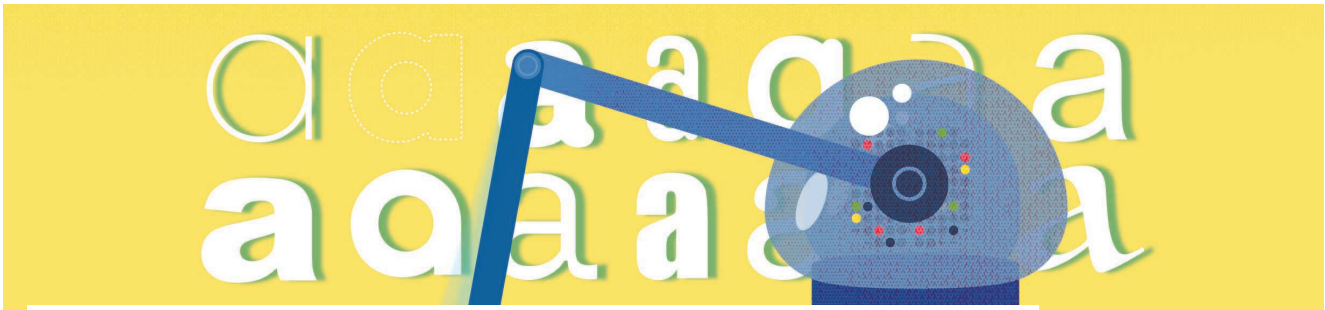

Deep Conversational Recommender Systems: Challenges and Opportunities

Dai Hoang Tran and Quan Z. Sheng, Macquarie University
Wei Emma Zhang, The University of Adelaide
Salma Abdalla Hamad, Macquarie University
Nguyen Lu Dang Khoa, Data61, CSIRO
Nguyen H. Tran, The University of Sydney

Unlike traditional recommender systems, the conversational recommender system (CRS) models a user's preferences through interactive dialogue conversations. Recently, deep learning approaches have been applied to CRSs, producing fruitful results. We discuss the development of deep CRSs and future research directions.

In recent years, natural language processing (NLP) techniques have advanced by leaps and bounds, and we are witnessing the booming of conversational user interfaces via virtual agents from big companies such as Microsoft Cortana, Amazon Alexa, Apple Siri, and Google Assistant. These agents can perform multiple tasks via voice or text commands. Even though their capabilities are still primitive, the array of actions they can carry out is impressive.

In another area, recommender systems emerged as a separate research field in the 1990s and are now acting as the core functionality of some of the largest online services in the world such as YouTube, Netflix,

Digital Object Identifier 10.1109/MC.2020.3045426
Date of current version: 8 April 2022

and Amazon.[1] Initially, recommender system techniques mainly used content-based and collaborative filtering approaches.[2] However, the need for conversational systems that can provide good suggestions to users is essential to many online e-commerce services, thus establishing grounds for the development of conversational recommender systems (CRSs).[3] This can be seen as a natural extension of these conversational virtual agents. Researchers and industry practitioners are trying to make this new application a reality.

Over the years, we have seen several approaches to developing a CRS, from using contextual bandits to machine learning methods.[3,4] Nevertheless, we are seeing a recent trend in the field of CRS where deep learning approaches are being employed to provide end-to-end solutions for CRS, and these systems are considered as deep CRSs (DCRSs).[5] To better understand DCRSs, in this article, we provide an overview of what CRSs and DCRSs are, the challenges pertinent to the development of DCRSs, and the current state-of-the-art deep models for DCRSs as well as discuss future research directions.

All of the information is collected from papers in the top conferences over the past five years. We identified relevant papers by using different combinations of the keywords, including "conversational," "recommender system," and "goal-oriented," on the DBLP[1] computer science bibliography website (https://dblp.org). We studied these papers in detail to identify the main challenges, their model structure, and potential future research directions. To the best of our knowledge, our work is the first one that tries to summarize and understand the DCRS in detail.

DCRSs

Background

In daily life activities, humans' most natural interactive actions are communicating with others via conversations. We converse about our work, gossip about other people's relationships, and recommend things we like to our friends. When it comes to recommendations—from seeking advice from friends about good movies to watch to looking for suggestions from travel agents regarding enjoyable holiday destinations—we can express our preferences and quickly get ideas from others through just a few exchanges in simple conversations.

Since multiple terms are often used interchangeably within the domain of conversational systems, we need to clarify their subtle differences. A conversational system is an intelligent system that can understand language and conduct verbal or written dialogue conversation between an information seeker (that is, a human being) and an information provider. *Virtual agent* and *virtual assistant* are other terms for an information provider, which is the intelligent component of a conversational system that interacts with the information seeker. In this article, we use the term *users* to denote information seekers and *agents* to denote information providers.

