Crowdsourcing a self-evolving dialog graph

Patrik Jonell
KTH Royal Institute of Technology
pjjonell@kth.se

Per Fallgren
KTH Royal Institute of Technology
perfall@kth.se

Fethiye Irmak Doğan
KTH Royal Institute of Technology
fidogan@kth.se

José Lopes
Heriot-Watt University
jd.lopes@hw.ac.uk

Ulme Wennberg
KTH Royal Institute of Technology
ulme@kth.se

Gabriel Skantze
KTH Royal Institute of Technology
skantze@kth.se

ABSTRACT

In this paper we present a crowdsourcing-based approach for collecting dialog data for a social chat dialog system, which gradually builds a dialog graph from actual user responses and crowd-sourced system answers, conditioned by a given persona and other instructions. This approach was tested during the second instalment of the Amazon Alexa Prize 2018 (AP2018), both for the data collection and to feed a simple dialog system which would use the graph to provide answers. As users interacted with the system, a graph which maintained the structure of the dialogs was built, identifying parts where more coverage was needed. In an offline evaluation, we have compared the corpus collected during the competition with other potential corpora for training chatbots, including movie subtitles, online chat forums and conversational data. The results show that the proposed methodology creates data that is more representative of actual user utterances, and leads to more coherent and engaging answers from the agent. An implementation of the proposed method is available as open-source code.

CCS CONCEPTS

• **Computing methodologies** → *Discourse, dialogue and pragmatics.*

KEYWORDS

datasets, dialog systems, crowdsourcing, human-computer interaction

ACM Reference Format:
Patrik Jonell, Per Fallgren, Fethiye Irmak Doğan, José Lopes, Ulme Wennberg, and Gabriel Skantze. 2019. Crowdsourcing a self-evolving dialog graph. In *1st International Conference on Conversational User Interfaces (CUI 2019), August 22–23, 2019, Dublin, Ireland*. ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3342775.3342790

CUI 2019, August 22–23, 2019, Dublin, Ireland

ACM ISBN 978-1-4503-7187-2/19/08…$15.00
https://doi.org/10.1145/3342775.3342790

1 INTRODUCTION

Dialog data is an important resource when implementing dialog systems. Such data can either be directly used to train a statistical dialog model, or as a source of inspiration and knowledge for experts handcrafting a rule-based dialog manager. Several dialog corpora have been explored for these purposes, such as OpenSubtitles [24], Switchboard [7], and the Microsoft Research Social Media Conversation Corpus [30]. However, it is not clear whether these corpora are actually well suited for dialog systems, since they might not be representative of such interactions. The use of generic, already existing, dialog datasets has reportedly resulted in systems that generate very general responses (e.g. I don't know) [29]. An alternative to using such datasets is to collect large amounts of data in more controlled conditions which are more similar to a human-computer interaction context. In the past, the Wizard-of-Oz (WoZ) methodology [14] has been used for this purpose, but WoZ data collections are typically expensive to conduct.

In this paper, we explore the use of crowdsourcing for the collection of coherent, engaging and structured dialogs, conditioned on a pre-defined persona. More specifically, we want to investigate *to what extent dialog authoring can be made into a collective process using a crowd of non-experts?* In our approach, the data is gradually collected and structured in a graph, as users interact with a dialog system. At the same time, the system automatically identifies parts of conversations that need more data and creates tasks for crowdworkers to author suitable system replies.

This methodology was tested during the second installment of the Amazon Alexa Prize 2018 (AP2018), a student competition funded by Amazon with the purpose of accelerating the field of conversational artificial intelligence [9]. Our contribution, called Fantom, was accepted as one of eight participating teams, chosen from a pool of 195 applications. During a period of 3 months, our dialog system interacted with approximately 75,000 unique Amazon customers in roughly 100,000 conversations, resulting in a dialog graph with 46,000 user nodes and 6,500 system nodes suggested by crowdworkers. By representing the data as a graph, we achieve a succinct representation of the corpus. However, collecting dialog data as a collective process has its own challenges, and it is not certain that crowdworkers are able to provide system responses which are coherent and leads to an engaging interaction. In this paper, we will describe how we approached this problem. To evaluate the merits of our approach, we have compared the data we collected during the competition, to other dialog corpora that have been used to build social chatbots.

2 RELATED WORK

Dialog systems can either be hand-coded or based on data-driven methods (or a combination of these). As examples of data-driven approaches, most chatbots of today are trained using example interactions. Hand-coded systems use rules to interpret user utterances and determine the appropriate system response. This approach has the advantage that an expert is in charge of what the system says, and thus avoids the issues with incoherent or offensive replies. Dialog rules can either be manually hand crafted or derived from example interactions. Although there is a wide-range of tools available to craft rule-based dialogs [4, 10, 11, 21, 34], hand-coding is extremely time-consuming and does not scale well with an open-domain chat system. Consequently a considerable amount of research has been done on how to learn from large datasets.

The interactions used for training such models can, for example, be collected from Twitter [12], movie subtitles [31], or the user responses in conversations between humans and an already existing bot, e.g. Cleverbot [3], and Microsoft Tay [32]. However, existing approaches based on these data sets have not resulted in systems capable of holding engaging interactions with a user for any length of time. One reason for this is that the dialog data they are based on are not specifically targeted toward engaging human-computer interaction with a coherent personality. This approach also risks resulting in a bot saying inappropriate things (as shown by the Microsoft Tay experience [32]). Therefore, a more controlled way of generating data might be desirable.

