Predicting User Intents and Satisfaction with Dialogue-based Conversational Recommendations

Wanling Cai
Hong Kong Baptist University
Hong Kong, China
cswlcai@comp.hkbu.edu.hk

Li Chen
Hong Kong Baptist University
Hong Kong, China
lichen@comp.hkbu.edu.hk

ABSTRACT

To develop a multi-turn dialogue-based conversational recommender system (DCRS), it is important to predict users' intents behind their utterances and their satisfaction with the recommendation, so as to allow the system to incrementally refine user preference model and adjust its dialogue strategy. However, little work has investigated these issues so far. In this paper, we first contribute with two hierarchical taxonomies for classifying user intents and recommender actions respectively based on grounded theory. We then define various categories of feature considering *content*, *discourse*, *sentiment*, and *context* to predict users' intents and satisfaction by comparing different machine learning methods. The experimental results for user intent prediction task show that some models (such as XGBoost and SVM) can perform well in predicting user intents, and incorporating *context features* into the prediction model can significantly boost the performance. Our empirical study also demonstrates that leveraging *dialogue behavior features* (i.e., including both user intents and recommender actions) can achieve good results in predicting user satisfaction.

CCS CONCEPTS

• **Human-centered computing** → *User models*; *User studies*; • **Information systems** → *Recommender systems.*

KEYWORDS

Dialogue-based conversational recommender systems; intent taxonomy; user intent prediction; user satisfaction prediction

ACM Reference Format:
Wanling Cai and Li Chen. 2020. Predicting User Intents and Satisfaction with Dialogue-based Conversational Recommendations. In *Proceedings of the 28th ACM Conference on User Modeling, Adaptation and Personalization (UMAP '20), July 14–17, 2020, Genoa, Italy.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3340631.3394856

1 INTRODUCTION

Over recent years, commercial conversational assistants, such as Google Assistant, Apple Siri, Amazon Alexa, and Microsoft Cortana, have emerged as powerful AI applications [18] that can converse with users for entertainment (non-task-oriented [7]) or accomplishing certain tasks (task-oriented [49]). Moreover, there is an increasing trend to integrate recommendation approaches into the task-oriented dialogue system to assist users in finding interesting items, so called *Dialogue-based Conversational Recommender Systems* (DCRSs) [11, 45]. However, most DCRSs can only handle one-shot recommendations [10, 45]. In other words, the system normally ends the conversation after presenting one or multiple recommendations to the user, no matter whether the user is satisfied with its recommendation or not, while in the real-world scenario, users may often interact with the recommender in multi-turn until they find the desired item(s) [47].

Seeker:	...
Recommender:	Another good one is Spaceballs.
Seeker:	**I did see that one, but I didn't really like it. I do love 80s movies though.**
Recommender:	Ok Well how about Planes, Trains and Automobiles.
Seeker:	**I may have seen that a long time ago but I can't remember. who is in that again?**
Recommender:	Steve Martin and John Candy. It is very funny.
Seeker:	**I love them both. I will try that one. Thanks so much!**

Figure 1: One human-human dialogue example from ReDial, a movie recommendation dialogue dataset [26], where the recommended movie is underlined and the seeker's feedback is highlighted in bold.

Indeed, one main challenging issue in multi-turn DCRSs is how to predict the user's intents behind her/his utterances [38], so that the system could incrementally refine its understanding of the user's preference and hence improve its recommendation in the subsequent conversation [36]. In addition, it is also critical to predict the user's satisfaction with the recommendation (that refers to whether the system can successfully fulfill the user's goal [24]), which may inform the system to adjust its dialogue strategy accordingly.

Unfortunately, so far little attention has been paid to predicting user intents and satisfaction in a DCRS, though there are some related studies in general domains [28] or question-answering (Q&A) systems [37]. In our view, users can behave differently when they interact with different types of system. For instance, in a Q&A system, users may often ask or rephrase their questions for expressing their information needs [37], but in DCRSs, users may be inclined to give feedback on the recommended item in order to receive better recommendations [49]. As shown in the dialogue example (see Figure 1), when the seeker did not like the recommended movie (e.g., "Spaceballs"), s/he gave feedback *"I didn't really like it. I do love 80s movies though."* Based on the seeker's feedback, the recommender may hence be able to resort to different strategies (e.g., giving explanation or providing another recommendation).

UMAP '20, July 14–17, 2020, Genoa, Italy

ACM ISBN 978-1-4503-6861-2/20/07…$15.00
https://doi.org/10.1145/3340631.3394856

Therefore, in this work, we have been engaged in understanding and predicting users' intents specific to conversational recommendations. Additionally, inspired by related work that incorporates user intents as features in general dialogue systems to identify user satisfaction [14, 19], we propose to particularly consider both user intents and recommender actions for better predicting user satisfaction with the recommendation. To be more specific, we have aimed to address the following three research questions:

RQ1: *How can we classify users' intents and recommenders' actions respectively in the dialogue conversation?*

RQ2: *How can we accurately predict a user's intents given her/his utterance in the recommendation dialogue?*

RQ3: *How does user satisfaction relate to their intents and recommender's actions in multi-turn interactions, and how can we accurately predict user satisfaction with the recommendation?*

To address these questions, we first examined a set of over 300 human-human dialogues centered around movie recommendations [26], in order to understand the language interaction between users (seekers) and human recommenders (see Figure 1), based on which we have developed two hierarchical taxonomies for user intents and recommender actions, respectively, using a grounded theory approach. Secondly, with the established taxonomies, we have further defined various categories of feature related to *content*, *discourse*, *sentiment*, and *context* information respectively, and have compared the performance of different machine learning and deep learning methods in terms of predicting user intents. To this end, we identify the most predictive features at both category-level and individual-level. The experimental results also show that some classical machine learning models (such as XGBoost and SVM) can perform well in predicting user intents. Besides, it shows that incorporating context features (like previous recommender actions) into the prediction model can significantly boost the performance. Thirdly, we have demonstrated the feasibility of integrating both user intents and recommender actions as dialogue behavior features to predict user satisfaction with the recommendation.

2 RELATED WORK

Dialogue-based Conversational Recommender Systems (DCRSs). Early examples of DCRS were mostly rule or frame based [6, 41, 47]. For example, in [41, 47], pre-defined questions were asked in order to quickly narrow down the space of relevant items, and a threshold number was used to determine what action (e.g., asking questions or proposing items) the system should carry out. Recent approaches have mainly emphasized on how to automatically select informative questions to obtain users' preferences before providing recommendations [11, 26, 45, 49, 51]. For instance, [51] implemented a system based on multi-memory network to ask aspect-based questions for understanding the user's need. However, most of recent systems normally end the conversation after presenting one or multiple recommendations to the user, which can not support multi-turn interactions, especially when the user is not satisfied with the current recommendation.

