2021 International Conference on INnovations in Intelligent SysTems and Applications (INISTA) | 978-1-6654-3603-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/INISTA52262.2021.9548636

Evaluation of Wizard-of-Oz and Self-Play Data Collection Techniques for Turkish Goal-Oriented Dialogue Agents

1st Doğukan Arslan
Department of AI & Data Engineering
Istanbul Technical University
Istanbul, Turkey
arslan.dogukan@itu.edu.tr

2nd Gülşen Eryiğit
Department of AI & Data Engineering
Istanbul Technical University
Istanbul, Turkey
gulsen.cebiroglu@itu.edu.tr

***Abstract*—As with all natural language processing tasks, the lack of open-source training data required for the development of dialogue agents is a major obstacle to research studies in the field. Especially languages that are not widely studied, such as Turkish, suffer more from this problem. This article introduces a comparison of Wizard-of-Oz and self-play data collection techniques for Turkish goal-oriented dialogue system generation. Three data sets have been prepared and introduced to the researchers by using these techniques. Being the first publicly available human-to-human Turkish dialogue data sets, although open for development, the created resources from the restaurant domain are very valuable for further research on Turkish dialogue systems. The mentioned methods are quantitatively compared on the produced data sets, in terms of dialog act classification and slot identification scores. Since it is costly to collect data with methods like Wizard-Of-Oz in every domain, an open-source flexible and easy-to-use framework is also provided implementing self-play which may be used to create machine-to-machine dialogue outlines and speed data collection for low-resource languages like Turkish. Besides, designed templates of annotation screens for crowdsourcing are provided for future studies.**

***Index Terms*—goal-oriented dialogue agent, wizard-of-oz, self-play**

I. Introduction

Dialogue systems may be categorized under 2 broad types: chatbots and goal-oriented dialogue agents. Chatbots are mostly used for entertainment purposes or making dialogue agents more natural [1] while goal-oriented dialogue agents are widely used in modern dialogue systems that interact with the users to accomplish a specific task such as booking tickets, reserving a restaurant table, querying account information, or making money transfers. Handcrafted data is needed to train them, and the quality of the data directly affects the success of the system.

In recent years, we see that two methods stand out for the data preparation stage; these are Wizard-of-Oz (WOZ) [2] and self-play methods [3]. WOZ is inspired by the idea of learning from human-to-human communications from natural dialogues (e.g., call center dialogues). Yet unfortunately, it is not easy to use the natural dialogues for training dialogue systems since humans may give conflicting answers with each other in similar cases. That's why the WOZ technique, where one person takes the place of an automated dialogue agent that helps the user to accomplish a goal, is preferred. This person is expected to mimic the decision-making process of a computer and give his/her answers according to predefined logical steps [4]. On the other hand, the WOZ technique is costly and time-consuming which makes it difficult to generate training data for different domains and languages. Additionally, the data produced might be biased to some sub-tasks. At this point, self-play shows up as an alternative. It is a method that helps us to generate balanced data for each sub-task. In self-play, based on simulated dialogues between the user and the system, first, dialogue outlines are generated. Later, those outlines are paraphrased by crowd workers which results in dialogues with better diversity while providing naturalness [3].

In the natural language processing field, the vast majority of the studies are on the English language, and it is needed to have language-specific data to create dialogue agents in any other particular language. Although translating dialogues to another language is theoretically an option, in practice existing examples do not show very successful results since translation errors propagate to the operation of the system. Hence, it is important to be able to quickly generate domain-specific and language-specific data sets. Current examples, such as mobile assistants, show that these systems are less successful in languages such as Turkish than in English versions. This once again reveals the importance of data preparation.

Although many dialogue agent projects on Turkish have been initiated in the industrial environment recently, the data preparation phase is still the most challenging part since it is expensive and time-consuming. Also, the mechanisms for data collection and tagging are not clear. In fact, since this is an emerging field, it is still an active research area for English studies as well. Different data preparation attempts are being made and existing methods are being updated [3]–[6]. Unfortunately, there is no open Turkish dialogue data set

that can be studied academically. In this article, we investigate different data preparation approaches for Turkish goal-oriented dialogue agent system development to close this gap. The main contributions of the article may be listed as follows:

1) two main Turkish dialogue data sets (WOZTR and SelfPlayTR) developed using Wizard-Of-Oz and self-play approaches and one additional data set which is an augmented set (Augmented SelfPlayTR),
2) an open-source flexible and easy-to-use framework implementing self-play (viz., a system bot and a user bot implementation which interacts with each other in a simulation environment) which may be used to create machine-to-machine dialogue outlines and speed data collection for low-resource languages like Turkish,
3) the design templates of crowdsourcing annotation screens (on Pybossa) used during the human annotations of the data sets,
4) quantitative evaluation of Wizard-Of-Oz and self-play approaches for dialogue data preparation.