A compromise between using a data-driven and a hand-coded approaches is to use crowdsourcing. This allows for collecting large amounts of data, while at the same time having control over how the data is collected. One of the first examples of crowdsourcing for dialog generation is Orkin et al. [25]. Similarly, Breazeal et al. proposed a data-driven approach to dialog generation for a social robot by crowdsourcing dialog and action data from an online multi-player game [2]. There has also been work on the collection of text-based corpora. For example, Filatova investigated irony and sarcasm by creating a corpus based on Amazon reviews [5]. More recently, Ben Krause et al. collected a dialog dataset using crowdworkers in order to train a generative RNN. They asked the crowdworkers to self-author complete dialogs [16]. Leite et al. presented a graph based approach using crowdworkers when letting participants interact with a robotic head [18]. The use of crowdworkers to build dialogs has also been done for task-oriented dialogs [33].

In order to avoid issues with utterances in dialog datasets being incoherent or contradictory, a persona can be used to guide the crowdworkers. Work has been done to condition a conversation model with a given persona [19] or profile information from the sender/receiver of a message [35]. Huang et al. let crowdworkers chat one-on-one, giving each one a list of personality traits to adhere to [8]. In this case, the user roles were symmetrical. While this can be desirable in certain cases, the roles in human-machine interaction are often asymmetrical, which should also be induced in the data collection.

3 BUILDING A DIALOG CORPUS USING CROWDSOURCING

In this section we will describe the approach used to build the dialog corpus. The dialog graph serves as a representation of the dialog corpus, but can also be used directly as a rudimentary dialog manager in a dialog system, which we did during the competition. The corpus is built using a graph structure, as depicted in Figure 1. The dialog graph differentiates between system utterances, which are authored by crowdworkers, and user utterances, which are obtained by using the dialog graph to guide the dialog system and letting users interact with the system. We will expand on these topics in more detail below. Further implementation details can be found in the code repository.

3.1 Dialog graph

The dialog graph contains nodes representing an utterance (or rather, a class of synonymous utterances), either from the system (system nodes) or from the user (user nodes). An edge from node X to node Y means that any utterance in Y is an appropriate answer to any utterance in X. An edge can therefore only connect a system node to a user node or vice versa; there is never an edge between two system nodes or two user nodes. A path through the graph thus represents a dialog between user and system. A path ending in a user node can easily be extended by adding a new system node to the last user node, using a graphical interface showing the preceding dialog to the crowdworker (see Figure 2).

With this approach, many workers can contribute and independently extend the graph. The graph is also extended with user nodes retrieved from real interactions with the system. Consequently, each path in the graph will represent a coherent dialog with system utterances. More specifically, the graph can be described as a collection of trees, each rooted in a user node. Each tree is typically a topic of its own, and the root node is context-independent (e.g. it can serve as a starting utterance in a conversation, or as a start of a new topic).

3.2 Crowdsourcing

In order to populate the dialog graph with system nodes, Amazon Mechanical Turk can be used by automatically posting tasks where workers are asked to continue the dialog based on a given dialog history (see Figure 2). The workers are then asked to both validate the previous dialog, and to author an appropriate response. To guide the worker, a brief system persona is shown on the right. For a more detailed description of the persona, see the next section.

The dialog history is created by traversing the graph backwards from a leaf node that requires a new system response by picking a random utterance from each node, up to 6 utterances back.

In order to ensure high quality of the responses, the worker is first asked to validate the dialog history, and to mark any utterances deemed to be incoherent. Failing to do so results in failing the task. Failing to report a dialog as incoherent if other workers have marked it incoherent, would result in the worker failing the task, and thus not getting paid. This is a strategy for 1) automatically finding inadequate utterances and workers not following instructions, and 2) letting workers know that their work is evaluated by other workers. The system also makes sure that a worker

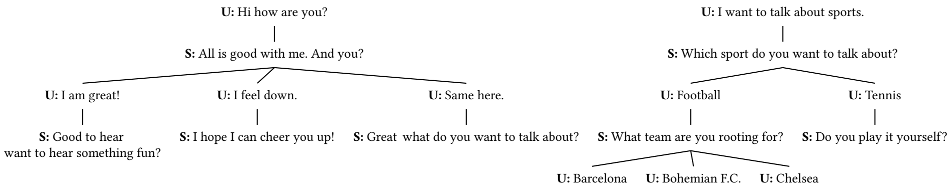

U: Hi how are you?
S: All is good with me. And you?
U: I am great!
U: I feel down.
U: Same here.
S: Good to hear want to hear something fun?
S: I hope I can cheer you up!
S: Great what do you want to talk about?
U: I want to talk about sports.
S: Which sport do you want to talk about?
U: Football
U: Tennis
S: What team are you rooting for?
S: Do you play it yourself?
U: Barcelona
U: Bohemian F.C.
U: Chelsea

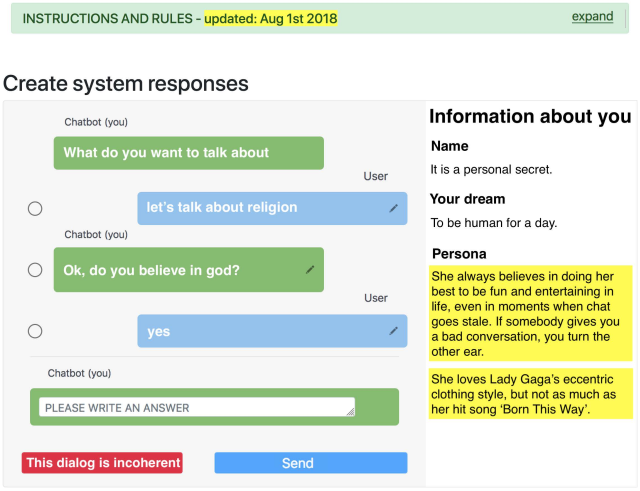
Figure 1: Part of the graph built for the social bot competing in AP2018 (hypothetical data), for viewing purposes only one utterance per node is shown. U: User utterance, S: System utterance.

never receives a dialog history which they have already contributed to themselves. Additionally the worker is required to complete a training procedure before being able to work on tasks.