On the other hand, little work on DCRS has explicitly investigated users' intents or goals when they interact with the system. In [49], the authors identified four most-frequent user intents in a shopping chatbot (i.e., *recommendation*, *comparison*, *ask opinion*, and *Q&A*), and three session-aware intents (i.e., *add filter condition*, *see-more*, and *negation*). In [22], the authors collected users' initial and follow-up queries when they ask for recommendations via speech or text based dialogues, and then classified users' initial queries into *objective*, *subjective*, and *navigation* goals, and follow-up queries into *refine*, *reformulate*, and *start over* categories. However, the follow-up query was only prompted when the user rated the current recommendation "fair" or "better", so it does not consider the user's negative feedback. In our previous work, we have built a taxonomy of user feedback intents for conversational recommendations [5], which lays the foundation of this work.

In the broader area of recommender systems, *critiquing-based systems* have been proposed to elicit users' feedback in graphical user interfaces (GUI) [9]. These systems provide a critiquing interface that enables users to give feedback on the recommendation (e.g., *"Lower Resolution and Cheaper"*). This type of system has mainly offered users three feedback modalities [8], i.e., *similarity-based*, *quality-based*, and *quantity-based*, but because the interaction is through GUI elements (e.g., menu, form, button), users do not have much freedom of posting any feedback that they wish.

User Intent Prediction. User intents behind their utterances indicate the goal they may want to achieve when they interact with a dialogue system [15]. In the field of natural language processing, user intent prediction has commonly been treated as utterance classification problem, for which different solutions have been proposed, such as those based on logistic regression model [3, 43, 44], support vector machine [2], hidden markov models [2, 46], or recent deep learning techniques like convolutional neural networks (CNN) [2, 25, 28, 37, 50], long short-term memory networks (LSTM) [29, 33], and recursive autoencoders (RAEs) [23]. However, their applications are mostly for open-domain conversations [43] and tutoring dialogues [34], not for DCRSs.

User Satisfaction Prediction. User satisfaction can indicate whether their goal is fulfilled or not [19]. Related work on web searching has investigated how to utilize interaction signals (i.e., clicking, dwell time, and mouse scrolling) and temporal sequence to predict user satisfaction, and demonstrated the effectiveness of using Markov model and deep neural models (e.g., LSTM) in solving the sequential modeling problem [20, 32, 48]. However, their studied interaction signals are not suitable for dialogue systems that primarily rely on natural language communication. Facing this challenge, some researchers have proposed to incorporate user intents as potential features [14, 19]. [19] also developed intent sensitive word embeddings for query representation of the sequence.

Compared to related work, our work has several novelties: 1). We establish two hierarchical taxonomies to classify user intents and recommender actions respectively for DCRS. 2). We perform user intent prediction specific to DCRS, which has been rarely studied in related work possibly due to the lack of a well established taxonomy and annotated dialogue data. 3). We leverage both user intents and recommender actions to predict user satisfaction with the recommendation, which is new in DCRS to the best of our knowledge.

3 RESEARCH DESIGN AND METHODOLOGY

In this section, we describe how the two taxonomies respectively for user intents and recommender actions are established, followed

Table 1: Statistics of our selected dialogue data (from ReDial)

Items	**SAT-Dial** (with user-satisfied recommendation)	**unSAT-Dial** (without user-satisfied recommendation)
# Conversations	253	83
# Human seekers	125 (# utterances: 1,711)	59 (# utterances: 550)
# Human recommenders	151 (# utterances: 1,747)	68 (# utterances: 575)
# Suggested movies per dialogue	4.57	4.51
# Turns per dialogue	mean=6.58, min=3, max=19	mean=6.49, min=3, max=12
# Words per utterance	mean=11.29, min=1, max=72	mean=10.72, min=1, max=69

by the experimental setups for performing user intent prediction and satisfaction prediction.

3.1 Recommendation Dialogue Data

3.1.1 Data Collection. The recommendation dialogue data we processed is from the ReDial[1] dataset [26], which is publicly available centered around dialogue-based movie recommendations according to [26]. The ReDial dataset was collected through an interface where workers (from Amazon Mechanical Turk) were paired to accomplish a movie recommendation task using natural language [26]. Specifically, for each pair, one worker was given the role "seeker" who was to seek for interesting movies, and the other played the role "recommender" who was responsible for giving recommendations. To ensure the dialogue quality, every conversation session involved at least four movies, and at the end both seeker and recommender were asked some questions for each mentioned movie so as to be able to check whether their answers are consistent (e.g., "*Was the movie suggested by the recommender?*" "*Has the seeker seen the movie?*" "*Did the seeker like the movie suggestion?*").

3.1.2 Data Selection. We cleaned out the raw dialogue data in the ReDial dataset (that contains 11,348 dialogues) by performing the following steps: 1). We filtered out dialogues that contain less than three dialogue turns[2] and less than four different recommended movies. 2). We removed those with inconsistent answers from seekers and recommenders to the post-conversation reflective questions. 3). We then randomly sampled some satisfactory recommendation dialogues (SAT-Dial) where one recommended movie was not liked by the seeker but a subsequent one was accepted by her/him. These dialogues were used to capture the seeker's feedback intents on recommendation when s/he was not satisfied with it, and furthermore the actions taken by the human recommender that helped the seeker find a satisfactory item later. 4). We further sampled some unsatisfactory recommendation dialogues (unSAT-Dial) by choosing the dialogues that do not contain any recommendations accepted by the seeker. These dialogues can be useful for detecting what kind of interaction may lead to unsuccessful recommendation. Finally, we got 253 satisfactory dialogues and 83 unsatisfactory dialogues (see Table 1 with the statistics).

3.2 Taxonomies for User Intents and Recommender Actions

3.2.1 Methodology for Taxonomy Development. We employed the grounded theory approach [16] to develop the taxonomy, by following its suggested iterative procedure [4, 13]. First, we developed the initial taxonomies respectively for user intents and recommender actions by examining 20 randomly sampled dialogue data (called the development set) from our selected dialogues. In this process, we performed open coding to identify any categories of intents/actions and their associated characteristics [16]. Next, we asked two annotators to independently label newly sampled 10 dialogue data (the preliminary test set) based on the initial taxonomies, with the purpose of refining those previously established categories. The result of this process is a set of categories and subcategories of intents/actions. Finally, we randomly sampled 10 new dialogue data (the validation set) for testing the coverage of our taxonomies. We repeated the whole process (i.e., *propose-refine-annotate*) 3 times, in order to make our taxonomies accommodate all of the possible situations that may exist in the sampled dialogue data.

3.2.2 Taxonomy for User Intents. The established taxonomy for user intents is aimed to classify the types of utterance inputted by recommendation seekers. Through the procedure mentioned above, we come up with 3 top-level intents (i.e., **Ask for Recommendation**, **Add Details**, and **Give Feedback**), and 15 sub-intents (see Table 2).

