IART: Intent-aware Response Ranking with Transformers in Information-seeking Conversation Systems

Liu Yang[1] Minghui Qiu[2] Chen Qu [1] Cen Chen[3] Jiafeng Guo[4] Yongfeng Zhang[5]
W. Bruce Croft[1] Haiqing Chen[2]
[1] Center for Intelligent Information Retrieval, University of Massachusetts Amherst [2] Alibaba Group
[3] Ant Financial Services Group [4] Institute of Computing Technology, Chinese Academy of Sciences
[5] Dept. of Computer Science, Rutgers University
{lyang,chenqu,croft}@cs.umass.edu,{minghui.qmh,haiqing.chenhq}@alibaba-inc.com,chencen.cc@antfin.com
guojiafeng@ict.ac.cn,yongfeng.zhang@rutgers.edu

ABSTRACT

Personal assistant systems, such as Apple Siri, Google Assistant, Amazon Alexa, and Microsoft Cortana, are becoming ever more widely used. Understanding user intent such as clarification questions, potential answers and user feedback in information-seeking conversations is critical for retrieving good responses. In this paper, we analyze user intent patterns in information-seeking conversations and propose an intent-aware neural response ranking model "**IART**", which refers to "**I**ntent-**A**ware **R**anking with **T**ransformers". IART is built on top of the integration of user intent modeling and language representation learning with the Transformer architecture, which relies entirely on a self-attention mechanism instead of recurrent nets [35]. It incorporates intent-aware utterance attention to derive an importance weighting scheme of utterances in conversation context with the aim of better conversation history understanding. We conduct extensive experiments with three information-seeking conversation data sets including both standard benchmarks and commercial data. Our proposed model outperforms all baseline methods with respect to a variety of metrics. We also perform case studies and analysis of learned user intent and its impact on response ranking in information-seeking conversations to provide interpretation of results.

ACM Reference Format:
Liu Yang, Minghui Qiu, Chen Qu, Cen Chen, Jiafeng Guo, Yongfeng Zhang, W. Bruce Croft, Haiqing Chen. 2020. IART: Intent-aware Response Ranking with Transformers in Information-seeking Conversation Systems. In *Proceedings of The Web Conference 2020 (WWW '20), April 20–24, 2020, Taipei, Taiwan.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3366423.3380011

1 INTRODUCTION

The recent boom of artificial intelligence has witnessed the emerging and flourishing of many intelligent personal assistant systems, including Amazon Alexa, Apple Siri, Alibaba AliMe, Microsoft Cortana and Google Assistant. This trend has led to an interest in conversational search systems, where users would be able to access information with conversational interactions. Existing approaches to building conversational systems include generation-based methods [24, 26], retrieval-based methods [11, 40, 42], and hybrid methods [28, 44]. Significant progress has been made on the integration of conversation context by generating reformulated queries with contexts [40], enhancing context-response matching with sequential interactions [38], and learning with external knowledge [45]. However, much less attention has been paid on the user intent in conversations and how to leverage user intent for response ranking in information-seeking conversations.

To illustrate user intent in information-seeking conversations, we show an example dialog from the Microsoft Answers Community[1] in Table 1. Microsoft Answers Community is a customer support QA forum where users can ask questions relevant to Microsoft products. Agents like Microsoft employees or other experienced users will reply to these questions. There could be multi-turn conversation interactions between users and agents. We define a taxonomy of user intent following previous research [20, 21]. We can observe that there are diverse user intents such as "Original Question (OQ)", "Information Request (IR)", "Potential Answers (PA)", "Follow-up Questions (FQ)", "Further Details (FD)", etc. in an information-seeking conversation. Moreover, several transition patterns can happen between different user intent. For example, given a question from the user, an agent could provide a potential answer directly or ask for some information as clarification questions before providing answers. Users will provide further details regarding the information requests from agents. At the beginning of a conversation, the agent would like to greet customers or express gratitude to users before they move on to next steps. Near the end of a conversation, the user may provide positive or negative feedback about answers from agents, or ask a follow-up question to continue the conversation interactions.

Such user intent patterns can be helpful for conversation models to select good responses due to the following reasons: (1) The intent sequence in conversation context utterances can provide additional signals to promote response candidates with correct intent and demote response candidates with wrong intent. For example, in Table 1, given the intent sequence **[OQ] → [IR/ PA] → [PA/ FQ] → [FD]**, we know that the user is still expecting an answer to solve her question. Although both Response-1 and Response-2 show some lexical and semantic similarities with context utterances, only Response-1 has the intent "Potential Answers" (PA). In this case, the model should have the capability to promote the rank of Response-1 and demote Response-2. (2) Intent information can help the model

[1] https://answers.microsoft.com

WWW '20, April 20–24, 2020, Taipei, Taiwan

ACM ISBN 978-1-4503-7023-3/20/04.
https://doi.org/10.1145/3366423.3380011

Table 1: An example dialog to illustrate user intent transition patterns from the Microsoft Answers Community. The user intent "OQ", "IR", "PA", "FQ", "FD", "GG" denote "Original Question", "Information Request", "Potential Answer", "Follow-up Question", "Further Details", "Greetings/ Gratitude" respectively. We highlight some lexical match between utterances and response candidates. This table is better readable in color.