While crowdsourcing is used to collect system responses, a dialog manager is needed for collecting user responses. During AP2018, a rudimentary system based on the dialog graph was set in place. In brief, the incoming user utterance is matched toward every root node. If the utterance matches a user node X with a high enough similarity score, the system's reply is chosen from the system nodes connected to X. The next user utterance is then matched with each of the children of that system node. If the user utterance does not match with a high enough similarity score, it is again compared to all root nodes (as the user might have changed topic). The method is described in more detail in work by Jonell et al. [13].

3.3 Persona

We discovered early on that users often wanted to ask the chatbot agent about its preferences when talking about for example music or sports, or ask where the agent lives, etc. – something that is typical for any social chat-oriented interaction. In order to acquire more coherent answers between crowdworkers, and a consistent personality of the system, a persona was designed. The persona was constructed with the aim of reflecting popular opinions, beliefs and sympathetic traits in general. A challenge here is to present the right amount of details about the persona to the crowdworker. On the one hand, providing more information allows the crowdworker to know more about the intended personality, and thus makes the suggested utterances more uniform and coherent. On the other hand, as noticed during AP2018, too long instructions may result in the crowdworkers paying less attention to them, or in the worst case, disregarding the persona completely. To balance this, we created a pool of 30 coherent persona attributes, and then randomly selected two of these attributes to show to the crowdworker for every task. An example of this can be seen to the right in Figure 2. As each crowdworker often did many tasks, they were over time given a large number of coherent attributes, without having to read and digest all of them at once.



INSTRUCTIONS AND RULES - updated: Aug 1st 2018

expand

Create system responses

Chatbot (you)

What do you want to talk about

User

○ let's talk about religion

Chatbot (you)

○ Ok, do you believe in god?

User

○ yes

Chatbot (you)

PLEASE WRITE AN ANSWER

This dialog is incoherent

Send

Information about you

Name

It is a personal secret.

Your dream

To be human for a day.

Persona

She always believes in doing her best to be fun and entertaining in life, even in moments when chat goes stale. If somebody gives you a bad conversation, you turn the other ear.

She loves Lady Gaga's eccentric clothing style, but not as much as her hit song 'Born This Way'.

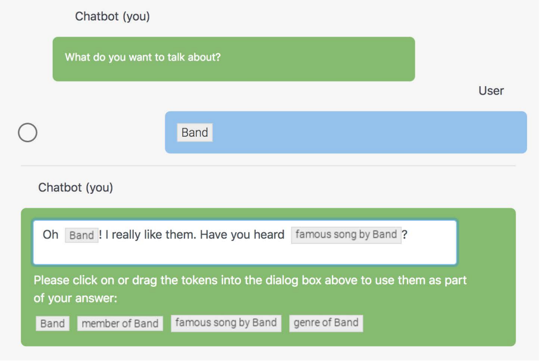
Figure 2: Interface used for authoring content in the graph

3.4 Scaling the Graph

It is important to consider scalability measures early on in the process, as the dialog graph otherwise might become difficult to maintain and contain vast amounts of redundant data. An example of this is if two semantically equivalent utterances (like "What is your favorite movie" and "What's the name of your favorite movie") would end up as two different nodes. The graph would then populate both branches with similar user answers and treat them as completely different dialogs, thereby resulting in two redundant branches. Below we describe a few measures attempting to address these issues.

3.4.1 Automatic management of nodes. The task of merging semantically equivalent nodes is not trivial, especially if the property of a fully valid graph is to be kept. During the AP2018 several measures were taken for merging nodes. One such measure was using *synonym nodes*; the system would look for synonyms (using a manually constructed list of semantically similar phrases) among a node's children and merge them together into one node, and also providing missing synonyms to these nodes. Another measure was to use common patterns for semantically equivalent nodes, such as '*Let's talk about X*' and '*Let's chat about X*' and merge them. The similarity score used in the dialog manager (See details in [13]) was used in order to calculate a score for potential merges between every node at the same level in the dialog graph. Depending on the output of the scoring function and how important the validity property of the dialog graph is, this can be done either automatically, manually or semi-automatically by posting merging tasks to Amazon Mechanical Turk.

Another crucial issue is to be able to distinguish between utterances that are contextually dependent and independent. A contextually independent utterance should make sense at any time in the conversation, for example to start a new topic, such as '*let's chat*



Chatbot (you)

What do you want to talk about?

User

○ Band

Chatbot (you)

Oh Band! I really like them. Have you heard famous song by Band?

Please click on or drag the tokens into the dialog box above to use them as part of your answer:

Band member of Band famous song by Band genre of Band

Figure 3: Crowdsourcing interface for tag nodes.

Table 1: Size of the graph and node visits (the number of times a user utterance matched the given node) at different depths of the dialog graph. User nodes at *depth* = 1 are root nodes. The two rightmost columns present median values together with the 90th percentile. Data from August 15th 2018.