Specifically, there are 4 sub-intents under **Ask for Recommendation**: The seeker asks for recommendations in the Initial Query (e.g., "*I like comedy. Do you know of any good ones?*"), or Continues to seek for more suggestions (e.g., "*Do you have any other suggestions?*"); Start Over indicates that the seeker starts a new query, possibly because s/he is unsatisfied with the current recommendation or s/he comes across a new goal; and the seeker may also Reformulate her/his previous query with or without clarifications or further constraints (e.g., "*Maybe I am not being clear. I want something that is in the theater now.*"). As for **Add Details**, the three sub-intents are: The seeker takes her/his initiative to Provide Preference (e.g., "*I usually enjoy movies with Seth Rogen and Jonah Hill.*"), Asks the recommender's opinion about an item (e.g., "*I really like Reese Witherspoon. How about you?*"), or Answers questions raised by the recommender. **Give Feedback** contains possible types of feedback the seeker may provide to the current recommendation, for which we find there are 8 major sub-intents: The seeker Accepts the recommended item when s/he likes it (e.g., "*Awesome, I will check it out.*"); s/he Inquires about the recommended movie for getting more details; s/he has Seen the recommendation before; s/he gives a Neutral Response without indicating her/his preference (e.g., "*I have actually never seen that one.*"); the seeker makes critique on the current recommendation by Critique-Add for adding more constraints (e.g., "*I would like something more recent.*"), Critique-Compare for requesting similar items to compare (e.g., "*Den of Thieves (2018) sounds amazing. Any others like that?*"), or Critique-Feature for critiquing a specific feature (e.g., "*That's a bit too scary for me.*"); and the seeker Rejects the recommended item if s/he dislikes it (e.g., "*I hated that movie.*").

3.2.3 Taxonomy for Recommender Actions. From recommenders' perspective, we have characterized their behavior into 4 top-level actions (i.e., **Request**, **Respond**, **Recommend**, and **Explain**) and 9 sub-actions (see Table 3).

Specifically, **Request** contains two sub-actions: The recommender may Request-Information about the seeker's preference for items (e.g., "*What kind of movies do you like?*") or feedback on the current recommendation; or ask a Clarifying Question (e.g., "*What kind of*

[1] https://redialdata.github.io/website/ [2] One dialogue turn denotes a consecutive utterance-response pair: Utterance is from seeker and response is from recommender.

Table 2: Taxonomy for user intents (sub-intents are sorted by their occurrence percentages in our dataset)

Intent (Code)	Description	Example	Percentage
Ask for Recommendation			**18.26%**
Initial Query (IQU)	Seeker asks for a recommendation in the first query.	"*I like comedy do you know of any good ones?*"	12.91%
Continue (CON)	Seeker asks for more recommendations in the subsequent query.	"*Do you have any other suggestions?*"	3.10 %
Reformulate (REF)	Seeker restates her/his query with or without clarification/further constraints.	"*Maybe I am not being clear. I want something that is in the theater now.*"	1.50%
Start Over (STO)	Seeker starts a new query to ask for recommendations.	"*Anything that I can watch with my kids under 10.*"	0.84%
Add Details			**18.58%**
Provide Preference (PRO)	Seeker provides specific preference for the item s/he is looking for.	"*I usually enjoy movies with Seth Rogen and Jonah Hill.*"	12.30%
Answer (ANS)	Seeker answers the question issued by the recommender.	"*Maybe something with more action.*" (Q: "*What kind of fun movie you look for?*")	4.91%
Ask Opinion (ASK)	Seeker asks the recommender's personal opinions.	"*I really like Reese Witherspoon. How about you?*"	2.39%
Give Feedback			**61.92%**
Seen (SEE)	Seeker has seen the recommended item before.	"*I have seen that one and enjoyed it.*"	21.14%
Accept (ACC)	Seeker likes the recommended item.	"*Awesome, I will check it out.*"	18.89%
Reject (REJ)	Seeker dislikes the recommended item.	"*I hated that movie. I did not even crack a smile once.*"	11.50%
Inquire (INQ)	Seeker wants to know more about the recommended item.	"*I haven't seen that one yet. What's it about?*"	6.55%
Critique-Feature (CRI-F)	Seeker makes critiques on specific features of the current recommendation.	"*That's a bit too scary for me.*"	6.50%
Critique-Add (CRI-A)	Seeker adds further constraints on top of the current recommendation.	"*I would like something more recent.*"	5.35%
Neutral Response (NRE)	Seeker does not indicate her/his preference for the current recommendation.	"*I have actually never seen that one.*"	4.29%
Critique-Compare (CRI-C)	Seeker requests sth similar to the current recommendation in order to compare.	"*Den of Thieves (2018) sounds amazing. Any others like that?*"	1.55%
Others	Greetings, gratitude expression, or chit-chat utterances.	"*Sorry about the weird typing.*"	14.55%

Table 3: Taxonomy for recommender actions (sub-actions are sorted by their occurrence percentages in our dataset)

Action (Code)	Description	Example	Percentage
Request			**13.87%**
Request Information (REQ)	Recommender requests for the seeker's preference or feedback.	"*What kind of movies do you like?*"	12.58%
Clarify Question (CLA)	Recommender asks a clarifying question for more details.	"*What kind of animated movie are you thinking of?*"	1.29%
Respond			**23.77%**
Respond-Feedback (RES)	Recommender responds to any other feedback from the seeker.	"*That's my favourite Christmas movie too!* " (U: "*My absolute favourite!!*")	15.89%
Answer (ANS)	Recommender answers the question asked by the seeker.	"*Steve Martin and John Candy.*" (Q: "*Who is in that?*")	7.88%
Recommend			**54.52%**
Recommend-Show (REC-S)	Recommender provides recommendation by showing it directly.	"*The Invitation (2015) is a movie kids like.*"	32.08%
Recommend-Explore (REC-E)	Recommender provides recommendation by inquiring about the seeker's preference.	"*Have you seen Cult of Chucky (2017) that one as pretty scary.*"	23.99%
Explain			**37.38%**
Explain-Introduction (EXP-I)	Recommender explains recommendation with non-personalized introduction.	"*What about Sleepless in Seattle (1993)? Hanks and Ryan?*"	22.83%
Explain-Preference (EXP-P)	Recommender explains recommendation based on the seeker's past preference.	"*Will Ferrell is also very good in Elf (2003) if you're in need of another comedy*"	13.01%
Explain-Suggestion (EXP-S)	Recommender explains recommendation in a suggestive way.	"*If you like gory then I would suggest The Last House on the Left (2009).*	2.37%
Others	Greetings, gratitude expression, or chit-chat utterances.	"*Have a good night.*"	29.80%

animated movie are you thinking of?"). Regarding **Respond**, the two sub-actions are: The recommender may Answer the seeker's question, or Respond-Feedback (for example, when a seeker gave feedback "*My absolute favourite!*", the recommender responded "*That's my favourite Christmas movie too!*"). **Recommend** includes two sub-actions to distinguish the ways of providing recommendations: Recommend-Show means that the recommender shows the recommendation directly (e.g., "*The Invitation (2015) is a movie kids like.*"); and Recommend-Explore indicates that the recommender provides an example item for acquiring the seeker's preference (e.g.,"*Have you seen Cult of Chucky (2017) that one as pretty scary?*"). Moreover, there are three sub-actions under **Explain**: Explain-Suggestion indicates that the recommender suggests the seeker to try the recommended item (e.g., "*If you like gory then I would suggest The Last House on the Left (2009).*"); Explain-Preference means that the explanation is in reference to the seeker's past preference (e.g., "*Will Ferrell is also very good in Elf (2003) if you're in need of another comedy.*"); and Explain-Introduction shows that the recommender introduces the recommendation in a non-personalized way (e.g., "*What about Sleepless in Seattle (1993)? Hanks and Ryan?*").