ID	Role	Utterances	Intent
Utterance-1	User	Windows downloaded this update "2018-02 Cumulative Update for Windows 10 ....." But during the restart it says "we couldn't complete the update, undoing changes". So what can I do to stop this? Thanks	**OQ**
Utterance-2	Agent	Is there any other pending updates? Try Download troubleshooter for Win 10.	**IR/ PA**
Utterance-3	User	Yes, pending updates the same one. I already used the built in troubleshooter, it did fix some 3 issues, but doing a restart the problem persists. Can I stop updates from installing this particular one? Thanks.	**PA/ FQ**
Utterance-4	User	Not sure if related but I just saw that Malicious Software Removal of March did not install	**FD**
Response-1 (***Correct***)	Agent	Try run troubleshooter and then restart your PC. If problem persist, open start and search for Feedback and open Feedback Hub app and report this issue.	**PA**
Response-2 (***Wrong***)	Agent	Glad to know that you fixed the issue, and as I said downloading the "Show or hide updates" troubleshooter and restarting the PC will help you. Thank you for asking questions and providing feedback here!	**GG**

to derive an importance weighting scheme over context utterances with attention mechanisms. In the given example dialog in Table 1, the model should learn to assign larger weights to utterances on question descriptions (OQ and FQ) and further details (FD) in order to address the information need of the user.

Most existing neural conversation models do not explicitly model user intent in conversations. More research needs to be done to understand the role of user intent in response retrieval and to develop effective models for intent-aware response ranking in information-seeking conversations, which is exactly the goal of this paper. There is some existing related work from the Dialog System Technology Challenge (formerly the Dialog State Tracking Challenge, DSTC)[2]. Many DSTC tasks focus on goal oriented conversations like restaurant reservation. These tasks are typically tackled with slot filling [8, 48], which is not applicable to information-seeking conversations because of the diversity of information needs. Recently in DSTC7 of 2018,[3] an end-to-end response selection challenge has been introduced, which shares similar motivation to our work. However, the evaluation treated response selection as a classification task and there was no explicit modeling of user intent.

In this paper, we analyze user intent in information-seeking conversations and propose neural ranking models with the integration of user intent modeling. Different user intent types are defined and characterized following previous research [20, 21]. Then we propose an intent-aware neural ranking model for response retrieval, which is built on top of recent breakthroughs in natural language representation learning with Transformers [3, 35]. We refer to the proposed model as "**IART**"[4], which is "**I**ntent-**A**ware **R**anking with **T**ransformers". IART incorporates intent-aware utterance attention to derive the importance weighting scheme of utterances in conversation context towards better conversation history understanding. We conduct extensive experiments with three information-seeking conversation data sets: **MSDialog**[5] [20], Ubuntu Dialog Corpus (**UDC**) [14], and another commercial customer service data from the AliMe assistant [13] in Alibaba group (**AliMe**). We compare our methods with various neural ranking models and baseline methods on response selection in multi-turn conversations including the recently proposed Deep Attention Matching Network (DAM) [51]. The results show our methods outperform all baselines. We also perform visualization and analysis of learned user intent patterns.

Our contributions can be summarized as follows: (1) We analyze user intent in information-seeking conversations for intent-aware response ranking. To the best of our knowledge, our work is the first to explicitly define and model user intent for response ranking in information-seeking conversations. (2) We propose an intent-aware response ranking model with Transformers to utilize user intent information for response ranking. (3) Experimental results with three different conversation data sets show that our methods outperform various baselines. We also perform analysis on learned user intent and ranking examples to provide insights. The code of our model implementation will be released on GitHub[6].

2 RELATED WORK

User Intent in Conversations. Some previous research studied utterance intent modeling in conversation systems [2, 15, 27, 31]. Stolcke et al. [31] performed dialog acts classification with a statistical approach on the SwitchBoard corpus, which consists of human-human chit chats conversations. In this paper, we explore how to combine utterance intent modeling with response ranking in conversations, so that the learned user intent of context utterances and response candidates can help the model select better responses in information-seeking conversations.

Conversational Search. Our research is relevant to conversational search [23, 33, 46, 49], which has received significant attention recently. Radlinski and Craswell described the basic features of conversational search systems [23]. Zhang et al. [49] introduced the System Ask User Respond (SAUR) paradigm for conversational search and recommendation. In addition to conversational search models, researchers have also studied the medium of conversational search [30, 34]. Our research targets at the response ranking of information-seeking conversations, with Transformer based ranking models and the integration of user intent modeling.

Neural Conversational Models. There is growing interest in research about conversation response generation and ranking with

[2]https://www.microsoft.com/en-us/research/event/dialog-state-tracking-challenge/
[3]http://workshop.colips.org/dstc7/
[4]IART is pronounced as "art".
[5]https://ciir.cs.umass.edu/downloads/msdialog/
[6]https://github.com/yangliuy/Intent-Aware-Ranking-Transformers

deep learning and reinforcement learning [5]. Existing work includes retrieval-based methods [22, 32, 38, 40–42, 45, 50], generation-based methods [4, 19, 24, 26, 29, 36], and hybrid methods [28, 44]. Our work is a retrieval-based method. Zhou et al. [51] investigated matching a response with conversation contexts with dependency information learned by Transformers. Our proposed models are also built with Transformer encoders. The main difference between our work and their research is that we explicitly define and model user intent in conversations. We show that the intent-aware attention mechanism can help improve response ranking in conversations.

Neural Ranking Models. Recent progress of research on neural approaches to IR has introduced a number of neural ranking models for information retrieval, question answering and conversation response ranking [7]. These models include representation focused models [10] and interaction focused models [6, 9, 18, 43, 47]. The neural ranking models proposed in our research adopt Transformers, which are solely based on attention mechanisms, as the encoder to learn representations.