From the perspective of online businesses, due to the natural and personal characteristics of direct communication via conversations, a large amount of modern services provide call or chat systems to deal with customer support. However, human resources are limited and costly; thus, the need for intelligent agents who are able to converse with users and give satisfactory recommendations is essential to them. On the other hand, from the user's perspective, the abilities to freely express one's preferences and retrieve tailored suggestions from the agents give the users a strong sense of satisfaction and confidence in the choices they make. These conversational systems that provide tailored suggestions to the users and can carry out intelligent conversations are called *CRSs*. An example of a CRS dialogue session is illustrated in Figure 1, which depicts a scenario where an agent is able to learn and provide a recommendation for matching shoes to a satisfied user.

In a typical CRS, there exist three components: a user intention understanding (UIU) module, a recommendation (REC) module, and a switching mechanism (SWM). The user's inputs and agent's responses are handled by the UIU module. Since the most common form of these is text, most of the models have trained a UIU module that can process and generate natural language data. However, we are seeing an increasing number of works on UIU that can handle multimodal inputs, such as both text and image inputs,[6,7] which is discussed in later sections.

The UIU module produces dialogue states that will be consumed and decided by the SWM to either keep asking the user more questions for clarification or pass the dialogue state to the REC module to generate recommendations. Notice that, in some cases, the SWM receives signals from both the UIU and REC modules to make a decision.[8] Optionally, certain research also proposed an improvement mechanism to improve the system's recommendations based on the user's feedback.[9]

To achieve these complex goals, the UIU and REC modules are usually trained by traditional machine learning approaches.[3] In recent years, the

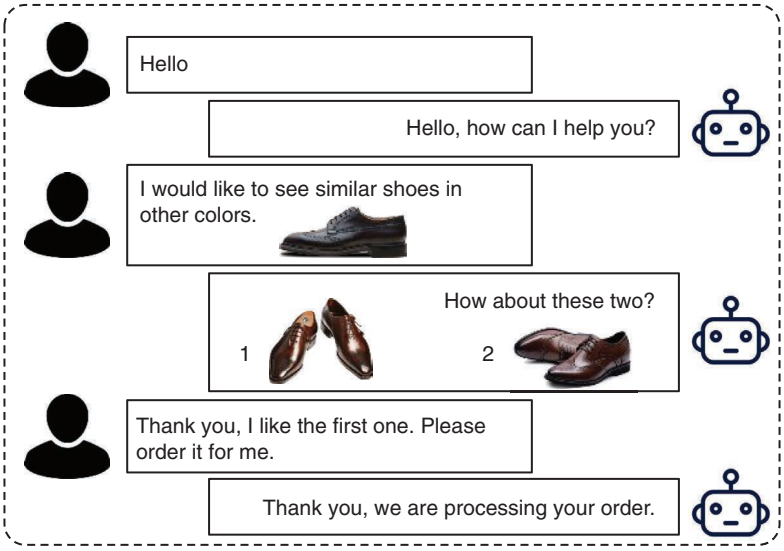

FIGURE 1. An example conversation between a user and an agent in a CRS in the e-commerce domain.

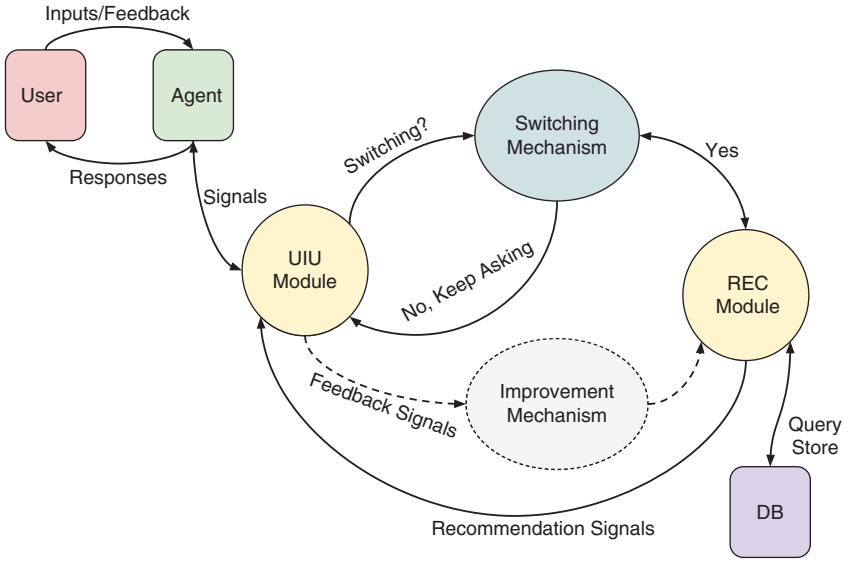

FIGURE 2. The main components of a DCRS. The dashed line shows optional components. DB: database.

tremendous success of deep learning methods in a variety of tasks has encouraged the development of DCRSs, in which the UIU's and REC's training methods are replaced by deep learning approaches, leading to performance improvement in these systems. Thus, we define a *DCRS* as a CRS that has at least one of its modules using a deep learning approach to develop the system. Figure 2 shows the main components of a DCRS.

