AppDialogue: Multi-App Dialogues for Intelligent Assistants

Ming Sun [*] **Yun-Nung Chen** [*] **Zhenhao Hua** [†]
Yulian Tamres-Rudnicky [‡] **Arnab Dash** [⋆] **Alexander I. Rudnicky** [*]
[*†⋆]School of Computer Science, Carnegie Mellon University [†]Pinterest [⋆]Ernst & Young
[*] {mings,yvchen,air}@cs.cmu.edu [†‡⋆] {troyhua0426,yulie355,arnabdash92}@gmail.com

Abstract

Users will interact with an individual app on smart devices (e.g., phone, TV, car) to fulfill a specific goal (e.g. find a photographer), but users may also pursue more complex tasks that will span multiple domains and apps (e.g. plan a wedding ceremony). Planning and executing such multi-app tasks are typically managed by users, considering the required global context awareness. To investigate how users arrange domains/apps to fulfill complex tasks in their daily life, we conducted a user study on 14 participants to collect such data from their Android smart phones. This document 1) summarizes the techniques used in the data collection and 2) provides a brief statistical description of the data. This data guilds the future direction for researchers in the fields of conversational agent and personal assistant, etc. This data is available at `http://AppDialogue.com`.

Keywords: spoken dialog system; multi-domain interaction; intelligent agent.

1. Introduction

Smart devices, such as smart phones or TVs, allow users to achieve their goals (intentions) through verbal and non-verbal communication. The intention sometimes can be fulfilled in one single domain (i.e., an app). However, it is possible to span multiple domains and requires information coordination among these domains. A human user, with the global context at hand, can well-organize the functionality provided by individual apps and coordinate information efficiently. In other word, user can mentally create his own virtual app on top of existing ones. On the other hand, although intelligent agents can be configured by developers to passively support (limited) types of cross-domain interactions, they are not capable of actively managing apps to satisfy a user's potentially complex intentions, because they do not consider the repeated execution of activities in pursuit of user intentions.

Currently, most human-machine interactions are carried out via touch-screen. Although the recognizable gestures have been expanded during the past decade (Harrison et al., 2014), interactive expressions are still restricted due to the limit of gestures and displays. This may affect usability for certain populations, such as older users or users with visual disabilities. By contrast, spoken language can effectively convey the user's high-level and complex intentions to a device (e.g., Apple Siri, Amazon Alexa, Google Now and Microsoft Cortana). However speech presents the challenges about 1) understanding both at the level of individual apps and at the level of tasks that span apps; and 2) communicating a task-level functionality between user and agent. This presented data may function as a testbed for smart devices to address aforementioned challenges.

In order to investigate how human users arrange apps together in their daily life, and also to understand how they would interact with intelligent assistants via speech instead, we designed this user study. We describe data collection and involved techniques in Section 2. Then we provide statistics of this corpus in Section 3. We discuss potential research with this data in Section 4.

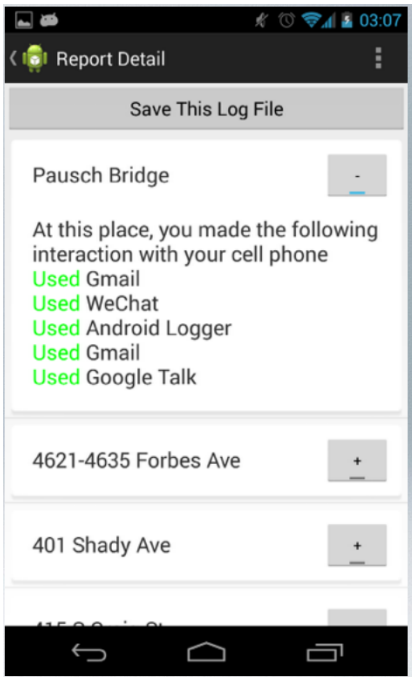

Figure 1: Logger privacy control.