3.2.4 Data Annotation. After the two taxonomies were established, we asked two annotators to label all of our selected dialogue data. Concretely, for each seeker utterance or recommender response, the annotator was encouraged to choose all suitable code(s) that s/he thinks can represent the seeker's intent(s) or the recommender's action(s). They first independently labeled 30 random dialogues, and then met to discuss and resolve any disagreements to ensure annotation quality and consistency, before they started to label the remaining dialogues. For all of the labeled dialogues, the average inter-rater agreement scores (Cohen's kappa [12]) across 15 sub-intents and 9 sub-actions are respectively 0.75 (min=0.50, max=0.95) and 0.82 (min=0.50, max=0.96), which indicate satisfactory agreement according to [31].

3.3 User Intent Prediction

3.3.1 Problem Definition. For the user intent prediction problem, our goal was to predict a subset of user intent labels y_i ($y_i \in \mathcal{L}$, where $\mathcal{L}$ refers to the set of all category labels in our user intent taxonomy) for each given user utterance u_i (the user's i-th utterance) in a recommendation dialogue, since one utterance may contain multiple intents (for example, "*I did see that one, but I didn't really like it. I do love 80s movies though.*" which implies two intents, i.e., *Reject* and *Critique-Add*). Therefore, this is essentially a multi-label classification problem.

3.3.2 Classification Methods. We compared several machine learning models that have popularly been used for text classification [1], including Logistic Regression, Support Vector Machine (SVM), Naive Bayes, Decision Tree, Random Forest, XGBoost, and Multilayer Perceptron (MLP), for which we need to transform our multi-label classification problem into single-label problem. Concretely, we adopted three typical transformation strategies [39]: 1). **Binary Relevance** that trains binary classifiers independently for each label; 2). **Classifier Chain** that also performs binary classification,

but predicts labels sequentially, because it leverages the output of the previous classifier as input features to the subsequent classifier; and 3). **Label Powerset** that trains one single-label classifier for every label combination that exists in the training data. In addition, we experimented with ML-kNN algorithm that is based on the k-Nearest Neighbors method to solve multi-label classification, and two deep learning methods, i.e., CNN and Bi-directional LSTM (Bi-LSTM)[3] that have achieved good results in text mining tasks [25, 37].

3.3.3 Features. We have summarized three categories of feature for intent prediction, as inspired by related work on dialogue act discovery [3, 15, 37, 40, 46]: **Content**, **discourse**, and **sentiment**. In addition, we add a new category, **context** features, which is used to capture the dialogue's context information (e.g., the previous utterance). These features are listed in Table 4.

Content features (Cont). The content features include the TF-IDF (short for Term Frequency-Inverse Document Frequency [30]) that considers the word frequency, the extracted name entity through spaCy[4], and the number of relevant items (i.e., movies) from the utterance. These features are meant to capture the content information of the utterance.

Discourse features (Disc). The discourse features include the Part-Of-Speech (POS) information of the utterance by counting the frequency of each POS tag, the occurrence of 5W1H question (i.e., *what*, *who*, *why*, *when*, *where*, and *how*), question mark and exclamation mark that are indicators of question-related intents (e.g., *Inquire*, *Ask opinion*), the length of utterance (i.e., the number of words) with or without duplication removal, stop words removal, and stemming. The discourse features can provide linguistic characteristics of the utterance.

Sentiment features (Sent). The sentiment features are expected to capture the user's positive/negative feedback and gratitude expressions. In the recommendation dialogue, positive or negative sentiments can be related to feedback intent *Accept* or *Reject*. Thus, we calculated the sentiment score (i.e., positive/negative/neutral score) of each utterance using VADER [21] and counted the number of positive and/or negative words within one utterance based on opinion lexicon [27].

Context features (Context). The context features capture context information of the utterance, such as the position of the current utterance, which may be related to some user intents like *Initial Query* (as it often appears at the beginning) and *Start Over* (that may be likely to occur after several interaction turns). Also, as suggested in [37], we computed the cosine similarities between the current utterance and the previous response from the recommender, the previous utterance from the seeker, the initial utterance, and all of the previous utterances, for indicating the coherence of the current utterance to the previous utterances. Furthermore, we leveraged the dialogue behaviors of the seeker and the recommender (i.e., user intents and recommender actions) from the previous interaction turn for predicting the seeker's current intents.

3.3.4 Evaluation and Experimental Setup. We adopted four commonly used evaluation metrics for the multi-label classification task [42]: **Accuracy (Acc)** that refers to the proportion of the predicted correct labels to the total number of predicted and actual labels for every utterance, **Precision (Prec)** that refers to the proportion of the predicted correct labels to the number of predicted labels, **Recall (Rec)** that refers to the proportion of the predicted correct labels to the number of actual labels, and **F1-score (F1)** that is the harmonic mean of precision and recall. The reported performance is the average evaluation over all utterances.

We implemented the classification models (as mentioned in Section 3.3.2) with four Python packages: scikit-learn[5], scikit-multilearn[6], TensorFlow[7], and Keras[8]. For each machine learning model, an inner cross validation (5-fold) is used to tune the hyper-parameters and the best values are selected based on **Accuracy** on the training data (80%). An outer cross validation (10-fold) is further used to evaluate the model as selected by the inner cross validation. For the two deep learning models, we represented each token in the utterance as the pre-trained word embedding vector (200 dimensions) using GloVe [35], and then tuned the hyper-parameters on the validation data (10%) and report the results on the testing data (10%).

Table 4: Features extracted for user intent prediction

Category	*Features*
Content	TF-IDF, Name Entity, # Relevant Items
Discourse	POS, 5W1H Question, Question Mark, Exclamation Mark, Utterance Length
Sentiment	Thanks, Sentiment Score, Opinion Lexicon
Context	Absolute Position, Utterance Similarity, Previous user intents & recommendation actions

3.4 User Satisfaction Prediction

Being different from related work that mainly leveraged users' intents to predict their satisfaction in general dialogue systems [14, 19], we have particularly considered both user intents and recommender actions for the satisfaction prediction specific to DCRS.

3.4.1 Problem Definition. Given a fixed number (N) of turns in the dialogue, the problem is how to predict if the user would eventually accept a recommendation, which is indeed a binary classification problem. Specifically, for satisfactory recommendation dialogues (SAT-Dial) (see Table 1), we extracted the previous N utterance-response pairs before the occurrence of intent label Accept as input data. For unsatisfactory recommendation dialogues (unSAT-Dial), we assume that the utterance with the task-irrelevant intent label (such as Others) may imply that the user is not interested in the recommendation, so we extracted the previous N utterance-response pairs before such kind of intent as input data.

3.4.2 Classification Methods. We employed the same machine learning methods (see Section 3.3.2) to solve the binary classification problem, but did not include the two deep learning models because they may under-fit the small dataset [17].

