DialPort: Real-World Data for Academia Spoken Dialog Systems

Tiancheng Zhao, Yulun Du, Kyusong Lee and Maxine Eskenazi
Department of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
{tianchez, yulund, kyusongl, max}@cs.cmu.edu

Abstract

This paper describes the DialPort spoken dialog Portal which gives academic dialog system creators the opportunity to connect with other dialog systems and run studies with real users. Indeed, the interaction with real users challenges system builders in ways that paid workers never do. The DialPort connection toolkit makes it easier for developers to link their systems to the Portal. With this ease in connecting and a flow of real data, DialPort paves the way for future academic research on an open platform.

1 Introduction

The public interest in Siri and other intelligent agents has created a class of potential users who can be of great use to the research community to create much-needed real user datasets. More and more people are willing to regularly talk to an agent, as we see by the six-digit sales of Alexa units at QVC and the corresponding customer feedback. To take advantage of this new popularity, we have created a Portal that joins numerous academic dialog systems so that users can experience a one-stop shop and developers are freed up from the search for a flow of users into their system. The Portal affords studies on many research topics from neural systems to chatbots to multi-topic mechanisms.

Industry has collected large datasets and sometimes retains pools of real users. They are viewed as strategic competitive resources and so are very rarely shared with the academic research community. Much fundamental research still remains to be done, such as signal processing in noisy conditions interaction with groups of users who are difficult for speech recognition to understand (e.g. like elderly and non-natives), management of complex dialogs (such as multi-party meetings, negotiations, and multimodal interaction), and the use of meta linguistic information, such as prosody.

It is difficult for any one academic research group to collect a significant amount of real user data. The users must be found and their interest maintained. At the same time, the user interface must be maintained and updated. By having one data-gathering Portal that all the dialog systems can connect to, the task for each participating site is considerably lighter. At the same time, the portal is of interest to more diverse potential users since the latter can select from a range of applications that fulfill their needs at any given time. Only the researchers maintaining the Portal itself are tasked with attracting the users and maintaining and updating the user-facing aspects.

DialPort went live in 2016. Since then, several academic systems, including some from CMU, have connected and the first real user data has started to flow.

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

2 DialPort and Procedure to Connect

Figure 1 gives an overview of the relationship between the Portal and the internal (CMU) and external agents [7]. Agents are defined here as all remote dialog systems that are connected to the Portal. The

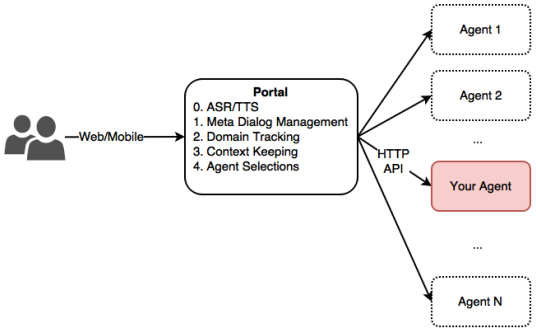

Figure 1: Overview of the DialPort Portal.

Portal handles the interface with the users via web or mobile devices. It also provides the following services to all agents:

- Automatic Speech Recognition (ASR) and Text-to-Speech (TTS) (powered by Google Chrome Speech API, more than 50 languages are supported).
- Domain Tracking: tracking the users' requested domains in a session.
- Context Keeping: remembering key slot-values across different remote agents.
- Agent Selection: selecting the best matching remote agent to answer the users.

An agent has to implement an HTTP API server that supports 3 RESTful APIs /init, /next, /end. Technical documentation can be found here[1]. Generally speaking, the Portal and an agent will interact as follows:

1. The Portal first interacts with a user to find out what the user is interested in (domain tracking).
2. The Portal then tries to select a remote agent that matches the user's needs the best (agent selection).
3. The Portal then starts a new remote session with the selected agent via (/init).
4. The Portal then passes each user utterance to the selected agent via (/next). The user is effectively talking to the selected agent.
5. After the selected agent decides to finish the conversation, the control is given back to the Portal and we go back to Step 1.
6. Sometimes a remote agent may perform poorly. In this case, the Portal ends the session via (/end) and goes back to Step 1.

2.1 Connected Systems and Collected Data

Several systems are already connected to DialPort. There are 5 different systems from CMU, including slot-filling systems (e.g. restaurant, weather, bus etc) and chat-oriented systems. The first external system to connect is the multi-domain dialog system from Cambridge University [5]. The Mr. Clue agent from University of Southern California was next [3]. Several other systems are presently in the process of being connected.