2. Data Collection

To collect data for understanding high-level intentions, we propose a framework including: 1) recording users' daily app usage by an mobile app, 2) letting users annotate their own high-level tasks, and 3) interacting with a wizard-of-oz system with speech to reenact their annotated tasks.

2.1. App Recording Interface

We implemented an Android app[1] that logs each app invocation as an event, together with date/time and the phone's location (if GPS is available). Episodes were defined as a sequence of app invocations separated by periods of inactivity; based on pilot data we determined that 3 minutes was a good minimum period duration for smartphone-based ac-

[1] `https://github.com/troyhua/AndroidLogApp`

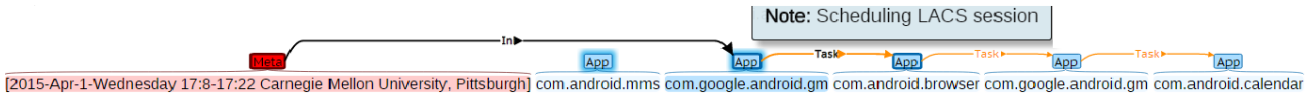

Figure 2: Multi-app annotation example; time and location are in red; constituent apps are blue. Users link apps into sequences corresponding to a particular activity (orange link).

tivities.

2.1.1. Privacy Control

Logs were uploaded by participants on a daily basis, after a privacy step that allowed them to delete episodes that they did not wish to share. As shown in Figure 1, activities happened in close time were grouped together first. Each such group was represented by an address. Participant can expand it to see further details such as apps involved (e.g., GMAIL, WECHAT in Figure 1). Participant can swipe this group to remove it from the log, if there is privacy concern. We were informed by participants that they made use of this feature. However we did not solicit further information about frequency of use or categories of events. Only information explicitly passed by the user was uploaded.

2.2. Task Annotation

Participants were invited to come to our lab on a regular basis (about once a week) to annotate their own logs and describe the nature of their smart phone activities. Uploaded data was formatted into episodes. Participants were presented with their own episodes with meta-information such as date, time, and location, to aid recall (see Figure 2). They were then asked to group events (apps) into sequences of activities (Lucchese et al., 2011) (which we will also refer to as tasks) as we had observed that episodes could include several unrelated activities. Participants were asked to produce two types of annotation, using the Brat tool (Stenetorp et al., 2012), configured for this project.

1. **Task Structure**: link applications that served a common goal/intention.
2. **Task Description**: type in a brief description of the goal or intention of the task.

For example, in Figure 2, the user first linked four applications (GMAIL, BROWSER, GMAIL and CALENDAR) in a row since they were used for the goal of planning a visit to LACS lab, and wrote a description "*scheduling LACS session*". As we observed in the collected data, some of the task descriptions are detailed. In other word, such descriptions themselves propose an app sequence (e.g., "*took a picture of my cat and then sent it to a friend*"). However, many of them are very abstract, such as "*look up math problems*" or "*schedule a study session*".

2.3. Task-Related Spoken Dialog

Participants were presented with tasks that they annotated earlier, including the meta-information (date, location, and time), their task description, and the apps that had been grouped (Meta, Desc, App lines in Figure 3). They were then asked to use a wizard system to perform the task by spoken language. The experiment took place in a lab setting. There was no effort to conceal the wizard arrangement

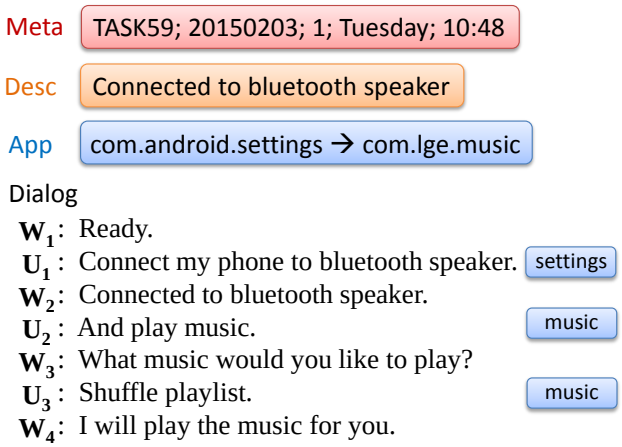

Figure 3: Multi-app task dialog example. The top display was shown to the participant; the resulting dialog is shown below.

