Investigating Proactivity in Task-Oriented Dialogues

Vevake Balaraman[1,2], **Bernardo Magnini**[1]
[1] Fondazione Bruno Kessler, Via Sommarive 18, Povo, Trento — Italy
[2] ICT Doctoral School, University of Trento — Italy.
{balaraman, magnini}@fbk.eu

Abstract

Proactivity (i.e., the capacity to provide useful information even when not explicitly required) is a fundamental characteristic of human dialogues. Although current task-oriented dialogue systems are good at providing information explicitly requested by the user, they are poor in exhibiting proactivity, which is typical in human-human interactions. In this study, we investigate the presence of proactive behaviours in several available dialogue collections, both human-human and human-machine and show how the data acquisition decision affects the proactive behaviour present in the dataset. We adopt a two-step approach to semi-automatically detect proactive situations in the datasets, where proactivity is not annotated, and show that the dialogues collected with approaches that provide more freedom to the agent/user, exhibit high proactivity.

1 Introduction

Proactivity is the collaborative attitude of humans to offer information in a dialogue even when such information was not explicitly requested. As an example, a travel operator may suggest points of interest and attractions in a certain area, even if the customer did not explicitly requested for them. The following portion of dialogue, extracted from the Nespole dataset (Mana et al., 2003), shows proactive contributions of the travel agent (displayed in italics).

Client: good morning; could you suggest any village in the Val di Fiemme to me; where it's possible to skate for example; that is does any skating rink exist in the Val di Fiemme;
Agent: yes; in the whole of Val di Fiemme there are some outdoor skating rinks; *where you can skate usually in the afternoon; in some rinks even in the morning*; and then right in Cavalese there's a skating rink an ice rink; *where even some courses are organized; where they also hold hockey or skating shows; and it's indoors.*

In this dialogue situation the travel agent provides indications both about the opening time of skating rinks and about skating courses, which were not requested by the customer. We may think proactivity as a guess of the agent with respect to the customer needs, with the purpose of anticipating expected requests, this way facilitating the achievements of the dialogue goals.

Proactivity is a crucial characteristics of human-human dialogues. It is related to the so called *principles of cooperative dialogue*, which have been summarized in the popular Grice's maxims (Grice, 1975). In particular, proactivity follows the maxim of quantity, where one tries to be as informative as one possibly can, and gives as much information as it is needed, and no more. Under this maxim, proactivity has to find a trade-off between providing useful not requested information and limiting excessive not needed information. For instance, in the context of our dialogue about skating in Val di Fiemme, an agent suggesting a good pizzeria would probably be perceived as a violation of the quantity maxim, as this information seems not enough needed in that context.

Despite the large use of proactivity that we note in everyday human-human dialogues, proactive behaviours are poorly represented in most of the models at the core of the last generation of task-oriented dialogue systems. Overall, we notice a general lack of cooperative phenomena (e.g., clar-

ification questions, explanatory dialogues, proactivity, etc.), that characterize, and somehow make efficient, task-oriented human-human dialogues. A notable exception are recommendation systems (Thompson et al., 2004; Sun and Zhang, 2018; Yoshino and Kawahara, 2015), where, however, the focus is on influencing the user towards a specific goal (e.g., buy a certain product). Instead, we intend proactivity to be a general collaborative strategy aiming at improving the quality and effectiveness of the conversation. As an example, proactivity can be used to anticipate future requests of the user (e.g., providing the telephone number of a certain restaurant), or to recover from failure situations (e.g., offering possible alternatives when there are no restaurants satisfying the user desires).

The main purpose of the paper is to conduct an empirical analysis over several existing task-oriented dialogue datasets, used to train dialogue models, in order to verify the presence of proactive behaviours. More specifically, we consider a human-human dialogue corpus collected with a role-taking methodology, i.e., Nespole, and compare it with other task-oriented dialogues collected either with Wizard of Oz or with bootstrapping methods. To conduct such a comparison, the major obstacle is that in both cases, proactivity is not annotated in any way, and we had to figure out methods (addressed in Section 2) to semi-automatically detect proactive situations. Results confirm that dialogues collected through role-taking methodology show a much richer presence of proactivity, which is a challenge for future dialogue systems.

