Toward Low-Cost Automated Evaluation Metrics for Internet of Things Dialogues

Kallirroi Georgila, Carla Gordon, Hyungtak Choi, Jill Boberg, Heesik Jeon and David Traum

Abstract We analyze a corpus of system-user dialogues in the Internet of Things domain. Our corpus is automatically, semi-automatically, and manually annotated with a variety of features both on the utterance level and the full dialogue level. The corpus also includes human ratings of dialogue quality collected via crowdsourcing. We calculate correlations between features and human ratings to identify which features are highly associated with human perceptions about dialogue quality in this domain. We also perform linear regression and derive a variety of dialogue quality evaluation functions. These evaluation functions are then applied to a held-out portion of our corpus, and are shown to be highly predictive of human ratings and outperform standard reward-based evaluation functions.

1 Introduction

The Internet of Things (IoT) is the network of physical devices (e.g., home appliances, health monitoring devices, etc.) connected to the Internet. IoT devices can be controlled each one separately by individual apps, or all together via an integrated

K. Georgila (✉) · C. Gordon · J. Boberg · D. Traum
USC Institute for Creative Technologies, Playa Vista, USA
e-mail: kgeorgila@ict.usc.edu

C. Gordon
e-mail: cgordon@ict.usc.edu

J. Boberg
e-mail: boberg@ict.usc.edu

D. Traum
e-mail: traum@ict.usc.edu

H. Choi · H. Jeon
Samsung Electronics Co., Ltd., Seocho-gu, Seoul, Korea
e-mail: ht777.choi@samsung.com

H. Jeon
e-mail: heesik.jeon@samsung.com

L. F. D'Haro et al. (eds.), *9th International Workshop on Spoken Dialogue System Technology*, Lecture Notes in Electrical Engineering 579,
https://doi.org/10.1007/978-981-13-9443-0_14

(1) User: Play music in 10 min.
(2) System: In which room shall I play music?
(3) User: Bedroom please.
(4) System: As you wish. I will play music in the bedroom in 10 min.
(5) User: Thanks. Please turn on the air-conditioner there too.
The user wants the air-conditioner to be turned on in 10 min but the user request is ambiguous and misenterpreted by the system.
(6) System: Okay, the air-conditioner in the bedroom is now on.
(7) User: Wait, I want the air-conditioner on in 10 min, not now.
(8) System: Oops my mistake. The air-conditioner in the bedroom is now off. I will turn it on in 10 min. Anything else?
(9) User: Yes, turn on the light.
The user has switched to a different task in a new room but the system thinks that the new command is related to the previous context (bedroom).
(10) System: I've turned on the light in the bedroom.
(11) User: Not in the bedroom, in the kitchen. I'm going to the kitchen now.
(12) System: I've turned on the light in the kitchen.
(13) User: Good. Move the TV up one channel.
(14) System: To be sure, you'd like me to move up one channel the TV in the kitchen, correct?
(15) User: Yes, this is exactly what I'd like you to do.
(16) System: Moving the kitchen TV up one channel.
(17) User: No, I mean down one channel.
(18) System: Got it. Moving the TV down one channel.
(19) User: Hmm. There is nothing interesting on.
(20) System: I totally sympathize. The quality of TV has gone downhill.
(21) User: Oh well, just turn it off.
There is a speech recognition error: "on" is recognized instead of "off".
(22) System: Sorry I can't do that. The TV is already on.
(23) User: I said turn the TV off.
(24) System: Done.
(25) User: Thanks.
(26) System: You're very welcome. I'm always here to help.

Fig. 1 Example system-user dialogue in the IoT domain

app. Alternatively, IoT devices can be controlled by a smart assistant via human-system dialogue interaction [11], which is the approach that we follow in this paper.

Building dialogue systems for the IoT domain is an area of research that has only recently emerged, and which poses many challenges. Figure 1 shows an example dialogue between a human user and a future smart home assistant that controls the IoT devices. As we can see, the human-system interaction can be quite complex. An intelligent dialogue system should be able to understand the user's requests and if needed follow up with smart clarification questions. For example, in utterance (2) the system behaves intelligently by asking the user a clarification question about which room it should play music in. Dialogue context is another factor that can affect dialogue quality, both regarding interpreting user requests and responding appropriately. For example, in utterances (5)–(6) the system misinterprets the user's ambiguous request, and instead of scheduling to turn on the air-conditioner in 10 min the air-conditioner is turned on immediately. A better system response would have been a clarification question “Shall I turn on the air-conditioner now or in 10 min?”. Furthermore, in utterance (14) the system’s confirmation request about the TV in the kitchen makes sense in this dialogue context, but it would sound weird in a different context. Regardless of the dialogue context, speech recognition errors can be another source of noise in the interaction as it is the case for all spoken dialogue systems, e.g., in utterances (21)–(22).

