Neural Language Generation for a Turkish Task-Oriented Dialogue System

Artun Burak Mecik*, **Volkan Ozer***, **Batuhan Bilgin***, **Tuna Cakar**, **Seniz Demir**†
Department of Computer Engineering,
MEF University, Istanbul, TURKEY
{mecika, ozerv, bilginba, cakart, demirse}@mef.edu.tr

Abstract

Rapidly growing language and speech-enabled technologies contribute to the development of task-oriented dialogue systems. The demand for better user engagement has been increasing at an accelerating pace and this brings new remarkable challenges including the generation of informative and natural system utterances. In this work, our ultimate goal is to develop a Turkish task-oriented dialogue system that enables users to navigate over a map in order to get informed about dining venues that best match their preferences and make reservations based on received recommendations. This paper presents the pipeline architecture of our dialogue system with a particular focus on the language generator. We utilize an open source framework for building the components of our system and develop a sequence-to-sequence (Seq2Seq) neural model for language generation. This pioneering work is the first that proposes the use of a neural generation model in a Turkish conversational system. Our evaluations suggest that Turkish neural generation from meaning representations given in the form of dialogue acts is effective, but still in need of further improvements.

1 Introduction

In the last decades, task-oriented dialogue systems with human-like communication capabilities (Chen et al., 2017; Zhao et al., 2019) have been widely deployed in applications with commercial value such as restaurant reservation (Henderson et al., 2019) and online shopping (Yan et al., 2017). As opposed to open-domain dialogue systems without a clear dialogue goal, these systems present adequate intelligence in understanding user utterances and taking actions in response to accomplish constrained tasks. Task-oriented dialogue systems that can converse naturally with users through text or auditory conversation have received increasing attention of language and speech communities. Conventional task-oriented dialogue systems combine different modules in a pipeline architecture (Raux et al., 2005): i) language understanding (Gupta et al., 2019), ii) dialogue state tracking (Lee and Stent, 2016), iii) dialogue policy (English and Heeman, 2005), and iv) natural language generation (Zhu et al., 2019). These modules are independently trained and optimized with separate objective functions. Pipeline architectures often suffer from cascaded error propagation and a change in the output representation of a previous module also affects subsequent modules. Recent end-to-end task-oriented dialogue systems (Liu and Lane, 2018; Wen et al., 2017) mitigate these problems by training a single model directly from data without distinguishing individual modules and optimizing a single objective function. Although end-to-end systems enable multi-domain adaptation by minimizing laborious feature engineering, they unfortunately might generate generic utterances or utterances that are repetitive.

End users face utterances generated by dialogue systems and their satisfaction heavily depends on the quality and semantic coherence of these productions. The natural language generation module is mainly responsible for producing informative and fluent utterances that engage users and improve their experiences. The input to this module is often a dialog act given in a semantic form that either conveys or requests information as directed by the dialogue policy (Zhao and Kawahara, 2019). A dialogue act is a meaning representation of an action (i.e., system or user) that can be realized using one or more sentences. Depending on the action type (e.g., greeting, inform, or confirm), dialog acts contain one or more slots (attributes) of different types (e.g., numeric or string) to fulfill the meaning (e.g., *inform(name="Green Food",phone=415986223)*).

*These authors contributed equally to the work.
†Corresponding author

Early research methods of language generation for task-oriented dialogue systems include manually-crafted rules and templates. This kind of generation is adequate to cover all information captured in a dialog act, but it lacks preferred flexibility, requires heavy manual effort, and necessitates domain expertise. Although these issues hinder scalability across different domains, they can be addressed by statistical generation approaches which can learn human writing patterns directly from annotated data. Recently, neural generation models have become a common approach for joint learning of sentence planning to cover all selected information and surface realization to incorporate that content in a fluent text. However, it is not straightforward to find large amounts of domain-specific labeled data (real conversational data) for training statistical or neural generation models, and it is yet infeasible for some languages including the morphologically rich language Turkish.

In this study, we describe our efforts towards building a task-oriented dialogue system for Turkish that enables users to navigate over a map and reach descriptive information of dining venues based on their preferences until a venue is booked for reservation. The system, implemented as a mobile application, interacts with users through an interface where textual and visual modalities are employed. In the current version, all venues that match user preferences are listed on a map and the user is presented with a single sentence description of any venue selected on that map. Although our goal is to enhance this work to a venue recommendation and reservation system where more sophisticated human-like conversations can take place, the system currently engages in a limited dialogue with end users mainly due to the lack of labeled conversational corpora for Turkish in this domain. We use the RASA open-source machine-learning based framework (Bocklisch et al., 2017) to develop natural language understanding and dialogue management components of the system. We also leverage knowledge obtained from a human-annotated English conversational data in restaurant reservation domain to imitate humans while building our dialogue policies.

In this paper, our focus is on the language generation component of the system which is implemented as a sequence-to-sequence (Seq2Seq) neural model. To our best knowledge, this work is the first that utilizes a neural generation model for producing task-oriented Turkish utterances. The literature does not report any study to show how effective neural models are in generating Turkish sentences from dialog acts in terms of coverage and correspondence to human generated texts. In this study, we report the system performance using automatic evaluation metrics over our corpus of 4200 pairs of dialog acts and reference sentences collected via crowdsourcing. In our experiments, we also assess the impact of delexicalization on the quality of generated utterances where verbalizations of rare words in dialogue acts are targeted.