WOZTR has been prepared by translating a portion of English MultiWOZ data set [4] and SelfPlayTR by rephrasing the outputs of the user and system bots' simulation. PyBossa[1] is a free, open-source crowdsourcing and micro-tasking platform which allows creating crowdsourcing tasks that require human cognition such as image classification, transcription, geocoding, and more. Unfortunately, annotation templates for text data are very scarce which makes the related contribution of this article a valuable resource for a wider audience. Qualitative evaluation of Wizard-Of-Oz and self-play approaches has previously been made in [3]. For the first time in the literature, we quantitatively compare these approaches by comparing the models, trained on their produced data sets, in terms of dialog act classification and slot[2] identification scores.

The article is structured as follows; related works are provided in Section II, data sets, their development strategies, crowdsourcing interfaces and data preparation tasks are introduced in in Section III, evaluations and results are given in Section IV and lastly, conclusion in Section V.

II. Related Work

Developing goal-oriented dialogue systems is an active research area with challenges, and many solutions have been proposed recently. For instance, [7] proposed Iterative Rectification Network to solve the slot consistency problem in dialogue systems. Meta-Dialog System is proposed by [8] to make dialogue systems more reliable when data resources are low. [9] proposed a framework (PARG) for paraphrase augmented response generation to enhance the performance of goal-oriented dialogue generation. [10] proposed Multi-Agent Dialog Policy Learning which aims to make two dialogue agents (the system and the user) to interact with each other, and to learn simultaneously which eliminates the need for an explicit user simulator.

Proposed methods to solve the problem of gathering and preparing appropriate training data for training dialogue systems can be categorized as human-to-machine, human-to-human, and machine-to-machine data collection [4]. For English, many training data sets are generated for goal-oriented dialogue agents with different methods. In "Let's Go Public!" [11], a human-to-machine data collection method is used to create a corpus from a bus schedule information system. Also, in Dialog State Tracking Challenge (DSTC2), the same kind of method is used to create a large corpus obtained from a telephone system which includes dialogues that aim to find a restaurant in Cambridge [12]. Besides, Ubuntu Dialogue Corpus is an example of a human-to-human data collection method, which is obtained from Ubuntu chat logs of technical support conversations [12]. MultiWOZ data set includes human-to-human dialogues over multiple domains for goal-oriented agents, obtained with the Wizard-Of-Oz method [4]. Also, Frames is a corpus collected with the Wizard-Of-Oz method too, aiming to add memory to goal-oriented systems [6]. For Turkish, proposed corpora and researches to collect corpus for goal-oriented dialogue agents are very limited. Multilingual LUNA Corpus contains human-to-machine dialogues in 5 languages including Turkish, obtained with the translation of Italian LUNA Corpus that consists of software/hardware help domain [13]. [14] provide a small data set for mobile personal assistant development with phone-specific intents such as making calls and sending SMS. Their introduced data set is just user input utterances/queries to a mobile phone rather than dialogues.

In [3], the data set that is generated via self-play is compared with the DSTC2 data set in terms of dialog flows and language variety. For the first time in the literature. In this article, we extend these comparisons to dialogue system performance by training and testing models with data sets similar to both.

III. Data Sets

This section presents our data sets together with their development strategies and our crowdsourcing methodology. All of the three data sets are from the restaurant reservation domain, with WOZTR and SelfPlayTR, including 300 dialogues each, and Augmented SelfPlayTR as an augmented version of SelfPlayTR[3].

A. WOZTR

To be on par with English studies, instead of preparing a Wizard-Of-Oz data set from scratch, we preferred to translate a common English dialogue data set. With this purpose, the data sets mentioned in the related work section were examined, and it is observed that the MultiWOZ data set [4] stands out among them with some of its features. First of all, it is being used in almost every recent English study as an evaluation data set. Secondly, since it is collected with the Wizard-Of-Oz method

[1]https://pybossa.com

[2]Slot refers to a critical entity value for a dialogue intent.