Definitions and formalization

A DCRS is primarily a goal-oriented conversational system that also incorporates the elements of chit-chat and question-answering conversational systems.[5] Figure 1 illustrates such an example. A DCRS is a goal-oriented conversational system because its main purpose is to provide recommendations to users. All interactions between the agent and user are steps for the system to understand the user's preferences and achieve its goal of proving a recommendation list. The elements of chit-chat and question answering are part of the processes of extracting the user's preferences. Hence, the DCRS is not an open-domain conversional system.[10]

This natural mode of interaction between users and agents presents considerable challenges when designing such a system. Concretely, a DCRS takes user $u \in U$ inputs about certain facets of possible items via an utterance; then the agent responds with either a new question to learn more about the current user's preferences or offers recommended items. At each turn t, the user u provides utterance s_{ut}, and the agent gives a responded utterance at the next turn $s_{a(t+1)}$. A dialogue session is a collection of these utterances $S_t = \{s_{u1}, s_{a2}, \ldots, s_{ut}, s_{a(t+1)}\}$ until the user stops responding or terminates the session.

In each session, there can be multiple rounds of conversation; each round ends when the agent provides recommended items. If the user initiates a new request after that, a new round starts. A multiround DCRS is a system where, after each round, the agent can improve its recommendation outputs based on the learned user preferences from the previous round.[9] Some DCRSs accept the user's feedback at the end of a round. A user's feedback is usually a positive or negative confirmation utterance after the agent's recommendation output.

At each turn t, the utterance set S_t is fed into the UIU module to produce a dialogue state embedding with d-dimension $z_t \in R^d$, usually via a recurrent neural network (RNN)-based encoder. The SWM module uses the z_t embedding to output an action score with n-dimension $p \in R^n$, which depends on the policy of the system; the p score will be classified into an action that is either a 1) more-question or 2) give-recommendation action.

For the first action of more questions, the UIU module generates the agent's responded question to ask the user for more specific information, usually an attribute-related question. For the second action of making recommendations, the REC module then queries its database to search for appropriate items based on dialogue state z_t and returns the recommendation signal r_t to the UIU module. The UIU then generates the agent's responded utterance containing the recommendation, and this marks the end of one round of conversation.

The work of Christakopoulou et al.[11] follows this flow. In certain cases, to provide more accurate responses, at turn t, both the UIU and REC modules receive the S_t signal and produce z_t and r_t signals, respectively. The signals are then combined and used by the SWM to decide the policy action.[12] The cycle repeats.

DCRS RESEARCH PROBLEMS AND CHALLENGES

Developing a DCRS is challenging due to its inherit nature of having two different tasks in one system, which are 1) understanding user intention and 2) giving relevant recommendations. The whole system needs to understand user intention; preferences; and, optionally, feedback through natural language, which is a challenging problem by itself. Apart from that, the DCRS also needs to know how to query its database to find relevant and personalized items based on the user's inputs. Table 1 summarizes key characteristics of prominent literature works. In this section, we outline the current research problems and challenges of DCRSs.

Understanding the user's intention

During a dialogue session in a DCRS, a user keeps expressing his or her intention via natural language; this can be as simple as initiating a conversation with a greeting, explaining his or her preferences for items, or giving feedback to the agent. It is critical for the agent to know what the current user's intention is for the system to act. However, text understanding and comprehension involve complicated ongoing research in the field of deep learning. Due to this issue, understanding the user's intention is one of the key challenges in developing a DCRS.

We observe that, currently, there are two groups of approaches being used to understand a user's intention. The first group, which is typically used by DCRSs that are trained on synthetic data sets, is attribute extraction from the user's utterances. Assume that, after each question asked by an agent, the user utterance will contain $V = \{v_1, v_2, \ldots, v_n\}$ attributes from a fixed set of attributes A. Then, the set V will be fed to the UIU module. This approach simplifies the encoding aspect of the user's utterance, as can be seen in work by Lei et al.[9] and Christakopoulou et al.[11] The second group tries to encode the utterances of the whole dialogue via an RNN-based neural network to extract a dialogue state $z \in R^d$ to feed to the UIU module, and this approach is better at generating fluent responses by an agent.[5,6]

Providing personalized recommendations

People might assume that a DCRS always provides personalized recommendations, but that is not always the case. The first priority of a DCRS is to provide relevant recommendations that match the user's preferences. However, certain DCRSs are designed to act more like search-and-filter engines that only consider the user's preferences in the current dialogue session. Thus, two users with the same preferences might receive the same recommendations. To overcome this issue of lacking personalized recommendations in the system, the DCRS needs to know the user's features (age, gender, and so on); remember the user's feedback as well as his or her past preferences; and, finally, incorporate those signals with the dialogue state to generate personalized recommendations. Hence, two users with the same preferences still receive different recommendations due to their varying histories. It is important to have a personalized DCRS for better user engagement.