The quality of IoT dialogues may also be influenced by potential side-effects of some actions. Consider utterances (9)–(12). The result of this system-user exchange is that both lights are now on (in the bedroom and the kitchen), even though the user wanted only the light in the kitchen to be on. Unless the bedroom light is turned off, this dialogue has a side-effect as well as the main effect on the light in the kitchen. This side effect might be undesirable, or desirable, or neutral. Furthermore, timing can be an important issue. For example, the system’s action in (18) depends on what the system did between (16)–(17). If the system already moved the TV up one channel as described in (16), then in (18) the system must move the TV down two channels. However, if (17) came before the system had a chance to change the channel, it would only need to move down one channel. Thus, not only the linguistic context is important, but also the context of device state and actions performed.

Given that users are bound to interact with this smart home assistant on a regular basis, it may not be enough that the system performs its tasks and behaves in a rational manner. People may have preferences about the system’s personality and expect it to be polite, nice, and adapt to their mood, needs, and desires. For example, some users may expect the system to sympathize with their opinions as in utterance (20) or be chatty as in utterances (8) and (26). Also, some users may prefer the system to be explicit about what it has done or is planning to do (e.g., in utterance (4) “I will play music in the bedroom in 10 min.”) whereas other users may appreciate brevity (e.g., in utterance (24) “Done.”).

As we can see from the above examples, there are similarities between dialogues in the IoT domain and multi-domain task-oriented dialogues. What makes the IoT domain particularly interesting is that the devices involved can work together in a synergistic way rather than being totally separate domains or tasks. For example, users may have a routine when they leave the house in the morning, e.g., locking the windows and doors, turning off the coffee machine and the toaster, etc. Or when they come back in the evening they may want the smart home assistant to create a relaxing atmosphere in the living room, e.g., by playing classical music at an appropriate speaker volume level, dimming the lights, etc.

The example dialogue of Fig. 1 illustrates that it is not clear at all what constitutes a successful system-user dialogue in the IoT domain. Our goal in this paper is to take the first steps toward developing low-cost evaluation metrics that are predictive of user perceptions about dialogue quality in this domain. By low-cost we mean that these metrics should be based on automatically extracted features or, if this is not possible, rely on simple annotations that can be performed by non-experts in linguistics or dialogue. Our contribution is two-fold. First, we provide insights about unique challenges in developing dialogue quality evaluation metrics in the IoT domain. We also present a novel annotation scheme for annotating a variety of both social and task-related aspects in this domain. Second, using a methodology similar to the PARADISE evaluation framework [20], we develop novel evaluation functions for the IoT domain. When applied to a held-out portion of our dialogue corpus, these evaluation functions are shown to be highly predictive of human ratings of dialogue quality, and outperform standard reward-based evaluation functions.

2 Related Work

Over the years, a variety of evaluation frameworks and metrics have been proposed for measuring the quality of human-system dialogue interaction, mainly for task-oriented dialogue systems [8]. Some metrics are subjective (e.g., user satisfaction, perceived task completion, etc.), and others are objective (e.g., word error rate, dialogue length, etc.). Objective measures can be calculated from the interaction logs while subjective assessments can be collected via surveys and questionnaires [10, 15].

PARADISE [20] is perhaps the most well-known framework for evaluating dialogue systems, and an attempt to automate the evaluation process. PARADISE seeks to optimize a desired quality such as user satisfaction by formulating it as a linear combination of a variety of metrics, such as task success and dialogue cost (e.g., dialogue length). The advantage of this method is that once a desired quality has been formulated as a realistic evaluation function, it can be optimized by controlling the factors that affect it. In the example above, user satisfaction can be optimized by increasing task success and minimizing dialogue length.

Reinforcement learning (RL) has become the standard technique for learning dialogue policies from data or simulated users (SUs). In RL, a typical reward function is for the system to earn a number of points for a fully successful dialogue (or for partial success, e.g., when some of the requested information is provided or confirmed), and subtract a penalty per system turn to ensure that the learned dialogue policies will not favor lengthy and tedious dialogues [9]. Note however that longer dialogue lengths are not necessarily indicative of poor dialogue quality but depending on the task they may actually indicate user engagement and satisfaction [2].