[3]https://github.com/TR-GODA/TrainingData

TABLE I
DIALOGUE DATA SETS GENERAL STATISTICS.

	MultiWOZ	WOZTR	SelfPlayTR	Augmented SelfPlayTR
Language	English	Turkish	Turkish	Turkish
Domain	Restaurant Subset	Restaurant	Restaurant	Restaurant
Dialogue Act Types	4	5	9	9
Slot Types	7	7	8	8
Dialogues	300	300	300	300
Utterances	939	939	1752	14904
Slots	1021	1021	1531	21532

TABLE II
DIALOGUE DATA SETS SLOT TYPE STATISTICS.

Slots	WOZTR	SelfPlayTR	Augmented SelfPlayTR
goal	-	69	276
restaurant	56	182	8690
location	209	211	1335
number of person	71	202	1464
price	217	187	1888
time	75	230	1794
date	72	201	1189
cuisine	321	249	4896

and it is an example of a human-to-human data set, it is more realistic and more suitable for translating to another language. MultiWOZ proposes ten thousand dialogues in domains such as restaurant reservation, attraction, taxi, police, train, hospital, and hotel. Among these, the restaurant reservation domain (which is mostly studied in academic literature) consists of the largest part of this data set. For the user part of the dialogues, MultiWOZ has 1 intent (restaurant reservation), and 4 dialog acts in this domain; viz. request, inform, bye and thank. The restaurant data sets come with an additional database where many restaurant records are stored to be queried by customers and agents. The database mainly consists of values for the slot types restaurant name (restaurant), food type (cuisine), address (location), price range (price).

300 dialogues were randomly chosen for the translation task from the restaurant domain of the MultiWOZ data set. Using crowdsourcing (Section III-C), these dialogues were translated from English to Turkish and re-annotated (slot tagged) correspondingly by crowd workers. Table I provides the statistics of WOZTR as well as the original MultiWOZ English restaurant domain section while Table II provides slot type statistics. Table III gives the dialog act distribution of the dialogue utterances. Since the original four dialog acts (listed above) are found to be limited when compared to the dialog acts used in the compared approach (self-play) [3], we also annotated dialogue classes with more detailed ones that are essential to manage a simple dialogue (Table III). It is seen that the MultiWOZ data set does not contain some dialog acts at all.

TABLE III
DIALOGUE DATA SETS DIALOGUE ACTS STATISTICS.

Dialog Acts	WOZTR	SelfPlayTR	Augmented SelfPlayTR
greet	-	142	142
inform	574	767	5392
choose	14	203	8711
bye	173	234	234
ask	157	117	117
ask for alternative	16	108	108
refuse	-	68	91
ask for repeat	-	91	91
change	-	13	35

B. *SelfPlayTR*

In our second data creation approach, a self-play framework is implemented to simulate dialogue outlines, which are later to be paraphrased by crowd workers. The below subsections explain the details of this framework, which is used to generate 300 additional dialogues (with 1752 utterances Table I) called SelfPlayTR, and the details of the Augmented SelfPlayTR, which is an automatically augmented data set with different slot values extracted from the same database used in both SelfPlayTR and WOZTR.

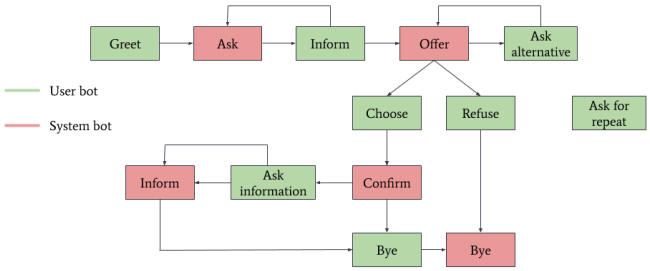

Fig. 1. Flow of the dialogue for self-play.

1) Selfplay Framework: Inspired by recent studies [15], [16] using self-play with reinforcement learning for mastering different games, [3] proposes a dialogue self-play architecture for building their dialogue system's data set. However, the study provides neither the implementation details nor the source code of the self-play framework. In this article, we introduce a similar self-play architecture and provide it as an open-source project via GitHub[4]. Our framework creates dialogue outlines for goal-oriented dialogue agents. It takes task-specific information (such as query database, dialog acts, slot types) as input to be able to create the outlines. We use the same database from MultiWOZ and change it according to our needs: i.e., restaurant names are used as is, food names (cuisines) - are translated into Turkish, addresses and location names are localized in a very simple and straightforward manner so that they only consist of different zones from

[4]https://github.com/TR-GODA/selfplay

Istanbul areas (viz. Europe or Asian sides), price ranges are also translated into Turkish.