Depth	User/System	Node count	Utterance count	Utterances per node	Node visits
1	User	2124	3355	1 (2)	4 (37.0)
	System	2058	2205	1 (1)	7 (68.4)
2	User	34188	92781	1 (1)	1 (2.0)
	System	3402	3567	1 (1)	2 (10.0)
3	User	7640	39221	1 (23)	1 (2.0)
	System	743	777	1 (1)	2 (8.0)
4	User	746	6771	1 (49)	1 (2.0)
	System	122	122	1 (1)	1 (4.9)
5	User	43	812	1 (49)	3 (2.0)
	System	2	2	1 (1)	1 (4.6)

about music' or '*what's your opinion on inhabiting Mars?*'. These can be contrasted with utterances that are contextually dependent, such as '*I like them*' or '*yeah, I think so too*'. The former are used as root nodes in the dialog graph, while the latter should only exist as children of other nodes in the graph. It is therefore important to be able to classify an utterance's contextual dependency in order to avoid populating the graph with context-dependent utterances at the root node level by mistake. In order to classify context dependent and independent nodes a root node classifier was trained and used, detailed in [13].

3.4.2 Tags for Named Entities. In order for the dialog graph to be able to store more generalized utterances in one node, tags were used to represent variable content, such as named entities. For example, if the system asked '*What is your favorite artist?*', the possible answers from users could potentially generate thousands of child nodes, if the answers were not generalized. A simple example of the tag mechanism is illustrated in Figure 3.

In order to generate such tags, a knowledge graph database such as Wikidata [6] can be used. This way, tags can also be associated with related attributes, as in the example above (*Band* and *famous song by Band*). A way of automatically finding potential tags in the dialog graph is by performing an entity information lookup for every node in the graph. If a certain percentage of sibling nodes belong to the same category, that category can become a tag, and related attributes retrieved and used as part of the crowdsourcing.

4 CORPUS COLLECTED DURING THE AMAZON ALEXA PRIZE 2018

In this section, observations regarding the system and experimentation results which were made during the course of AP2018 are described.

4.1 Analysis of the collected data

Toward the end of the competition the dialog graph consisted of approximately 50,000 active nodes. Detailed data about the size of the collected dialog graph is shown in Table 1 (for further details about the dialog graph see [13]). As can be seen, the number of user nodes at depth 2 is significantly higher than that of other levels. This is due to a large amount of system responses such as '*What is your favorite movie*' or '*Do you have a favorite rock band*' at depth 1 resulting in a wide variety of answers from users. This observation motivated work on scalability-enabling features such as tags for named entities and automatic merging of semantically similar nodes (see Section 3.4). The high percentiles of utterances per node on the user levels at depth 3, 4 and 5 is a result of the automatic merging. Most likely, the system found an utterance matching a simple phrase, e.g. '*yes*' or '*not really*', and populated these nodes with a number of semantically similar phrases.

As a result of Fantom's involvement in AP2018, the data in the dialog graph was heavily influenced by what topics users found interesting at the particular time of the interaction. Therefore many utterances correspond to trending topics such as popular music, video games or movies, national holidays, sports results, greetings depending on time of day, and also the occasional profane dialog from a user harassing the chatbot. Furthermore, as encouraged by Amazon when marketing the Alexa Prize, many dialogs start with '*Let's chat about X*' or '*What do you know about Y?*', which set the topical trend for many of the dialog paths in the graph. For instance, as many users were interested in talking about music, celebrities and movies, a lot of the data correspond to these topics. This is something to keep in mind when building the dialog graph: the utterances of users, specifically the root nodes, impact the topical nature of the collected data to a high degree.

Given the exposure of the AP2018, over 75,000 users interacted with the system and thereby contributed to the dialog graph. To utilize the user interaction data in our crowdsourcing system we first had to make sure that the data was properly anonymized so that no personal data was released to the public. For this purpose, we built an anonymization tool which prioritized utterances in respect to frequency, but each utterance had to be manually deemed anonymous in order to be used. During the course of the competition, Amazon formalized the process for anonymization of user data. The substantial user base gave rise to a significantly higher amount of user nodes compared to system nodes, and further motivates the importance of smart solutions that help maintain a balanced and coherent dialog graph. The magnitude of this user base is not the typical use case however, and it's likely that a dialog graph deployed in a different environment will generate a more balanced ratio. Ultimately, the size and balance of the graph comes down to an issue of time and money, and/or to what extent crowdsourcing is being used.

Table 2: Comparison of both qualitative and quantitative measures between different dialog corpora.

	Dialog graph	**Reddit**	**Twitter**	**Switchboard**	**BNC**	**Self-dialog**	**OpenSubtitles**
Control over input data	Yes	No	No	No	No	Yes	No
Yields large datasets	No	Yes	Yes	Yes	Yes	No	Yes
Cost	High	Low	Low	High	High	High	Low
Spontaneous/ Scripted	Both	Spontaneous	Spontaneous	Spontaneous	Spontaneous	Scripted	Scripted
Collective authoring	Yes	Yes	Yes	Yes	Yes	No	N/A
Linear branching structure	Branching	Linear	Linear	Linear	Linear	Linear	Linear
Trending topics	Yes	Yes	Yes	No	No	Yes	No
# Tokens	118,304	2,053,916	9,586	868,731	11,887,763	4,401,338	353,387
# Token Types	9,147	62,854	1,591	12,926	51,791	41,141	9,278
# Utterances	19,701	100,178	784	128,790	1,037,619	375,386	48,951
TTR	0.077	0.031	**0.166**	0.015	0.004	0.009	0.026
MSTTR	**0.811**	0.756	0.727	0.616	0.667	0.684	0.600
LS	**0.464**	0.298	0.255	0.167	0.228	0.207	0.328
% High quality utterance pairs	**75%**	19%	36%	48%	23%	54%	28%