from the participant. An assistant (21-year-old male native English speaker) interacted with the participant and was in the same space (albeit not directly visible). The wizard was instructed to respond directly to participant's goal-directed requests and to not accept out-of-domain inputs. The participants were informed that it was not necessary to follow the order of the applications used on the smart phones. Other than for remaining on-task, we did not constrain expression.

The wizard can perform very simple tasks such as "*finding a restaurant using browser*", "*composing a text message*", "*pointing and shooting a picture*", etc. When being asked to take some action, the wizard may request additional information from the user or disambiguate user's intent if necessary according to common sense. For example, the wizard would ask "*what would you like to say in the message?*" or "*which phone would you like to connect to, cell phone or work phone?*". Otherwise, the wizard simply informs the user the completion of the task, e.g., "*Ok, I will upload this picture to Facebook*".

Conversations between users (U) and the wizard (W) were recorded by Microsoft Kinect device. Recordings were manually segmented into user and wizard utterances. Each utterance is manually transcribed and also decoded by cloud speech recognizer (Google ASR). One example dialog is shown in Figure 3.

Each user utterance was later manually associated with the corresponding apps/domains that would handle it. As shown in Figure 3, SETTINGS would deal with U_1 to setup bluetooth connection and MUSIC would take care of U_2 and U_3. However, sometimes users produce utterances which may involve several apps, e.g., "*Boost my phone so I can play [game] spiderman*" requires CLEANMASTER to clear the RAM and the game SPIDERMAN. Among the total 1607 utterances, 154 (9.6%) were associated with more