There are four main classes for dialog acts such as constatives, directives, commissives, and acknowledgments [1]. These four main classes were considered while defining dialog acts of self-play in order to cover a simple dialogue. The framework consists of 10 dialog acts in total (for user and system bots); these are *greet*, *inform*, *choose*, *ask alternative*, *ask*, *bye*, *refuse*, *ask for a repeat*, *change*, and *offer*. As may be seen, these slightly differ from MultiWOZ dialog acts which shows that the Wizard-of-Oz technique may not cover all the necessary stages to understand the dialogue. Also, it has 8 slot types; viz. *goal*, *restaurant*, *location*, *number of person*, *price*, *time*, *date* and *cuisine* (Table II). Note that the *goal* sloat which specifies the *intent* of the user, does not exist in WOZTR.

Our self-play framework (hereinafter referred to as Selfplay for short) relies on the idea to communicate two bots (i.e., the system bot and the user bot) with each other and simulate dialogues. Selfplay's working mechanism may be shortly defined as follows; first, it creates a specific goal for the user bot based on the pre-provided task-specific information. However not all the user goals may be satisfied in real scenarios. To simulate this, as a second stage, the system temporarily deletes some goal related values from its query database with a predetermined probability (i.e., delete rate) and creates a back-up plan for the user with again a predetermined probability (i.e., backup rate). Then, the program randomly determines whether the user or system starts the conversation. Finally, it will make two bots speak till the end condition is satisfied. Flow architecture of the Selfplay may be seen from Figure 1. Since the "ask for repeat" act may appear anytime, it is not linked to any of the flow elements.

Dialogue flows are affected by the personality of the user which is determined by four parameters. The *verbose* parameter indicates how talkative the user is. It shows how many slots will be informed at once. The *flexibility* parameter represents the user's response when there aren't matching results for the goal. Depending on this, the user might choose another option rather than his/her goal. *Ask for a repeat* parameter indicates how frequently the user will ask for repetition and finally, *randomness* parameter indicates the probability of the illogical behavior of the user. For instance, high randomness probability might cause the user to leave the conversation in the middle of it. At the end of this cycle, the framework produces system-generated dialogue utterances (for both the system and the user) which are not as natural as human sentences and needs to be rephrased by humans. Produced and paraphrased utterance samples may be seen from Table IV. The prior stage is accomplished relying on natural language grammar and automatic language generation. The produced data set statistics may again be investigated from Tables I, II, and III. As one may observe from Table I, SelfPlayTR, having the same number of dialogues, contain more utterances when compared to WOZTR, which means that its dialogues are longer. This may be attributed to the modeling of user behaviors and their resemblance to the real-world scenarios.

2) Augmented SelfPlayTR: When using machine learning algorithms to train a model, one needs a sufficient amount of training data, and the lack of it can lead to weak models. To investigate the data size impact on our training models, we used a data augmentation technique[5] and investigated its impact in our evaluation scenarios. Since we had a database of possible slot values such as different restaurant names, addresses, etc., it was possible for us to slightly change the slot values of the existing dialogue utterances and increase the size of our training set.

For data augmentation, we changed the value of *cuisine* and *restaurant* slot types with new ones extracted from our database. For instance, for the sentence "*[Kohinoor](restoran) güzel görünüyor*", we replaced other available restaurant names from our database with "Kohinoor" and added the new sentences to the training data set. As a result, we observed an increase in the number of sentences mostly in dialog acts such as "*inform*" and "*choose*" (Table II & Table III). General statistics of the Augmented SelfPlayTR are also provided in Table I. Note that the *# of dialogues* statistic in Table I is not relevant for Augmented SelfPlayTR due to augmentation process that occur at sentence level.

C. Crowdsourcing Interfaces & Data Preparation Tasks

To prepare and annotate dialogues for WOZTR and SelfPlayTR data sets, annotation screens were designed and implemented using an open-source crowdsourcing framework called Pybossa, which is used for creating crowdsourcing tasks, presenting these tasks to annotators, and saving the results. Two undergraduate volunteers annotated the data sets using these implemented screens. We believe these crowdsourcing screens (and their templates[6]) will be beneficial for developing similar data sets in future studies.