The Portal went public at the beginning of 2017, and as of the writing of this paper, it has collected 1146 dialogs from real users, from the internet as well as Facebook. The detailed breakdown is shown in Table 2.1. The agent developers retain the ownership of the collected data for their system. They

[1] `https://dialrc.org/resources.html`

Source	No. Session	Avg No. Turns / Session
Google Ad Words	9	9.11
First author's personal website	68	6.21
DialPort Website	805	7.21
Facebook Messenger	13	6.38
MTurk studies	186	2.9
Facebook Ad	43	7.49
SIGDIAL 2017 Demo	18	7.89
University of Pittsburgh Class	4	14
Total	1146	6.5

Table 1: Sources of collected data

are also responsible for their (Institutional Review Board) IRB application and for distributing their data. The collected corpora will be anonymized and released to the public and they will be accessed through one main access point.

3 Services from Portal

There are two key benefits that come from joining the Portal which can significantly reduce the cost of setting-up a real-world dialog system for an academic research group: the interface and user access.

3.1 Cross Platform User Interface

DialPort currently supports two types of interfaces to interact with human users: Web-Portal and Messenger-Portal. Web-Portal is a browser-independent, no plug-in, desktop-based public website[2], which has two versions: Web-Speech-Portal and Web-Text-Portal. Web-Speech-Portal is equipped with ASR/TTS and a turn-taking interaction manager [4], which enables continuous spoken language interaction between the Portal agent and human users. A real-time streaming audio server will be available shortly. This will support sending raw audio data to DialPort. The Web-Text-Portal shares the graphical interface equivalent of the speech alternative, and expects typed input. The text version is automatically triggered if the system detects the following conditions: 1) The user browser does not support Web Speech API The user computer has no microphone or fails to pass the microphone test. A recent survey of AMT workers we conducted shows that half of the workers preferred using a text interface and half a speech interface.

Users can also access DialPort from Facebook Messenger[3]. This interface works on both mobile and desktop hardware and can be easily used via the Facebook Messenger chatting interface. Messenger-Portal provides rather different user experience from the web version because of its similarity to text messaging with friends rather than talking to an intelligent agent. This may open up interesting research on the impact on users and on user experience. The current version of Messenger-Portal only supports text input and output between computers and humans. Fortunately, thanks to the flexible design of Messenger API, a more powerful version that supports multi-media input/output, including speech messages, is presently under construction.

3.2 Two Sources of Users

DialPort is connected to two outside sources of users: real users and crowdworkers from Amazon Mechanical Turk (AMT). In reality, there are several types of portal users. The developers themselves try out the system. Then they ask "friends and family" to try it. There are also real users from our advertising using Google AdWords and the creation of a Facebook page that is advertised to our "friends" and "followers". The Facebook page attracts explorers and real users through both organic (friends and friends-of-friends) and paid distribution. We also have crowdworker users from Amazon Mechanical Turk to easily collect large amount of dialog data or evaluate spoken dialog systems. The latter workers are not real users and we need quality control to reduce noisy data, yet we can, if needed, collect data at a lower cost and more quickly than via our other approaches.

[2] `https://skylar.speech.cs.cmu.edu`

[3] `https://www.messenger.com/t/TheDialportProject`

4 Benefits for Developers

To simplify the life of researchers and developers who work on agents connecting to Portal, the following resources are provided.

4.1 Server Templates for Joining Portal

To connect to DialPort, an agent needs to have an HTTP server waiting for utterances directed from DialPort through specific protocols. We are making connecting to DialPort extremely easy for anyone with basic programming knowledge by providing them with off-the-shelf server wrapper templates in three mainstream programming languages: Java, Python, and JavasSript. We have included the templates in the PortalAPI tutorials[4].

Following the DialPort communication protocols, the Java, Python, and JavaScript server wrappers are implemented with the Spark, Flask, and Node.js frameworks respectively. In this way, we completely hide the implementation of RESTful APIs from the developers. As a result, developers of potential agents only need to modify two methods in the wrapper to connect to DialPort, making the whole connecting process pain-free.

4.2 Log Viewer and Annotation Tools

We also provide a web-interface for DialPort members to view and annotate their dialog data. The log viewer displays every session of dialog history between the system and the users. Besides the plain dialog history, the viewer also indicates the source of users and other meta-information, such as average turn-taking delay, number of turns, average response length, etc.