In the current literature, we have seen a mix of these solutions. Some

works solely act as a search or filter engine without taking the user's attributes into account,[7,12] while others do provide personalized recommendations using the user's attributes in their systems.[8,9]

Developing a suitable SWM

It is imperative for a DCRS to know when to ask questions to learn more about a user's preferences or give responses with recommendation results because, if this mechanism behaves wrongly, it will lead to a lengthy conversation or inappropriate recommendation turn, which results in the user's dissatisfaction. Not only that, the SWM is also the connection between major modules in a DCRS, as illustrated in Figure 2. Hence, this is a key challenge in developing a high-performance DCRS, that is, to provide an accurate SWM.

So far, we have observed three groups of approaches for the SWM, namely, rule based, pointer softmax probability, and reinforcement policy score. For rule-based SWM, there is no specific methodology. A rule can be just a simple constraint, such as, for each turn, always providing a recommendation,[11] or a designed choice, for example, the confident score of the top k recommended items over a threshold.[12]

Pointer softmax probability originates from the work by Gűlçehre et al.[17] where the SWM uses the gated recurrent unit (GRU) cell to decode the hidden state of the current dialogue context and decides whether to generate response tokens with or without recommended item via a softmax probability score. This leads to natural and fluent generative responses from the agent, as seen in the works of Li et al.[5] and Liao et al.[13]

The third approach is the reinforcement policy score, where, at each turn in

TABLE 1. Summarization of prominent DCRS in the literature with their characteristics.

DCRS	Data set	Question space	FR	SWM	P	Multimodal	Multiround	Neural networks
VisualDialog,[7] KDD 2019	Synthetic	Text	No	Rule based	No	Yes	Yes	RNN and CNN based
KMD,[6] ACM-MM 2018	MMD[15]	Text	Yes	Policy network	No	Yes	Yes	HRED, EI-Tree, and reinforcement
ReDIAL-DCR,[5] NeurIPS 2018	ReDIAL	Text	Yes	Pointer softmax	Yes	No	Yes	HRED and autoencoder
DCR,[13] CoRR 2019	MultiWOZ[16]	Text	Yes	Pointer softmax	No	No	No	HRED and GCN
SAUR,[12] CIKM 2018	Synthetic	Attributes	No	Rule based	No	No	Yes	Memory network
CRM,[8] SIGIR 2018	Synthetic	Attributes	No	Policy network	Yes	No	No	LSTM and reinforcement
EAR,[9] WSDM 2020	Synthetic	Attributes	No	Policy network	Yes	No	Yes	Reinforcement
Q&R,[11] KDD 2018	YouTube	Attributes	No	Rule based	Yes	No	No	RNN based
CEI,[14] AI*IA 2017	Synthetic	Text	No	Policy network	No	No	Yes	RNN based and reinforcement

ACM-MM: ACM multimedia; AI*IA: International Conference of the Italian Association for Artificial Intelligence; CEI: Converse-Et-Impera; CIKM: International Conference on Information and Knowledge Management; CNN: convolutional neural network; CoRR: Computing Research Repository (arxiv.org); CRM: Conversational recommender model; DCR: Deep conversational recommender; EAR: estimation-action-reflection; FR: fluent response of the agent; GCN: graph convolutional neural network; HRED: hierarchical recurrent encoder-decoder; KDD: knowledge discovery and data mining; KMD: Knowledge-aware multimodal dialogue systems; LSTM: long short-term memory; MMD: multimodal domain-aware conversation systems; MultiWOZ: multi-domain wizard-of-Oz dataset; NeurIPS: The Conference and Workshop on Neural Information Processing Systems; P: personalization; Q&R: Question & Recommendation; ReDIAL: recommendations through dialog; SAUR: System Ask, User Respond; SIGIR: Computing Machinery Special Interest Group in Information Retrieval; WSDM: International Conference on Web Search and Data Mining.

a dialogue session, the SWM generates an action vector based on embedding signals from the UIU and REC modules. This action vector then is fed into a reinforcement policy network, which outputs a softmax class score to reflect which action to take. Henceforth, the policy network is trained to maximize the action reward based on the labels of the data set. This SWM approach is chosen by Sun and Zhang,[8] Lei et al.,[9] and Greco et al.[14] who focus on generating mechanical responses.

Handling of multimodal inputs/outputs

The user's input to a DCRS is often text. However, modern conversational systems allow multiple input types, such as text, image, or audio files (for example, Facebook Messenger). In this regard, it is also natural for DCRSs to be able to process multimodal inputs aside from textual input, and the agents need to be able to provide multimodal outputs as well. This is a real challenge for DCRSs since it introduces another complexity into the UIU module, where the agent has to understand the semantic meaning of other nontextual input types. For instance, if a user expresses she wants to find dresses that resemble one from a clothing image, the agent needs to understand the various features of that dress to find similar ones to recommend. Different input types need specialized methods to extract their semantic meanings, which can also require additional training of the submodule from the whole model.