2 Methodology

In this Section, first we define proactivity behaviours in the context of task-oriented dialogues, and then we describe the methodology we use to detect proactivity in available dialogue corpora.

2.1 Defining Proactivity

Our starting point is the work on proactivity presented in (Balaraman and Magnini, 2020), where a pro-active behaviour is defined as any information that: (i) is introduced by the system; (ii) was not previously introduced in the dialogue by the user; and (iii) is assumed to be relevant to achieve the user needs. According to this definition, system turns like the following are all proactive:

- System: *We have good reviews for restaurant X.*
- System: *There a no Eritrean restaurants in the city center, but there are several of them in the south of the city.*
- System: *In case it might be useful, the telephone of the restaurant is X*, after a certain restaurant has been chosen by the user.
- System: *There is a metro station close to the restaurant you have chosen*.

As the examples show, proactive information is strictly related to domain knowledge (e.g., knowledge about restaurants in a city). Moreover, the system may decide to be proactive only in certain dialogue situations, where there is need to help the user to positively conclude a dialogue. In our second example, for instance, the user needs do not match any instance in a domain Knowledge Base (i.e., there are no Eritrean restaurants in the city center), and the system informs the user that there are Eritrean restaurants in the south of the city, this way avoiding a longer follow up interaction.

2.2 Detecting Proactivity

Unfortunately, proactivity was not a designing feature of any of the datasets considered in this study. This means that proactivity is not marked, and that we need to figure out how it can be detected at a reasonable cost. The approach taken in this paper detects proactivity occurring in *intermediate failure situations*, when the system tries to recover from a dialogue failure. There are two reasons for this choice: (i) failure situations are easy to be detected through simple patterns (e.g., *I am sorry...*, *We do not have...*); (ii) as the capacity to recover from failure situations is crucial to maximize the final success of the dialogue, we assume that the attitude of a system to be proactive is particularly revealed in failure situations. In other words, we look at failure situations as typical situations where proactivity should be applied by a system. Given a dialogue collection, we can consider the proportion of proactivity within intermediate failures as a sort of upper bound of proactivity in the whole collection.

Under this assumption, we adopted a failure-based, two-step methodology for detecting proactivity. At the first step we detect as much as possible turns where the system inform the user that

his/her request cannot be satisfied. This step is implemented through either pattern-based search of typical linguistic expressions indicating failure (e.g., *I am sorry...*, *We do not have...*, *There are no...*, etc.) or patterns in dialogue acts of the system. At the second step, we focus on system failure responses, and check whether the response contains any proactive information (see Section 2.1: if any proactive information is present, we mark the system turn as proactive, otherwise as non-proactive. This second step is either performed manually or by finding patterns in the dialogue acts of the system response.

3 Experimental Data

In this section we describe the different data acquisition approaches used for the collection of task-oriented dialogue datasets, and provide details about them.

3.1 Data Acquisition Approaches

We consider three data acquisition approaches that are widely used for dialogue collection.

Wizard of Oz (WoZ) is the most popular approach to collect task-oriented dialogues, possibly using crowd workers (Fraser and Gilbert, 1991; Kelley, 1984). This involves a pair of crowd workers who are provided with respective dialogue goals and are asked to communicate in natural language to achieve the goal. Each crowd worker, acting either as the wizard or the user, is provided with the instructions to achieve the dialogue goal.

The following is an example of a dialogue script provided to the crowd worker in the MultiWoZ (Budzianowski et al., 2018) dataset.

1. You are looking for a *place to stay*. The hotel should be in the *cheap* price range and should be in the type of *hotel*
2. The hotel should *include free parking* and should *include free wifi*
3. Once you find the *hotel* you want to book it for *6 people* and *3 nights* starting from *tuesday*
4. If the booking fails how about *2 nights*
5. Make sure you get the *reference number*

The dialogue script is typically filled in using placeholders in a template (shown in *italics* in our example). We notice the amount of details present in the dialogue description, which could influence the crowd worker utterance for a given turn, and induce to follow a structure similar to the dialogue script.