3 OUR APPROACH

3.1 Problem Formulation

The research problem of response ranking in information-seeking conversations is defined as follows. We are given an information-seeking conversation data set $\mathcal{D} = \{(\mathcal{U}_i, \mathcal{R}_i, \mathcal{Y}_i)\}_{i=1}^{N}$, where $\mathcal{U}_i = \{u_i^1, u_i^2, \ldots, u_i^{t-1}, u_i^t\}$ in which u_i^t is the utterance in the t-th turn of the i-th dialog. $\mathcal{R}_i$ and $\mathcal{Y}_i$ are a set of response candidates $\{r_i^1, r_i^2, \ldots, r_i^k\}_{k=1}^{M}$ and the corresponding labels $\{y_i^1, y_i^2, \ldots, y_i^k\}$, where $y_i^k = 1$ denotes r_i^k is a true response for $\mathcal{U}_i$. Otherwise $y_i^k = 0$. For user intent information, there are sequence level user intent labels for both dialog context utterances and response candidates $\mathcal{E} = \{(\mathcal{I}_i^u, \mathcal{I}_i^r)\}_{i=1}^{N}$, where $\mathcal{I}_i^u$ and $\mathcal{I}_i^r$ are user intent labels for context utterances and response candidates for the i-th dialog respectively. Our task is to learn a ranking model $f(\cdot)$ with $\mathcal{D}$ and $\mathcal{E}$. For any given $\mathcal{U}_i$, the model should be able to generate a ranking list for the candidate responses $\mathcal{R}_i$ with $f(\cdot)$. Note that in practice, $\mathcal{E}$ can come from predicted results of user intent classifiers to reduce human annotation costs. In our paper, $\mathcal{E}$ are predicted results of the user intent classifier [21] for MSDialog and Ubuntu Dialog Corpus. For AliMe data, $\mathcal{E}$ is the output of the intention classifier which is a probabilistic distribution over 40 intention scenarios [13].

3.2 Method Overview

In following sections, we describe the proposed method for intent-aware response ranking in information-seeking conversations. The model incorporates intent-aware utterance attention to derive the importance weighting scheme of different context utterances. Given input context utterances and response candidates, we first generate representations from two different perspectives: user intent representations with a trained neural classifier and semantic information encoding with Transformers. Then self-attention and cross-attention matching will be performed over encoded representations from Transformers to extract matching features. These matching features will be weighted by the intent-aware attention mechanism and aggregated into a matching tensor. Finally a two-layer 3D convolutional neural network will distill final representations over the matching tensor and generate the ranking score for the conversation context/ response candidate pair.

3.3 User Intent Taxonomy

We use the MSDialog data that consists of technical support dialogs for Microsoft products developed by Qu et al. [20]. Over 2,000 dialogs with 10,020 utterances were sampled for user intent annotation on Amazon Mechanical Turk.[7] A taxonomy of 12 labels presented in Table 2 were developed to characterize the user intent in information-seeking conversations. The user intent labels include question related labels (e.g., Original Questions, Clarifying Question, etc.), answer related labels (e.g., Potential Answer, Further Details, etc.), feedback related labels (e.g., Positive Feedback, Negative Feedback) and greeting related labels (e.g., Greetings/ Gratitude), which cover most of the user intent types in information-seeking conversations. In addition to MSDialog, we also consider the Ubuntu Dialog Corpus (UDC) [14]. User intent annotation is also performed for randomly sampled 4,063 UDC utterances. More details can be found in Qu et al. [20].

Table 2: Descriptions of user intent taxonomy.

Code	Label	Description
OQ	Original Question	The first question that initiates a QA dialog
RQ	Repeat Question	Questions repeating a previous question
CQ	Clarifying Question	Users or agents ask for clarification
FD	Further Details	Users or agents provide more details
FQ	Follow Up Question	Follow-up questions about relevant issues
IR	Information Request	Agents ask for information from users
PA	Potential Answer	A potential solution to solve the question
PF	Positive Feedback	Positive feedback for working solutions
NF	Negative Feedback	Negative feedback for useless solutions
GG	Greetings/Gratitude	Greet each other or express gratitude
JK	Junk	No useful information in the utterance
O	Others	Utterances that cannot be categorized

3.4 Utterance/ Response Input Representations

Given a response candidate r_i^k and an utterance u_i^t in the context $\mathcal{U}_i$, we represent the utterance/ response pair from two different perspectives: 1) user intent representation with intent classifiers (Section 3.4.1); 2) utterance/ response semantic information encoding with Transformers (Section 3.4.2).

3.4.1 ***User Intent Representation.*** To represent user intent, we adopt the best setting of the neural classifiers CNN-Context-Rep proposed by Qu et al. [21] for user intent classification. Specifically, given sequences of embedding vectors for context utterances and response candidate $\mathbf{E}(u_i^t)$ and $\mathbf{E}(r_i^k)$, convolutional filters with the shape (f, d) are applied to a window of f words to produce a new feature c_i. This operation is applied to every possible window of words in the utterance u_i^t and generates a feature map $\mathbf{c} = \{c_1, c_2, \ldots, c_{n-f+1}\}$. Max pooling is applied to select the most salient feature. The model uses multiple filters with varying window sizes to obtain multiple features in different granularity. These features will be concatenated and flattened into an output tensor, which will be projected into a tensor with shape $(l_t, 1)$ with a fully connected layer. l_t is the number of different user intent labels.[8]

[7] https://www.mturk.com/

[8] In our experiments for MSDialog and UDC, $l_t = 12$ as presented in Section 3.3.

