CHATWITHREC: TOWARD A REAL-TIME CONVERSATIONAL RECOMMENDER SYSTEM

[1]RANIA ALBALAWI, [2]TET HIN YEAP

School of Information Technology and Engineering, University of Oaawa, Ottawa, Ontario, Canada
E-mail: [1]ralba028@uoAawa.ca, [2]tet@eecs.uoAawa.ca

Abstract - Nowadays, the Internet helps to increase the demand for the development of commercial applications and services that can provide better shopping experiences and commercial activities for customers around the world. Research has investigated many approaches to produce optimal results in conversational recommendation systems (CRSs) that aim to analyze user-generated content (UGC) in online social networks (OSNs) landscape to suggest appropriate recommendations. However, it is hard to obtain the user's preferences and actual requirements at the beginning of the interaction, which makes users unsatisfied with recommending items and services. In this paper, we propose a conversational recommendation system called ChatWithRec that aims to solve the cold-start problem by detecting topics and matching them with appropriate content, like ads, after analyzing the user's contextual conversation dynamically, to increase the accuracy of recommendations. We apply the Latent Dirichlet Allocation topic model (LDA) to analyze user's conversation and observe topics dynamically. We integrated Google AdMob and customized ads databases to deliver conversation-related tasks. We evaluated the performance of our CRS by applying several statistical metrics. The results are encouraging and indicate that the system is fast and satisfies users by getting what they seek without interrupting their conversation flow, and solve the cold-start problem.

Keywords - Conversational Recommendation System, Real-time, Advertisement, NLP,Topic Modelling, Latent Dirichlet Allocation, Cold-start Problem.

I.INTRODUCTION

Social recommendation systems partially, the text-based form, have become a significant stream of research in e-Commerce [1]. They can be used within mobile Apps and/or desktop platforms, to increase the benefits of commercial activities and enhance the user's satisfaction, such as the movies recommendation like in [2]. Definitely, offering the right items at the right time only to interested users is a challenging subject, since predicting items is based on numerous processes and attributes like the user's history, profile, community trends, location, and social relations. For example, suggesting friends on Facebook based on the friendship strength like in [3], who provides the functionality of suggesting friends to people by utilizing their personalized information. Recently, there has been a high demand to develop conversational recommendation systems and services. On the other hand, there is a need to extract more hidden information from different online sources that are stored as text and written in natural language in OSNs domain, such as Instagram, LinkedIn, and Facebook. Numerous approaches and algorithms have been proposed in natural language processing (NLP) domain to mine and analyze texts in OSNs platforms, most based on classification, clustering, and probabilistic techniques. Yet, many existing approaches fail to satisfy users, specifically mobile Internet users. Moreover, there is a need for effective methods and tools that can help in analyzing social media content, mainly for those using online UGC. Our work is different from other existing works since we aim to enrich the conversation-related tasks, such as recommendation, by defining the user's requirements and needs in the real-time mode without considering their previous activities like purchase history and items rating. Furthermore, we aim to solve the cold-start problem that occurs when the system has no information about the user or does not rate any object yet, and insufficient user reviews about items. Finally, one of our goals is to provide a system that can help users to find valuable objects immediately without disturbing their conversation flow. We adopted the Google AdMob and customized databases to request and present an advertisement to involved users. Besides applied the LDA model to detect topics from the user's online conversation. The rest of the paper is structured as follows. In Section II, we explore the related research on CRSs in addition to the topic modelling subjects. Section III proposed the main architecture of our real-time conversational recommendation system. Section IV describes the first version of our proposed real-time conversational recommendation system. Section V, describes the enhanced version of our CRS. And Section VI presents the conclusion along with an outlook on future work.

II.RELATED WORK

Recent researches have investigated approaches to develop conversational recommendation systems that become a hot subject, and many kinds of researches study its different aspects such as social context, time context, and so forth. Following, we present some related works in the field of conversational recommender systems and topic modelling approach.

A. Conversational Recommendation Systems CRSs

CRSs have become very popular, and they implemented their strategies depending on the user's goals, especially in this age of global explosion in social media and the social networking services where individuals and groups participate in the sharing of opinions and information. This system supports a natural-like interaction mode and interactive dialogue, which is similar to the human interaction. CRSs have many interaction approaches in their requirements elicitation processes to identify recommendations. It also contains many approaches to display the best recommendation lists to users. For example, [4] presented an approach called FindMe that uses implicit user's model and displays recommended items based on the user's initial query and user's critiques of the current best item.