For WOZTR, two tasks were given to the crowd workers. The former was the translation of the dialogues and the latter was the annotation of those dialogues. In the translation task, machine translations of the given sentences were obtained from Yandex.Translate API[7] and provided to the crowd workers as an additional help for speeding up the process. The crowd workers were then expected to translate the given dialogues on their own, sentence by sentence without corrupting the flow of the entire dialogue. For the annotation task, slot types from the English sentences were provided and it was expected from crowd workers to find corresponding words in Turkish sentences and annotate them. For example, for the word "italian" which was annotated as "cuisine" in an English sentence, it was expected from crowd workers to annotate "italyan" (Turkish) as "cuisine" too in the Turkish sentence. For SelfPlayTR, the only task was paraphrasing the given dialogue outlines. It was expected from crowd workers to

[5]Data augmentation in data analysis is a technique to increase the amount of data by either adding slightly modified copies of already existing ones or by newly creating synthetic data.

[6]https://github.com/TR-GODA/PybossaTemplates

[7]https://yandex.com/dev/translate

TABLE IV
A SAMPLE DIALOGUE OUTLINE PRODUCED BY SELF-PLAY (ON THE LEFT) AND PARAPHRASED BY CROWD WORKERS (ON THE RIGHT). ENGLISH TRANSLATIONS ARE PROVIDED BELOW EACH DIALOGUE UTTERANCE WITH AN ITALIC FONT.

Outline	Paraphrase
Sistem: iste(amaç) *System: ask(goal)*	Size nasıl yardım edebilirim? *How can i help you?*
Kullanıcı: bilgilendir(amaç=restoran rezervasyonu, gün=cuma, mutfak=fark etmez, fiyat=pahalı, saat=13) *User: inform(goal=restaurant reservation, date=friday, cuisine=does not matter, price=expensive, time=13)*	Cuma günü saat 13 için pahalı bir yerde restoran rezervasyonu yaptırmak istiyorum, mutfak fark etmez. *I want to book a restaurant for 13 o'clock on friday in an expensive place, the cuisine does not matter.*
S: iste(yer) *S: ask(location)*	Neredeki restoranlarla ilgileniyorsunuz? *Which area are you looking for a restaurant?*
K: bilgilendir(yer=avrupa yakası,kişi sayısı=3) *U: inform(location=european side, number of people=3)*	Avrupa Yakası'ndaki bir restoranda 3 kişilik yer ayırtmak istiyorum. *I want to reserve a place for 3 people in a european side restaurant.*
S: teklif_et(restoran=the good luck chinese food takeaway, restoran=yu garden, restoran=chiquito restaurant bar) *S: offer(restaurant=the good luck chinese food takeaway, restaurant=yu garden, restaurant=chiquito restaurant bar)*	The Good Luck Chinese Food Takeaway, Yu Garden ve Chiquito Restaurant Bar'dan birine ne dersiniz? *How about one of these; The Good Luck Chinese Food Takeaway, Yu Garden and Chiquito Restaurant Bar?*
K: tekrar_sor() *U: ask_for_repeat()*	Anlamadım? *Excuse me?*
S: teklif_et(restoran=the good luck chinese food takeaway, restoran=yu garden, restoran=chiquito restaurant bar) *S: offer(restaurant=the good luck chinese food takeaway, restaurant=yu garden, restaurant=chiquito restaurant bar)*	The Good Luck Chinese Food Takeaway, Yu Garden ve Chiquito Restaurant Bar'dan birine ne dersiniz? *How about one of these; The Good Luck Chinese Food Takeaway, Yu Garden and Chiquito Restaurant Bar?*
K: alternatif_sor() *U: ask_for_alternative()*	Başka seçenekler var mı? *Are there any other options?*
S: teklif_et(restoran=darrys cookhouse and wine shop, restoran=cocum, restoran=michaelhouse cafe) *S: offer(restaurant=darrys cookhouse and wine shop, restaurant=cocum, restaurant=michaelhouse cafe)*	Cookhouse and Wine Shop, Cocum ve Michealhouse Cafe sizin için uygun görünüyor. *Cookhouse and Wine Shop, Cocum and Michealhouse Cafe seem suitable for you.*
K: seç(restoran=cocum) *U: choose(restaurant=cocum)*	Cocum olsun lütfen. *Cocum please.*
S: onaylandı(cocum) *S:confirm(cocum)*	Cocum için rezervasyonunuzu onaylıyorum. *I confirm your reservation for the Cocum.*
K: veda_et() *U: bye()*	Teşekkürler, hoşça kalın. *Thanks, bye.*
S: veda_et() *S: bye()*	Güle güle. *Good bye.*

TABLE V
5-FOLD CROSS-VALIDATION RESULTS.