As such, currently, there are only a handful of research works that address such challenges; a few notable ones are by Liao et al.[6] and Yu et al.[7] The DCRS described by Liao et al.[6] can incorporate image semantic meaning through an exclusive and independent-tree (EI-tree) neural network as well as external domain knowledge. The work of Yu et al.,[7] on the other hand, provides a unique user response via item clicking; then, all of these signals, including the user's utterances and requested images, are passed to an augmented cascading bandit module to provide the agent's responses. We expect to see more research tackling multimodal data in the future.

Training multitask models

A DCRS contains multiple modules to handle different tasks, notably, the UIU and REC modules. Even though the main objective is to provide relevant recommended items via the agent's responses, it does include subgoals, such as the agent utterance generation and finding of the top k recommended items based on the user's utterances. Henceforth, how to incorporate the multiple-objective functions of different tasks for an end-to-end model training is one of the key challenges in developing a DCRS.

From our investigation, the majority of research works contain the element of optimizing the multitask loss function in the form of $L_{\{t1,\ldots,tn\}} = L_{t1} + \ldots + L_{tn}$. Usually, the multitask loss function $L_{\{t1,\ldots,tn\}}$ is placed at the SWM to optimize the agent's best action to be taken at each turn of the dialogue session, whereas the other loss functions L_t are optimized separately from their respective modules.[6,9,14]

Training on limited fluent dialogue data sets

The first and foremost challenge for any deep learning problem is having a large and accurate data set, and, in the field of DCRSs, this is a major issue. Through our study of recent works, DCRSs are lacking fluent dialogue data sets, which contain natural and fluent human-to-human conversation. However, only a handful of them are available, such as the ReDIAL[5] and MultiWOZ[16] data sets. As a deep learning model needs lots of training data, a synthetic DCRS data set is the preferable choice for several research works because it is simple to bootstrap and generate, as shown in Table 1. Due to this lack of natural fluent dialogue conversation data sets, it is unavoidable that solutions based on synthetic ones will get exposed to biased or nonfluent/mechanical responses. As such, having a big data set of fluent conversion is a key challenge when developing a DCRS.

This has allowed us to observe an interesting fact: the research works that utilize fluent dialogue data sets tend to focus on generating seamless responses in a multiround setting, where the agent's response can include zero or more recommended items, while the user can keep carrying out the conversation in a natural way.[5,13] On the other hand, the approaches that use synthetic data sets for validation tend to center on generating mechanical yet relevant questions to exploit the user's preferences. Thus, if the user's answer does not match with the training templates, the agent will not understand and may ask the same question again.[9,12] Therefore, having a big dialogue data set with natural conversation is important for developing a fluent and accurate DCRS. We are seeing several works trying to address this issue by using machine learning to generate more realistic dialogues, such as that by Suglia et al.[18]

DCRS DEEP LEARNING MODELS

In this section, we provide an in-depth look at the deep learning approaches for developing a DCRS. When we are categorizing the technical aspects

into buckets, we realize that it is not easy to divide them based on the whole architecture of the system since a DCRS contains multiple components, and each of them can have its own deep learning model. Thus, we make the categorization based on deep learning models applied for the three main components of a DCRS: the UIU module, the REC module, and SWM, as shown in Figure 3. Next, we elaborate on the details of how these deep learning models act in their respective modules.

UIU deep learning models

A required task of a DCRS is to understand a user's intention via his or her inputs. Due to its conversational nature, most of the UIU models are deep learning models for understanding textual natural language inputs. However, a few notable works also try to tackle multimodal data, such as both text and image inputs.[6,7] We outline the most popular deep learning approaches to handle the user's utterances in DCRS in the following.

RNN-based models. Due to the popularity of RNN-based models for dealing with natural languages, the majority of UIU models are RNN based in the DCRSs of our investigation. The most popular one is the hierarchical recurrent encoder–decoder model, as seen in recent works.[5,6,13] Other RNN variants are the memory network with attention weight[6,12] and GRU encoder.[14] The basic operation of these models is to encode the current dialogue session of S_t of turn t into a dialogue state vector z_t, which will be further processed by other modules of the DCRS.

Convolutional-neural-network-based models. A multimodal DCRS uses the convolutional neural network (CNN)-based deep learning model to extract image input features for encoding the dialogue state. The extracted image features are usually concatenated with utterance features to be processed further in the pipeline. Yu et al.[7] used a pretrained ResNet CNN model to extract image features, while the work of Liao et al.[6] had its CNN model extract the semantic meaning of an image, called *EI-tree*.

REC deep learning models

Many REC modules of different DCRSs do not use deep learning models. Instead, they opt for the matrix factorization approach, due to its capability to take advantage of all users and item attributes.[8,9] Nevertheless, we observe a few variants of deep learning models used for making recommendations in a DCRS.