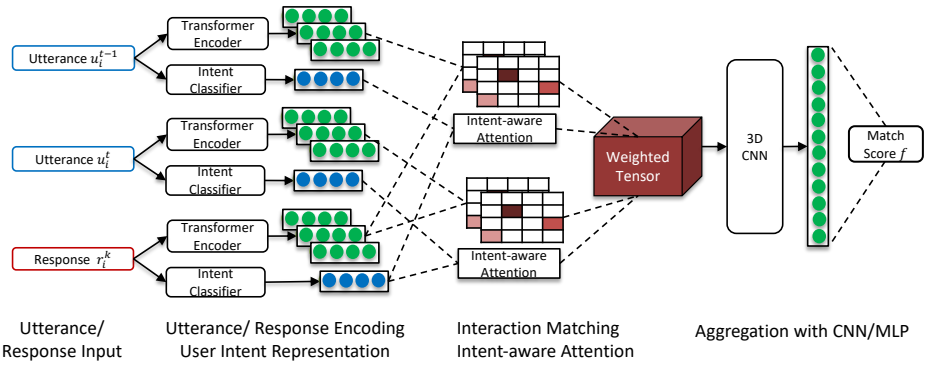

Figure 1: The architecture of the IART model for intent-aware conversation response ranking.

3.4.2 ***Utterance/ Response Encoding and Matching with Transformers.*** We adopt the encoder architecture in Transformers [35] to encode the semantic dependency information in utterance/ response pairs. Transformers are built with Scaled Dot-Product Attention, which performs transformation from a query and a set of key-value pairs to an output. Following the design of Transformers, we also add a feed-forward network FFN with ReLU activation over the layer normalized [1] sum of the output Attention$(\mathcal{Q}, \mathcal{K}, \mathcal{V})$ and the query $\mathcal{Q}$. We refer to this module as the TransformerEncoder module, which will be used as a feature extractor for utterances and responses to capture both the dependency information within words in the same sequence and interactions between words in two different sequences. We consider both self-attention and cross-attention based interaction matching to learn representations for context utterance/ response candidate pairs.

3.5 Intent-aware Attention Mechanism

Given the self-attention/ cross-attention interaction matching matrices for different utterances/ response pairs from a dialog, we first stack them to aggregate them as a 4D matching tensor as follows:

$$\mathcal{B} = \{\mathbb{B}_{t,p,q,l}\}_{l_c \times l_u \times l_r \times (2L+2)} \tag{1}$$

where l_c, l_u, l_r, L are the number of utterance turns in conversation context, number of words in the context utterance, number of words in the response candidate and number of stacked layers in TransformerEncoder. t, p, q, l are indexes along these 4 dimensions of the matching tensor.

We propose an intent-aware attention mechanism to weight matching representations of different utterance turns in a conversation context, so that the model can learn to attend to different utterance turns in context. The motivation is to incorporate a more flexible way to weight and aggregate matching features of different turns with intent-aware attention. Specifically, let $\mathbf{I}_u^t \in \mathbb{R}^{l_t \times 1}, \mathbf{I}_r^k \in \mathbb{R}^{l_t \times 1}$ denote the intent representation vectors defined in Section 3.4.1 for context utterances and response candidates, we design three different types of intent-aware attention as follows:

Dot Product. We concatenate the two intent representation vectors of the utterance/ response pair, and compute the dot product between the parameter $\mathbf{w}$ and the concatenated vector: $\mathcal{A}_t = \text{softmax}(\exp(\mathbf{w}^T[\mathbf{I}_u^t, \mathbf{I}_r^k]))$, where $\mathbf{w} \in \mathbb{R}^{2l_t \times 1}$ is a model parameter.

Bilinear. We compute the bilinear interaction between $\mathbf{I}_u^t$ and $\mathbf{I}_r^k$ and then normalize the result: $\mathcal{A}_t = \text{softmax}(\exp(\mathbf{I}_u^{t\,T}\mathbf{w}\mathbf{I}_r^k))$, where $\mathbf{w} \in \mathbb{R}^{l_t \times l_t}$ is the bilinear interaction matrix to be learned.

Outer Product. We compute the outer product between $\mathbf{I}_u^t$ and $\mathbf{I}_r^k$ and then flatten the result matrix to a feature vector. Finally we project this feature vector into an attention score with a fully connected layer and a softmax function: $\mathcal{A}_t = \text{softmax}(\exp(\mathbf{w}^T \cdot \text{flat}(\mathbf{I}_u^t \otimes \mathbf{I}_r^{k\,T})))$, where flat and $\otimes$ denote the flatten layer which transforms a matrix with shape $(l_t \times l_t)$ into a vector with shape $(l_t^2 \times 1)$ and outer product operation. $\mathbf{w} \in \mathbb{R}^{l_t^2 \times 1}$ is a model parameter.

Note that the normalization in the softmax function is performed over all utterance turns within a conversation context. Thus the result $\mathcal{A}_t$ is the attention weight corresponding to the t-th utterance turn in a conversation context. We also add masks over the padded utterance turns to avoid introducing noise matching feature representations. With the computed attention weights over context utterance turns, we can scale the 4D matching tensor to generate a weighted matching tensor:

$$\widehat{\mathcal{B}} = \{\mathbb{B}_{t,p,q,l} \cdot \mathcal{A}_t\}_{l_c \times l_u \times l_r \times (2L+2)} \tag{2}$$

Finally IART adopts a two layer 3D convolution neural network (CNN)[9] to extract important matching features from this weighted matching tensor $\widehat{\mathcal{B}}$. A 3D CNN requires 5D input and filter tensors, as we can add one more input dimension corresponding to the batched training examples over the 4D weighted matching tensor. We compute the final matching score $f(\mathcal{U}_i, r_i^k)$ with a MLP over the flattened output of the 3D CNN. For model training, we compute the cross-entropy loss between the predicted matching scores $f(\mathcal{U}_i, r_i^k)$ and the ground truth matching labels. The parameters of IART are optimized using back-propagation with *Adam* algorithm [12].

