DialCrowd: A toolkit for easy dialog system assessment

Kyusong Lee, Tiancheng Zhao, Alan W. Black and Maxine Eskenazi
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA
{kyusongl,tianchez,awb,max+}@cs.cmu.edu

Abstract

When creating a dialog system, developers need to test each version to ensure that it is performing correctly. Recently the trend has been to test on large datasets or to ask many users to try out a system. Crowdsourcing has solved the issue of finding users, but it presents new challenges such as how to use a crowdsourcing platform and what type of test is appropriate. DialCrowd makes system assessment using crowdsourcing easier by providing tools, templates and analytics. This paper describes the services that DialCrowd provides and how it works. It also describes a test of DialCrowd by a group of dialog system developers.

1 Introduction

The development of a spoken dialog system involves many steps and always ends in system tests. As our systems have become more complicated and the statistical methods we use demand more and more data, proper system assessment becomes an increasingly difficult challenge. One of the easier approaches to goal-oriented system assessment is to employ user simulation (Jung et al., 2009; Pietquin and Hastie, 2013; Schatzmann et al., 2005). It aims at the overall assessment of the system by measuring goal completion. While this is a useful first approach, it can't reveal what a human user would actually say. Thus this approach is usually used as a first approximation, quickly followed up with some assessment using humans. SOme chatbot systems use machine learning metrics to compare a model-generated response to a golden standard response. However, those metrics assume that a valid response has a significant word overlap with the golden response, which is often not the case. Liu et al. (2016) showed that these metrics correlate very weakly with human judgment. Other approaches used to assess non-task oriented dialog systems include word similarity metrics, next utterance classification, word perplexity, and response diversity (Serban et al., 2015). They are limited since they can't reproduce the variety found in actual user behavior.

Crowdsourcing platforms, such as Amazon Mechanical Turk (AMT), have shown promise in assessing spoken dialog systems (Eskenazi et al., 2013; Jurčíček et al., 2011). But for most developers it is not trivial to set up the crowdsourcing process and obtain usable results. Jurčíček et al. (2011) noted that this process must be cheap to operate and easy to use. Researchers (the requesters) have to overcome the following difficulites: learning how to use the crowdsourcing entity interface, learning how to create an understandable and attractive task, deciding on the correct form that the task should take (the template), connecting the dialog systems that are to be assessed to the crowdsourcing platform, paying the workers, assessing the quality of the workers' production, getting solid final results. To solve the connection issue, researchers have used the telephone to connect their dialog systems, relying on a crowdsourcing web interface to present the task, then sending the worker to the dialog system and finally bringing them back to the interface to collect their production and schedule payment. This connection issue is one example of these hurdles. Researchers are also faced with the choice of the form of assessment. The types of tests may vary. One form that is often found in the literature is to compare two versions of the same system (A/B text). The literature shows that a small number of test types covers most publications.

DialCrowd (https://dialrc.org/dialcrowd.html) is a toolkit that makes crowdsourced evalua-

Proceedings of the SIGDIAL 2018 Conference, pages 245–248,
Melbourne, Australia, 12-14 July 2018.

tion studies easy to run. We have identified a small number of standard evaluation experiment types and provided templates that generate web interfaces for these studies in a crowdsourcing environment. The DialCrowd interface first has the researcher choose the type of study (or she can make up her own). Once the type is chosen, the corresponding template appears and is filled in. This generates the task (HIT on AMT) that the worker will see. This considerably lowers preparation time, and guides those who are new to the field to commonly-accepted study types. DialCrowd presently has a small set of templates which will soon expand to include those suggested by our users or that we find in the literature. Other aspects of crowdsourced assessment that DialCrowd presently addresses are:

- Explaining the overall goal of the assessment to the worker
- Instructing the worker on how to accomplish the task
- Reminding a requester to post a consent form for explicit permission to use the data
- Helping calculate how much to pay for a HIT
- How to make a HIT less susceptible to BOTs
- Help in designing the appearance of the HIT.

Going forward, DialCrowd will also provide tools to:

- Assess an individual worker
- Create a golden data set
- Assess the final outcome with basic analytics
- Ensure that results are collected ethically and are made available to the community with as few restrictions as possible that do not compromise the worker's privacy.

2 Related Work

The performance of dialog systems can be measured via: task success, the number of turns per dialog, ASR accuracy, system response delay, naturalness of the speech output, consistency with the users expectations, and system cooperativeness (Moller and Skowronek, 2003). These metrics are both subjective and objective. Subjective metrics often come in the form of exit polls following the worker's interaction with a system. They often measure how much a worker liked interacting with a system or whether the worker would like to use the system again. Objective metrics can be extracted automatically or labeled manually by experts.