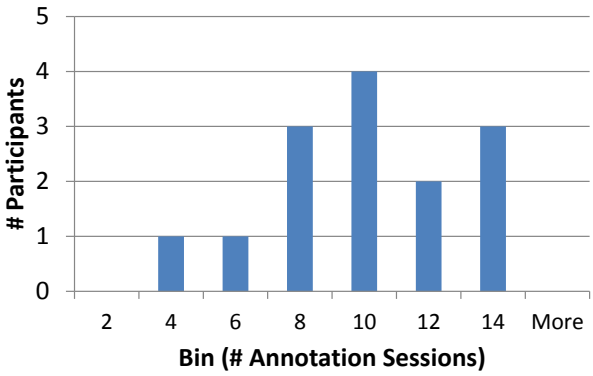

Figure 4: Histogram of number of annotation sessions

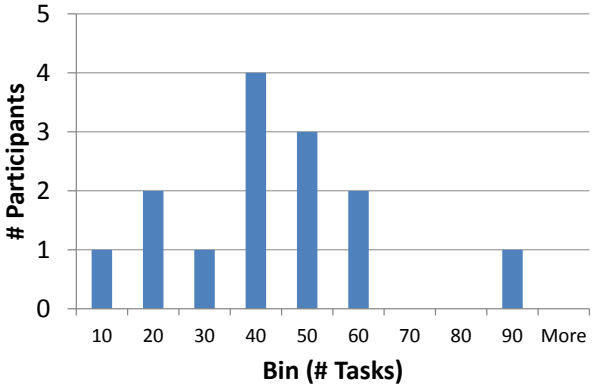

Figure 5: Histogram of number of tasks

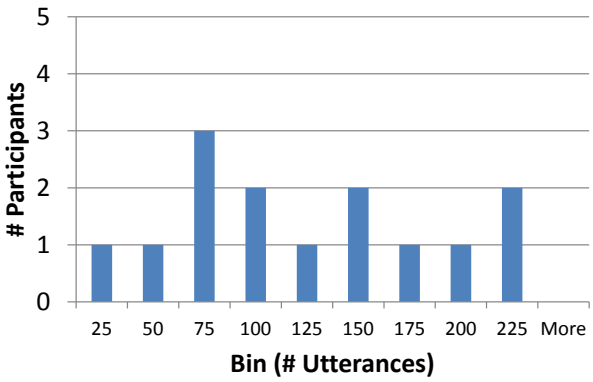

Figure 6: Histogram of number of utterances

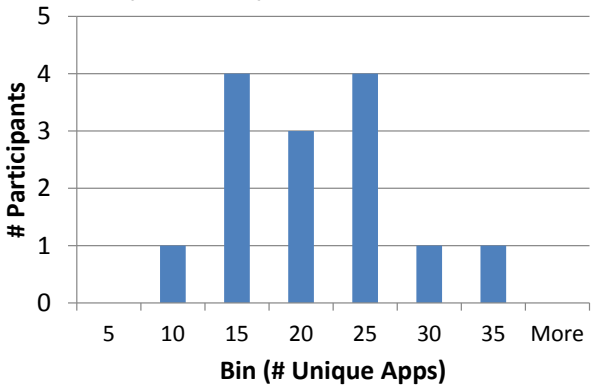

Figure 7: Histogram of number of apps

than one app — 146 requires two apps and 8 requires three.

3. Data Statistics

We recruited 14 participants who already owned Android smartphones, with OS version 4. The participants were recruited via two main channels: 1) flyers on Carnegie Mellon Pittsburgh campus and 2) Carnegie Mellon Center for Behavioral and Decision Research[2] (CBDR) participation pool. Table 1 provides the demographic breakdown.

Category	#	Age	#Apps	#Tasks	#Multi
Male	4	23.0	19.3	42.5	33.3
Female	10	34.6	19.1	36.3	32.2
Age $<$ 25	6	21.2	19.7	44.8	36.3
Age $\geq$ 25	8	38.9	18.8	33.0	29.6
Native	12	31.8	19.3	34.8	28.8
Non-native	2	28.5	18.0	57.5	55.0
Overall	14	31.3	19.1	38.1	32.5

Table 1: Corpus characteristics. A native Korean and Spanish speaker participated; both are fluent in English.

3.1. Corpus Characteristics

We collected 533 multi-app spoken dialogs with 1607 utterances (on average 3 user utterances per dialog). Among these sessions, we have 455 multi-turn dialogs (involving 2 or more user turns). The breakdown of the total 533 dialogs is shown in Table 1, where we list the number of participants (#), average age (Age), number of unique apps involved (#Apps), number of all dialogues (#Tasks) and multi-turn dialogues (#Multi). We used a cloud-based ASR engine (Google ASR) to decode all 1607 utterances, and observed a top-1 word error rate (WER) of 23% (with text normalization). On average, there are 6.6±4.6 words per user utterance. After removing stop-words[3], there are 4.1±2.5 words per utterance. Most frequent words across 14 participants are shown in Table 2.

Word	Frequency (%)
open	6.08
text	1.99
go	1.74
please	1.74
send	1.52
picture	1.50
call	1.44
check	1.20
facebook	1.16
message	1.16

Table 2: Top content words and frequency.

3.2. Analysis

Participants dropped out of the study at different times (see Figure 4). On average, each participant annotated 42.4±21.6 logs during the study. Note that participant submitted one log per day. In each visit to our lab (less than

[2] http://cbdr.cmu.edu/

[3] http://www.nltk.org/book/ch02.html

Field	Detailed Explanation
user	Annoymized User ID
phone_interaction_day	Day of a week when the real life smart phone interaction (not speech) happened
phone_interaction_time	Time when the real life smart phone interaction (not speech) happened
phone_interaction_location	Address when the real life smart phone interaction (not speech) happened (street number removed). N/A indicated unavailability of GPS.
gender	male or female
age	young (≤ 25) or old (> 25)
native	first language is English or not
speech_interaction_date	Date when the speech version of the real life smart phone interaction happened
tarsk_ord	The ordinal task ID for one user's multi-app tasks
utt_id	Utterance ID within one multi-app task
apps	Apps (in real-life smart phone interaction) to handle the current user utterance
googleplay_categories	Google Play categories for current apps
transcription	Manually transcribed user utterance by a native English speaker
asr_hypothesis	Google Automatic Speech Recognition (Google ASR) top-1 hypothesis
sys_response	System (wizard) response to the user utterance
task_description	Task description user annotated regarding the nature of the multi-app task