Autoencoder-based models. An autoencoder has been used in the research of general recommender systems to overcome the cold-start problem, and a notable work using this model is Autorec.[19] Based on this approach, Li et al.[5] trained their REC module using a deep denoising autoencoder network to predict a user's ratings that have not been observed in the training set.

Graph CNN-based models. Graph CNN (GCN) deep learning models allow us to solve problems based on graph-structured data, such as a social network or recommendation item relationship. The principle of the GCN network is that it takes into account both nodes' attributes as well as the nodes' neighborhood attributes using a graph convolution operation, thus allowing the model to learn a better local representation of each node and achieving a state-of-the-art performance in graph-structured data tasks. Based on the GCN principle, the work of Liao et al.[13] builds a GCN recommendation model for a travel DCRS application that constructs the graph-structured data to connect hotels, restaurants, and other travel facilities to provide additional services to customers that go well together.

SWM deep learning models

The SWM plays an important role in keeping the DCRS behaving correctly at each user utterance input, as explained in the "Developing a Suitable SWM" section. Certain works applied simple rule-based methods,

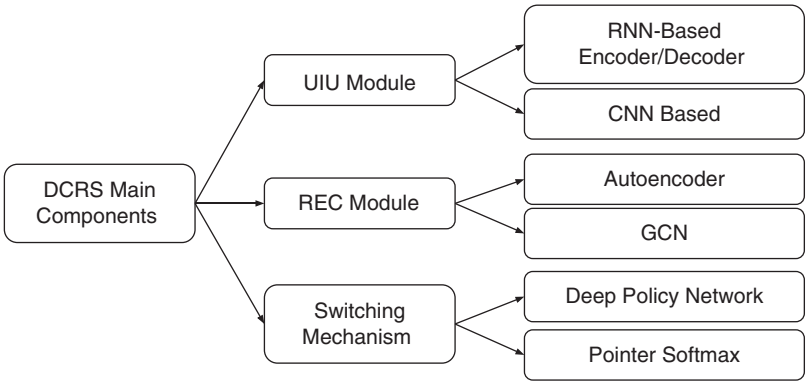

FIGURE 3. Deep learning approaches for each DCRS component. GCN: graph CNN.

such as providing a recommendation after each turn[11] or returning recommendations when the first top k items ranking probability reach a certain threshold.[12] However, researchers are employing more sophisticated approaches to train the SWM, and the deep learning models they often use are deep policy network and pointer softmax probability.

Deep policy network. A deep policy network is a straightforward usage of reinforcement learning to maximize the reward of an SWM action based on the current dialogue state. The action space is usually either more-question or give-recommendation action, as mentioned in the "Definitions and Formalization" section. For each dialogue turn during the training phase, a reward score is given to the SWM to make the correct action choice, and a negative reward is given otherwise. Both Christakopoulou et al.[8] and Lei et al.[9] used a two-layer feed-forward neural network to optimize the network parameters, while Liao et al.[6] used the BLEU score as the reward signal.

Pointer softmax probability. This technique is based on the work of Gűlçehre et al.[17] The principle of this method is that, during the inference process, if a certain rare condition is met, the neural network can point to other data points that it knows how to handle instead of processing the current rare condition. By utilizing this approach in the SWM, a DCRS can generate fluent agent responses with recommendations. As proven in the work of Li et al.,[5] at step m in the dialogue D, the UIU module keeps generating sentence tokens while asking the SWM if it should point to a movie name or not, based on the current dialogue state. If the SWM decides to point to a movie name, the UIU will generate recommended movie names and add to the generative response of the agent. The work of Liao et al.[13] also uses a similar principle.

OPEN RESEARCH DIRECTIONS

The prosperity of deep learning in advancing NLP and recommendation tasks has brought more development to the field of DCRSs. Especially in the past five years, the number of works on DCRSs has increased tremendously. Some challenges still remain unsolved and need more investigation. We have identified further research directions and discuss these in the following sections.

Synthetic fluent dialogue data sets

As one of the current challenges in developing a DCRS, we need better data sets for training the DCRS deep learning models. However, given the high cost and complication of making a real human-to-human dialogue data set, it is more feasible to create high-quality synthetic dialogue data sets. With the advancement of deep learning in handling natural language tasks, such as language generation, text comprehension, and question answering,[10] we strongly believe that researchers can use deep learning techniques to make better synthetic training data sets for DCRS development. By using advanced NLP techniques to generate a comprehensive and human-like DCRS dialogue data set, we can train a DCRS model in certain domains with high precision when extracting user preferences and improve its fluent responses. Additionally, this also eliminates the costly effort of manually constructing training data sets.