Toolkits must support both interactive and non-interactive studies. There are offline datasets that could be used to run some system studies. But they can't be used if success depends on how the user responds to a system utterance. In this case, only interactive tests can do the job. On the other hand, some researchers may have sets of responses that their systems have produced for which they need to know the appropriateness, given recent dialog context. Non-interactive tests are used in this case. DialCrowd provides support for both forms.

Non-interactive tests are the simplest to implement since the actual dialog system is not involved. Here the worker often sees a portion of a real dialog and passes some sort of judgment. Yang et al. (2010) for example used the Let's Go dialog logs (Raux et al., 2005) and identified several cue phrases that afforded the development of a set of heuristics to automatically classify those logs into five categories in terms of task success: too short, multi-task, task complete, out of scope, and task-incomplete.

Interactive tests usually have instructions and a scenario to enact that constrain the worker's behavior. Jurčíček et al. (2011)), for example, conducted real user evaluations of the Cambridge Restaurant Information system using AMT.

Crowdsourcing has several advantages. The crowd has been shown to be substantially more efficient in accomplishing assessment tasks (Munro et al., 2010). No time is spent recruiting users. Jurčíček et al. (2011) note that it took several weeks to recruit users for the Cambridge trial while it only took several days to get this done using crowdsourcing and the cost was much lower.

3 DialCrowd

The inspiration for DialCrowd comes from the TestVox toolkit (Parlikar, 2012) for speech synthesis evaluation. TestVox enables any developer to quickly upload data in a standard format and then deploy it on AMT or some other crowdsourcing site, or to a controlled set of developer-selected workers and get results easily and rapidly. TestVox is easy to deploy on AMT.

Several tools have recently been proposed to

connect non-speech dialog systems to AMT. DialCrowd is different in that it is speech-enabled. DialCrowd is designed to make it easy to connect to spoken dialog systems using Google Chrome's speech recognition. It also provides audio testing to ensure that workers have a working microphone, speakers, and headset. DialCrowd is designed to eliminate common crowdsourcing mistakes that affect results such as giving the worker too much information, creating a task with an unreasonably high cognitive load and proposing a task that a bot can easily be created to do. It provides off-the-shelf dialog systems that can be used as a baseline, such as DialPort's Let's Forecast (weather), Let's Eat (restaurants), Let's Go (bus information) and Qubot (question answering chatbot) (Zhao et al., 2016). Requesters can use their own dialog systems as the baseline.

DialCrowd uses test design techniques such as Latin Square in a set of templates (Cochran and Cox, 1950)). It uses timed sandbox trials to suggest correct, respectful payment for a HIT with the following equation $= \frac{M \times T}{60min}$ where M is the hourly minimum wage in the requester's state. T is the average amount of time on task during internal testing for 10 people. Requesters pay using their own accounts with the crowdsourcing platform of choice.

4 Overall Architecture of DialCrowd

DialCrowd has two components: DialCrowd Admin (requester view) and DialCrowd Worker (worker view). Although not restricted to AMT, this paper explains the overall process on AMT as an example. Given a dialog log format, the requester selects the set of turns and the context the worker should see. This section describes the process on DialCrowdAdmin.

1. Creating a project on Amazon MTurk: DialCrowd's requester site provides 10 sample templates that cover common uses of AMT. For interactive assessment, a survey template is chosen and DialCrowdAdmin automatically generates the link to a dialog system.

2. Create a project on DialCrowd Admin: After creating a project, the study is designed in detail. DialCrowd can help assess a single dialog system with Likert feedback ratings. It can also compare more than one dialog system, for example using an A/B template. In the latter case, dialog systems are presented in random order or in a Latin Square format. For non-interactive tests, JSON data, such as dialog logs, is added by the requester. DialCrowd also supports various types of exit polls: Likert scale, open-ended, and A/B, with random order presentation. For interactive tests, there are two types of testing: "1 to N" and "N to 1" where "1 to N" means one worker tests and individually scores N dialog systems (Likert Scale or select the best one). "N to 1" means N workers test one dialog system that DialCrowd has randomly selected amongst several.

3. Connect one or more dialog systems: 1. At the end of the DialCrowdAdmin setup, the DialCrowd Worker webpage is available. To connect to DialCrowd, a dialog system has an HTTP server waiting for utterances that DialCrowd directs to it using some simple specific protocols. This makes connecting to DialCrowd easy for anyone with basic programming knowledge. DialCrowd provides off-the-shelf server wrapper templates in three mainstream programming languages: Java, Python, and JavasSript https://github.com/DialRC/PortalAPI. The API protocol is the same as for DialPort.

4. Testing the task and then deploying it: After running the backend RESTful APIs, the requester inputs the backend API URL and checks the DialCrowd connection. The requester can then preview the website automatically generated by DialCrowdAdmin. DialCrowdAdmin provides log viewers and survey results. Requesters can also download data. DialCrowdWorker is the website through which workers talk to dialog systems and carry out the assigned task. The website is automatically generated by DialCrowdAdmin.