Moreover, in [5] researcher developed a restaurant's recommendation system called Entrée system, which suggests restaurants based on similar restaurants that the user knows and has liked before. Entrée presents users with a fixed set of critiques to accompany a suggested restaurant case. For example, the user may request another restaurant that is cheaper by evaluating its price and style features. Furthermore, the time factor can lead to more appropriate recommendations and enhance the CRSs by analyzing the needs of users at a particular time [6][7] Proposed a time-aware collaborative filtering to evaluate what the users might hold interest in when they browse through the Twitter social network platform. Most existing CRSs apply the text mining approaches to extract relevant data, like keywords. For example, [8] developed a method to extract information and then generate appropriate recommendations to a particular group of people. This method employs interactive and conversational approaches to gather information and use it to influence a group's decision-making process.

B. Topic Modelling

In recent times, there have been massive developments in the NLP domain, including topic modelling, which are based on machine learning algorithms for text mining, information extraction, sentiment analysis, etc. Some typical NLP real-world applications currently in use include automatically summarizing documents, named entity recognition, topic extraction, and topic modelling [9]. Actually, there are several topic modelling methods like LDA, and Term Frequency-Inverse Document Frequency (TF-IDF) that are used to extract topics from texts, such as emails, documents, blogs, etc. The choice of the method to extract topics is based on many features. Indeed, the LDA model is the most common and highly-used model in numerous fields and toolkits such as Gensim [10], Mallet [11], and Stanford TMT [12]because it is able to solve other models' limitations, such as LSI [13] and PLSI [14].

Many cited works in CRSs field applied the topic modelling method to recommend items and services. For example,[15] Developed a document recommendation system; that converted text from the Automatic Speech Recognition System, they used many methods which were semantic methods, cosine similarity, and LDA methods that applied in the MALLET toolkit environment to extract the most important and significant terms for short conversation fragments. [16] Proposed a news-topic recommender system based on extracting keywords from Internet news for a specific time. Researchers applied different keyword extraction algorithms such as TF-IDF and Rapid Automatic Keyword Extraction. However, the interest in CRSs still remains high, and there is more need for improvements like providing users with further appropriate and personalized recommendations. Indeed, in diverse existing studies, users are often not satisfied with the initial recommendations because presented systems usually they did not define the users' preferences at the beginning of the interaction [17]. Other studies proposed numerous methods and algorithms but without considering the user's actual requirements and interests. Furthermore, many current NLP algorithms and methods mostly detect the most significant and/or repeated keywords, like in document recommendation, without considering if they show the correct text topic. We believe that offering a CRS that can recognize the meaning of the user's conversation is important to recommend more interesting objects at the right time. However, there is more need to study the conversation context to provide better conversation-related tasks such as an advertisement that we involve as an output action in our proposed ChatWithRec CRS that is built based on the research presented in this work.

III. PROPOSED CHATWITHREC SYSTEM

The ChatWithRec CRS aims to suggest related-conversation tasks to users by analyzing their contextual conversations in real-time efficiently and effortlessly. Fig.1 shows the overall architecture of the proposed ChatWithRec CRS that has several layers, which are, the online social network layer, and the personal assistant server layer that contains two components, the dialogue analysis, and the advertisement recommendation components. In our previous work [18] we explained the system architecture in more details. The following section explains the ChatWithRec recommendation system architecture in details.

A. The OSNs Layer

Analyzing UGC such as users' posts and their profile has become an important topic today since ONSs help users easily communicate with others and share their thoughts, trends, etc. (e.g., recommending relevant product). However, OSNs have numerous limitations, such as the cold-start problem, privacy, and data sparseness, among others. Several researchers have addressed the cold-start problem for both users and items, these researchers have taken several approaches in order to overcome this drawback, such as users' social connections, profiles, and demographic information. However, it is challenging to define and predict users' preferences and current requirements from unstructured data, like in online conversation, that we aim to solve in this work. In this work, we aim to incorporate the users' personal social network applications and then suggest related content such as ads after analyzing their text-based interactions. Furthermore, we developed a chat app called ChatWithRec as a proof of concept in our proposed architecture that presents a possible solution to solve the cold-start problem, which is the main focus of this work. Our future goal will be to integrate other OSNs for a specific user who subscribes to our server to suggest more personal and accurate recommendations dynamically.