Bootstrapping, also referred to as Machines talking to Machines (M2M), is a simulation-based approach for generating *outlines* for a number of dialogues via self-play (Shah et al., 2018), a methodology that takes advantage of a task-specific information input provided by the developer. The task-specification defines the schema of intents, the slot names and the slot values for a certain domain. Based on the task-specification, the framework first generates a set of dialogue outlines containing natural language utterances and their corresponding annotations. The obtained dialogues are then paraphrased using crowd workers in order to obtain linguistic variations. This approach reduces the resources required to collect a large dialogue dataset and enables the developer to control for the diversity both in the dialogue flow and in the user behaviors. Table 1 shows an example of a dialogue outline generated through a bootstrapping approach, which is then paraphrased using crowd workers.

Role-Taking. This methodology involves people playing two roles, typically with minimum training, interacting in order to achieve a given goal (e.g., a travel agent and a customer with the goal of organizing a trip; an applicant and a job operator with the goal of finding a job opportunity). For both the participants responses are unscripted and are supposed to be natural as expected in a real-world conversation. This is similar to the MAP task approach (Anderson et al., 1991; Meena et al., 2013), which allows to collect unscripted dialogues with specific communication goals. Both the participants can be trained workers, acting respectively as the user and the expert, and are provided with a dialogue goal and information they can use (e.g. an applicant with a CV and a job operator with job offers). Table 2 shows a sample interaction for this approach.

3.2 Datasets

We have analysed proactivity in five available collections of task-oriented dialogue datasets in English, all of them used to train dialogue models. In addition, we have compared them with Nespole (Mana et al., 2003), a human-human dia-

Dialogue Outline		Paraphrase
Annotation	**Template utterances**	**NL utterances**
S: greeting()	Greeting.	Hi, how can I help you?
U: inform(intent=book_movie, name=Inside Out, date=tomorrow, num_tickets=2)	Book movie with name is Inside Out and date is tomorrow and num tickets is 2.	I want to buy 2 tickets for Inside Out for tomorrow.
S: ack() request(time)	OK. Provide time.	Alright. What time would you like to see the movie?
U: inform(time=evening)	Time is evening.	Anytime during the evening works for me.
S: offer(theatre=Cinemark 16, time=6pm)	Offer theatre is Cinemark 16 and time is 6pm.	How about the 6pm show at Cinemark 16?
U: affirm()	Agree.	That sounds good.
S: notify_success()	Reservation confirmed.	Your tickets have been booked!

Table 1: A sample dialogue collected through the *bootstrapping* approach (Shah et al., 2018).

logue dataset which was collected to study real-world human-human interactions. Table 3 reports the main characteristics of the six datasets, including the method of data acquisition.

WOZ2.0 includes textual conversations for restaurant booking in Cambridge and was collected using Wizard of Oz by pairing users in Amazon Mechanical Turk. The user and the wizard contribute a single turn to each dialogue (Wen et al., 2017). (Mrkšić et al., 2017) expanded the original WoZ dataset producing the WoZ2.0 dataset, consisting of 1200 dialogues.

MultiWOZ2.1 includes dialogues in multiple domains collected via Wizard of Oz. The developers explicitly encouraged goal changes, in order to model realistic conversations (Budzianowski et al., 2018). Different versions of the dataset have been published recently, addressing annotation errors occurring in the original dataset (Ramadan et al., 2018; Budzianowski et al., 2018; Eric et al., 2020; Zang et al., 2020). We use the MultiWoZ2.1 dataset, containing 10438 dialogues.

Schema-Guided Dataset (SGD) consists of 22825 dialogues in multiple domains collected using the Machine Talking to Machine (Bootstrapping) approach (Rastogi et al., 2019). Dialogues generated via simulation are then paraphrased by the crowd workers for language variability. SGD promotes research towards dialogue systems that can handle dynamic schemas.

Microsoft Dialogue dataset (Li et al., 2016; Li et al., 2018) consists of dialogues collected via Amazon Mechanical Turk using a bootstrapping approach for three different domains *Movie-Ticket Booking*, *Restaurant Reservation* and *Taxi Ordering* with 2890, 4103 and 3094 dialogues, respectively.