5 COMPARISON TO OTHER DIALOG DATA SOURCES

In this section, we compare the corpus collected during AP2018 with a variety of other common dialog data sources, in terms of how useful they are for developing social chat systems. The columns of Table 2 show examples of some of the most common corpora with regard to conversational data (as there is an array of different corpora we do not claim that the list is exhaustive) and the rows represent certain properties of each method. *Reddit* refers to extracted user posts from January 2015 that were deemed usable as conversation data [28]. The Twitter data was collected and curated through crowdsourcing by [30], available as the Microsoft Research Social Media Conversation Corpus[1]. British National Corpus [26] and SwitchBoard [7] contain transcribed human-human conversations. Self-dialog refers to a crowd-sourced corpus collected by Edina [16], a team that took part in the first instalment of the Alexa Prize. The corpus was built using a self-authoring strategy, where an author wrote both parts of the conversation. Finally, subtitles from television and movies can be extracted and used as conversational data. An example of this is OpenSubtitles [20].

To build an engaging and coherent chatbot (which was the goal of AP2018), it is essential to have control over the data collection process. For such applications, methods that generate large quantities quickly, e.g. scraping forums or extracting subtitles, might be less attractive - as they bring less control. The dialog graph and self-authoring approaches provide higher control, but at the cost of yielding lower quantities of data, in combination with the monetary cost and longer setup time. The naturalness of the dialog data might be affected by the text source (written or transcribed from speech) and how the data was authored; a single author that produces a dialog might produce a narrow type of narrative because they are limited to their own creativity ([16] does however claim that a single author still can provide interesting dialog). Furthermore, a potential benefit with the dialog graph approach is that it differs in terms of its branching structure to other corpora (as seen in Figure 1). In other words, a branching dialog can take many different directions depending on what is said, whereas a linear dialog only has one specific path. An additional strength with approaches that have high control over the data collection process is that trending events or other time-based phenomena can be utilized and incorporated into the data. By only extracting subtitles, the data will for instance not reflect specific dialogs about the 2018 FIFA World Cup or Elon Musk sending cars to space (topics that may be of interest depending on application area).

To further highlight the benefits of our approach we have computed the measures specified in [22] for the different corpora used. These metrics were already used in [23] to assess the richness of corpora created to be used in language generation tasks for task oriented dialog systems. Table 2 shows the number of tokens, number of different types of tokens, two versions of the token-type ratio (the original one (TTR) and the mean segmental one (MSTTR)) and the lexical sophistication (LS), which accounts for the percentage of lexical word types not in the list of the 2,000 most frequent words from the British National Corpus. The higher the figures are, the more complex the dataset is. The results show that the dialog graph corpus, compared to the other corpora, had the highest MSTTR and LS scores, supporting that the proposed method can generate varied and rich lexical content. The metric "High quality utterance pairs" will be discussed in Section 6.

5.1 Evaluation

An evaluation was conducted in order to determine if utterances from the various corpora were (1) similar to what experts would expect users to say in a conversation with a social chat bot, and if so (2) whether the given response by the system was coherent with what the user said, and (3) if the system response was engaging (i.e. interesting and helpful for continuing the conversations, as defined by [15]).

5.1.1 Evaluators. 14 evaluators ($age = 34 \pm 11, female = 3, male = 10, other = 1$) working in the fields of speech, dialog and machine learning participated in rating the corpora on the four metrics.

5.1.2 Task. 200 utterances from each corpus were evaluated, totalling 1,400 utterance pairs, evenly distributed among the evaluators. An utterance pair consisted of a user utterance followed by a system response. For the dialog graph, each utterance belonged to one of these categories, but for the other corpora these categories were consecutively assigned. Each utterance pair was presented to one evaluator, who was asked to answer (on a likert scale, 0-4) how likely they thought it would be to find the given user utterance in a social chatbot dialog. If they answered with a score of 2 ("neither

[1] The corpus is actually bigger compared to the numbers presented in the table, but the corpus is based on tweet ids, and many tweets are since removed.

Table 3: Summary of four linear mixed effect models with each metric as a dependent variable comparing the dialog graph (Intercept) with the other corpora. Results that are significant ($p < 0.05$) are marked in bold.

	Likely user utterance				*Coherent system response*				*System resp. helpful to continue conv.*				*Interesting system response*			
	Estimate	Std. Error	t value	p value	Estimate	Std. Error	t value	p value	Estimate	Std. Error	t value	p value	Estimate	Std. Error	t value	p value
Dialog graph (Intercept)	**3.20**	**0.18**	**17.57**	**<0.001**	**3.41**	**0.11**	**29.94**	**<0.001**	**3.29**	**0.12**	**27.51**	**<0.001**	**2.48**	**0.11**	**23.37**	**<0.001**
BNC	**-1.27**	**0.10**	**-12.85**	**<0.001**	**-1.27**	**0.12**	**-10.96**	**<0.001**	**-1.25**	**0.10**	**-12.13**	**<0.001**	**-0.57**	**0.10**	**-5.94**	**<0.001**
Self-dialog	**-0.60**	—"—	**-6.02**	**<0.001**	**-0.23**	**0.11**	**-2.21**	**0.03**	**-0.50**	**0.09**	**-5.35**	**<0.001**	0.02	0.09	0.20	0.85
OpenSubtitles	**-1.16**	—"—	**-11.73**	**<0.001**	**-0.90**	**0.11**	**-7.99**	**<0.001**	**-1.02**	**0.10**	**-10.20**	**<0.001**	-0.10	0.09	-1.09	0.28
Reddit	**-1.48**	—"—	**-14.97**	**<0.001**	**-0.94**	**0.12**	**-7.68**	**<0.001**	**-1.01**	**0.11**	**-9.38**	**<0.001**	0.11	0.10	1.04	0.30
Switchboard	**-0.79**	—"—	**-7.99**	**<0.001**	**-0.41**	**0.11**	**-3.73**	**<0.001**	**-0.75**	**0.10**	**-7.73**	**<0.001**	**-0.28**	**0.09**	**-3.09**	**<0.01**
Twitter	**-1.21**	—"—	**-12.19**	**<0.001**	**-0.58**	**0.12**	**-4.97**	**<0.001**	**-0.70**	**0.10**	**-6.74**	**<0.001**	0.12	0.10	1.19	0.23

