource management system (HRMS). The primary goal was to understand how the user experience (UX) can be improved when a user navigates their organization's data and identify technical limitations and considerations that can arise when building large and diverse systems. The process involved a design phase, followed by developing wireframes and technical prototypes to understand the integration of voice assistants into the HRMS. The voice assistant required natural language processing and intent recognition. The recommendation system required an iterative user-interaction log. The mainstream workflow required cross-platform integration capabilities for seamless transition from smartwatches, mobile devices, and desktop computers. We evaluated our prototype with a user experience study as well as a synthetic-data analysis of conversational recommendations.

Our research contributes to understanding the usability and feasibility of a conversational context-aware recommendation system for information retrieval that assists users with finding and using needed information in their context. In particular, our prototype evaluation provides insights into the technical requirements of such a system to help users generate intelligent business analytics in HRMS.

II. Related Work

The highest priority in our research was to design and implement the conversational voice assistant because of its unique potential to increase accessibility to insights and to provide context-aware recommendations to the user through the voice assistant to make the experience as natural as possible. In this section, we discuss existing work related to the essential components of the conversational context-aware information recommendation.

A. Voice Assistance

Prior research has shown that despite high awareness of voice assistance not many people who are aware of the technology are actually using it (PwC, 2018). The most influential factors to user adoption of voice assistants revolve around trust and attitude towards the technology. Conversational voice assistants can be accomplished with better natural language processing (NLP). If the NLP system is able to understand user intentions seamlessly and respond

978-1-6654-5111-6/22/$31.00 ©2022 IEEE

with humanlike conversations with natural back and forth flow, the system is more likely to achieve a strong anthropomorphism and encourage user trust. Voice assistance technology has been studied in different contexts to discover its utility for different user types. For example, in the medical environment, surveys given to 154 expert health care panelists showed that voice-controlled intelligent personal assistants would be useful to support elderly people self-therapy, and communicational tools by patients and health care professionals (Ermolina & Tiberius, 2021). However, many did not trust voice assistance to outperform the value of a personal medical staff such as remembering medical records. This study proves the utility of voice assistance and also implies how a human-like conversational voice assistant accomplished by higher level NLP could improve upon the current state of the art.

B. *Interactive Machine Learning*

Interactive machine learning (IML) is the design and implementation of algorithms that operate machine learning with the aid of human interaction. IML allows users to interactively examine the impact of their actions and adapt subsequent inputs to obtain desired behaviors. This modeling technique has been used for visual pattern mining, visual topic analysis, interactive information retrieval, and interactive anomaly detection (Jiang et al. 2018). In our research, we leverage IML in generating recommendations. A log file is generated at each interaction the user has with the assistant. This logs are used during interactions to generate analytical insights based on metric preferences, and metric suggestions recommended to users in past usage.

C. *Context-aware Recommendation*

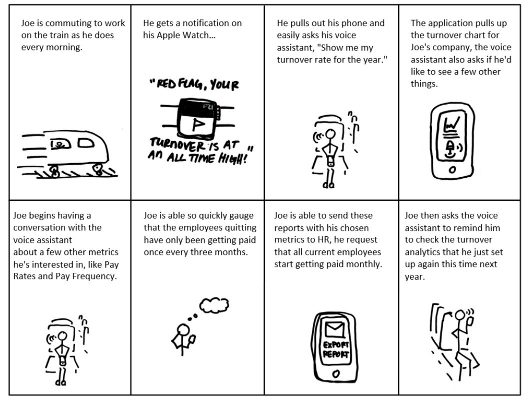
Context includes any information that is relevant to the user's situation and task e.g., location and time (Abowd et al., 1999). Context-aware recommendations are personalized to the user, which is an essential component of a good user experience design. A good recommender system has to be aware of what the user has been doing in the past and how the user's preferences differ from others. Context-awareness in our approach facilitates generating personalized analytical insights according to the metrics the client wants to evaluate. Combining context awareness with interactive machine learning creates a product lifecycle that learns from its own experience; this allows the system to increase its user experience at each iteration.