4 EXPERIMENTS

4.1 Data Set Description

We evaluated our method with three data sets: Ubuntu Dialog Corpus (UDC), MSDialog, and a commercial data collected from the AliMe assistant at Alibaba group. The statistics of different experimental data sets are shown in Table 3. The Ubuntu Dialog Corpus (UDC) [14] contains multi-turn technical support conversation chat logs on the Ubuntu system. We used the data copy shared by Xu et al.

[9] https://www.tensorflow.org/api_docs/python/tf/nn/conv3d

Table 3: The statistics of experimental datasets, where C denotes context and R denotes response. # Cand. per C denotes the number of candidate responses per context. Note that we did not filter any stop words or words with low frequency for computing the average length of contexts or responses.

Data	UDC			MSDialog			AliMe		
Items	Train	Valid	Test	Train	Valid	Test	Train	Valid	Test
# C-R pairs	1000k	500k	500k	173k	37k	35k	51k	6k	6k
# Cand. per C	2	10	10	10	10	10	15	15	15
# + Cand. per C	1	1	1	1	1	1	2.9	2.8	2.9
Avg # turns per C	10.1	10.1	10.1	5.0	4.9	4.4	2.4	2.1	2.2
Avg # words per C	116.8	116.3	116.7	451.3	435.2	375.1	38.3	35.3	34.2
Avg # words per R	22.2	22.2	22.3	106.1	107.4	105.5	4.9	4.7	4.6

[39]. It is also used in several previous related works [38, 45, 51].[10] MSDialog is released from previous related work by Qu et al. [20]. It contains QA dialogs on various Microsoft products crawled from the Microsoft Answer community. For the AliMe dataset, it contains the chat logs between customers and the AliMe assistant bot at Alibaba. For each query of the dataset, it contains several response candidates from the chatbot engine which are labeled by a business analyst. The details about these data sets are in Yang et al. [45]. Note that the proposed model is more on response re-ranking instead of response retrieval in one step.

4.2 Experimental Setup

4.2.1 ***Baselines.*** We consider different baselines as follows [11]:

Traditional retrieval models: these methods treat the dialog context as the query to retrieve response candidates for response selection. We consider BM25 [25] as the retrieval model. We also consider BM25-PRF [45], which matches conversation context with the expanded responses using BM25.

Neural ranking models: we consider several representative neural ranking models: MV-LSTM [37], DRMM [6] and Duet [17]. We also consider models based on Deep Matching Networks (DMN) with external knowledge [45], which incorporate external knowledge with pseudo-relevance feedback (DMN-PRF) and QA correspondence knowledge distillation (DMN-KD).

Deep Attention Matching Network (DAM) [51]: DAM is a strong baseline method for response ranking in multi-turn conversations with open source code released[12] until this paper. DAM also represents and matches a response with its multi-turn context using dependency information learned by Transformers. It does not explicitly model user intent in conversations.

For evaluation metrics, we adopted mean average precision (MAP) and $R_n@k$ which is the recall at top k ranked responses from n available candidates for a given conversation context following previous related works [14, 38, 45, 51].

4.2.2 ***Parameter Settings and Implementation Details.*** All models are implemented with TensorFlow[13] and the MatchZoo[14] toolkit. Hyper-parameters are tuned with the validation data. For the hyper-parameter settings of IART, we set the size of the convolution and pooling kernels as $(3, 3, 3)$. The number of stacked Transformers layers is set as 5 for UDC and 4 for MSDialog. The batch size is 128 for UDC and 32 for MSDialog. All models are trained on a single Nvidia Titan X GPU. Learning rate is initialized as 1e-3 with exponential decay during training process. The decay steps and decay rate are set as 400 and 0.9. The maximum utterance length is 50 for UDC and 200 for MSDialog. The maximum number of context utterance turns is set as 9 for UDC and 6 for MSDialog. We padded zeros if the number of utterance turns in a context is less than the maximum number of utterance turns. For user intent labels, there are 12 different types for UDC/ MSDialog, and 40 different types for AliMe data. For the word embeddings, we trained word embeddings with the Word2Vec tool [16] with the CBOW model using our training data following previous work [38, 51]. The max skip length between words and the number of negative examples is set as 10 and 25. The dimension of word embeddings is 200. Word embeddings will be initialized by these pre-trained word vectors and updated during the training process.

4.3 Evaluation Results

We present evaluation results over different methods in Table 4. We summarize our observations as follows: (1) On MSDialog, all three variations of IART with dot, outer product and bilinear based intent-aware attention mechanism show significant improvements over all baseline methods, including the recently proposed strong baseline method DAM. On UDC, IART with three different intent-aware attention mechanisms also show improvements under all metrics except for R10@5. With the comparison between the results of DAM and IART, we can find that incorporating user intent modeling and intent-aware attention weighting scheme can help improve the response ranking performance. (2) If we compare three variations of IART, we can find that the bilinear based intent-aware attention mechanism works better for MSDialog and outer product based intent-aware attention mechanism works better for UDC. The overall performances of these three model variations are close to each other. Overall our proposed model IART shows larger performance improvements on MSDialog. One possible reason is that the intent classifier on MSDialog is more accurate due to the larger annotated training data of MSDialog for user intent prediction and more formal language used in MSDialog, as shown in evaluation results by Qu et al. [21]. (3) On AliMe data, all three variations of IART also show comparable or better results than all baseline methods including the strong baseline DAM. These results on real product data further verify the effectiveness of our proposed methods.