unlikely or likely") or above, they were presented with three questions regarding the system answer: (1) "How coherent is the system answer with the user's utterance?", (2) "How interesting/fun is the system answer?", and (3) "How well does the system answer help to continue the conversation?". These answers were also given on a likert scale, 0-4.

5.1.3 Utterance selection. User—system utterance pairs were extracted from all corpora. This was done by randomly selecting an utterance in the corpus that had a response by another person following it. Furthermore, in order to avoid particularly short or long utterances, only utterances with 5 to 10 words were chosen. A filtering process was also applied before presenting the utterance to the annotator by lowercasing all utterances, fix white space issues around punctuations, removal of common internet symbols (e.g. smileys) and substituting username handles (starting with an @) or references to "Alexa" with a random name from a list of common names.

5.1.4 Results. R [27] and lme4 [1] were used to perform a linear mixed effects regression on the four metrics. A separate model was used for each metric with the corpus as a fixed effect and the evaluator as a random intercept. Per item random intercept was not used, since utterance pairs were unique and only viewed by one evaluator once. A model with random slope and intercept for evaluator resulted in a singular fit (for all four models) and was therefore not used. The "Likely user utterance" model had 1400 observations while the other models had 995 observations provided by 14 evaluators. P-values were estimated via the Satterthwaite approximation with lmerTest [17] and a detailed summary is provided in Table 3. All reported differences below are statistically significant ($p < 0.05$).

The dialog graph corpus was rated significantly higher than all the other corpora on how likely the user utterance was to being said to a social chatbot. The dialog graph responses were rated more coherent than all of the other corpora. System answers from the dialog graph were rated higher in being helpful to continue the dialog compared to all other corpora. There was no difference between most corpora regarding how interesting the system utterance was, except BNC and Switchboard which were rated lower than the dialog graph. The results of the evaluation are also shown in Figure 4.

6 DISCUSSION

6.1 Analysis of results

The dialog graph corpus was rated as having the most likely user utterances. This was expected, since it was in fact the only corpus which had utterances that were said to a machine as opposed to a human. More interestingly, it also scored highest on having system responses that were deemed to be coherent and helpful in continuing the conversation. To write such responses was in fact part of the instructions to the crowdworkers, so this shows that crowdworkers are indeed capable of following these instructions and writing such responses. This provides some support for the viability of the crowdsourcing approach. Although the dialog graph corpus was given a high score on the "interesting/fun" dimension, it was not significantly higher than several of the other corpora. However, we would argue that an "interesting" response in itself is not sufficient. The combination of the metrics thus gives support for our data collection approach.

The metric "High quality utterances", seen in Table 2, is defined as the proportion of utterance pairs (out of 200) where the user utterance was considered at least likely to be a user utterance and the system response at least coherent in the corpora evaluation (a score of 3 or 4 out of 4). The results show that the dialog graph is substantially higher than all the other corpora, with Self-dialog and Switchboard around the 50% mark. In contrast Reddit and BNC, both relatively large corpora, has a low percentage only reaching around 20% casting serious concerns on their usefulness in the social chatbot context, if not rigorous filtering is applied first.

6.2 Scaling

The branching factor for some nodes can become high if there are many ways to reply to an utterance, especially if it is an open question. Therefore the number of utterances that workers need to write responses to scales exponentially. Looking at Table 1, we see that on the first level of our graph the ratio of user to system nodes is almost 1-1. However, already on the second level we have over 30,000 user nodes, which are replies to the 2000 system nodes preceding them. At the second level, only a tenth of the user nodes have a system reply. One way to decrease the branching factors is limiting the number of open questions. Closed questions (e.g. yes/no-questions) and questions which ask for specific entities that can be handled with tags would allow for deeper paths in the tree. Which question should be added to the graph can then be reflected in the instructions to the crowdworkers. We have taken several measures to address these issues (described in Section 3.4), but there is still room for improvement of automatic merging of nodes.