III. Methods

A. *Iterative User Experience Prototyping*

User experience (UX) design is the process of creating meaningful and relevant experiences for the users of an interface. Prototyping was centric to the use cases formulated with the HRMS company. These use cases were building blocks for what would be useful in the context of an end-user. We used storyboarding as a visual tool to explore how a user would interact with specific design features. Throughout the many iterations of designing this voice assistant for business analytics, new storyboards were created to establish the scope of what users could ask of the assistant. Through this process, the team was able to recognize the value of having a voice assistant suggest related metrics or ways to filter or facet the data to identify underlying organizational issues. Additionally, constructing realistic scenarios helped identify the need to keep a flexible solution that could provide value in a large variety of situations.

Figure 1: Use Case Story Board



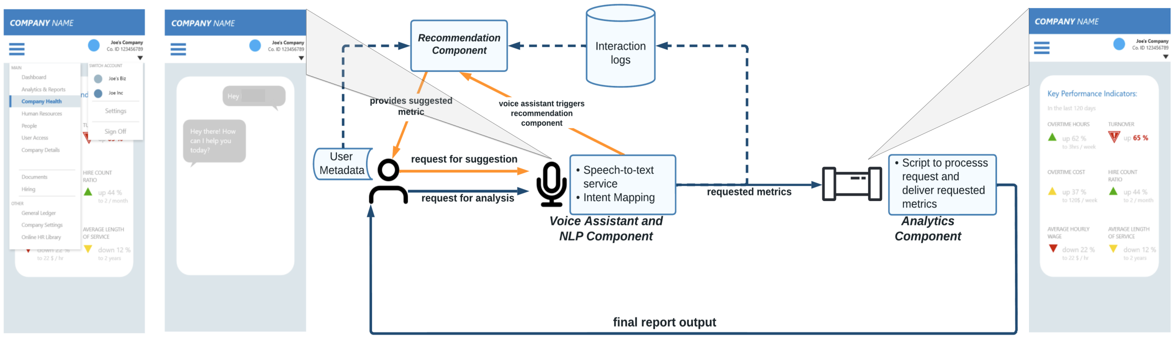
B. *Natural Language Processing*

Natural language processing (NLP) is the computational ability of computers to understand human language as it is spoken and written. NLP was used in our model to interpret what the user was directly asking for, but also to process the user's true intentions. The intent of the user was used to deliver helpful information to the user, even if they are asking for it indirectly. For example, a user might ask "Can you show me my turnover rate?" or "What is my employee retention?" when searching for the same metric. The NLP design processes language and user intentions to deliver the appropriate data insights. When a user speaks into the voice assistant, the Google Speech API can be used to accurately convert the audio into text. This text can be processed by the intent models to trigger actions. This connection between spoken audio and NLP enriches the communication the user has with the assistant by making it feel more natural.

C. *Recommendation Algorithms*

We leverage item-based, user-based, context-based and hybrid recommendation methods to generate and predict the needed information during the user interaction with the conversational voice assistant. All methods are based on nearest-neighbor. Item-based recommendation matches items based on their similarities with items the user has previously worked with. We generated item-based recommendations for our users where the items were metrics they had previously worked with. *User-based recommendation* matches users based on their similar preferences and profiles and aims to automate the common principle of word-of-mouth (Ricci et al., 2010). Users in our case are the ones that have been using the system and interacting with the voice assistant. Groupings of users can happen based on similar job roles or departments. For example, CEO's, administrators, and human resource representatives would be given recommendations for metrics or reports based on what another CEO, administrator, or human resource representative searched for. *Hybrid recommendation* combines content and collaborative filtering, by generating recommendations both based on profile matching and metric preference matching. This

Figure 2: Recommendation Generation Process



method becomes more accurate as the number of users and number of items in the system increases, because with more resources to work with it is able to match more content with similar attributes.

Context-based recommendation gets in depth with personalization by taking into account additional contextual information such as time and location. This method can be seen as a more personalized version of hybrid recommendation methods. Contextual information in our case is date time and location. The system provides more detailed recommendations to the user by filtering according to days of the week, month to generate trends as well as location to capture trends that vary by location.

D. Apriori Algorithm and Association Rule Mining

To mine frequent and preferred metrics used by the users, we leverage Association Rule Mining and more specifically the Apriori Algorithm (Agrawal et al. 1996) . Apriori Algorithm was initially designed for retail and market analysis based on buying patterns. This research extends the usage of the Apriori Algorithm and Association Rule Mining methodology to identify what metrics an individual user may prefer based on previously requested metrics. In this system, Apriori Algorithm processes metrics to generate quantitative values about their frequency in log data which are then fed into the Association Rule Mining process to generate quantitative results about the strength of relationship between different metrics.