4.4 Case Study and User Intent Visualization

We perform a case study in Table 5 on the top ranked responses by different methods including the best baseline DAM and our proposed model IART with bilinear based intent-aware attention mechanism. We show the conversation context utterances and top-1 ranked response by each method. In this example, IART produced the correct top ranked response. We visualized the learned user intent representation of context utterances and returned top-1 ranked response by DAM and IART in Figure 2. The predicted user intent of conversation utterances is **[OQ] → [IR] → [PA] → [IR] → [FD/**

[10]The data can be downloaded from https://www.dropbox.com/s/2fdn26rj6h9bpvl/ubuntu%20data.zip?dl=0

[11]Note that the experimental setup where we compare our method with baselines without user intent modeling is reasonable. User intent modeling should be only added into the treatment instead of baselines for controlled experimental comparison to show the effectiveness of the incorporation of user intent.

[12]https://github.com/baidu/Dialogue/tree/master/DAM

[13]https://www.tensorflow.org/

[14]https://github.com/NTMC-Community/MatchZoo

Table 4: Comparison of different models over Ubuntu Dialog Corpus (UDC), MSDialog, and AliMe data sets. Numbers in bold font mean the result is better compared with the best baseline DAM. † and ‡ means statistically significant difference over the best baseline DAM with $p < 0.1$ and $p < 0.05$ measured by the Student's paired t-test respectively.

Data	UDC				MSDialog				AliMe			
Methods	R10@1	R10@2	R10@5	MAP	R10@1	R10@2	R10@5	MAP	R10@1	R10@2	R10@5	MAP
BM25 [25]	0.5138	0.6439	0.8206	0.6504	0.2626	0.3933	0.6329	0.4387	0.2371	0.4204	0.6407	0.6392
BM25-PRF [45]	0.5289	0.6554	0.8292	0.6620	0.2652	0.3970	0.6423	0.4419	0.2454	0.4209	0.6510	0.6412
MV-LSTM [37]	0.4973	0.6733	0.8936	0.6611	0.2768	0.5000	0.8516	0.5059	0.2480	0.4105	0.7017	0.7734
DRMM [6]	0.5287	0.6773	0.8776	0.6749	0.3507	0.5854	0.9003	0.5704	0.2212	0.3616	0.6575	0.7165
Duet [17]	0.4756	0.5592	0.8272	0.5692	0.2934	0.5046	0.8481	0.5158	0.2433	0.4088	0.6870	0.7651
DMN-KD [45]	0.6443	0.7841	0.9351	0.7655	0.4908	0.7089	0.9304	0.6728	0.3596	0.5122	0.7631	0.8323
DMN-PRF [45]	0.6552	0.7893	0.9343	0.7719	0.5021	0.7122	0.9356	0.6792	0.3601	0.5323	0.7701	0.8435
DAM [51]	0.7686	0.8739	0.9697	0.8527	0.7012	0.8527	0.9715	0.8150	0.3819	0.5567	0.7717	0.8452
IART*Dot*	**0.7703**	**0.8746**	0.9688	**0.8535**	**0.7234**‡	**0.8650**‡	**0.9772**‡	**0.8300**‡	**0.3821**	0.5547	**0.7802**†	**0.8454**
IART*Outerproduct*	**0.7717**‡	**0.8766**‡	0.9691	**0.8548**‡	**0.7212**‡	**0.8664**‡	**0.9749**	**0.8289**‡	**0.3901**‡	**0.5649**‡	**0.7812**†	**0.8493**†
IART*Bilinear*	**0.7713**‡	**0.8747**	0.9688	**0.8542**†	**0.7317**‡	**0.8752**‡	**0.9792**‡	**0.8364**‡	**0.3892**†	**0.5592**†	**0.7801**†	**0.8471**

Table 5: A case study and examples of Top-1 ranked responses by different methods. y_i^k means the label of a response candidate.

Context	**[User]** Hi, I have the new Outlook which updated a few days ago. I cannot find how to add senders to my blocked senders list manually. How do I do this on the new Outlook? Thanks **[Agent]** Hi, There are different ways to block senders on Outlook depending on the version of Outlook that you are using. May we know what version of Outlook are you using? **[User]** Hi, I'm using the desktop website beta version. Thanks. **[Agent]** Desktop Website beta version? Are you referring to the Outlook Web App or the Windows mail? **[User]** I go to Outlook.com and sign in on there.	
Context Intent	**[OQ] → [IR] → [PA] → [IR] → [FD/ OQ]**	
Method	y_i^k	Top-1 Ranked Response
DAM	0	Thanks for the reply. Some email domain needs to be manually added to Outlook. However, it's good to know that the issue is resolved from your end. Should you need further assistance in the future, please do let us know. **[PF]**
IART*Bilinear*	1	In Outlook Web App ..... to manually block an email address, follow these steps: ..... Let us know how things go. **[PA]**

OQ]. The agent performed "Information Request (IR)" to confirm whether it is the Outlook Web app or the Windows desktop app. The user confirmed "Further Details (FD)" that the problem was related to the Outlook Web app (Outlook.com). Given such a user intent pattern in the conversation context, a reasonable response can be with intent "Potential Answers (PA)" on providing potential solutions to the user's question, which is captured by IART due to the integration of user intent modeling. The DAM model, without user intent modeling, failed in such cases and selected a response candidate with "Positive Feedback (PF)" intent. The response returned by DAM assumed that "the issue is resolved", but actually the user was expecting an answer to her unsolved technical problem. This gives an example and interpretation of why user intent modeling can be helpful for response ranking in conversations.