Incorporating external domain knowledge

A particular work of Liao et al.[6] proposes a unique approach by using external domain knowledge to improve their DCRS. The authors embed the external knowledge domain into a memory network and then generate the agent's responses based on the extracted domain knowledge and current dialogue context. Given the lack of adequate training data sets and difficulty of incorporating personalization into the DCRS, using external domain knowledge from free knowledge bases such as DBPedia (https://www.dbpedia.org) or NELL (rtw.ml.cmu.edu) can definitely bring benefits to the DCRS area, especially when addressing the challenge of the dearth of training data sets. Henceforth, one main direction is to develop methods for better integrating external domain knowledge into the DCRS architecture to increase system performance.

Improvement from the user's feedback

In the current literature regarding DCRSs, we find that not many works consider the user's feedback to improve the system performance. The work of Lei et al.[9] addresses this concern via a reflection phase, where negative feedback is collected and stored for future retraining of the system. It is a simple and effective technique. Another approach is learning the user's feedback via intent taxonomy.[20] Therefore, future researchers can develop online methods to improve DCRSs directly from user feedback for a better system performance. This method can effectively make the DCRS

ABOUT THE AUTHORS

DAI HOANG TRAN is an honorary research fellow at Macquarie University, Sydney, 2109, Australia. His research focuses on machine learning and recommender systems. Tran received a Ph.D. in computer science from Macquarie University, Australia. Contact him at dai-hoang.tran@hdr.mq.edu.au.

QUAN Z. SHENG is a full professor and head of the Department of Computing at Macquarie University, Sydney, 2109, Australia. His research interests include the Internet of Things and service-oriented, distributed, Internet, and pervasive computing. Sheng received a Ph.D. in computer science from the University of New South Wales. Contact him at michael.sheng@mq.edu.au.

WEI EMMA ZHANG is a lecturer at the School of Computer Science, University of Adelaide, Adelaide, SA 5005, Australia, and honorary lecturer at the Department of Computing, Macquarie University. Her research interests include text mining, natural language processing, information retrieval and Internet of Things applications. Zhang received a Ph.D. in 2017 from the School of Computer Science, University of Adelaide. She is the vice chair of the Executive Committee of the IEEE Technical Community on Services Computing. Contact her at wei.e.zhang@adelaide.edu.au.

SALMA ABDALLA HAMAD is an honorary research fellow at Macquarie University, Sydney, 2109, Australia. Her research focuses on Internet of Things security. Hamad received a Ph.D. in computer science from Macquarie University, Australia. Contact her at salma-abdalla-ibrahim-mah.h@hdr.mq.edu.au.

NGUYEN LU DANG KHOA is a research team leader and a senior research scientist in the Analytics and Decision Sciences Program at Data61 (The Commonwealth Scientific and Industrial Research Organization). His research interests include machine learning and data mining, with a focus on anomaly detection, tensor decomposition, online learning, spectral graph embedding, clustering, and predictive modeling. Khoa received a Ph.D. in computer science from the University of Sydney. Contact him at khoa.nguyen@data61.csiro.au.

NGUYEN H. TRAN is a senior lecturer with the School of Computer Science, University of Sydney, Sydney, 2006, Australia. His research interests include distributed computing, machine learning, and networking. Tran received a Ph.D. in electrical and computer engineering from Kyung Hee University. He is a Senior Member of IEEE. Contact him at nguyen.tran@sydney.edu.au.

better at understanding the user's intention and providing personalized recommendations, thus solving two of the aforementioned challenges.

Unified evaluation metrics

A noticeable observation from our study is the discrepancy of the measurement metrics when evaluating a DCRS. This occurs due to the inherent multitasking nature of the DCRS. To the best of our knowledge, there is no single metric to evaluate the DCRS as a whole. Therefore, researchers rely on both goal-oriented dialogue and recommendation measurement metrics to evaluate their works. Some metrics are used more than the others, such as the BLUE score to evaluate the fluency of an agent's generative responses,[6,14] while several others use the success-rate metric to measure their recommendation's effectiveness.[8,9] We hope to see more unified measurement metrics for DCRS evaluation, and we believe that having these can help us address several aforementioned challenges in developing a DCRS, such as better understanding the user's intention and providing personalized recommendations.

CRSs are a practical application domain for modern online services, and DCRSs are the next evolution of them. In recent years, we have seen a rising effort of research works that aim to improve this new exciting field. By taking a detailed look at the current state of the field from different angles; summarizing the characteristics, problems, and challenges of DCRSs; and proposing future research directions, we hope that our study provides useful information and elicits excitement for more researchers to contribute to this vibrant research area.

ACKNOWLEDGMENT

This work was supported in part by the Australian Research Council Discovery Grant DP200102298.