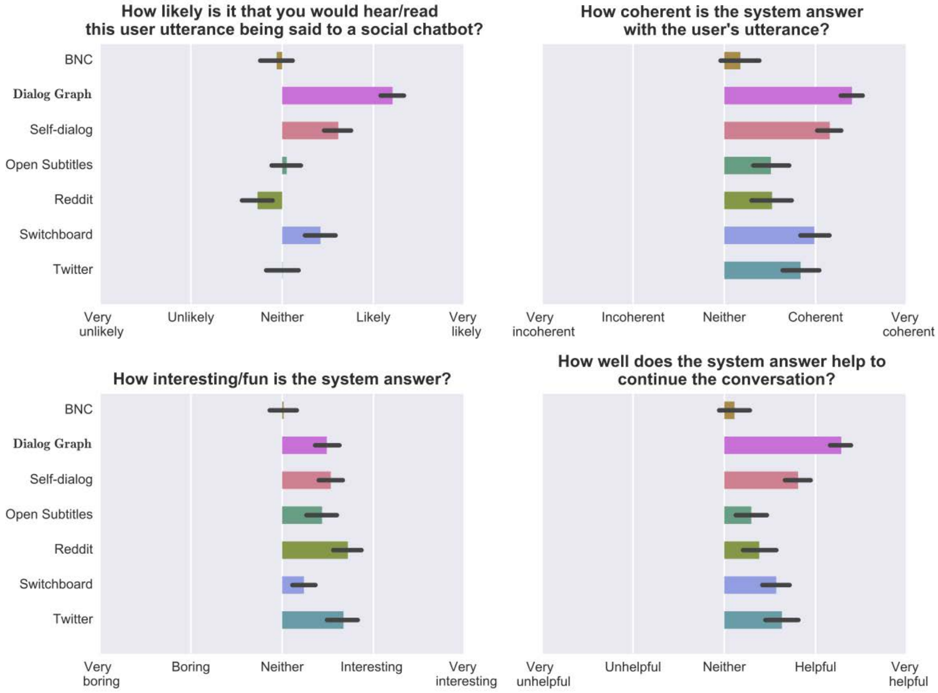


Figure 4: The mean and the 95% confidence intervals of the evaluated metrics across datasets. The method presented in this paper is denoted as the dialog graph.

6.3 Time dependent conversations

Depending on how one decides to populate the dialog graph, certain obstacles may appear. As described, the graph is constantly being updated based on what topics users want to talk about. They may choose to talk about anything, and it is quite common that conversations regarding time specific events arise. Consequently, a number of outdated conversations not suitable for training a social chatbot, need to be dealt with. An example from AP2018 are conversations that were gathered during the 2018 FIFA World Cup. During this period the dialog graph was updated with utterances like: '*Are you watching the world cup?*'. As users tend to talk about trends and news quite often, this can become troublesome if not handled correctly. This could potentially be handled using an automatic or crowdsourced approach. For example, certain conversation paths in the graph that are only relevant during a certain time periods could be tagged, and when that period is over the paths are removed.

6.4 Persona

A proper evaluation of the effect of the persona would be an interesting direction for future work, both in terms of how to construct it, but also how to properly convey it to the crowdworker. The approach described in this paper presented 2 random attributes from a pool of 30, which seemed to have a positive effect on the produced utterances. However, further evaluation of this method is necessary.

7 CONCLUSION

This paper presents a crowdsourcing-based method for obtaining and storing dialog data by using the format of a graph. This dialog graph showed great potential during the course of AP2018, both in terms of storing data and as the basis for a simple dialog manager. In a comparison with other types of dialog corpora we show that the dialog graph corpus collected with the presented data collection method outperforms the other corpora in several important aspects in the context of social chatbots. We believe that the presented method is a good alternative for collecting datasets that reflect engaging and coherent conversations suitable for developing social chatbots. The code is available at: https://github.com/kth-social-robotics/fantombot.

ACKNOWLEDGMENTS

We are grateful to Mattias Bystedt, Jonas Ivarsson, Marketa Slukova, Hannes Kindbom and Johan Boye for being a part of the Fantom team and your contributions during the Alexa Prize competition. A special thanks is extended to Zofia Malisz for your valuable input on statistics and David House for being the instructor in our instruction videos. The authors would also like to acknowledge the support from the Swedish Research Council (2017-05189) and the Swedish Foundation for Strategic Research (RIT15-0107). Finally, we would like to thank the Alexa Prize staff for feedback and support during the Alexa Prize challenge.

REFERENCES

[1] Douglas Bates, Martin Mächler, Ben Bolker, and Steve Walker. 2015. Fitting Linear Mixed-Effects Models Using lme4. *Journal of Statistical Software* 67, 1 (2015), 1–48. https://doi.org/10.18637/jss.v067.i01

[2] Cynthia Breazeal, Nick DePalma, Jeff Orkin, Sonia Chernova, and Malte Jung. 2013. Crowdsourcing human-robot interaction: New methods and system evaluation in a public environment. *Journal of Human-Robot Interaction* 2, 1 (2013), 82–111.

[3] Cleverbot. 2018. https://www.cleverbot.com. Last accessed 2018-08-14.

[4] Microsoft Corporation. 2018. Luis. https://www.luis.ai. Last accessed 2018-08-14.

[5] Elena Filatova. 2012. Irony and Sarcasm: Corpus Generation and Analysis Using Crowdsourcing. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*.

[6] Wikimedia Foundation. 2019. Wikidata. https://www.wikidata.org. Last accessed 2019-04-12.

[7] J. J. Godfrey, E. C. Holliman, and J. McDaniel. 1992. SWITCHBOARD: telephone speech corpus for research and development. In *[Proceedings] ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 1. 517–520 vol.1. https://doi.org/10.1109/ICASSP.1992.225858

[8] Ting-Hao 'Kenneth' Huang, Joseph Chee Chang, and Jeffrey P. Bigham. 2018. Evorus: A Crowd-powered Conversational Assistant Built to Automate Itself Over Time. (2018). https://doi.org/10.1145/3173574.3173869 arXiv:1801.02668

[9] Amazon.com Inc. 2018. The Amazon Alexa Prize. https://developer.amazon.com/alexaprize. Last accessed 2018-10-24.