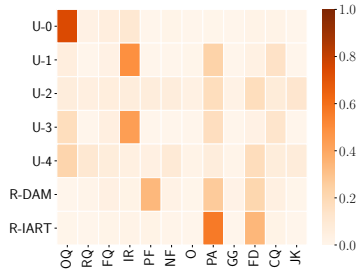

Figure 2: Visualization of learned user intent representation of context utterances and returned top-1 ranked response by DAM and IART from the case study in Table 5. U-0 to U-4 denotes the 0-th turn to the 4-th utterance turn in the context. R-DAM and R-IART denotes the top-1 ranked response returned by DAM and IART respectively. Darker spots mean higher predicted probabilities.

5 CONCLUSIONS

In this paper, we analyze user intent in information-seeking conversations and propose an intent-aware neural ranking model with Transformers. We first define and characterize different user intent types, and then propose an intent-aware neural ranking model for response retrieval which incorporates intent-aware utterance attention to derive the importance weighting scheme of different utterances to improve conversation history understanding. Our proposed methods outperform all baseline methods on three different data sets including both standard benchmarks and commercial data. We also perform case studies and analysis of the learned user intent with their impact on response ranking in information-seeking conversations to provide insights.

ACKNOWLEDGMENTS

This work was supported in part by the Center for Intelligent Information Retrieval, in part by NSF IIS-1715095, and in part by China Postdoctoral Science Foundation (No. 2019M652038). Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

REFERENCES

[1] J. Ba, R. Kiros, and G. E. Hinton. 2016. Layer Normalization. *CoRR* (2016).

[2] S. Bhatia, P. Biyani, and P. Mitra. 2014. Summarizing Online Forum Discussions-Can Dialog Acts of Individual Messages Help?. In *EMNLP '14*.

[3] J. Devlin, M. Chang, K. Lee, and K. Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018).

[4] B. Dhingra, L. Li, X. Li, J. Gao, Y. Chen, F. Ahmed, and L. Deng. 2017. Towards End-to-End Reinforcement Learning of Dialogue Agents for Information Access. In *ACL'17*.

[5] J. Gao, M. Galley, and L. Li. 2018. Neural Approaches to Conversational AI. *CoRR* abs/1809.08267 (2018).

[6] J. Guo, Y. Fan, Q. Ai, and W. B. Croft. 2016. A Deep Relevance Matching Model for Ad-hoc Retrieval. In *CIKM '16*.

[7] J. Guo, Y. Fan, L. Pang, L. Yang, Q. Ai, H. Zamani, C. Wu, W. B. Croft, and X. Cheng. 2019. A Deep Look into Neural Ranking Models for Information Retrieval. *CoRR* abs/1903.06902 (2019).

[8] C. Hori, J. Perez, R. Higashinaka, T. Hori, Y. Boureau, M. Inaba, Y. Tsunomori, T. Takahashi, K. Yoshino, and S. Kim. 2019. Overview of the sixth dialog system technology challenge: DSTC6. *Computer Speech & Language* 55 (2019).

[9] B. Hu, Z. Lu, H. Li, and Q. Chen. 2014. Convolutional Neural Network Architectures for Matching Natural Language Sentences. In *NIPS '14*.

[10] P. Huang, X. He, J. Gao, L. Deng, A. Acero, and L. P. Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *CIKM '13*.

[11] Z. Ji, Z. Lu, and H. Li. 2014. An Information Retrieval Approach to Short Text Conversation. *CoRR* abs/1408.6988 (2014).

[12] D. P. Kingma and J. Ba. 2014. Adam: A Method for Stochastic Optimization. *CoRR* (2014).

[13] F. Li, M. Qiu, H. Chen, X. Wang, X. Gao, J. Huang, J. Ren, Z. Zhao, W. Zhao, L. Wang, and G. Jin. 2017. AliMe Assist: An Intelligent Assistant for Creating an Innovative E-commerce Experience. In *CIKM '17*.

[14] R. Lowe, N. Pow, I. Serban, and J. Pineau. 2015. The Ubuntu Dialogue Corpus: A Large Dataset for Research in Unstructured Multi-Turn Dialogue Systems. *CoRR* abs/1506.08909 (2015).

[15] D. Madan and S. Joshi. 2017. Finding Dominant User Utterances And System Responses in Conversations. *CoRR* abs/1710.10609 (2017).

[16] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *NIPS '13*.

[17] B. Mitra, F. Diaz, and N. Craswell. 2017. Learning to Match Using Local and Distributed Representations of Text for Web Search. In *WWW '17*.

[18] L. Pang, Y. Lan, J. Guo, J. Xu, S. Wan, and X. Cheng. 2016. Text Matching as Image Recognition. In *AAAI '16*.

[19] M. Qiu, F. Li, S. Wang, X. Gao, Y. Chen, W. Zhao, H. Chen, J. Huang, and W. Chu. 2017. AliMe Chat: A Sequence to Sequence and Rerank based Chatbot Engine. In *ACL '17*.

[20] C. Qu, L. Yang, W. B. Croft, J. R. Trippas, Y. Zhang, and M. Qiu. 2018. Analyzing and Characterizing User Intent in Information-seeking Conversations. In *SIGIR '18*. 989–992.

[21] C. Qu, L. Yang, W. B. Croft, Y. Zhang, J. R. Trippas, and M. Qiu. 2019. User Intent Prediction in Information-seeking Conversations. In *CHIIR '19*. 25–33.

[22] C. Qu, L. Yang, M. Qiu, Y. Zhang, C. Chen, W. B. Croft, and M. Iyyer. 2019. Attentive History Selection for Conversational Question Answering. In *CIKM '19*.

[23] F. Radlinski and N. Craswell. 2017. A theoretical framework for conversational search. In *CHIIR '17*.

[24] A. Ritter, C. Cherry, and W. B. Dolan. 2011. Data-Driven Response Generation in Social Media. In *ACL '11*.