Maluuba Frames dataset (El Asri et al., 2017) consists of 1369 dialogues collected via Wizard of Oz using a Slack bot for travel vacation domain. Users were assigned a tasks using a template where placeholder values are filled by drawing values from a database. If the task is successful, the user either ended the dialogue or received an alternate task. In case of no match, suggestions were sometimes provided to the wizards, who then decided whether to use or not the suggestion for the user.

Nespole (Mana et al., 2003; **?**) is a VoIP (Voice over Internet Protocol) corpus consisting of spoken interactions between a professional agent and a recruited worker acting as a user or client. We use the DB-1 part of the Nespole dataset, consisting of 39 dialogues (in the transcribed version of the dataset 3 client side dialogues were missing, leaving 36 dialogues for a total of 1549 turns). Dialogues are about vacation planning in the Trentino region and, unlike other datasets, they do not have a fixed user-side goal, but rather a collaborative goal. Specifically, the user and the agent collaborate via a spoken conversation to achieve a goal that satisfies the user.

Speaker	Utterance
System	Could you help me to find my way to the bus stop?
User	start from the department store
System	yeah
User	and eh
System	Should I start by going west?
User	yeah do that
User	then you will get to a meadow and when you get to the meadow
System	Eh, could you repeat that?
User	you go straight and you see a meadow on your right side
System	A green field?
User	ehm yeah a field
System	mhm
User	pass the meadow and turn right so you are going north
System	okay
...	...
User	at the junction go south and then you will get to the bus stop
System	okay, thanks a lot.

Table 2: A sample dialogue collected through the Role-Taking approach (Meena et al., 2013).

4 Results and Discussion

We have applied the methodology described in Section 2 to detect proactivity in the six datasets. First we detect the number of failure turns in each dataset and then, among failures, we identify the turns that exhibit proactivity.

Table 4 reports the number of failure turns we were able to detect for each dataset, and the proportion of them that exhibit a proactive behaviour, according to our definition in Section 2.1. We can notice that the datasets collected via Wizard of Oz (WoZ) typically exhibit very low proactivity. This could be due to the fact that in the WoZ approach users are provided with a task description detailing how to proceed with the dialogue. This indirectly influences the users to use certain formats as defined in the description. The MultiWoZ2.1 dataset shows the highest proactivity among the datasets collected via WoZ approach: this is due the explicit encouragement of goal changes in task-descriptions. As for the SGD and Microsoft dialogue datasets, collected via a *bootstrapping* approach, we can notice that over 50% of the failure turns exhibiting proactivity. This is because of the choice of the developers to specifically include such failure and recovery scenarios in the dialogue flow.

Datesets collected via WoZ and bootstrapping have different approaches in adopting proactivity. Since WoZ is collected by pairing humans, proactive turns often contain information that would lead to a dialogue success. However, in the bootstrapping approach, as it is based on a script, the proactive turns contain information that are possible for the user to request but may not lead to dialogue success. An example in MultiWoz2.1 is the following: *"There are no hotels that fit your criteria in the South, but there are two Guesthouses. Would you like to book one of those?"*. Here the crowd-worker acting as a wizard has already looked the availability of two Guesthouses and is providing this information to another crowd-worker who is acting as the user. If the user chooses the guesthouse, the dialogue would be a success. A similar example in Microsoft Dialogue dataset is the following: *"I'm sorry The Other Side of The Door is not playing in your area on Tuesday. I am able to find show times for The Witch and Triple 9"*. Here, the system-agent is providing information that the user-agent can choose as alternatives, but the alternatives may not always directly lead to dialogue success. When the user-agent responds *"The Witch will be fine"*. the system-agent searches the knowledge-base and responds *"I'm sorry they are only showing The Witch at 4:40 pm. Would that be acceptable for you?"* which again is a proactive response.