B. The Personal Assistant Server Layer

This layer is considered as the essential part of our proposed ChatWithRec CRS, it has the conversational analysis component, which was developed to monitor the user's conversation in a real-time manner, and the advertisement database. Conversational analysis module applied the LDA topic modelling to analyze the conversation session then detect topics. After extracting the conversation's topic, it sends to the AdMob database to filter the most suitable advertisements, then sends it to the only interested user after showing a pop-up notification message in his chat window to check if he is interested in displaying advertisements as a banner advertisement. Users can access the advertisement page by clicking on the banner advertisement message directly, and may also modify or cancel receiving specific ads. The Conversational analysis component is managed by using Python, a high-level programming language that supports multiple programming paradigms, which contains enormous comprehensive standard libraries. We should mention that most of our reviewed works implemented LDA model in Python and Java environment, not in the Xcode environment that we used in our implementation.

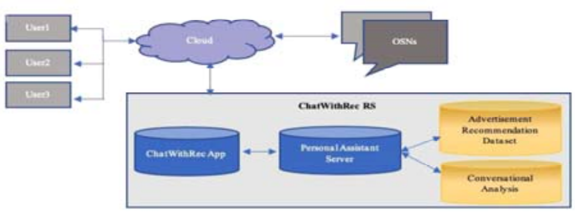

Fig.1: The main architecture of the proposed ChatWithRec CRS.

As a solution, we implemented the LDA model in Python and then integrated it into our Xcode environment that available in the ChatWithRec1 GitHub account. Also, during our work, we tested various topic modelling tools and algorithms to define the most appropriate approach that can present a relevant conversational topic in our proposed ChatWithRec recommendation system. We should mention that each tool has different processes to filter its topics and keywords. We applied the data pre-processing, which is a text-cleaning method to prepare it before sending it to the LDA model. It contains the following process: stop-word removing, stemming, lemmatizing, tokenizing, and identifying n-gram procedure such as bigrams and trigrams. Also, we applied the Bag-of-Word (BoW) algorithm that was treated in many previous related works. The LDA define what topic is discussed by online users, and then determine what kind of advertisements match the user's real requirements and interests. We applied the LDA model, since it is the most used model in real-life applications to extract keywords from text. Topics extracted are passed to the chat App using socket.io that used them to request an advertisement from the Google mobile ad called AdMod that is a popular ad-server and capable of delivering advertisement based on user's location with a low-performance cost.

IV.CHATWITHREC VERSION ONE

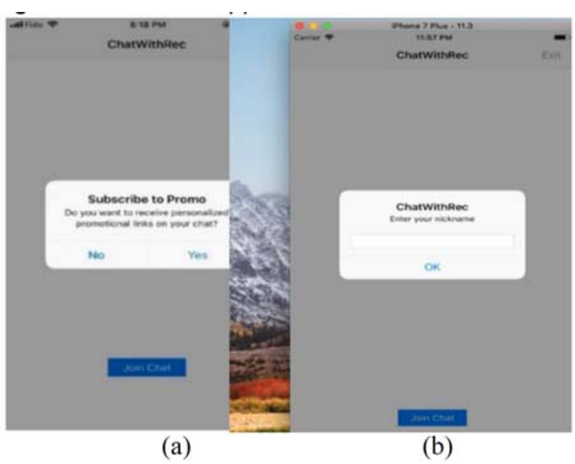

Fig.2: A screenshot from ChatWithRec app subscription agreement.

In this stage of our implementation, we developed our chat app, the ChatWithRec CRS, it is compatible with the iOS mobile operating system. The ChatWithRec functionalities include collecting the user's conversations, then sending them to our assistance server while displaying results from the server to the user. After installing the ChatWithRec App, a notification message displays to the user, requesting their agreement to receive relevant content. We considered extracting four diverse numbers of topics to define which number of acquired topics can give us a clear understanding of users' online conversation sessions. We evaluated our results by applying several statistical metrics such as coherence, perplexity, and F-score in our evaluation. Fig.2 shows a screenshot of the user's agreement and the app interface.