An association rule is an expression X→Y, where X and Y are a set of items. Given a database D of transactions - where each transaction T in set D is a set of items, X →Y expresses that whenever a transaction T contains X than T probably also contains Y (Hipp et al, 1970). Items X and Y for this case are different metrics selected by the users and are used in the analysis. The strength of the association is measured with three main metrics: support, confidence, and lift.

Support is the measure of how frequent a metric is in all available log data of past usage, confidence measures the likelihood of occurrence of Y given that X is present, and lift indicates the importance of a rule by calculating the increase in probability of having Y in log data with the knowledge of X being present, over the probability of having Y in the log data without any knowledge about presence of X

E. Conversational Context-aware Information Recommendation

The overall architecture of the system that facilitates Conversational Context-aware Information Recommendation consists of five components: Front End, NLP, Voice Assistant, Analytics, and Recommendation.

1) Mobile Interface

We followed an iterative process to design a mobile interface for the conversational context-aware information recommendations using Adobe XD. The goal was to efficiently deliver the most important information to the user through conversation and visualization. The prototype includes the conversational voice assistant and representational analytics for the user based on their company's data. The system is designed to take the job role of the user into account to provide recommendations based on the user profile. The system also uses time and the user's location to provide insights, for example, if the user is located in a southern region, the voice assistant asks if the analytics should be specific to that region. The voice assistant is easily accessed by a simple "Hey Voice Assistant" command. The design will then automatically initiate the chat box screen providing the ability to the user to visualize and validate the conversation that they are having with the voice assistant. The bubbles playback the voice commands and voice assistant's responses in real-time.

2) Voice Assistant

The backend system processes commands and triggers the desired action. Initial stages of development began with converting the user's audio outputs into a string text through a speech-to-text processor. We used the pyttsx3 package to connect to the Google Cloud Speech API and respond with an auditory response. While this can help users navigate through explicit commands, using it alone fails to capture the wide variety of commands which yield a similar action. To encapsulate this conversational element within our system, we leveraged natural language processing and natural language understanding capabilities offered through the Neuralintents package developed by Florian Dedov. This free package uses libraries from TensorFlow and NLTK to train the underlying models enabling Natural Language Processing (NLP) and Natural Language Understanding (NLU) for the voice assistant. Intent models could be trained with a sample of trigger phrases formatted as a JSON fileFor example, an analytics intent was established with some associated phrases being "how is my company doing", "company's performance", "is my company healthy" or "what is my status". Any of these phrases or similar phrases could start the analytics process. It is more difficult to obtain desired metrics or parameters for reporting and analytics through the voice due to the nature of the package resulting in UX differences between the technical prototype and design prototype. Technology like Google's DialogFlow may allow for obtaining parameters and should be explored to improve this technical prototype's capabilities.

3) Analytics Generation

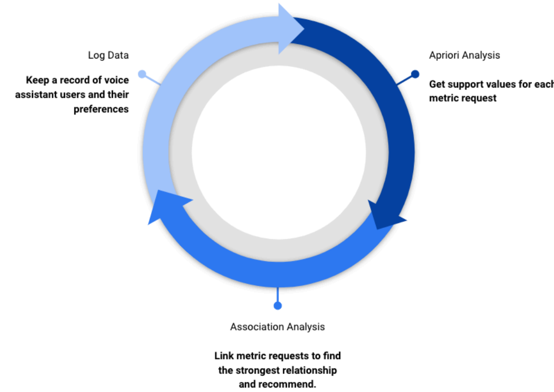
The team obtained raw data representative of employee data that an HRMS holds, and a script was developed to process the data to obtain relevant key metrics such as new hire count, turnover rate, average hourly wage, and the average length of service. After the user interacts with the voice assistant, the requested metrics are calculated and outputted to the user. Notably, this pipeline needed to be able to filter and facet data based on variables like location or department. In line with the technology stack used for this research, these scripts were developed in Python using the Pandas package.