A variety of metrics have been employed for measuring the quality of SUs used for training and evaluating dialogue policies. The idea is that the action generated by the SU is compared against the user action in a human-human or human-system reference corpus (in the same dialogue context), and measures such as precision, recall, accuracy, and perplexity are used [5, 6, 18]. However, these metrics can be problematic because if a SU action is not the same as the user action in the reference corpus, this does not necessarily mean that it is a poor action. Also, once a user or system response deviates from the corresponding action in the reference corpus, the remaining dialogue will unfold in an entirely different way than the fixed dialogue in the reference corpus, which will make further comparisons meaningless.

In non-task-oriented dialogue systems (e.g., chatbots) developing robust evaluation metrics can be even harder than for task-oriented dialogue. Here it is not clear what success means and thus task-specific objective metrics are not appropriate. Instead subjective evaluations for appropriateness of responses can be much more meaningful, which has led to the development of coding schemes for response appropriateness and scoring in such cases [17, 19]. Another approach is to evaluate dialogue systems in a semi-formal manner, using human judges to rate the coherence of a conversational agent and correlating these judgements with measures extracted from within the system [1]. Dialogue coherence can also be formulated as an information ordering task [4, 16]. In [16], a binary classifier was built for distinguishing between coherent and incoherent dialogues using local transition patterns that span over adjacent dialogue turns encoding lexical and semantic information. In [4], random permutations of dialogue segments were generated and rated by human judges on a Likert scale in terms of coherence. It was found that Kendall's τ correlated well with human judgements.

Word-overlap similarity metrics such as BLEU, METEOR, and ROUGE (originally employed in machine translation and summarization) are currently widely used for measuring chatbot dialogue quality. However, BLEU, METEOR, and ROUGE suffer from the same problems as the aforementioned SU evaluation metrics. In fact it has been shown that BLEU, METEOR, and ROUGE do not correlate well with human judgements of dialogue quality [13]. Note that BLEU has also been used for evaluating SUs [12]. Discriminative BLEU, a variation of BLEU where reference strings are scored for quality by human raters, was found to correlate better with human judgements than standard BLEU [3]. To address the issues with BLEU, METEOR, and ROUGE, next utterance classification was introduced as a method for evaluating chatbots [14], but the proposed metric recall@k is very similar to the recall metric previously used for evaluating SUs, and consequently has the same limitations. Recently, topic-based metrics for chatbot evaluation (topic breadth and topic depth) were found to correlate well with human judgements [7].

3 Our Dialogue Corpus

Our corpus currently consists of approximately 6200 dialogues in the IoT domain between a smart home assistant and a user. These dialogues were written by several linguists, and obviously they are not as realistic as the dialogues we would get by having humans interact with a real system or in a Wizard of Oz setting, which is something we plan to do for future work. However, despite this limitation, our dialogues are designed to capture a variety of phenomena that we would encounter in real human-system dialogues, including speech recognition errors, misunderstandings, clarification requests, timing, context and scheduling issues, and generally all the phenomena that we discussed in the dialogue example of Fig. 1. For this reason we consider our corpus to be a valuable resource for bootstrapping our research in the IoT dialogue domain.

Our corpus includes system-user conversations regarding one home appliance at a time (washer, speaker, bulb, TV, air-conditioner) or multiple devices at the same time (e.g., the air-conditioner and the TV). Also, in our IoT ontology we have multiple

Table 1 Statistics of our corpus (232 dialogues)

Dialogue feature	Mean	Standard deviation
Number of tasks per dialogue	1.41	0.68
Number of system turns per dialogue	2.80	0.98
Number of user turns per dialogue	2.80	0.98
Number of all turns per dialogue	5.60	1.96
Number of system words per dialogue	14.08	8.01
Number of user words per dialogue	15.32	6.00
Number of all words per dialogue	29.40	12.15
Average number of system words per utterance	5.14	2.61
Average number of user words per utterance	5.69	1.88
Average number of all words per utterance	5.41	1.81

devices of the same type (e.g., there can be a TV in the bedroom, a TV in the living room, and a TV in the guest room). Our corpus contains dialogues where the system and the user need to make sure that they are both referring to the same device, which in turn leads to very realistic and complex system-user interactions.

For the experiments presented in this paper we selected 232 dialogues from 6 categories (washer, speaker, bulb, TV, air-conditioner, multiple devices in the same dialogue), taking care to include as many realistic dialogue phenomena as possible. For each category we have 2 sub-categories: dialogues without any misunderstandings and dialogues with misunderstandings. Thus in total we have 12 dialogue categories with about 20 dialogues per category.

Our corpus of 232 dialogues has been annotated manually, automatically, and semi-automatically with a variety of features both on the utterance level and the full dialogue level. More specifically, we have automatically calculated the following features: number of system and user turns per dialogue, number of total words from system and user per dialogue, average number of words per system and user utterance in a dialogue, and number of occurrences of specific words and expressions, e.g., “yes/yeah/yep/yup”, “no/nope”, “ok/okay”, “alright/all right”, “done”, “system”, “thanks/thank you”, “good/great”, “not at all”, “sure”, “sure thing”, “got it”, “no problem”, “sorry/apologize/apologies”, “naturally”, “obviously”, etc. Table 1 shows statistics of our corpus of 232 dialogues.