[3] For these two models, we used the sigmoid activation function at the output layer to model the probability of each label as Bernoulli distribution to address the multi-label classification problem. [4] https://spacy.io/

[5] https://scikit-learn.org/ [6] http://scikit.ml/ [7] https://www.tensorflow.org/ [8] https://keras.io/

3.4.3 Features. In order to predict user satisfaction with the recommendation, we first investigated the relationship from user intents and recommender actions respectively to user satisfaction. The results (that will be given in Section 4.1) show that sub-intents under *Add Details* occur more frequently in unSAT-Dial than in SAT-Dial, and sub-actions under *Explain* more frequently occur in SAT-Dial. Motivated by such observations, we define them as **Dialogue Behavior features (Dial)** for performing the satisfaction prediction task, by counting the occurrence of each intent/action label from input data. In addition, we included utterance-level features (i.e., content, discourse, and sentiment features as described in Section 3.3.3) for all of the involved seeker utterances and recommender responses.

3.4.4 Evaluation and Experimental Setup. **Precision (Prec)**, **Recall (Rec)**, and **F1-score (F1)** metrics were still adopted to evaluate the prediction result. Accuracy was not used because it tends to be high in our imbalanced data (as SAT-Dial is about three times more than unSAT-Dial). The reported result is the average performance over all dialogues.

We implemented the classification models with scikit-learn. The procedure of hyper-parameters tuning and model evaluation is the same as that for user intent prediction.

4 RESULTS

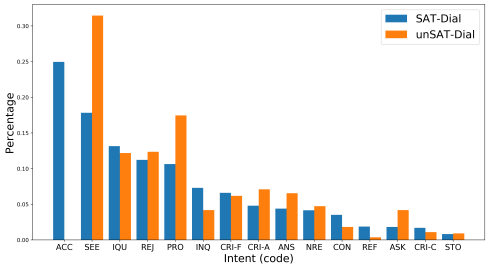
4.1 Dialogue Data Analysis

We first analyzed the annotated dialogue data in order to gain some insights of their characteristics.

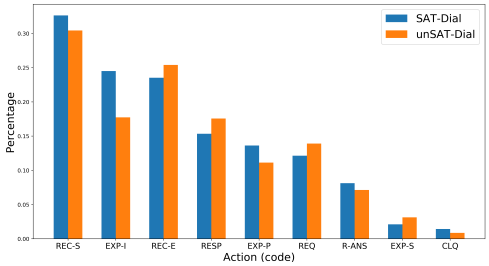
4.1.1 User Intent Analysis. The distribution of user intents in our dataset is shown in Table 2, from which we can see that *Give Feedback* more frequently occurs than other two top-level intents *Ask for Recommendations* and *Add Details*, inferring that those seekers often conversed with recommenders by giving feedback on the recommended item. Regarding the second-level sub-intents, we computed the intent distributions for satisfactory dialogues (SAT-Dial) and unsatisfactory dialogues (unSAT-Dial) respectively. As shown in Figure 2(a), in unSAT-Dial, three sub-intents under *Add Details*, i.e., *Provide Preference*, *Answer* and *Ask*, more frequently occur than those in SAT-Dial (e.g., the frequency of *Provide Preference* is 17.5% in unSAT-Dial vs. 10.6% in SAT-Dial). Moreover, we find that seekers have often *Seen* the recommended item in unsatisfactory dialogues, but *Inquire* more often in satisfactory dialogues.

In addition, we find that 23.57% of utterances contain more than one intent label and there are 136 combinations of multiple intents (e.g., *CRI-F+PRO*) that occur in all of the utterances.

4.1.2 Recommender Action Analysis. The recommender action distribution is shown in Table 3, from which we can see human recommenders tend to *Recommend* items to users in nearly half of the cases (54.52%) and *Explain* the recommended item (37.38%), suggesting that these two actions are quite common in recommendation dialogues. Similar to user intent analysis, we also compared the distribution of second-level sub-actions in satisfactory dialogues with that in unsatisfactory dialogues (see Figure 2(b)), which shows that sub-actions under *Explain* more frequently occur in SAT-Dial (e.g., 24.50% vs. 17.74% in unSAT-Dial regarding *Explain-Introduction*, 13.62% vs. 11.13% regarding *Explain-Preference*). This may imply that providing explanation to the recommendation is likely to increase users' acceptance of the recommended item.



(a) User intents



(b) Recommender actions

Figure 2: Distribution comparison between satisfactory (SAT-Dial) and unsatisfactory dialogues (unSAT-Dial).

4.2 User Intent Prediction

We examine the performance of user intent prediction from three aspects: Comparison of classification models, comparison of feature categories, and prediction performance in respect of each intent.

4.2.1 Comparison of Classification Models. As it is shown in Table 5, **XGBoost** can achieve the best overall performance no matter of which problem transformation strategy is used. **SVM** (with the transformation strategy Classification Chain) and Logistic Regression (with Label Powerset) can achieve performance comparable to XGBoost, followed by two tree-based methods Decision Trees and Random Forest.

As for the three problem transformation strategies, **Classification Chain** and **Label Powerset** are more effective in handling the multi-label classification task than Binary Relevance, which might be because they both consider the label dependency.

Table 6 further shows that the two deep learning methods (CNN and Bi-LSTM) do not perform better than XGBoost, possibly due to the lack of sufficient training data. Besides, the k-Nearest Neighbor algorithm for multi-label classification (i.e., ML-kNN) performs worst, which may be because the label density of our data (the average number of labels per utterance divided by the number of all labels) is low (i.e., 0.0921).

4.2.2 Comparison of Feature Categories. Table 7 shows the experimental results with different combinations of feature categories by using the best performing model XGBoost, from which we can

Table 5: Performance of machine learning methods with three problem transformation strategies for user intent prediction, where the best Accuracy (Acc), Precision (Prec), Recall (Rec), and F1 results are underlined

Methods	Binary Relevance				Classification Chain				Label Powerset			
	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
Logistic Regression	0.5796	0.7160	0.6148	0.6612	0.6111	0.6898	0.6322	0.6596	0.6198	0.6791	0.6053	0.6400
SVM	0.5597	0.6701	0.6047	0.6332	**0.6293**	0.7179	0.6340	0.6730	0.6048	0.6004	0.6123	0.6056
Naive Bayes	0.4438	0.5137	0.5705	0.5400	0.4567	0.5137	0.5793	0.5439	0.5365	0.5989	0.5542	0.5755
Decision Tree	0.5264	0.5187	0.6778	0.5871	0.5356	0.5513	0.6325	0.5887	0.4515	0.4706	0.4755	0.4729
Random Forest	0.5742	0.5962	**0.7029**	0.6449	0.5968	0.6372	**0.6817**	0.6583	0.4794	0.4748	0.5096	0.4913
XGBoost	**0.5970**	**0.8169**	0.6007	**0.6919**	0.6274	**0.7957**	0.6268	**0.7010**	**0.6199**	**0.6868**	**0.6109**	**0.6463**
MLP	0.4773	0.7922	0.4743	0.5928	0.5079	0.7780	0.5045	0.6115	0.6157	0.6837	0.6029	0.6407

Table 6: Performance of ML-kNN model and two deep learning methods (CNN and BiLSTM) for user intent prediction

Methods	Acc	Pre	Rec	F1
ML-KNN	0.3960	0.4347	0.4335	0.4340
CNN	0.5698	0.6773	0.5618	0.6141
BiLSTM	0.5720	0.6747	0.5794	0.6234

see that the prediction performance of considering **content features** alone is better than considering each of the other three feature categories, and discourse features are more effective than context features. Moreover, **combining context features and content/discourse features** can achieve better performance than combining content and discourse features, and the combination of context features with other two categories (i.e., content and discourse) can further boost the performance. Lastly, the results of combining all of the four feature categories obtain the best prediction accuracy among all settings, suggesting that each feature category brings certain contribution to enhancing the intent prediction.