A. Experimental Setup and Results

We utilized a set of conversations dialogue from both the Facebook platform and Wikimedia databases (two sets of data, 100 and 1000 articles collection) to test the conversational analysis segment by operating the LDA topic model. Our data was a collection of user-generated dialogue about different topics such as travelling, ordering food, and shopping. We applied the text pre-processing part, text cleaning, then we tested our pure conversation model through different scenarios to check if our application ran properly and evaluated our system quality and usability. As a result of our initial ChatWithRec version, the user receives ads that were related to his conversation in real-time manner. Thus, we considered displaying the recommended ads as a banner ad, not an interstitial ad, to make sure that the conversation flow does not disturb or annoy users. From our experiment, we observed that extracting fewer number of topics lead to a high coherence score that helps to define the main topic in less time, which is useful in our developing system. Also, there is no need to check the user's preferences or even item ratings before pushing the advertisement's notification to an actively interested user, which solves the cold-start problem. However, during the long time of conversation sessions, more than 10 minutes, we received several advertisement recommendations; some of them were related to the user's interests, and other were random ads. That happened because we cannot control the AdMob's manufacturing action side after we send the extracted topic from our server side. To overcome this limitation, we decided to modify our ChatWithRec CRS by considering another approach to recommend related advertisements to users.

V.CHATWITHREC VERSION TWO

We considered two approaches as output results: that are; the AdMob database that shows random advertisement sometimes as a result in some test scenarios, and the modified database that includes a specific number of linked advert websites as a recommendation. Based on the AdMob database issues, we decided to reduce our targeted areas and filter our recommendations by considering just some fields which are travels, flights, hotels, and foods. In our future work, we will increase our fields to have more subjects, like shopping and entertainment.

Accordingly, instead of relying on AdMob facility as a core output-related action, we selected some popular and widely used websites (3 for each included area), and we are going to suggest them all to the interested users to choose which he prefers. By clicking on the link, the user can leave the conversation window and come back later. For example, included travelling websites regarding the travel topic are, Expedia, Kayak, and Trivago. We should mention that users have full control over what type of advertisements they are interested in receiving by selecting them from the app setting.

A. Experimental Setup and Results

We followed the same processes as in the initial ChatWithRec CRS version such as text cleaning. We also explored other NLP algorithms, such as TF-IDF and integrated the unsupervised algorithm LDA to detect topics that related to the users' conversation sessions. Adding more algorithms lead to better results and helped to understand the conversation more as well as obtain better results that improve our ChatWithRec CRS's accuracy and usability. In addition, we calculated several statistical metrics, like in section-4. As a result, we acquired a higher F-score, perplexity, and coherence score with fewer extracted topic, which enhances the ChatWithRec CRS performance and extracted topic quality, Fig.3 and 4 shows both coherence and perplexity results with different extracted topics. In addition, we manually extracted keywords that reflected the conversation topic to make sure the ChatWithRec CRS could understand the conversation and extract the correct topic from it.

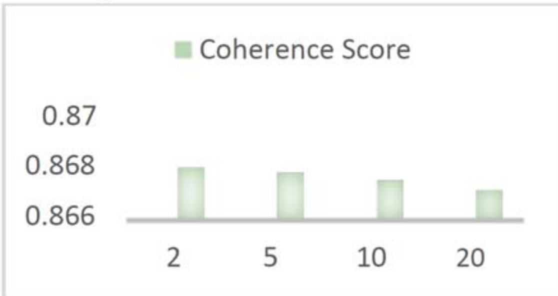

Fig.3: The Coherence score results.

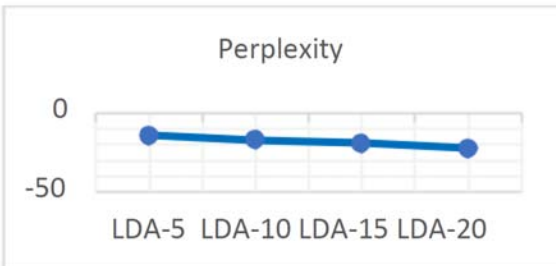

Fig.4: The perplexity score results.