We have also developed a novel annotation scheme and performed the following annotations on the utterance level for both system and user utterances:

- System utterances:
  - Assess action:
    A-something (system does something: “I’m connecting the speaker.”)
    A-nothing (system does nothing: “Which speaker?”)
    A-valid (system does requested thing: “U: Turn on the kitchen light. S: I’m turning on the kitchen light.”)

    A-invalid (system does not do requested thing: “U: Turn on the kitchen light. S: I’m turning on the porch light.”)
  - Describe current understanding:
    CU-confirm (confirm request before doing: “Shall I turn on the light?”)
    CU-lack (describe lack of understanding: “Sorry I don’t understand.”)
  - Action acknowledge:
    AA-past (action specified in the past: “The light has been turned on.”)
    AA-present (action specified in the present: “I’m turning on the light.”)
    AA-future (action specified in the future: “I’ll turn on the light in 5 min.”)
    AA-ANS (action not specified: “U: Turn on the light. S: Done.”)
    AA-AI (action impossible: “I can’t open the door while the cycle is running.”)
    AA-null (action is done but not acknowledged: “U: Turn on the light. S: Anything else?”)
  - Specify state:
    SS-done (explicit action, done: “The light is now on.”)
    SS-NA (explicit action, not applicable: “The light is already on.”)
    SS-unclear (explicit action, unclear: “The light is on.”—it is not clear whether the light was already on or the system performed the action)
  - Requests:
    Req-location (missing parameter, location: “Which light?”)
    Req-dev (missing parameter, device: “What should I connect to Wifi?”)
    Req-time (missing parameter, time: “When should I do that?”)
    Req-temp (missing parameter, temperature: “What temperature?”)
    Req-other (missing parameter, other: “What should I connect it to?”)
    Req-action (request more actions: “Anything else?”)
    Req-repeat (request repeat: “Could you repeat?”)
  - Other response:
    O-null (equivalent to silence)
    O-pleasant (system pleasantry: “You are welcome.”)
  - Level of specificity:
    explicit (parameters explicit: “U: Turn on the light. S: The light has been turned on.”)
    implicit (parameters implicit: “U: Turn on the light. S: It’s been turned on.”)
  - Register:
    Reg-direct (direct: “U: Turn on the light. S: I’m turning on the light.”)
    Reg-conv (conversational: “U: Turn on the light. S: Sure thing, the light is now on.”)
  - Grammaticality:
    gram (grammatical responses: “Which light shall I turn on?”)
    ungram (ungrammatical responses: “Which light shall I open?”)

- User utterances:

  - Request action:
    RA-dev (RA-dev-wash, RA-dev-speaker, etc., depending on the device: “Turn on the speaker.”)

  RA-location (specified location: “Turn on the speaker in the bedroom.”)
  RA-time (specified time: “Turn on the TV in 10 min.”)
  RA-temp (specified temperature: “Decrease temperature to 40°.”)
  RA-end-state (specified end state: “I feel like listening to music.”)
  RA-other (specified other: “Connect the speaker to Bluetooth.”)
  RA-action (specified action: “Turn it off.”)
- Response to system:
  RS-yes (yes)
  RS-no (no)
  RS-null (silence)
  RS-restate (restate request: “S: I don’t understand. Which light should I turn on? U: The one in the bedroom.”)
  RS-decline (decline further action: “S: Anything else? U: No, thanks.”)
  RS-param (provide parameters: “S: When? U: Today at 4 pm.”)
- Pleasantries:
  P-greet (greeting: “Hello system!”)
  P-thank (thanks: “Thanks.”)
- Level of specificity:
  explicit (parameters explicit: “The washer in the bathroom.”)
  implicit (parameters implicit: “The one in the bathroom.”)

Our corpus of 232 dialogues was annotated for the features presented above by our principal annotator. It was these annotations that were used to generate some of the feature correlations which ultimately informed our dialogue quality evaluation functions (see Sect. 5). To measure inter-annotator agreement, a smaller subset of 30 dialogues was annotated by a second annotator, and used for comparison with the annotations performed by the principal annotator. Overall there was raw agreement of 97% and a Krippendorff’s alpha value of 0.867, including cases where neither annotator annotated anything for a category. If we look only at cases where at least one annotator entered a tag, we still have 83.3% agreement. In those cases raw agreement was above 80% for most categories, ranging from 58.6% for “Response to system” to 100% for “Other response”.