REFERENCES

1. S. Wang, L. Hu, Y. Wang, L. Cao, Q. Z. Sheng, and M. A. Orgun, "Sequential recommender systems: Challenges, progress and prospects," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, 2019, pp. 6332–6338, doi: 10.24963/ijcai.2019/883.
2. S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Comput. Surv.*, vol. 52, no. 1, pp. 1–38, 2019, doi: 10.1145/3285029.
3. K. Christakopoulou, F. Radlinski, and K. Hofmann, "Towards conversational recommender systems," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD'16)*, 2016, pp. 815–824, doi: 10.1145/2939672.2939746.
4. S. Lee, R. J. Moore, G.-J. Ren, R. Arar, and S. Jiang, "Making personalized recommendation through conversation: Architecture design and recommendation methods," in *Proc. Workshops 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 727–730.
5. R. Li, S. E. Kahou, H. Schulz, V. Michalski, L. Charlin, and C. Pal, "Towards deep conversational recommendations," in *Proc. 32nd Conf. Neural Inf. Process. Syst. (NeurIPS)*, 2018, pp. 9748–9758.
6. L. Liao, Y. Ma, X. He, R. Hong, and T.-S. Chua, "Knowledge-aware multimodal dialogue systems," in *Proc. 26th ACM Int. Conf. Multimedia (MM'18)*, 2018, pp. 801–809, doi: 10.1145/3240508.3240605.
7. T. Yu, Y. Shen, and H. Jin, "A visual dialog augmented interactive recommender system," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2019, pp. 157–165, doi: 10.1145/3292500.3330991.
8. Y. Sun and Y. Zhang, "Conversational recommender system," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval (SIGIR)*, 2018, pp. 235–244, doi: 10.1145/3209978.3210002.
9. W. Lei *et al.*, "Estimation-action-reflection: Towards deep interaction between conversational and recommender systems," in *Proc. 13th Int. Conf. Web Search Data Mining (WSDM)*, 2020, pp. 304–312, doi: 10.1145/3336191.3371769.
10. M. Hu, F. Wei, Y. Peng, Z. Huang, N. Yang, and D. Li, "Read + verify: Machine reading comprehension with unanswerable questions," in *Proc. 33rd AAAI Conf. Artif. Intell. 31st Innovative Appl. Artif. Intell. Conf. 9th AAAI Symp. Educ. Adv. Artif. Intell. (AAAI)*, 2019, pp. 6529–6537, doi: 10.1609/aaai.v33i01.33016529.
11. K. Christakopoulou, A. Beutel, R. Li, S. Jain, and E. H. Chi, "Q&R: A two-stage approach toward interactive recommendation," in *Proc. 24th ACM SIGKDD Int. Conf. (KDD)*, 2018, pp. 139–148, doi: 10.1145/3219819.3219894.
12. Y. Zhang, X. Chen, Q. Ai, L. Yang, and W. B Croft, "Towards conversational search and recommendation: System ask, user respond," in *Proc. 27th ACM Int. Conf. Inf. Knowl. Manage. (CIKM'18)*, 2018, pp. 177–186, doi: 10.1145/3269206.3271776.
13. L. Liao, R. Takanobu, Y. Ma, X. Yang, M. Huang, and T.-S. Chua, "Deep conversational recommender in travel," 2019, *arXiv:1907.00710*.
14. C. Greco, A. Suglia, P. Basile, and G. Semeraro, "Converse-Et-Impera: Exploiting deep learning and hierarchical reinforcement learning for conversational recommender systems," in *Proc. 2017 Adv. Artif. Intell. (AI*IA)*, pp. 372–386, doi: 10.1007/978-3-319-70169-1_28.
15. A. Saha, M. Khapra, and K. Sankaranarayanan, "Towards building large scale multimodal domain-aware conversation systems," in *Proc. AAAI Conf. Artif. Intell.*, May/Jun. 2018, pp. 696–704.
16. P. Budzianowski *et al.*, "MultiWOZ —A large-scale multi-domain wizard-of-Oz dataset for task-oriented dialogue modelling," 2018, *arXiv:1810.00278*.
17. Ç. Gülçehre, S. Ahn, R. Nallapati, B. Zhou, and Y. Bengio, "Pointing the unknown words," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics (ACL)*, 2016, pp. 140–149, doi: 10.18653/v1/P16-1014.
18. A. Suglia, C. Greco, P. Basile, G. Semeraro, and A. Caputo, "An automatic procedure for generating datasets for conversational recommender systems," in *Proc. CLEF*, in *Working Notes of CLEF 2017—Conference and Labs of the Evaluation Forum* (Workshop Proc. Ser. 1866), Dublin, Ireland, Sept. 11–14, L. Cappellato, N. Ferro, L. Goeuriot, and Thomas Mandl, Eds., 2017. [Online]. Available: http://ceur-ws.org/Vol-1866/paper_197.pdf
19. S. Sedhain, A. K. Menon, S. Sanner, and L. Xie, "Autorec: Autoencoders meet collaborative filtering," in *Proc. 24th Int. Conf. World Wide Web (WWW)*, 2015, pp. 111–112, doi: 10.1145/2740908.2742726.
20. W. Cai and L. Chen, "Towards a taxonomy of user feedback intents for conversational recommendations," in *Proc. ACM RecSys Late Breaking Results*, 2019, pp. 51–55.