Table 3: Detailed explanation of the fields in the data

1 hour), we asked participants to annotate as much as possible. On average, they annotated 4.3±1.5 logs per visit. Some participants have more than one multi-app tasks per day, while others have less. On average, in our collection, each participant has 1.03±0.70 such tasks per day.

Figure 5 and Figure 6 show the distribution of number of tasks and utterances over 14 participants. The correlation between individual participant's total number of tasks and total number of utterances is strong (r=0.92), which is intuitive.

In total, there are 130 unique apps across 14 users. On average, each user has 19.1±6.1 unique apps. The distribution of number of apps in shown in Figure 7. The correlation between individual's number of unique apps and number of tasks is moderate (r=0.65). The more multi-app tasks a user performs, the more unique apps are involved across these tasks.

3.3. Meta-data Description

In the raw data, the first row is the header and the rest is the content. Table 3 explains each field in the header. We anonymize user names by replacing them with unique IDs. Due to its sensitivity Location (address) information is anonymized by removing some of the detail (such as street number).

4. Discussion

In this section, we briefly discuss two of the several potential research directions based on this dataset, namely follow-up app prediction and virtual app construction in the context of multi-domain personal assistant.

It has been shown that based on simple context such as time or location, intelligent agents on smart phones that anticipate a user's needs can surface or launch specific apps before they are needed. This can significantly improve the efficiency of navigating the apps (Shin et al., 2012; Vetek et al., 2009). However, language interaction may provide additional context. For example, by hearing "please find a restaurant", the agent not only knows that the user wants a recommendation of a restaurant *now*, it may also offer the bus information or navigation to some restaurant since it is likely to be the *next* action the user may want to take. There are several different ways to provide assistance based on the predicted next app or domain, e.g., the agent may proactively offer information of the predicted domain (the restaurant-bus example above) or launch app in the background and promote the app to the top position in OS stack. This dataset provides contexts such as language, date, time to build context-aware prediction models (Sun et al., 2015). This data can also helps the agent to understand the current intent within apps (Chen et al., 2015; Chen et al., 2016).

Nowadays, users mentally arrange a set of functionality provided by individual apps to fulfill more complex task/intention for which an app does not exist. For example, to *schedule meeting*, CALENDAR and EMAIL would be involved. Providing users with the ability to construct a virtual app composed from existing apps will allow them to develop custom functionality suited to their needs. User should be able to command the agent with abstract high-level language such as "can you schedule me a meeting with Alex for next week", instead of providing detailed individual steps such as "check my availability for next week", "send Alex an email and ask if he can meet on

Cluster	Item Examples (task descriptions supplied by participant)
1	"Picture messaging XXX", "Take picture and send to XXX"
2	"Look up math problems", "Doing physics homework", "Listening to and trying to buy a new song"
3	"Talking with XXX about the step challenge", "Looking at my step count and then talking to XXX about the step challenge"
4	"Playing [game] spiderman", "Allocating memory for spiderman"
5	"Using calculus software", "Purchasing Wolfram Alpha on the play store"
6	"Texting and calling XXX", "Ask XXX if she can talk then call her"
7	"Talking and sharing with group mates", "Emailing and texting group members"

Table 4: Intention clustering of tasks based on utterances, with typical descriptions.