Other manual or semi-automatic annotations that we performed at the whole dialogue level are as follows. Note that semi-automatic annotations were based on manual annotations, and were manually checked afterwards.

- Number of all tasks and successful tasks in the dialogue.
- Number and list of misunderstandings in the dialogue.
- Number of system confirmation requests in the dialogue. (This can be derived from “CU-confirm” above.)
- Number of system requests for repetition in the dialogue. (This can be derived from “Req-repeat” above.)
- Number of system requests for more information in the dialogue. (This can be derived from “Req-dev”, “Req-location”, “Req-temp”, “Req-time”, and “Req-other” above.)
- Number and list of immediate and scheduling tasks in the dialogue.

- Number and list of devices in the dialogue.
- Number of cases in which the system says that it cannot perform an action. (This can be derived from AA-AI above.)

Note that for each of the annotations above we compute two features. One feature gives us the exact count of e.g., misunderstandings, system confirmation requests, scheduling tasks, etc. The other feature has a binary value (yes/no) and keeps track of whether e.g., misunderstandings, system confirmation requests, scheduling tasks, etc., are present in the dialogue or not regardless of their frequency.

4 Collection of Human Ratings Via Crowdsourcing

We grouped our 232 dialogues in sets of 5 dialogues and asked human raters on Amazon Mechanical Turk (MTurk) to rank them. Each rater had to perform 4 tasks. In each task, raters were presented with 8 sets of 5 dialogues and asked to rank them (best dialogue to worst dialogue in the set) based on which system they would most like to interact with. The types of dialogues and contexts of the individual tasks are described below:

- Task 1: All dialogues in a set had the same task goal and device (i.e., turn on TV, connect speaker to Bluetooth, etc.).
- Task 2: Dialogues in a set represented a mix of task goals and devices.
- Task 3: Dialogues in a set represented a mix of task goals and devices, and raters were presented with a description of the pre-conversation system status, i.e., the state and location of each device before the dialogue starts.
- Task 4: Dialogues in a set represented a mix of task goals and devices, and raters were presented with a description of the pre-conversation and post-conversation system status, i.e., the state and location of each device before the dialogue starts and after the dialogue ends.

We collected rankings from 199 people on MTurk. From these rankings we generated pairwise comparisons for all dialogues. For example, assuming that we have 3 dialogues D1, D2 and D3, we can generate the pairs D1-D2, D1-D3, D2-D3 and calculate for each pair how many times the first dialogue of the pair is ranked higher than the second dialogue of the pair and vice versa. Thus for each dialogue we can generate a score by dividing the number of times this dialogue wins in all pairwise comparisons by the number of all comparisons of this dialogue. So in the example above, the score for dialogue D1 would be (“number of times D1 beats D2” + “number of times D1 beats D3”)/(“number of times D1 competes with D2” + “number of times D1 competes with D3”). This score (from now on referred to as “Score”) is a real number between 0 and 1 and will be used in the calculations of correlations and evaluation functions in Sect. 5.

5 Dialogue Quality Evaluation Functions

We calculated pairwise Pearson correlations between all features in our annotations (automatic, semi-automatic, and manual) as well as Pearson correlations between the "Score" and each feature in our annotations. Note that "***" means that the correlation is very significant ($p < 0.001$), "**" means that the correlation is significant ($p < 0.01$), and "*" means that the correlation is borderline significant ($p < 0.05$). Table 2 shows a list of some of the most indicative (higher and/or more significant) correlations that we found between the "Score" and dialogue features.

For each feature we experimented both with binary values (the feature exists or not) and counts (frequency of occurrence). Counts may be affected by the number of tasks and multiple devices in the interaction and we wanted to account for that. From

Table 2 Correlations of "Score" with features (***: $p < 0.001$, **: $p < 0.01$, *: $p < 0.05$)

Dialogue feature	Pearson's r
Number of misunderstandings	−0.76***
Misunderstandings exist or not? (Binary)	−0.77***
Number of system confirmation requests	−0.50***
System confirmation requests exist or not? (Binary)	−0.50***
Number of system requests for more information	0.27***
System requests for more information exist or not? (Binary)	0.28***
Number of silence occurrences	−0.67***
Silence exists or not? (Binary)	−0.68***
Number of times the user says "I mean/I meant"	−0.17**
User says "I mean/I meant" or not? (Binary)	−0.20**
Number of times the user says "I said"	−0.23***
User says "I said" or not? (Binary)	−0.23***
Number of times the user says "no/nope"	−0.72***
User says "no/nope" or not? (Binary)	−0.73***
Number of system turns per dialogue	−0.61***
Number of user turns per dialogue	−0.61***
Number of all turns per dialogue	−0.61***
Number of system words per dialogue	−0.44***
Number of user words per dialogue	−0.45***
Number of all words per dialogue	−0.51***
Number of times the system does nothing (A-nothing)	−0.54***
System does nothing (A-nothing) or not? (Binary)	−0.36***
Number of times the system does something invalid (A-invalid)	−0.33***
System does something invalid (A-invalid) or not? (Binary)	−0.33***
System has a conversational style or not? (Binary)	0.17**
Number of times the user specifies the location of a device (RA-location)	−0.22**
User specifies the location of a device (RA-location) or not? (Binary)	−0.26***