	Dialogue Act Classification	Slot Identification
WOZTR	0.84	0.77
SelfPlayTR	0.82	0.89
Augmented SelfPlayTR	0.89	0.99

TABLE VI
SLOT IDENTIFICATION SCORES.

	WOZTR	SelfPlayTR	Augmented SelfPlayTR
goal	-	0.87	0.99
resturant	0.33	0.93	0.99
location	0.76	0.87	0.98
number of person	0.94	0.92	0.98
price	0.75	0.90	0.99
time	0.95	0.93	0.98
date	0.93	0.84	0.98
cuisine	0.75	0.83	0.99

paraphrase the given dialogues, sentence by sentence without corrupting the flow of the dialogue.

IV. EXPERIMENTAL RESULTS AND DISCUSSIONS

We evaluate the performances in terms of dialog act classification and slot identification scores. To accomplish these tasks, we use the DIET classifier [17] (an architecture for dialog act and slot identification based on Transformers [18]) on RASA (an open-source language understanding and dialogue management platform) [19]. We use FastText pre-trained word embeddings [20] as Turkish word vectors. 5-fold cross-validation is applied to each data set.

Table V gives the overall evaluations as macro average F1 scores. Detailed slot identification and dialog act classification scores are provided in Table VI and Table VII. It may be seen from the tables that WOZTR is slightly better than SelfPlayTR in terms of dialog act classification, yet, SelfPlayTR is far better in slot identification. Also, the augmentation process shows very good performance and boosts the SelfPlayTR model both for dialog act classification and slot identification stages.

As a second set of experiments, we make cross-evaluation of the models on different data sets; as explained above although these data sets are from the same domain, they

TABLE VII
DIALOGUE ACT CLASSIFICATION SCORES.

	WOZTR	SelfPlayTR	Augmented SelfPlayTR
greet	-	0.55	0.67
inform	0.95	0.98	0.99
choose	0.89	0.95	0.99
bye	0.98	0.71	0.80
ask	0.90	0.99	0.99
ask for alternative	0.51	0.93	0.96
refuse	-	0.53	0.92
ask for repeat	-	0.84	0.83
change	-	0.39	0.90

TABLE VIII
CROSS EVALUATION RESULTS.

Train	Test	Dialog Act Classification	Slot Identification
WOZTR	SelfPlayTR	0.27	0.49
SelfPlayTR	WOZTR	0.43	0.55
Augmented SelfPlayTR	WOZTR	0.52	0.59

contain different types of dialog acts and slot values due to translation and localization. As a result, it is not expected that they perform much better than the original models developed with the data from the same set (Table V). But it still makes sense to make their cross-evaluation to see which model is more generalizable. For this, we train our models with data sets from each group (WOZTR and SelfPlayTR) and test it with a data set from the other group.

In [3], it is also shown that the self-play approach results in higher diversity in dialogue flows and language when compared to MultiWOZ. In line with the [3], we showed in our experiments that a model which is trained with SelfPlayTR performs better on WOZTR rather than a model which is trained with WOZTR and tested with SelfPlayTR (Table VIII). Additionally, the model trained on augmented SelfPlayTR performs much better on WOZTR.

V. Conclusion

In this article, we introduced our contributions to the Turkish goal-oriented dialogue system research. These are 3 Turkish data sets for the evaluation of goal-oriented dialogue agents, consisting of around 600 dialogues, 2.7K dialogue utterances and 15K augmented dialogue utterances, comparison of two different data preparation methods (viz. self-play and Wizard-of-Oz) and their quantitative analysis.

To the best of our knowledge, these are the first publicly available human-to-human Turkish data sets for goal-oriented dialogue systems. We believe that self-play is a more promising and cost-efficient technique for developing dialogue data sets especially in low-resource languages such as Turkish. In addition to these main contributions, we also provided our self-play framework as an open-source GitHub project as well as our crowdsourcing platform templates with the hope to pave the way for the Turkish research studies in the field.