Notice that in the above comparisons we only utilized the previous recommender response as the dialogue behavior in the category of context features. In the following analysis, we varied the number of considered dialogue turns. Figure 3 shows the performance of the best two models and the worst model, which shows that considering the seeker's intents in the previous utterance is slightly better than only considering the previous response from the recommender for both XGBoost and Naive Bayes models, while the **performance decreases when more turns are involved**, which may be because the latest conversation is more relevant to the current intent.

Another analysis focuses on revealing the effect of individual features. We still tested on the best model XGBoost, and used Label Powerset for problem transformation. There are **several interesting observations**: 1). Content features take a large proportion among the top-20 most important features (12 out of 20). In particular, some vocabularies (e.g., *good*, *bad*, *need*, *check*) are informative content features for identifying user intents. 2). Some context features (6) are also among the top-20, such as the absolute position of the utterance in the dialogue and the previous recommender actions. 3). As for discourse features, the length of utterance with unique words is important, together with question mark and exclamation mark. 4). The sentiment features are relatively less useful for predicting user intents.

Table 7: Experimental results with different combinations of feature categories for user intent prediction

	Cont	Disc	Sent	Context	Acc	Prec	Rec	F1
1 Category	✓				**0.4726**	**0.7165**	**0.4868**	**0.5793**
		✓			0.3918	0.5224	0.3841	0.4426
				✓	0.3407	0.5020	0.3343	0.4011
			✓		0.1993	0.3241	0.2044	0.2498
2 Categories	✓			✓	**0.5603**	**0.7669**	**0.5627**	**0.6488**
		✓		✓	0.5438	0.6946	0.5346	0.6039
	✓	✓			0.5291	0.7381	0.5350	0.6201
	✓		✓		0.4921	0.7289	0.5067	0.5972
			✓	✓	0.4587	0.6209	0.4518	0.5229
		✓	✓		0.4268	0.5553	0.4208	0.4787
3 Categories	✓	✓		✓	**0.6119**	**0.7913**	**0.6112**	**0.6896**
	✓		✓	✓	0.5870	0.7760	0.5887	0.6692
		✓	✓	✓	0.5698	0.7188	0.5569	0.6275
	✓	✓	✓		0.5415	0.7418	0.5500	0.6313
All	✓	✓	✓	✓	**0.6274**	**0.7957**	**0.6268**	**0.7010**

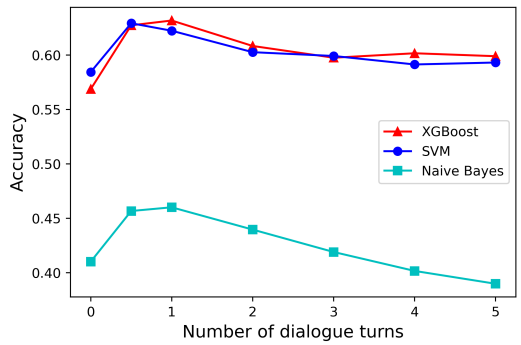


Figure 3: Comparison by varying the considered number of dialogue turns for user intent prediction. X-axis value 1 indicates only considering the previous utterance-response pair, and value 0.5 means only considering the previous recommender response.

4.2.3 Individual Intent Prediction. In addition to reporting the average performance over all intent labels, we investigated the prediction accuracy in respect of each intent (note this is a binary classification problem). The best prediction results using XGBoost

Table 8: Performance for individual intent prediction

Intent Code	Cont	Disc	Sent	Context	Prec	Rec	F1
OTH	✓	✓	✓	✓	0.9325	0.9134	0.9224
IQU	✓	✓		✓	0.8985	0.8933	0.8941
SEE	✓	✓	✓	✓	0.7859	0.6798	0.7270
ACC	✓	✓	✓	✓	0.8391	0.6416	0.7239
CON	✓			✓	0.8014	0.5429	0.6294
INQ	✓	✓	✓	✓	0.6910	0.5352	0.5923
PRO	✓		✓	✓	0.7302	0.4930	0.5821
ANS	✓	✓		✓	0.6182	0.5053	0.5471
REJ	✓	✓	✓		0.6704	0.4500	0.5357

Table 9: Performance of machine learning methods for user satisfaction prediction

Methods	Cont	Disc	Sent	Dial	Prec	Rec	F1
Logistic Regression	✓	✓		✓	0.8488	**0.5806**	0.6795
SVM		✓		✓	0.8778	0.5556	0.6629
Naive Bayes				✓	0.8833	0.5556	0.6651
Decision Tree				✓	0.7109	0.5528	0.6167
Random Forest				✓	0.8862	0.5306	0.6503
XGBoost				✓	0.7897	0.5653	0.6426
MLP				✓	**0.8990**	0.5681	**0.6884**
KNN			✓	✓	0.8850	0.5181	0.6427

are given in Table 8, from which we can see that both Initial Query and Others are easiest to be predicted, with F1-scores greater than 89%. Seen and Accept can also achieve relatively high accuracy (F1-scores greater than 72%). Some of the other intents, including Continue, Inquire, Provide Preference, Answer, and Reject, are predicted with F1-scores ranging from 53% to 63%, inferring that it might still be challenging to identify these intents. Of note, we omit seven intents with poor prediction results (F1-scores lower than 38%) in Table 8.

From the presented results, we can also see that all of the four feature categories contribute to the prediction, especially **content and context features**.

4.3 User Satisfaction Prediction

4.3.1 Comparison of Classification Models. Table 9 shows the results of comparing machine learning methods for predicting user satisfaction. We can see that **MLP** achieves the best precision and F1-score, while **Logistic Regression** obtains the highest recall. Other methods, such as SVM, Naive Bayes, and Random Forest, can also perform relatively well in solving this classification task.

4.3.2 Comparison of Feature Categories. We used the best performing model MLP to conduct this analysis. Table 10 shows that the performance of using **dialogue behavior features (Dial)** (i.e., involving both user intents and recommender actions) is higher than that of using content features, followed by discourse and sentiment features; while the combination of all the feature categories cannot outperform dialogue behavior features. Moreover, by varying the number of dialogue turns, we find that the performance still tends to decrease and fluctuate when involving more turns (see Figure 4).

Table 10: Experimental results with feature category for user satisfaction prediction

Method	Cont	Disc	Sent	Dial	Prec	Rec	F1
MLP				✓	**0.8990**	**0.5681**	**0.6884**
	✓				0.6551	0.4944	0.5501
		✓			0.5570	0.3486	0.4122
			✓		0.6067	0.2681	0.3606
	✓	✓	✓	✓	0.7995	0.5444	0.6292

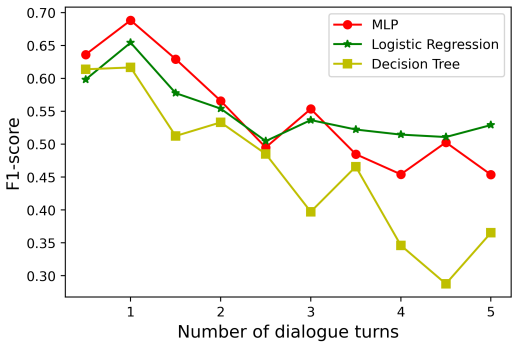


Figure 4: Comparison by varying the considered number of dialogue turns for user satisfaction prediction.