Table 3 Correlations of misunderstandings with features (***: $p < 0.001$, **: $p < 0.01$, *: $p < 0.05$)

Dialogue feature	Pearson's r (for counts)	Pearson's r (for binary values)
Occurrences of silence	0.42***	0.44***
Occurrences of "I mean/I meant"	0.22***	0.24***
Occurrences of "I said"	0.25***	0.29***
Occurrences of "no/nope"	0.66***	0.64***
Number of system turns per dialogue	0.58***	–
Number of user turns per dialogue	0.58***	–
Number of all turns per dialogue	0.58***	–
Number of system words per dialogue	0.55***	–
Number of user words per dialogue	0.47***	–
Number of all words per dialogue	0.60***	–
System does nothing (A-nothing)	0.47***	0.33***
System does something invalid (A-invalid)	0.41***	0.40***
User specifies the location of a device (RA-location)	0.17**	0.24***

the above correlations we can see that misunderstandings and system confirmation requests are indicative of poor quality dialogues. On the other hand when the system asks for more information (e.g., "U: Can you connect the speaker? S: What should I connect it to?") this is indicative of having understood what the user wants and it is something that the human raters liked. Also, human raters seem to like a more conversational style for the system (i.e., when the system says "sure thing", etc.). The user action of specifying the location of a device usually occurred together with misunderstandings and thus negatively correlated with the "Score".

Table 3 shows a list of some of the most indicative (higher and/or more significant) correlations that we found between misunderstandings and dialogue features. Note that the correlations for counts are derived when we compare the number of misunderstandings with the counts of the dialogue features. In the same way, the correlations for binary values are derived when we compare whether misunderstandings exist or not with the binary values of the dialogue features. The high (or relatively high) correlation between misunderstandings and each one of these features entails that once we include misunderstandings in our evaluation function we do not also need to consider all these features.

The next step was to perform regression experiments to come up with evaluation functions that are predictive of human ratings. We excluded as redundant the aforementioned features that were highly correlated with misunderstandings, and we experimented with variations of the following features that as we saw in Table 2 were highly correlated with the "Score":

- Number of misunderstandings (Mu)
- Misunderstandings exist or not? (Binary-Yes/No) (Mu_b)

Table 4 Evaluation functions and corresponding root mean square error (RMSE) values (the best, i.e., lowest, RMSE values are shown in bold); reward-based function is normalized; Mu: number of misunderstandings, Mu_b: misunderstandings exist or not (binary), Conf: number of system confirmation requests, Conf_b: system confirmation requests exist or not (binary), Inf: number of system requests for more information, Inf_b: system requests for more information exist or not (binary), Conv_b: system has a conversational style or not (binary)