The analysis for proactivity in Nespole differs from the other datasets, as Nespole is not modeled to find an exact match for the user needs, and, as a consequence, there are no clear failure situations. In addition, while the other datasets were collected focused towards using them for training dialogue systems, Nespole was collected to analyze linguistic features in real-world dialogues. However, we manually analysed the 36 dialogues (1549 turns) of vacation planning and identified the turns where the agent exhibits proactivity. We found that 49 turns in 26 dialogues are proactive responses, where the agent provides information not explicitly requested by the user (see the example in the Introduction). Since Nespole is a VoIP dataset, the number of turns are not comparable to the other datasets as they contain frequent in-

Dataset	Data Acquisition	#Dialogues	#Turns	Avg. Turn length
WoZ2.0	Wizard of Oz	1,200	8,824	11.27
MultiWoZ2.1	Wizard of Oz	10,438	143,048	13.18
Maluuba Frames	Wizard of Oz	1,369	19,986	12.60
Schema-Guided Dataset	Bootstrapping	22,825	463,284	9.86
Microsoft Dialogue				
- Movie-Ticket Booking	Bootstrapping	2,890	21,656	10.96
- Restaurant Reservation		4,103	29,719	11.45
- Taxi Ordering		3,094	23,311	11.04
Nespole	Role-Taking	36	1,549	18.48

Table 3: Statistics about the datasets used for the proactivity analysis.

Dataset	#Failure	#Proactive	%
WoZ2.0	414	26	5.9
MultiWoZ2.1	2,127	325	15.3
Maluuba Frames	1,214	77	6.3
Schema-Guided	3,362	1,737	51.7
Microsoft			
- Movie	318	161	50.6
- Restaurant	775	323	41.7
- Taxi	104	38	36.5
Nespole	–	49	–

Table 4: Number of failure situations (turns) and corresponding proactivity, for each dataset.

terruptions and fillers. An example of proactive turn in Nespole is the following: *"no; there's no entertainment for the kids; entertainment for the kids would be at the Olimpionic Hotel; but it's a 3 star one already"*. We can see that the agent provides information for a scenario that was requested by the user with a piece of proactive information (*entertainment for the kids would be at the Olimpionic Hotel; but it's a 3 star one*). We notice that proactive turns in Nespole exhibit much richer information compared to the other datasets, which could be attributed to the freedom of expression provided to the agent, unlike to the other approaches considered.

We now discuss a few research questions that arise from our study on proactivity in dialogue collections.

Does our failure-based methodology provide reasonable coverage about proactivity in our datasets? We assume that task-oriented dialogue systems should maximize their success rate (i.e., matching the user needs), and that recovering from intermediate failure situations potentially increases their success rate. Under this assumption failure situations act as an upper bound for the situations in which the systems is expected to be proactive. As an example, having found that 5.9% of intermediate failures in WoZ2.0 are proactive, we infer that the amount of proactivity in the whole WoZ2.0 will not be higher that 5.9%.

Does proactivity correlate with the method of collection of the dataset? As seen in Table 4, the Wizard of Oz approach consistently has very low proactivity, while the *bootstrapping* approach exhibits high proactivity. While the amount of proactivity in each dataset depends on the developer choice about the dialogue goals and on the instructions provided to users, we can conclude that the WoZ approach indirectly influences the user to deviate from a collaborative approach and to follow a scripted dialogue.

5 Conclusion

Task-oriented dialogue systems have shown to be effective in providing services to users with a high success rate. However, the interaction still lacks an effective proactive approach, which is typical in human-human conversations. In this study, we compare proactive behaviours in several available dialogue datasets, and show that the dialogues collected through Wizard of Oz contain a small proportion of system proactive responses, while dialogues collected through simulation-based and role-taking methodologies contain higher degree of proactivity. To sum up, we suggest that data collection strategies should be better aware that their designing principles have strong influence on the quality of the dialogues. Particularly, we recommend higher attention to proactive behaviours, and, in general, to collaborative phenomena.

References

Anne H. Anderson, Miles Bader, Ellen Gurman Bard, Elizabeth Boyle, Gwyneth Doherty, Simon Garrod, Stephen Isard, Jacqueline Kowtko, Jan McAllister, Jim Miller, Catherine Sotillo, Henry S. Thompson, and Regina Weinert. 1991. The hcrc map task corpus. *Language and Speech*, 34(4):351–366.