For future work, the framework may be extended to cover dialogues that are not goal-oriented. Also, multi-intent support should be implemented to handle multiple domains in one dialogue. As is the case for every NLP tasks, the data set sizes should be increased to obtain comparable results with widely-studied languages.

Acknowledgment

Thanks to Erdinç Kandemir for his help in Pybossa tasks.

References

[1] D. Jurafsky and J. Martin, "Speech and language processing - an introduction to natural language processing, computational linguistics, and speech recognition," in *Prentice Hall series in artificial intelligence*, 2000.

[2] N. Dahlbäck, A. Jönsson, and L. Ahrenberg, "Wizard of oz studies: why and how," in *Proceedings of the 1st international conference on Intelligent user interfaces*, 1993, pp. 193–200.

[3] P. Shah, D. Hakkani-Tür, G. Tür, A. Rastogi, A. Bapna, N. N. Kennard, and L. Heck, "Building a conversational agent overnight with dialogue self-play," *ArXiv*, vol. abs/1801.04871, 2018.

[4] P. Budzianowski, T.-H. Wen, B. Tseng, I. Casanueva, S. Ultes, O. Ramadan, and M. Gasic, "Multiwoz - a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling," in *EMNLP*, 2018.

[5] H. Chen, X. Liu, D. Yin, and J. Tang, "A survey on dialogue systems: Recent advances and new frontiers," *SIGKDD Explor.*, vol. 19, pp. 25–35, 2017.

[6] L. E. Asri, H. Schulz, S. Sharma, J. Zumer, J. Harris, E. Fine, R. Mehrotra, and K. Suleman, "Frames: a corpus for adding memory to goal-oriented dialogue systems," in *SIGDIAL Conference*, 2017.

[7] Y. Li, K. Yao, L. Qin, W. Che, X. Li, and T. Liu, "Slot-consistent nlg for task-oriented dialogue systems with iterative rectification network," in *ACL*, 2020.

[8] Y. Dai, H. Li, C. Tang, Y. Li, J. Sun, and X.-D. Zhu, "Learning low-resource end-to-end goal-oriented dialog for fast and reliable system deployment," in *ACL*, 2020.

[9] S. Gao, Y. Zhang, Z. Ou, and Z. Yu, "Paraphrase augmented task-oriented dialog generation," *ArXiv*, vol. abs/2004.07462, 2020.

[10] R. Takanobu, R. Liang, and M. Huang, "Multi-agent task-oriented dialog policy learning with role-aware reward decomposition," *ArXiv*, vol. abs/2004.03809, 2020.

[11] A. Raux, B. Langner, D. Bohus, A. Black, and M. Eskénazi, "Let's go public! taking a spoken dialog system to the real world," in *INTERSPEECH*, 2005.

[12] M. Henderson, B. Thomson, and S. Young, "Word-based dialog state tracking with recurrent neural networks," in *SIGDIAL Conference*, 2014.

[13] E. Stepanov, G. Riccardi, and A. Bayer, "The development of the multilingual luna corpus for spoken language system porting," in *LREC*, 2014.

[14] G. Çelikkaya and G. Eryiğit, "Use of nlp techniques for an enhanced mobile personal assistant: The case of Turkish," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 5, no. 3, pp. 94–104, 2017.

[15] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel *et al.*, "Mastering chess and shogi by self-play with a general reinforcement learning algorithm," *arXiv preprint arXiv:1712.01815*, 2017.

[16] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan, and D. Hassabis, "A general reinforcement learning algorithm that masters chess, shogi, and go through self-play," *Science*, vol. 362, pp. 1140 – 1144, 2018.

[17] T. Bunk, D. Varshneya, V. Vlasov, and A. Nichol, "Diet: Lightweight language understanding for dialogue systems," *ArXiv*, vol. abs/2004.09936, 2020.

[18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *ArXiv*, vol. abs/1706.03762, 2017.

[19] T. Bocklisch, J. Faulkner, N. Pawlowski, and A. Nichol, "Rasa: Open source language understanding and dialogue management," *arXiv preprint arXiv:1712.05181*, 2017.

[20] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of tricks for efficient text classification," *ArXiv*, vol. abs/1607.01759, 2017.