4) Recommendation

At a high level, users' interaction with the assistant will create log data that will undergo apriori analysis to mine rules for association analysis. These rules then help provide recommendations for the user (Figure 3). Using the user profile containing information about the user's name, position, and location where they operate, a log of user interactions and metric requests were generated. User metadata was stored in a CSV file; interaction logs were stored individually in a folder as JSON files. These files were imported into Pandas data frames, which allowed for a custom voice assistant based on a number of factors interactive user experience. Within a more robust architecture, data would be stored within a database, and these data frames could be made using queries to their respective databases. With this data, the recommendation component uses a hierarchical model with item-based, profile-based, and context-based recommendations. Furthermore, item-based recommendations can directly use contextual information by subsetting the dataset towards a specific location, or another type of context. This hierarchical structure allows for the recommendation component to support two kinds of users: users who do not know what to look for and users who have some idea of what to look for. Users who already have a precise idea of what to look for may not need the voice assistant to provide suggestions. If the user does not quite know what to look for, the system can default to the user metadata; users with the same or similar roles can be synthesized to give a recommendation. In the cases where the user has some information on what to look for, the specific items already requested can follow the strongest association rules. The rules can be generated by the total dataset, or a subset based on time, location, or profile, allowing for more relevant rules.

Our research and designs were being continually evaluated using feedback from UX Design working on current HRMS. Direct communications and weekly meetings with company managers and user experience designers were influential on the design of the system. After completing the prototype, a user study was performed to evaluate the user experience of the Adobe XD model. Separately, another study was performed by conducting synthetic data simulations to evaluate the backend of the model and test the feasibility of recommendation generation from interaction logs.

Figure 3: Recommendation Generation Process



IV. Evaluation

Our research and designs were being continually evaluated using feedback from UX Design working on current HRMS. Direct communications and weekly meetings with company managers and user experience designers were influential on the design of the system. After completing the prototype, a user study was performed to evaluate the user experience of the Adobe XD model. Separately, another study was performed by conducting synthetic data simulations to evaluate the backend of the model and test the feasibility of recommendation generation from interaction logs.

V. Results

A. User Experience Study

Ultimately our research questions for the user experience design were: "Is a real user likely to use this voice assistant?" and "Does the user experience design help facilitate helpful interactions with the voice assistant?" We conducted a study with 12 people from three different groups in the partner company to test the viability of our prototype and determine its practical value. The users were asked to generate thoughts about the system after testing the Adobe XD prototype. Users were able to give voice commands and receive voice assistance speech playbacks to walk through applicable client case studies. Major takeaways from this study were relevant to use cases, ideas, potential roadblocks, and limitations for the application and conversational voice assistant.

Overall, the users expressed that they would appreciate the conversational NLP and context-aware recommendations to make analytic insight delivery more natural for the user. It was noted that the recommendation system would be a great way to have a more natural and helpful conversation with a voice assistant and seven out of 12 users indicated they would be likely to use the voice assistant for generating analytics. The sales engineers focused more on navigation features. The engineers liked the idea of linking analytics with reports. They also thought that cross platform usage can be helpful for ease of use and accessibility. The HR consultants thought the conversational agent presenting analytic results both in natural language and via visualization graphs would help get quantitative and qualitative insights. The consultants also emphasized the need for flexibility in choosing metrics. The study presented more nuanced use cases for the conversational agent such as requesting compliance documents, time off requests, task lists, reminders, attendance reports, and payroll processing. These are all cases that can be easily implemented to the model with our current structure due to the streamlined data pipeline. Adding metrics to user databases (like payroll processing dates, time off request dates, or punch in times to account for attendance) can functionalize these new use cases. Ultimately we discovered that real users were enthusiastic about the new design and would be likely to use the voice assistant in practice. It was noted that conceptually the conversational voice assistant and recommendation systems would be helpful to the users. The UX design has some areas for improvement which were noted specifically in the limitations and roadblocks provided by the users, but overall they expressed that they would use the voice assistant and they appreciated its potential for helping them with their tasks in the application.

B. *Synthetic Data Analysis: Recommendation System*

To evaluate the feasibility of using Apriori analysis to extract frequent metrics used in the past user interactions for recommending during conversations, we leveraged a simulated data generation approach that mimicked real-world user interactions with our system. In a report generation scenario, the user interacts with the voice assistant to specify which metrics should be included in the analytic report. If the user is not sure what metrics to include, s/he asks for recommendation. This interaction is logged and used to update the set of frequent metrics that were used for generating that type of analytic report (Figure 4). If no metrics are chosen before the user asks for a recommendation, profile-based matching is used to select the metric most used by the user's role. Additionally, context-based suggestions are provided such that if a user wants to limit the calculated metrics to a specific region, the recommendation system will first suggest the user's current location. If a metric is already requested in the report and the user asks for another suggestion, frequent metrics mined by the Apriori analysis are used to suggest metrics.