5 DISCUSSION AND FUTURE WORK

Therefore, in this work, we have not only contributed with two taxonomies established for user intents and recommender actions respectively through examining human-human recommendation dialogue data, but also investigated the possibility of automatically predicting user intents and satisfaction with the recommendation by comparing different machine learning methods. Specifically, for user intent prediction, some methods such as XGBoost and SVM can achieve outperforming accuracy by unifying four feature categories (i.e., content, sentiment, discourse, and context). Moreover, we find that incorporating context features (such as previous recommender actions) can indeed help boost the performance. Regarding user satisfaction prediction, leveraging both user intents and recommender actions (as dialogue behavior features) enables some classification model like MLP to achieve competitive accuracy.

In the future, we plan to address three limitations of the current work: 1). We just analyzed a set of dialogue data about movie recommendations, so the taxonomies' generalizability to other dialogues and product domains need to be further verified. 2). The deep learning (DL) methods did not perform well in our experiment, which might be due to the small dataset we currently have, so we will continue to label more dialogue data and identify whether DL methods would become superior when the dataset is enlarged. 3). We did not consider the temporal sequence of utterances/responses within a dialogue, which however might be treated as potentially useful context features to further improve the prediction accuracy.

ACKNOWLEDGMENTS

This work was partially supported by Hong Kong Baptist University IRCMS Project (IRCMS/19-20/D05). We also thank Ms. Yangyang Zheng for her assistance in annotating.

REFERENCES

[1] Charu C. Aggarwal and ChengXiang Zhai. 2012. *Mining Text Data*. Springer Science & Business Media.

[2] A. Bhargava, A. Celikyilmaz, D. Hakkani-Tür, and R. Sarikaya. 2013. Easy Contextual Intent Prediction and Slot Detection. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '13)*. 8337–8341.

[3] Sumit Bhatia and Prasenjit Mitra. 2012. Classifying User Messages For Managing Web Forum Data. In *WebDB*. 13–18.

[4] Andrei Broder. 2002. A Taxonomy of Web Search. *SIGIR Forum* 36, 2 (2002), 3–10. http://doi.acm.org/10.1145/792550.792552

[5] Wanling Cai and Li Chen. 2019. Towards a Taxonomy of User Feedback Intents for Conversational Recommendations. In *Proceedings of ACM RecSys 2019 Late-breaking Results co-located with the 13th ACM Conference on Recommender Systems*.

[6] Joyce Yue Chai, Malgorzata Budzikowska, Veronika Horvath, Nicolas Nicolov, Nanda Kambhatla, and Wlodek Zadrozny. 2001. Natural Language Sales Assistant - A Web-Based Dialog System for Online Sales. In *Proceedings of the Thirteenth Conference on Innovative Applications of Artificial Intelligence Conference (IAAI '01)*. 19–26. http://dl.acm.org/citation.cfm?id=645453.653001

[7] Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. A Survey on Dialogue Systems: Recent Advances and New Frontiers. *ACM SIGKDD Explorations Newsletter* 19, 2 (2017), 25–35. http://doi.acm.org/10.1145/3166054.3166058

[8] Li Chen and Pearl Pu. 2006. Evaluating Critiquing-based Recommender Agents. In *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 1 (AAAI '06)*. 157–162. http://dl.acm.org/citation.cfm?id=1597538.1597564

[9] Li Chen and Pearl Pu. 2012. Critiquing-based Recommenders: Survey and Emerging Trends. *User Modeling and User-Adapted Interaction* 22, 1-2 (2012), 125–150. http://dx.doi.org/10.1007/s11257-011-9108-6

[10] Konstantina Christakopoulou, Alex Beutel, Rui Li, Sagar Jain, and Ed H. Chi. 2018. Q&R: A Two-Stage Approach Toward Interactive Recommendation. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '18)*. 139–148. http://doi.acm.org/10.1145/3219819.3219894

[11] Konstantina Christakopoulou, Filip Radlinski, and Katja Hofmann. 2016. Towards Conversational Recommender Systems. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*. 815–824. http://doi.acm.org/10.1145/2939672.2939746

[12] Jacob Cohen. 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement* 20, 1 (1960), 37–46.

[13] Jane F Dye, Irene M Schatz, Brian A Rosenberg, and Susanne T Coleman. 2000. Constant Comparison Method: A Kaleidoscope of Data. *The Qualitative Report* 4, 1 (2000), 1–10.

[14] Klaus-Peter Engelbrech, Florian Gödde, Felix Hartard, Hamed Ketabdar, and Sebastian Möller. 2009. Modeling User Satisfaction with Hidden Markov Model. In *Proceedings of the SIGDIAL 2009 Conference: The 10th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL '09)*. 170–177. http://dl.acm.org/citation.cfm?id=1708376.1708402

[15] Elena V. Epure, Dario Compagno, Camille Salinesi, Rebecca Deneckere, Marko Bajec, and Slavko Žitnik. 2018. Process Models of Interrelated Speech Intentions from Online Health-Related Conversations. *Artificial Intelligence in Medicine* 91 (2018), 23–38.

[16] Barney G. Glaser. 1998. *Doing Grounded Theory: Issues and Discussions*. Sociology Press.

[17] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. The MIT Press.

[18] Jonathan Grudin and Richard Jacques. 2019. Chatbots, Humbots, and the Quest for Artificial General Intelligence. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. 1–11. https://doi.org/10.1145/3290605.3300439

[19] Seyyed Hadi Hashemi, Kyle Williams, Ahmed El Kholy, Imed Zitouni, and Paul A. Crook. 2018. Measuring User Satisfaction on Smart Speaker Intelligent Assistants Using Intent Sensitive Query Embeddings. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM '18)*. 1183–1192. http://doi.acm.org/10.1145/3269206.3271802

[20] Ryuichiro Higashinaka, Yasuhiro Minami, Kohji Dohsaka, and Toyomi Meguro. 2010. Modeling User Satisfaction Transitions in Dialogues from Overall Ratings. In *Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL '10)*. 18–27. http://dl.acm.org/citation.cfm?id=1944506.1944510

[21] C.J. Hutto and Eric Gilbert. 2014. VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. In *International AAAI Conference on Web and Social Media (ICWSM '14)*. https://www.aaai.org/ocs/index.php/ICWSM/ICWSM14/paper/view/8109

[22] Jie Kang, Kyle Condiff, Shuo Chang, Joseph A. Konstan, Loren Terveen, and F. Maxwell Harper. 2017. Understanding How People Use Natural Language to Ask for Recommendations. In *Proceedings of the Eleventh ACM Conference on Recommender Systems (RecSys '17)*. 229–237. http://doi.acm.org/10.1145/3109859.3109873