Description	Evaluation function	RMSE
Reward-based function	100*Task_success−5*Num_system_turns	0.5224
Mu	−0.21*Mu+0.62	**0.0902**
Mu_b	−0.23*Mu_b+0.62	**0.0920**
Conf	−0.16*Conf+0.56	0.1318
Conf_b	−0.17*Conf_b+0.56	0.1325
Inf	0.09*Inf+0.49	0.1281
Inf_b	0.09*Inf_b+0.48	0.1270
Conv_b	0.05*Conv_b+0.49	0.1320
Mu+Conf	−0.18*Mu−0.05*Conf+0.62	**0.0919**
Mu_b+Conf_b	−0.2*Mu_b−0.05*Conf_b+0.63	0.0929
Mu+Inf	−0.2*Mu+0.02*Inf+0.61	**0.0899**
Mu_b+Inf_b	−0.23*Mu_b+0.01*Inf_b+0.62	**0.0919**
Conf+Inf	−0.15*Conf+0.04*Inf+0.55	0.1281
Conf_b+Inf_b	−0.16*Conf_b+0.04*Inf_b+0.55	0.1283
Mu+Conf+Conv_b	−0.19*Mu−0.05*Conf−0.02*Conv_b+0.63	0.0944
Mu_b+Conf_b+Conv_b	−0.2*Mu_b−0.05*Conf_b−0.01*Conv_b+0.63	0.0944
Mu+Inf+Conv_b	−0.21*Mu+0.02*Inf−0.01*Conv_b+0.62	**0.0911**
Mu_b+Inf_b+Conv_b	−0.23*Mu_b+0.01*Inf_b−0.01*Conv_b+0.62	**0.0924**
Conf+Inf+Conv_b	−0.15*Conf+0.04*Inf+0.01*Conv_b+0.55	0.1269
Conf_b+Inf_b+Conv_b	−0.16*Conf_b+0.04*Inf_b+0.01*Conv_b+0.55	0.1271
Mu+Conf+Inf	−0.18*Mu−0.05*Conf+0.01*Inf+0.62	**0.0915**
Mu_b+Conf_b+Inf_b	−0.2*Mu_b−0.05*Conf_b+0.01*Inf_b+0.63	0.0928
Mu+Conf+Inf+Conv_b	−0.18*Mu−0.05*Conf+0.01*Inf−0.02*Conv_b+0.63	0.0939
Mu_b+Conf_b+Inf_b+Conv_b	−0.2*Mu_b−0.05*Conf_b+0.005*Inf_b−0.01*Conv_b+0.63	0.0943

- Number of system confirmation requests (Conf)
- System confirmation requests exist or not? (Binary-Yes/No) (Conf_b)
- Number of system requests for more information (Inf)
- System requests for more information exist or not? (Binary-Yes/No) (Inf_b)
- System has a conversational style (i.e., at least half of system responses are annotated as "Reg-conv") or not? (Binary-Yes/No) (Conv_b)

We randomly split our corpus in a training set and a test set (75% for training and 25% for testing). We applied linear regression to the training set, calculated our evaluation functions, and then measured how these evaluation functions performed on the test set (i.e., how predictive they were of the actual human ratings). To do that we calculated the root mean square error (RMSE) as shown in Eq. (1) where *n* is the number of dialogues, $Score_{i\,Predicted}$ is the predicted "Score" for dialogue i (calculated by our evaluation function), and $Score_{i\,Actual}$ is the actual "Score" for dialogue i (derived from the human ratings).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Score_{i\,Predicted} - Score_{i\,Actual})^2} \tag{1}$$

Obviously the lower the RMSE the better. We also constructed a reward-based evaluation function based on the literature of reinforcement learning of dialogue policies. In this case a typical reward function is to give 100 points for a fully successful dialogue minus a penalty (−5) per system turn. We normalized so that the resulting scores of this function range from 0 to 1, and thus they are comparable to the rest of the "Scores" derived from the human ratings and the evaluation functions.

Table 4 shows the derived evaluation functions and the corresponding RMSE values. As we can see, the evaluation functions that include "misunderstandings", and to a lesser extent "system confirmation requests", "system requests for more information", and "conversational style" are all good predictors of the real "Scores" (derived from the human ratings). This is not true for the reward-based evaluation function which results in a much higher RMSE.

Note that we did not find any statistically significant correlation between the "Score" and the "level of specificity" in the system's responses (explicit vs. implicit). As part of our human ratings' data collection process, we asked raters on MTurk to provide qualitative feedback about which features of dialogues they liked or disliked. Some raters consistently mentioned the ability of the system to explicitly state the action that it was about to perform (grounding) and request more information (as a clarification request) as positive dialogue features. However, other people appreciated brevity and preferred more implicit system responses. This means that it is hard to come up with one evaluation function that captures the preferences of all users, and that we may need to develop dialogue quality evaluation functions tailored to specific types of users.

6 Conclusion and Future Work

We analyzed a corpus of system-user dialogues in the IoT domain. Our corpus was automatically, semi-automatically, and manually annotated with a variety of features both on the utterance level and the full dialogue level. The corpus also includes human ratings of dialogue quality collected via crowdsourcing. We calculated correlations between annotated features in our corpus and human ratings, and developed dialogue quality evaluation functions that were shown to be highly predictive of human ratings when tested on a held-out portion of our corpus.

For future work, we plan to develop models that can recreate rankings within a set of dialogues and see whether these derived rankings agree with the actual human rankings in our data set. Furthermore, we would like to collect more realistic system-user dialogues in a Wizard of Oz setting and/or with a real dialogue system. We also plan to develop evaluation functions that are tailored to specific users or groups of users (user modeling). User modeling will also include studying how to make the system-user interaction more engaging, which in turn will facilitate establishing rapport between the system and the user.