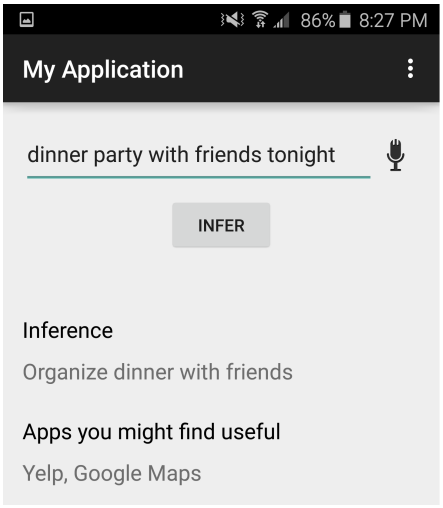

Figure 8: Example of understanding user's intention. **Inference** indicates the recognized intention represented by natural language; supportive apps are also displayed.

Monday". To learn such virtual app (aka high-level intentions), intelligent agent can either ask user for instruction, or implicitly observe recurring activities and abstract their structure (sequence of apps). This dataset can be used to investigate the latter approach (Sun et al., 2016a; Sun et al., 2016b). Users label each activity with a language description such as "contacting group members". Although the language may vary even for similar tasks, it is possible to use clustering techniques on top of semantic relatedness to group related tasks together (see Table 4). Thus, new speech input can be associated with one of the learned virtual apps. Thus, the agent can smoothly transit user to the next domain, or even create a unified conversation from each elemental app to reduce redundancy. One example is shown in Figure 8.

5. Conclusion and Future Work

In this paper we described a corpus on natural interactions with a smartphone, based on a user's actual everyday activity. We present a process of 1) logging user's real-life smart phone activities and 2) collecting their paired speech version through a wizard-of-oz system. This parallel corpus provides insight into how users would interact with personal assistant to fulfill complex intentions, which involve the coordination among multiple domains (apps). Our long-term goal is to build such intelligent agent to understand user's intentions and facilitate the interaction accordingly. This dataset provides a test bed for such research.

6. Acknowledgment

This research was supported in part by YAHOO! InMind project, and by the General Motors Advanced Technical Center. We thank Chenran Li and Avnish Saraf for creating the demo[4] as shown in Figure 8.

7. Bibliographical References

Chen, Y.-N., Sun, M., and Rudnicky, A. I. (2015). Leveraging behavioral patterns of mobile applications for personalized spoken language understanding. In *Proceedings of 2015 International Conference on Multimodal Interaction (ICMI)*.

Chen, Y.-N., Sun, M., Rudnicky, A. I., and Gershman, A. (2016). Unsupervised user intent modeling by feature-enriched matrix factorization. In *Proceedings of The 41st IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

Harrison, C., Xiao, R., Schwarz, J., and Hudson, S. E. (2014). Touchtools: leveraging familiarity and skill with physical tools to augment touch interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2913–2916.

Lucchese, C., Orlando, S., Perego, R., Silvestri, F., and Tolomei., G. (2011). Identifying task-based sessions in search engine query logs. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 277–286. ACM.

Shin, C., Hong, J.-H., and Dey, A. K. (2012). Understanding and prediction of mobile application usage for smart phones. In *ACM Conference on Ubiquitous Computing*, pages 173–182. ACM.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). BRAT: a web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107. Association for Computational Linguistics.

[4]Video: `https://youtu.be/FvQto8pP1OU`

Sun, M., Chen, Y.-N., and Rudnicky, A. I. (2015). Understanding user's cross-domain intentions in spoken dialog systems. In *NIPS workshop on Machine Learning for SLU and Interaction*.

Sun, M., Chen, Y.-N., and Rudnicky, A. I. (2016a). HELPR a framework to break the barrier across domains in spoken dialog systems. In *International Workshop on Spoken Dialog Systems*.

Sun, M., Chen, Y.-N., and Rudnicky, A. I. (2016b). An intelligent assistant for high-level task understanding. In *ACM Conference on Intelligent User Interfaces*.

Vetek, A., Flanagan, J. A., Colley, A., and Kernen, T. (2009). Smartactions: Context-aware mobile phone shortcuts. In *Human-Computer Interaction*, pages 796–799.