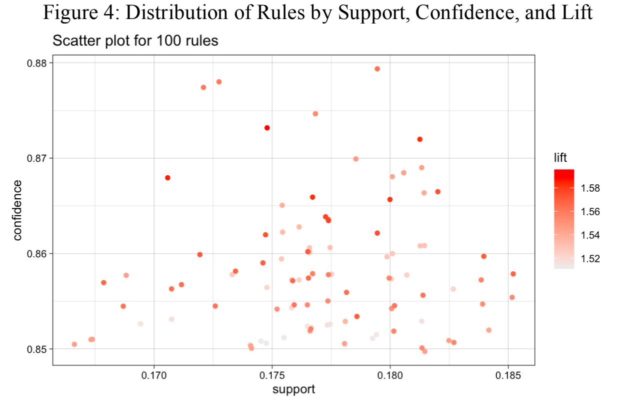
We generated the simulated interaction logs using ten unique metrics (turnover rate, head count, termination count, pay rate, average wage, available budget, overtime, labor cost, onboarding cost, and total hours worked). A random number (between one to 10) of these metrics was chosen as the set of requested metrics for a report. This metric list was assigned a random profile among four choices (Payroll Administrator, Business Analyst, Recruiter, and Project Manager). The Apriori algorithm allowed us to construct metric groupings according to three different statistical metrics, support, confidence, and lift as described in the Methods section. We set the minimum support and confidence thresholds for the frequently mined metric groups to 0.15 and 0.85 respectively. These numbers were empirically chosen based on the number of logs and the distribution of values. Based on the thresholds, the maximum number of metrics in antecedents were 5 for each profile.

Figure 4: Frequently Mined Metrics by Profile (P), Lift (L), Confidence (C), and Support (S)

Metric Sets	P	L	C	S
(Overtime, Pay Rate, Average Wage) → (Termination Count)	Overall	1,402	0.672	0.087
(Available Budget, Labor Cost, Onboarding Cost, Total Hours Worked) → (Overtime)	Recruiter	1,701	0.823	0.073
(Average Wage, Labor Cost, Onboarding Cost) → (Available Budget)	Project Manager	1,673	0.761	0.099
(Total Hours Worked, Labor Cost, Turnover Rate) → (Available Budget)	Business Analyst	1,725	0.821	0.086
(Available Budget, Turnover Rate, Head Count) → (Total Hours Worked)	Payroll Administrator	1,665	0.775	0.079

Figure 4 includes the two most frequently mined metric groups from each profile-base subsets and the overall log data. Support, confidence and lift statistics for the first and second recommendations are strictly close for each profile. Although interaction data was randomly generated and may not reflect actual interactions in the real-world, the figure provides initial insights into the expected outcome from the process. If the logs were created from actual users, they would frequently include at least 6 metrics in their reports. For example, the recruiter often includes budget, head count, labor cost, overtime rate and turnover rate whereas the payroll admin uses average wage, overtime, termination count, total hours, head count, and turnover rate. Figure 5 shows the distribution of rules with support, confidence, and lift above the threshold. While the support for most rules is above the mean value of 0.177, the confidence and lift of those rules remains below their means of 0.86 and 1.54.

Figure 4: Distribution of Rules by Support, Confidence, and Lift



In a second scenario, we generated another set of log data with a bias towards generating groups of metrics that we knew should happen together. The first group included turnover rate, head count, and termination count. The second group included pay rate, head count, and average wage. The third group included available budget, overtime, and headcount. The fourth group included labor cost, onboarding cost, average wage, and total hours worked. For any log file, there was a 88.9% chance that any one of these groups was chosen. For the other 11.1% of the time, the metrics generated on the log would be a random number of metrics from the list used in the first group.