Acknowledgements This work was funded by Samsung Electronics Co., Ltd. Some of the authors were partly supported by the U.S. Army Research Laboratory. Any statements or opinions expressed in this material are those of the authors and do not necessarily reflect the policy of the U.S. Government, and no official endorsement should be inferred.

References

1. Artstein R, Gandhe S, Gerten J, Leuski A, Traum D (2009) Semi-formal evaluation of conversational characters. In: Grumberg O, Kaminski, M, Katz, S, Wintner S (eds) Languages: from formal to natural. Essays dedicated to Nissim Francez on the occasion of his 65th birthday. Lecture Notes in Computer Science 5533. Springer, pp 22–35
2. Foster ME, Giuliani M, Knoll A (2009) Comparing objective and subjective measures of usability in a human-robot dialogue system. In: Proceedings of ACL, pp 879–887. Suntec, Singapore
3. Galley M, Brockett C, Sordoni A, Ji Y, Auli M, Quirk C, Mitchell M, Gao J, Dolan B (2015) DeltaBLEU: a discriminative metric for generation tasks with intrinsically diverse targets. In: Proceedings of ACL (short papers), pp 445–450. Beijing, China
4. Gandhe S, Traum D (2008) Evaluation understudy for dialogue coherence models. In: Proceedings of SIGDIAL, pp 172–181. Columbus, Ohio, USA
5. Georgila K, Henderson J, Lemon O (2005) Learning user simulations for information state update dialogue systems. In: Proceedings of Interspeech, pp 893–896. Lisbon, Portugal
6. Georgila K, Henderson J, Lemon O (2006) User simulation for spoken dialogue systems: learning and evaluation. In: Proceedings of Interspeech, pp 1065–1068. Pittsburgh, Pennsylvania, USA
7. Guo F, Metallinou A, Khatri C, Raju A, Venkatesh A, Ram A (2017) Topic-based evaluation for conversational bots. In: Proceedings of NIPS Workshop on Conversational AI: Today's Practice and Tomorrow's Potential. Long Beach, California, USA
8. Hastie H (2012) Metrics and evaluation of spoken dialogue systems. In: Lemon O, Pietquin O (eds) Data-driven methods for adaptive spoken dialogue systems. Springer, pp 131–150
9. Henderson J, Lemon O, Georgila K (2008) Hybrid reinforcement/supervised learning of dialogue policies from fixed datasets. Comput Linguist 34(4):487–511
10. Hone KS, Graham R (2000) Towards a tool for the subjective assessment of speech system interfaces (SASSI). J Nat Lang Eng 6(3–4):287–303
11. Jeon H, Oh HR, Hwang I, Kim J (2016) An intelligent dialogue agent for the IoT home. In: Proceedings of the AAAI Workshop on Artificial Intelligence Applied to Assistive Technologies and Smart Environments, pp 35–40. Phoenix, Arizona, USA
12. Jung S, Lee C, Kim K, Jeong M, Lee GG (2009) Data-driven user simulation for automated evaluation of spoken dialog systems. Comput Speech Lang 23(4):479–509
13. Liu CW, Lowe R, Serban IV, Noseworthy M, Charlin L, Pineau J (2016) How NOT to evaluate your dialogue system: an empirical study of unsupervised evaluation metrics for dialogue response generation. In: Proceedings of EMNLP, pp 2122–2132. Austin, Texas, USA

14. Lowe R, Serban IV, Noseworthy M, Charlin L, Pineau J (2016) On the evaluation of dialogue systems with next utterance classification. In: Proceedings of SIGDIAL, pp 264–269. Los Angeles, California, USA
15. Paksima T, Georgila K, Moore JD (2009) Evaluating the effectiveness of information presentation in a full end-to-end dialogue system. In: Proceedings of SIGDIAL, pp 1–10. London, UK
16. Purandare A, Litman D (2008) Analyzing dialog coherence using transition patterns in lexical and semantic features. In: Proceedings of FLAIRS, pp 195–200. Coconut Grove, Florida, USA
17. Robinson S, Roque A, Traum D (2010) Dialogues in context: an objective user-oriented evaluation approach for virtual human dialogue. In: Proceedings of LREC, pp 64–71. Valletta, Malta
18. Schatzmann J, Georgila K, Young S (2005) Quantitative evaluation of user simulation techniques for spoken dialogue systems. In: Proceedings of SIGDIAL, pp 45–54. Lisbon, Portugal
19. Traum DR, Robinson S, Stephan J (2004) Evaluation of multi-party virtual reality dialogue interaction. In: Proceedings of LREC, pp 1699–1702. Lisbon, Portugal
20. Walker M, Kamm C, Litman D (2000) Towards developing general models of usability with PARADISE. J Nat Lang Eng 6(3–4):363–377