[25] S. Robertson and S. Walker. 1994. Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval. In *SIGIR '94*.

[26] L. Shang, Z. Lu, and H. Li. 2015. Neural Responding Machine for Short-Text Conversation. In *ACL '15*.

[27] S. Shiga, H. Joho, R. Blanco, J. R. Trippas, and M. Sanderson. 2017. Modelling Information Needs in Collaborative Search Conversations. In *SIGIR '17*. 715–724.

[28] Y. Song, C. Li, J. Nie, M. Zhang, D. Zhao, and R. Yan. 2018. An Ensemble of Retrieval-based and Generation-based Human-computer Conversation Systems. In *IJCAI '18*.

[29] A. Sordoni, M. Galley, M. Auli, C. Brockett, Y. Ji, M. Mitchell, J. Nie, J. Gao, and B. Dolan. 2015. A Neural Network Approach to Context-Sensitive Generation of Conversational Responses. In *NAACL '15*.

[30] D. Spina, J. R Trippas, L. Cavedon, and M. Sanderson. 2017. Extracting audio summaries to support effective spoken document search. *JAIST '17* 68, 9 (2017).

[31] A. Stolcke, N. Coccaro, R. Bates, P. Taylor, C. Van Ess-Dykema, K. Ries, E. Shriberg, D. Jurafsky, R. Martin, and M. Meteer. 2000. Dialogue Act Modeling for Automatic Tagging and Recognition of Conversational Speech. *Comput. Linguist.* 26, 3 (Sept. 2000), 339–373.

[32] C. Tao, W. Wu, C. Xu, W. Hu, D. Zhao, and R. Yan. 2019. Multi-Representation Fusion Network for Multi-Turn Response Selection in Retrieval-Based Chatbots. In *WSDM '19*.

[33] P. Thomas, D. McDu, M. Czerwinski, and N. Craswell. 2017. MISC: A data set of information-seeking conversations. In *CAIR '17*.

[34] J. Trippas, D. Spina, M. Sanderson, and L. Cavedon. 2015. Towards understanding the impact of length in web search result summaries over a speech-only communication channel. In *SIGIR '15*.

[35] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł . Kaiser, and I. Polosukhin. 2017. Attention is All You Need. In *NIPS '17*.

[36] O. Vinyals and Q. V. Le. 2015. A Neural Conversational Model. *CoRR* abs/1506.05869 (2015).

[37] S. Wan, Y. Lan, J. Guo, J. Xu, L. Pang, and X. Cheng. 2016. A Deep Architecture for Semantic Matching with Multiple Positional Sentence Representations. In *AAAI '16*.

[38] Y. Wu, W. Wu, C. Xing, M. Zhou, and Z. Li. 2017. Sequential Matching Network: A New Architecture for Multi-turn Response Selection in Retrieval-Based Chatbots. In *ACL '17*.

[39] Z. Xu, B. Liu, B. Wang, C. Sun, and X. Wang. 2016. Incorporating Loose-Structured Knowledge into LSTM with Recall Gate for Conversation Modeling. *CoRR* (2016).

[40] R. Yan, Y. Song, and H. Wu. 2016. Learning to Respond with Deep Neural Networks for Retrieval-Based Human-Computer Conversation System. In *SIGIR*.

[41] R. Yan, Y. Song, X. Zhou, and H. Wu. 2016. "Shall I Be Your Chat Companion?": Towards an Online Human-Computer Conversation System. In *CIKM '16*.

[42] R. Yan, D. Zhao, and W. E. 2017. Joint Learning of Response Ranking and Next Utterance Suggestion in Human-Computer Conversation System. In *SIGIR '17*.

[43] L. Yang, Q. Ai, J. Guo, and W. B. Croft. 2016. aNMM: Ranking Short Answer Texts with Attention-Based Neural Matching Model. In *CIKM '16*.

[44] L. Yang, J. Hu, M. Qiu, C. Qu, J. Gao, W. B. Croft, X. Liu, Y. Shen, and J. Liu. 2019. A Hybrid Retrieval-Generation Neural Conversation Model. *CIKM'19*.

[45] L. Yang, M. Qiu, C. Qu, J. Guo, Y. Zhang, W. B. Croft, J. Huang, and H. Chen. 2018. Response Ranking with Deep Matching Networks and External Knowledge in Information-seeking Conversation Systems. In *SIGIR '18*.

[46] L. Yang, H. Zamani, Y. Zhang, J. Guo, and W. B. Croft. 2017. Neural Matching Models for Question Retrieval and Next Question Prediction in Conversation. *CoRR* abs/1707.05409 (2017).

[47] J. Yu, M. Qiu, J. Jiang, J. Huang, S. Song, W. Chu, and H. Chen. 2018. Modelling Domain Relationships for Transfer Learning on Retrieval-based Question Answering Systems in E-commerce. *WSDM '18*.

[48] X. Zhang and H. Wang. 2016. A Joint Model of Intent Determination and Slot Filling for Spoken Language Understanding. In *IJCAI '16*.

[49] Y. Zhang, X. Chen, Q. Ai, L. Yang, and W. B. Croft. 2018. Towards Conversational Search and Recommendation: System Ask, User Respond. In *CIKM '18*.

[50] X. Zhou, D. Dong, H. Wu, S. Zhao, D. Yu, H. Tian, X. Liu, and R. Yan. 2016. Multi-view Response Selection for Human-Computer Conversation. In *EMNLP*.

[51] X. Zhou, L. Li, D. Dong, Y. Liu, Y. Chen, W. X. Zhao, D. Yu, and H. Wu. 2018. Multi-Turn Response Selection for Chatbots with Deep Attention Matching Network. In *ACL '18*.