VI.CONCLUSIONS AND FUTURE WORK

Researchers have demonstrated the importance of social recommender systems in the e-Commerce platform. They developed numerous methods to help users receive recommendations that are relevant to them socially and personally in the appropriate time. However, many of the existing approaches fall short, and users are often unsatisfied with the initial recommendations, because some of those models will not elicit the user's preferences at the beginning of the interaction or understand user's actual needs; this is known as the cold-start problem. We proposed a real-time conversational recommendation system called ChatWithRec that aims to solve the cold-start problem, enhance the user's social interaction, and improve the accuracy of users' advertisement recommendations. The cold-start problem refers to the difficulty in bootstrapping the recommendation systems for new users or new items. The uniqueness of our system is the real understanding of the users' conversations by extracting and detecting topics dynamically that involve user's dialogue, and then matching them with suitable tasks, like advertisements. As a result of our testing, we acquired better conversation-related advertisements by applying the ChatWithRec version.2 that has customized database comparing to the AdMob. In our future work, we plan to explore further advertisement regions and as well as add more topic modelling algorithms such as Word2vec, RASA, and non-negative matrix factorization (NMF) to understand the conversational context better. Currently, we have begun to build some of the previous algorithms, but we have not yet tested them thoroughly. Furthermore, we are planning to add both of the user's previous chat history and the user's preference model that we are going to create by collecting information from the user's profile.

REFERENCE

[1] Jannach. D, Zanker. M, Felfernig. A, Friedrich. G. "Recommender systems: an introduction," Cambridge University Press, New York (2011).

[2] Yera Toledo. R. and L. Martinez. "Fuzzy tools in recommender systems: A Survey". Volume 10, Issue 1, January 2017, Pages 776 - 803 (2017).

[3] Agarwal. V, and Bharadwaj. K, "A collaborative filtering framework for friends recommendation in social networks based on interaction intensity and adaptive user similarity," Social Network Analysis and Mining 3(3): 359-379. Bharadwaj (2013).

[4] Burke. R, Kristian J. Hammond, and Benjamin C. Young. "The findme approach to assisted browsing," IEEE Expert Volume:12, Issue: 4, Jul/Aug 1997,: 32- 40 (1997).

[5] Burke, R. "Hybrid recommender systems: survey and experiments. User Modeling and User-Adapted Interaction". 12(4): 331.370 (2002).

[6] Akermi, I., M. Boughanem and R. Faiz. Just-in-time recommendation approach within a mobile context. (2016).De Maio, C, G. Fenza, M. Gallo, V. Loia and M. "Social media marketing through time-aware collaborative filtering". Concurrency and Computation: Practice and Experience 30(1): e4098. Parente (2018).

[7] Nguyen, T. N. "Conversational group recommender systems". Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization. Bratislava, Slovakia, ACM: 331-334 (2017).

[8] Farzindar, A. and D. Inkpen. "Natural language processing for social media". Synthesis Lectures on Human Language Technologies 8(2): 1-166 (2015).

[9] Radim, R. Sojka, P. "Software Framework for Topic Modelling with Large Corpora". (2010).

[10] McCallum, A. K. "MALLET: A Machine Learning for Language Toolkit." http://mallet.cs.umass.edu. (2002).

[11] Ramage, D. "Topic modeling for the social sciences". J.

[12] Evan Rosen, Christopher D. Manning and Daniel A. McFarland. Stanford University Stanford, CA 94305, Workshop on Applications for Topic Models:Text and Beyond. NIPS (2009).

[13] Deerwester, S. "Indexing by latent semantic analysis". C., S. T., Landauer, T. K., Furnas, G. W., & Harshman, A., Journal of the American Society of Information Science. 41(6), 391–407 (1990).

[14] Hofman, T. "Unsupervised learning by probabilistic latent semantic analysis". Manufactured in The Netherlands., Machine Learning, 42, 177–196, 2001. 42, 2001 177–196 (2001).

[15] Keerthana S. "Recommended search of documents from conversation with relevant keywords using text similarity". Journal of Network Communications and Emerging Technologies (JNCET). www.jncet.org. Volume 7, Issue 2, February (2017).

[16] Zihuan Wang, K. H., Youngsam Kim. "A news-topic recommender system based on keywords extraction". Multimedia Tools and Applications 77, Number 4, : 4339 (2018).

[17] Bridge, D. and J. P. Kelly. "Ways of Computing Diverse Collaborative Recommendations", Berlin, Heidelberg, Springer Berlin Heidelberg (2006).

[18] Rania, Albalawi, Tet Hin Yeap, and Morad Benyoucef, "Toward a real-time social recommendation system," MEDES'19, November 12-14, Limassol, Cyprus, (2019). "In Press".

★★★