Vevake Balaraman and Bernardo Magnini. 2020. Proactive systems and influenceable users: Simulating pro-activity in task-oriented dialogues. In *Proceedings of the 24th Workshop on the Semantics and Pragmatics of Dialogue - Full Papers*, Virually at Brandeis, Waltham, New Jersey, July. SEMDIAL.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - a large-scale multi-domain wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026, Brussels, Belgium, October-November. Association for Computational Linguistics.

Layla El Asri, Hannes Schulz, Shikhar Sharma, Jeremie Zumer, Justin Harris, Emery Fine, Rahul Mehrotra, and Kaheer Suleman. 2017. Frames: a corpus for adding memory to goal-oriented dialogue systems. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 207–219, Saarbrücken, Germany, August. Association for Computational Linguistics.

Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyang Gao, Adarsh Kumar, Anuj Goyal, Peter Ku, and Dilek Hakkani-Tur. 2020. MultiWOZ 2.1: A consolidated multi-domain dialogue dataset with state corrections and state tracking baselines. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 422–428, Marseille, France, May. European Language Resources Association.

Norman M. Fraser and G.Nigel Gilbert. 1991. Simulating speech systems. *Computer Speech Language*, 5(1):81 – 99.

Herbert P Grice. 1975. Logic and conversation. In *Speech acts*, pages 41–58. Brill.

J. F. Kelley. 1984. An iterative design methodology for user-friendly natural language office information applications. *ACM Trans. Inf. Syst.*, 2(1):26–41, January.

Xiujun Li, Zachary C Lipton, Bhuwan Dhingra, Lihong Li, Jianfeng Gao, and Yun-Nung Chen. 2016. A user simulator for task-completion dialogues. *arXiv preprint arXiv:1612.05688*.

Xiujun Li, Yu Wang, Siqi Sun, Sarah Panda, Jingjing Liu, and Jianfeng Gao. 2018. Microsoft dialogue challenge: Building end-to-end task-completion dialogue systems. *arXiv preprint arXiv:1807.11125*.

N. Mana, S. Burger, Roldano Cattoni, L. Besacier, V. MacLaren, J. McDonough, and F. Metze. 2003. The nespole! voip multilingual corpora in tourism and medical domains. In *INTERSPEECH*.

Raveesh Meena, Gabriel Skantze, and Joakim Gustafson. 2013. The map task dialogue system: A test-bed for modelling human-like dialogue. In *Proceedings of the SIGDIAL 2013 Conference*, pages 366–368, Metz, France, August. Association for Computational Linguistics.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Tsung-Hsien Wen, Blaise Thomson, and Steve Young. 2017. Neural belief tracker: Data-driven dialogue state tracking. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1777–1788, Vancouver, Canada, July. Association for Computational Linguistics.

Osman Ramadan, Paweł Budzianowski, and Milica Gasic. 2018. Large-scale multi-domain belief tracking with knowledge sharing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, volume 2, pages 432–437.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2019. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. *arXiv preprint arXiv:1909.05855*.

Pararth Shah, Dilek Hakkani-Tür, Bing Liu, and Gokhan Tür. 2018. Bootstrapping a neural conversational agent with dialogue self-play, crowdsourcing and on-line reinforcement learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*, pages 41–51, New Orleans - Louisiana, June. Association for Computational Linguistics.

Yueming Sun and Yi Zhang. 2018. Conversational recommender system. In *The 41st International ACM SIGIR Conference on Research Development in Information Retrieval*, SIGIR '18, page 235–244, New York, NY, USA. Association for Computing Machinery.

Cynthia A. Thompson, Mehmet H. Göker, and Pat Langley. 2004. A personalized system for conversational recommendations. *J. Artif. Int. Res.*, 21(1):393–428, March.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gasic, Lina M. Rojas Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 438–449. Association for Computational Linguistics.

Koichiro Yoshino and Tatsuya Kawahara, 2015. *News Navigation System Based on Proactive Dialogue Strategy*, pages 15–25. Springer International Publishing, Cham.

Xiaoxue Zang, Abhinav Rastogi, Srinivas Sunkara, Raghav Gupta, Jianguo Zhang, and Jindong Chen. 2020. Multiwoz 2.2: A dialogue dataset with additional annotation corrections and state tracking baselines. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI, ACL 2020*, pages 109–117.