We also provide two types of annotation tools: entity and dialog act. The entity and dialog act annotation are similar to the Microsoft LUIS interface [6], which enables the developer to label every user utterance with a set of dialog acts and includes words that are slot-values. Moreover, an active learning system was developed to speed up the annotation process. Two main functions are implemented: 1. data ranking: rank unlabelled utterances in terms of the confidence of the current prediction models and rank the ones with low confidence higher, similar to the active exploration approach used in reinforcement learning [1]. 2. After a certain number of new utterances are labelled, a new NLU is trained on-the-fly and labels all the unlabeled utterances. This can lead to significant gain in speed because the developers then only need to confirm the model prediction on the correctly classified utterances.

4.3 Daily and Weekly Reports

We provide automated daily and weekly reports to DialPort contributors by email. We maintain two versions: one is a master branch for real users and the other is develop branch for internal testing. we include the numbers from master branch, not from develop branch. The report contains the number of sessions, the average number of turns per session, the total number of turns, and the number of turns of each remote agent (Table 4.3). We also provide several other statistics to show trends over time with a line chart such as the number of sessions on each interface (typing vs. speaking), average turn-taking delay, and the number of sessions on each remote agent over time.

5 Future Research Challenges

The purpose of DialPort is to provide infrastructure to enable academic research labs to pursue their own research programs by providing them real-user data and powerful interfaces. For instance, a researcher can conduct A/B test to compare the performance of a modular-based agent with an end-to-end neural system by connecting them to real users from the Portal. At the same time, developing DialPort is a research quest as well. Many interesting and unexplored research topics are raised from DialPort. For example, we can view the learning of all remote agents plus the meta dialog manager as an online hierarchical reinforcement learning problem, where the meta dialog manager has to balance between the exploration and the exploitation [2], so that the users can have overall good user

[4] `https://dialrc.github.io/PortalAPI/`

DialPort Daily Report for 2017-X-X
Number of sessions: 20 Average number of turns per session: 6.3 Total number of turns: 145 Number of turns (Meta DM) : 65 Number of turns (Agent 1) : 30 ... Number of turns (Agent N) : 32 View logs: http://xxxx.xxx.xxx This is an automatically generated email. Contactxxx@xxx.xxx.xxx for comments/requests.

Table 2: DialPort Portal daily report

satisfaction by talking to strong systems (exploitation) and also directing enough of a data stream to poorly performing systems for them to learn better policy (exploration). Another research topic is how to model long-term memory for a dialog agent to remember salient information from previous dialog sessions with itself and other agents in order to better serve the users. Moreover, when there are multiple remote systems that offer the same domain, how can a meta dialog manager fuse their output and generate the best possible response. Since similar dialog platforms do not exist for the academic community with real users, the possibilities of new and interesting research topics are limitless.

6 Conclusions

In conclusion, this paper presents DialPort, a platform to collect real-world data for the academia community. DialPort offers the user interface, connection tool-kits and data management services to reduce the cost of an academic lab to setup a dialog system and collect data with human users. DialPort also opens up challenging and important research questions for developing better future dialog systems.

References

[1] Milica Gasic and Steve Young. Gaussian processes for pomdp-based dialogue manager optimisation. 2013.

[2] Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. Reinforcement learning: A survey. *Journal of artificial intelligence research*, 4:237–285, 1996.

[3] Eli Pincus and David R Traum. Towards automatic identification of effective clues for team word-guessing games. In *LREC*, 2016.

[4] Antoine Raux and Maxine Eskenazi. A finite-state turn-taking model for spoken dialog systems. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 629–637. Association for Computational Linguistics, 2009.

[5] Stefan Ultes, Lina M Rojas Barahona, Pei-Hao Su, David Vandyke, Dongho Kim, Inigo Casanueva, Paweł Budzianowski, Nikola Mrkšić, Tsung-Hsien Wen, Milica Gasic, et al. Pydial: A multi-domain statistical dialogue system toolkit. *Proceedings of ACL 2017, System Demonstrations*, pages 73–78, 2017.

[6] Jason D Williams, Eslam Kamal, Mokhtar Ashour, Hani Amr, Jessica Miller, and Geoffrey Zweig. Fast and easy language understanding for dialog systems with microsoft language understanding intelligent service (luis). In *SIGDIAL Conference*, pages 159–161, 2015.

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