[23] Tsuneo Kato, Atsushi Nagai, Naoki Noda, Ryosuke Sumitomo, Jianming Wu, and Seiichi Yamamoto. 2017. Utterance Intent Classification of a Spoken Dialogue System with Efficiently Untied Recursive Autoencoders. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*. 60–64. https://www.aclweb.org/anthology/W17-5508

[24] Diane Kelly. 2009. Methods for Evaluating Interactive Information Retrieval Systems with Users. *Foundations and Trends in Information Retrieval* 3, 1-2 (2009), 1–224. https://doi.org/10.1561/1500000012

[25] Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP '14)*. 1746–1751. https://www.aclweb.org/anthology/D14-1181

[26] Raymond Li, Samira Ebrahimi Kahou, Hannes Schulz, Vincent Michalski, Laurent Charlin, and Chris Pal. 2018. Towards Deep Conversational Recommendations. In *Advances in Neural Information Processing Systems 32*. 9748–9758. http://papers.nips.cc/paper/8180-towards-deep-conversational-recommendations.pdf

[27] Bing Liu, Minqing Hu, and Junsheng Cheng. 2005. Opinion Observer: Analyzing and Comparing Opinions on the Web. In *Proceedings of the 14th International Conference on World Wide Web (WWW '05)*. 342–351. https://doi.org/10.1145/1060745.1060797

[28] Chunxi Liu, Puyang Xu, and Ruhi Sarikaya. 2015. Deep Contextual Language Understanding in Spoken Dialogue Systems. In *Sixteenth Annual Conference of the International Speech Communication Association*.

[29] Yang Liu, Kun Han, Zhao Tan, and Yun Lei. 2017. Using Context Information for Dialog Act Classification in DNN Framework. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP '17)*. 2170–2178. https://www.aclweb.org/anthology/D17-1231

[30] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

[31] Mary L McHugh. 2012. Interrater Reliability: the Kappa Statistic. *Biochemia Medica* 22, 3 (2012), 276–282.

[32] Rishabh Mehrotra, Imed Zitouni, Ahmed Hassan Awadallah, Ahmed El Kholy, and Madian Khabsa. 2017. User Interaction Sequences for Search Satisfaction Prediction. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*. 165–174. http://doi.acm.org/10.1145/3077136.3080833

[33] Lian Meng and Minlie Huang. 2018. Dialogue Intent Classification with Long Short-Term Memory Networks. In *National CCF Conference on Natural Language Processing and Chinese Computing (NLPCC '18)*. Springer, 42–50.

[34] Andrew Olney, Max Louwerse, Eric Matthews, Johanna Marineau, Heather Hite-Mitchell, and Arthur Graesser. 2003. Utterance Classification in AutoTutor. In *Proceedings of the HLT-NAACL 03 Workshop on Building Educational Applications Using Natural Language Processing - Volume 2 (HLT-NAACL-EDUC '03)*. 1–8. https://doi.org/10.3115/1118894.1118895

[35] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP '14)*. 1532–1543. https://www.aclweb.org/anthology/D14-1162

[36] Bilih Priyogi. 2019. Preference Elicitation Strategy for Conversational Recommender System. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining (WSDM '19)*. 824–825. http://doi.acm.org/10.1145/3289600.3291604

[37] Chen Qu, Liu Yang, W. Bruce Croft, Yongfeng Zhang, Johanne R. Trippas, and Minghui Qiu. 2019. User Intent Prediction in Information-seeking Conversations. In *Proceedings of the 2019 Conference on Human Information Interaction and Retrieval (CHIIR '19)*. 25–33. http://doi.acm.org/10.1145/3295750.3298924

[38] Dimitrios Rafailidis and Yannis Manolopoulos. 2019. The Technological Gap Between Virtual Assistants and Recommendation Systems. *CoRR* abs/1901.00431 (2019). http://arxiv.org/abs/1901.00431

[39] Jesse Read, Bernhard Pfahringer, Geoff Holmes, and Eibe Frank. 2011. Classifier Chains for Multi-label Classification. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. 254–269.

[40] Daniel E. Rose and Danny Levinson. 2004. Understanding User Goals in Web Search. In *Proceedings of the 13th International Conference on World Wide Web (WWW '04)*. 13–19. http://doi.acm.org/10.1145/988672.988675

[41] Hideo Shimazu. 2001. ExpertClerk: Navigating Shoppers' Buying Process with the Combination of Asking and Proposing. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 2 (IJCAI '01)*. 1443–1448. http://dl.acm.org/citation.cfm?id=1642194.1642287

[42] Mohammad S. Sorower. 2010. A Literature Survey on Algorithms for Multi-label Learning. *Oregon State University, Corvallis* 18 (2010), 1–25.

[43] Andreas Stolcke, Noah Coccaro, Rebecca Bates, Paul Taylor, Carol Van Ess-Dykema, Klaus Ries, Elizabeth Shriberg, Daniel Jurafsky, Rachel Martin, and Marie Meteer. 2000. Dialogue Act Modeling for Automatic Tagging and Recognition of Conversational Speech. *Computational Linguistics* 26, 3 (2000), 339–373. https://doi.org/10.1162/089120100561737

[44] Ming Sun, Yun-Nung Chen, and Alexander I. Rudnicky. 2015. Understanding User's Cross-domain Intentions in Spoken Dialog Systems. In *NIPS workshop on Machine Learning for SLU and Interaction (NIPS-SLU)*.

[45] Yueming Sun and Yi Zhang. 2018. Conversational Recommender System. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR '18)*. 235–244. http://doi.acm.org/10.1145/3209978.3210002

[46] Dinoj Surendran and Gina-Anne Levow. [n.d.]. Dialog Act Tagging with Support Vector Machines and Hidden Markov Models. In *Ninth International Conference on Spoken Language Processing (ICSLP-Interspeech '06)*.

[47] Cynthia A. Thompson, Mehmet H. Göker, and Pat Langley. 2004. A Personalized System for Conversational Recommendations. *Journal of Artificial Intelligence Research* 21, 1 (2004), 393–428. http://dl.acm.org/citation.cfm?id=1622467.1622479

[48] Kyle Williams and Imed Zitouni. 2017. Does That Mean You're Happy?: RNN-based Modeling of User Interaction Sequences to Detect Good Abandonment. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management (CIKM '17)*. 727–736. http://doi.acm.org/10.1145/3132847.3133035

[49] Zhao Yan, Nan Duan, Peng Chen, Ming Zhou, Jianshe Zhou, and Zhoujun Li. 2017. Building Task-oriented Dialogue Systems for Online Shopping. In *Thirty-First AAAI Conference on Artificial Intelligence (AAAI '17)*. https://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14261

[50] Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level Convolutional Networks for Text Classification. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1 (NIPS '15)*. 649–657. http://dl.acm.org/citation.cfm?id=2969239.2969312

[51] Yongfeng Zhang, Xu Chen, Qingyao Ai, Liu Yang, and W. Bruce Croft. 2018. Towards Conversational Search and Recommendation: System Ask, User Respond. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM '18)*. 177–186. http://doi.acm.org/10.1145/3269206.3271776