5 A user study of DialCrowd

This section describes a study of the use of DialCrowd by a set of requesters. The DialCrowd toolkit was made available to 10 dialog researchers. We gave them survey links and asked them to use DialCrowd. After they used it, we collected feedback. When asked how long it took to build a crowdsourcing study in their previous research, over 50% said more than one day and less than one week. For DialCrowd, 50% said they finished the whole process in between one and three hours. When asked how they set up the evaluation pipeline previously, 90% said they did it themselves without a toolkit. When asked how easy it was to use the DialCrowd toolkit and if it was

useful, answers averaged above 4 on a scale from 1 to 5 where 5 was best.

- The instructions were clear to follow. [AVG:4.4, STD:0.69]
- The toolkit is useful. I want to use this toolkit in the future to run other studies [AVG:4, STD:0.94]
- I will use this toolkit in the future to run other studies [AVG:4.2, STD:0.78]

They also said that it took a lot less time to run a study using DialCrowd (100%), and that the toolkit is well documented (80%). They used interactive tests on their dialog systems or chatbot and non-interactive tests for classifying intent or entity labeling in specific domains. Among the open-ended questions, we received several questions about whether future versions of DialCrowd could include turn-based assessments and full systems that include other ASRs and TTSs, not just Google Chrome APIs. Participants also asked about adding more question types/more support for custom question types through an API. We are working on this function at present.

6 Conclusion

DialCrowd is a spoken dialog system crowdsourcing assessment toolkit. It is designed for use by the research community. Most users have found DialCrowd easy to use and would like to use it again in the future.

Acknowledgments

This work is partly funded by National Science Foundation grant CNS-1512973. The opinions expressed in this paper do not necessarily reflect those of the National Science Foundation.

References

William G Cochran and Gertrude M Cox. 1950. Experimental designs. .

Maxine Eskenazi, Gina-Anne Levow, Helen Meng, Gabriel Parent, and David Suendermann. 2013. *Crowdsourcing for speech processing: Applications to data collection, transcription and assessment*. John Wiley & Sons.

Sangkeun Jung, Cheongjae Lee, Kyungduk Kim, Minwoo Jeong, and Gary Geunbae Lee. 2009. Data-driven user simulation for automated evaluation of spoken dialog systems. *Computer Speech & Language* 23(4):479–509.

Filip Jurčíček, Simon Keizer, Milica Gašić, Francois Mairesse, Blaise Thomson, Kai Yu, and Steve Young. 2011. Real user evaluation of spoken dialogue systems using amazon mechanical turk. In *Twelfth Annual Conference of the International Speech Communication Association*.

Chia-Wei Liu, Ryan Lowe, Iulian V Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. *arXiv preprint arXiv:1603.08023* .

Sebastian Moller and Janto Skowronek. 2003. Quantifying the impact of system characteristics on perceived quality dimensions of a spoken dialogue service. In *Eighth European Conference on Speech Communication and Technology*.

Robert Munro, Steven Bethard, Victor Kuperman, Vicky Tzuyin Lai, Robin Melnick, Christopher Potts, Tyler Schnoebelen, and Harry Tily. 2010. Crowdsourcing and language studies: the new generation of linguistic data. In *Proceedings of the NAACL HLT 2010 workshop on creating speech and language data with Amazon's Mechanical Turk*. Association for Computational Linguistics, pages 122–130.

Alok Parlikar. 2012. Testvox: Web-based framework for subjective evaluation of speech synthesis. *Open-source Software* page 13.

Olivier Pietquin and Helen Hastie. 2013. A survey on metrics for the evaluation of user simulations. *The knowledge engineering review* 28(1):59–73.

Antoine Raux, Brian Langner, Dan Bohus, Alan W Black, and Maxine Eskenazi. 2005. Let's go public! taking a spoken dialog system to the real world. In *Ninth European Conference on Speech Communication and Technology*.

Jost Schatzmann, Kallirroi Georgila, and Steve Young. 2005. Quantitative evaluation of user simulation techniques for spoken dialogue systems. In *6th SIGdial Workshop on DISCOURSE and DIALOGUE*.

Iulian Vlad Serban, Ryan Lowe, Peter Henderson, Laurent Charlin, and Joelle Pineau. 2015. A survey of available corpora for building data-driven dialogue systems. *arXiv preprint arXiv:1512.05742* .

Zhaojun Yang, Baichuan Li, Yi Zhu, Irwin King, Gina Levow, and Helen Meng. 2010. Collection of user judgments on spoken dialog system with crowdsourcing. In *Spoken Language Technology Workshop (SLT), 2010 IEEE*. IEEE, pages 277–282.

Tiancheng Zhao, Kyusong Lee, and Maxine Eskenazi. 2016. Dialport: Connecting the spoken dialog research community to real user data. In *Spoken Language Technology Workshop (SLT), 2016 IEEE*. IEEE, pages 83–90.