Figure 6: Metric Based Rule Mining for Different Sample Sizes

	Turnover			Pay Rate			Available Budget			Labor Cost		
No. Interaction	Top Rule	Supp	Conf	Top Rule	Supp	Conf	Top Rule	Supp	Conf	Top Rule	Supp	Conf
20	Yes	0.16	1.00	No	0.37	1.00	No	0.16	1.00	No	0.11	1.00
50	Yes	0.13	0.86	No	0.25	0.80	No	0.21	1.00	No	0.13	1.00
100	Yes	0.14	0.93	Yes	0.27	0.87	Yes	0.23	0.79	No	0.11	0.85
500	Yes	0.23	0.77	Yes	0.25	0.84	Yes	0.25	0.85	No	0.11	0.88

VI. DISCUSSION

The two-part evaluation process provided a variety of feedback about different aspects of the prototype. Frontend evaluation provided reactions about UX design, system functionality, use cases, integration to the main product, and analytical ideas. The front-end evaluation also provided quantitative responses from different user groups about their likelihood to use a conversational recommendation system for business analytics. The user study validated the design process in terms of UX improvements, while also highlighting concerns with flexibility in how users can search for information. The recommendation system was an essential part of the system and context-aware recommendations benefitted the user experience flow, since it would provide newfound data insights accessible to both novice and skilled users with helpful recommendations. This means that there is a potential to reduce needed training for using new applications, which saves HRMS companies money in regard to training new clients on how to use their system.

The backend evaluation tested the feasibility of processing past interaction logs between users and the conversational agent to provide needed information during interactions. Synthetic data generation aims to facilitate testing implementations for the recommender system to know if mining past user interaction logs for the recommendation process is technically feasible. Evenly distributing synthetic data throughout different user profiles allowed to create an unbiased logging process. The results showed that even in randomly generated data with even distribution, the mined frequent metrics for each profile seem reasonable and realistic and can be used to guide the voice assistant with metric recommendations during the conversation. Overall, this evaluation shows it is feasible for the backend architecture to generate recommendations based on the rules generated by the Apriori algorithm.

Beyond the scope of companies, a conversational voice assistant and recommendation system based on our model has the potential to be implemented in many different contexts. Voice assistance and recommendation learning systems can help provide modern accessibility to users who may not have physical or visual capabilities to communicate with their devices. Our model introduces the potential to navigate an interface and receive information from verbal queue and audio responses alone. Interactive recommendation introduces helpful information to users before they even knew they were looking for it. The context-aware recommendations save time and frustration by making conversation flow with the voice assistant an intuitive experience. Context-awareness in our proposed system can also address some of the privacy and security concerns about using a conversational agent by adjusting the information delivery to the situation of the user. For example, asking the user whether certain information should be delivered in a public place. Overall these functions created by our model could help optimize the user experience of all types of interfaces. The model also has the potential to be scaled to diverse and widespread contexts.

VII. CONCLUSION

We leveraged a novel combination of natural language processing, interactive machine learning, context awareness, and user experience design to implement a conversational voice assistant to accomplish the overarching goal of delivering digestible data insights with context-aware recommendations. Through a user study, we gathered useful information and proved our model to be of value for the client company and potentially other systems. Our frequent item mining analysis also showed that leveraging user interaction history to generate information recommendations during interaction is technically feasible. Larger scoped testing and implementation with actual user interaction logs can be done to scale the system for more complex use cases. Future iterations of our prototype can boost current HRMS applications with a conversational information recommendation voice assistant. It can also be used to increase accessibility to data insights and optimize user workflows in all types of systems.

REFERENCES

[1] Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., & Verkamo, A. I. (1996). Fast discovery of association rules. Advances in knowledge discovery and data mining, 12(1), 307-328.

[2] Ermolina, A., & Tiberius, V. (2021). Voice-Controlled Intelligent Personal Assistants in Health Care: International Delphi Study. Journal of Medical Internet Research, 23(4), e25312.

[3] Gupta, J., & Gadge, J. (2015). Performance analysis of recommendation system based on collaborative filtering and demographics.

[4] Jiang, L., Liu, S. & Chen, C. Recent research advances on interactive machine learning.

[5] Pitardi, V., & Marriott, H. (2020). Alexa, she's not human but...Unveiling the drivers of consumers' trust in voice-based artificial intelligence. Wiley, 632-633.

[6] PwC. (2018). Consumer Intelligence Series: Prepare for the voice revolution. PwC. Retrieved March 24, 2022.

[7] Ricci, F., Rokach, L., & Shapira, B. (2011). Introduction to recommender systems handbook. In Recommender systems handbook (pp. 1-35). Springer, Boston, MA.