[10] Amazon.com Inc. 2018. Lex. https://aws.amazon.com/lex. Last accessed 2018-08-14.

[11] Wit.AI Inc. 2018. Wit. https://wit.ai. Last accessed 2018-08-14.

[12] Sina Jafarpour, Christopher JC Burges, and Alan Ritter. 2010. Filter, rank, and transfer the knowledge: Learning to chat. *Advances in Ranking* 10 (2010), 2329–9290.

[13] Patrik Jonell, Mattias Bystedt, Fethiye Irmak Doğan, Per Fallgren, Jonas Ivarsson, Marketa Slukova, Ulme Wennberg, José Lopes, Johan Boye, and Gabriel Skantze. 2018. Fantom: A Crowdsourced Social Chatbot using an Evolving Dialog Graph. *Alexa Prize Proceedings* (2018).

[14] John F Kelley. 1983. An empirical methodology for writing user-friendly natural language computer applications. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. ACM, 193–196.

[15] Chandra Khatri, Behnam Hedayatnia, Anu Venkatesh, Jeff Nunn, Yi Pan, Qing Liu, Han Song, Anna Gottardi, Sanjeev Kwatra, Sanju Pancholi, Ming Cheng, Qinglang Chen, Lauren Stubel, Karthik Gopalakrishnan, Kate Bland, Raefer Gabriel, Arindam Mandal, Dilek Hakkani-Tür, Gene Hwang, Nate Michel, Eric King, and Rohit Prasad. 2018. Advancing the State of the Art in Open Domain Dialog Systems through the Alexa Prize. *CoRR* abs/1812.10757 (2018). arXiv:1812.10757 http://arxiv.org/abs/1812.10757

[16] Ben Krause, Marco Damonte, Mihai Dobre, Daniel Duma, Joachim Fainberg, Federico Fancellu, Emmanuel Kahembwe, Jianpeng Cheng, and Bonnie L. Webber. 2017. Edina: Building an Open Domain Socialbot with Self-dialogues. *CoRR* abs/1709.09816 (2017). arXiv:1709.09816 http://arxiv.org/abs/1709.09816

[17] Alexandra Kuznetsova, Per B. Brockhoff, and Rune H. B. Christensen. 2017. lmerTest Package: Tests in Linear Mixed Effects Models. *Journal of Statistical Software* 82, 13 (2017), 1–26. https://doi.org/10.18637/jss.v082.i13

[18] Iolanda Leite, André Pereira, Allison Funkhouser, Boyang Li, and Jill Fain Lehman. 2016. Semi-situated Learning of Verbal and Nonverbal Content for Repeated Human-robot Interaction. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction (ICMI 2016)*. ACM, New York, NY, USA, 13–20. https://doi.org/10.1145/2993148.2993190

[19] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A Persona-Based Neural Conversation Model. *arXiv* (2016), 10. arXiv:1603.06155 http://arxiv.org/abs/1603.06155

[20] Pierre Lison and Jörg Tiedemann. 2016. OpenSubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)* (23-28), Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis (Eds.). European Language Resources Association (ELRA), Paris, France.

[21] Google LLC. 2018. Dialogflow. https://dialogflow.com. Last accessed 2018-08-14.

[22] Xiaofei Lu. 2009. Automatic measurement of syntactic complexity in child language acquisition. *International Journal of Corpus Linguistics* 14, 1 (2009), 3–28.

[23] Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2017. The E2E dataset: New challenges for end-to-end generation. *arXiv preprint arXiv:1706.09254* (2017).

[24] OpenSubtitles. 2018. https://www.opensubtitles.org. Last accessed 2018-08-14.

[25] Jeff Orkin and Deb Roy. 2009. Automatic learning and generation of social behavior from collective human gameplay. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems, 385–392.

[26] Oxford University press. 2018. British National Corpus (BYU-BNC). https://corpus.byu.edu/bnc. Last accessed 2018-08-23.

[27] R Core Team. 2018. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/

[28] Reddit. 2017. Reddit corpus. https://files.pushshift.io/reddit/comments. Last accessed 2019-04-11.

[29] Iulian Vlad Serban, Alessandro Sordoni, Yoshua Bengio, Aaron C Courville, and Joelle Pineau. 2016. Building End-To-End Dialogue Systems Using Generative Hierarchical Neural Network Models.. In *AAAI*, Vol. 16. 3776–3784.

[30] Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. A Neural Network Approach to Context-Sensitive Generation of Conversational Responses. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, Denver, Colorado, 196–205. https://doi.org/10.3115/v1/N15-1020

[31] Oriol Vinyals and Quoc V. Le. 2015. A Neural Conversational Model. *CoRR* abs/1506.05869 (2015). arXiv:1506.05869 http://arxiv.org/abs/1506.05869

[32] Jane Wakefield. 2016. Microsoft chatbot is taught to swear on Twitter. *BBC* (Mar 2016). https://www.bbc.com/news/technology-35890188

[33] Tsung-Hsien Wen, David Vandyke, Nikola Mrksic, Milica Gasic, Lina M Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2016. A network-based end-to-end trainable task-oriented dialogue system. *arXiv preprint arXiv:1604.04562* (2016).

[34] Bruce Wilcox. 2015. ChatScript. https://github.com/bwilcox-1234/ChatScript. [Online; accessed 2018-08-14].

[35] Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing Dialogue Agents: I have a dog, do you have pets too? *arXiv preprint arXiv:1801.07243* (2018).