2 Related Work

Previous research on pipelined dialogue systems has focused on improving the performance of individual components in the architecture. Rule-based parsing methods (Denis et al., 2006), multiclass classification algorithms such as SVMs (Sarikaya et al., 2016), and deep convex networks (Tur et al., 2012) were shown to be effective in detecting user's intent. Promising results were also achieved with the use of recurrent (Yao et al., 2013) and recently hierarchical (Zhao and Kawahara, 2019) neural networks. Mapping textual spans of an utterance to slots in a dialogue act was often considered as a sequence tagging problem and quite good results were achieved with maximum entropy models such as conditional random fields (CRFs) and stochastic finite state transducers (Raymond and Riccardi, 2007). Deep belief networks (Deoras and Sarikaya, 2013), convex networks (Deng et al., 2012), and bidirectional long short-term memory networks (Jaech et al., 2016) were later shown to outperform CRF-based approaches. A variety of different approaches have emerged for dialogue state tracking. A tracker that benefits from domain independent rules and basic probability (Wang and Lemon, 2013), and a CRF-based discriminative approach (Ren et al., 2013) achieved comparable performances to machine-learning based methods. The effectiveness of neural models was also exploited for state tracking task. One pioneering work combined an RNN model with delexicalized feature representations in order to generalize it to unseen slots and values, and with an online unsupervised adaptation approach to exploit unlabeled data (Henderson et al., 2014). An RNN model was later used to train a state tracker capable of working across different domains (Mrkšić et al., 2015). Recently, dialogue state tracking was tackled as

a reading comprehension problem and addressed using an attention-based neural network (Gao et al., 2019). Reinforcement learning was heavily utilized for learning dialogue policies (Cuayáhuitl, 2017; Shah et al., 2016; Weisz et al., 2018). Recent experiments suggested that utilizing pre-trained language models in task-oriented dialogue components is a promising approach (Wu et al., 2020).

Although many generation methods have been proposed so far, they can be broadly classified into three types. Rule or template based approaches require significant expertise and human effort, and the number of manually constructed templates is limited (Jurčíček et al., 2014; Mitchell et al., 2014). On the other hand, stochastic or statistical approaches enable less monotonic generation by training a generator from data directly (Mairesse et al., 2010; Mairesse and Walker, 2011; Oh and Rudnicky, 2000). Recent developments in neural networks have enabled generation to be handled as a transformation from meaning representations to system responses via a single model. In a work that simulates the few-shot learning setting with scarce annotated data, a multilayer transformer model was trained for generating responses and generalization to new domains was achieved by utilizing pre-trained language models (Peng et al., 2020). The work of Wen et al. (Wen et al., 2015a) jointly utilized recurrent and convolutional neural networks for realizing the content of a dialog act, and the RNN-based generator that encodes one-hot representation of the dialog act as its initial state was trained with semantically unaligned data. Semantically controlled long short-term memory was also explored for training a generator from unaligned data where sentence planning and surface realization are jointly optimized (Wen et al., 2015b). A recent work employed a Seq2Seq generator with attention using GRU cells to capture the semantic content of dialog acts and used a language model to achieve naturalness in generated utterances (Zhu et al., 2019). Our work is most similar to the work of Dušek and Jurčíček (Dušek and Jurčíček, 2016) but their dialog act representation formed by concatenating triples of act type, slot name, and slot value differs from our input representation.

Turkish, a morphologically rich language with free-constituent order, has been in focus of language processing research for many years (Oflazer and Saraclar, 2018). However, Turkish language generation has been relatively less-studied up to now. Scarcity of available data and lack of annotations are some of the obstacles to developing robust systems with high performances. Previous generation literature is restricted to some well-known problems of surface form generation (Cicekli and Korkmaz, 1998; Ayan, 2000) and text summarization (Nuzumlalı and Özgür, 2014; Çagdas Can Birant et al., 2016). Recently, template-based language generation was employed in a venue recommendation system (Elifoğlu and Güngör, 2018) where a distinct template for each venue property is used. To our best knowledge, Turkish text generation from structured data has not been yet exploited. Moreover, there is no prior knowledge as to whether the use of neural models in generating utterances from dialog acts is effective or not, especially in domains with a very limited amount of annotated data. Our work reports first empirical evaluations that measure the usability and effectiveness of a neural model in this task.

3 System Architecture

Our task-oriented dialogue system is implemented as a mobile application and exhibits the traditional pipeline architecture. A user utterance is processed by three downstream components before a dialog act is transferred to the language generation component. In the rest of this section, the mobile application, and the language understanding and dialogue management components are described in detail.

3.1 Mobile Application

Users interact with our mobile application through an interface where they rely on menus that display listings of choices for different properties of dining venues. At any time while using the application, users can search for venues exhibiting different properties by choosing any of these alternatives. As shown in Figure 1-a, a user is initially asked to specify venue properties being sought (i.e., its location, customer rating, price range, and type of served food). All venues that exhibit these properties are listed on a map of the selected region (Figure 1-b) and the user can navigate between these venues. If the user selects a listed venue on the map, a single sentence description of the venue along with some of the matching properties are presented to the user in a separate window at the bottom of the screen. That description is produced by our neural generator using the meaning representation passed from the system. On this map view, the user can

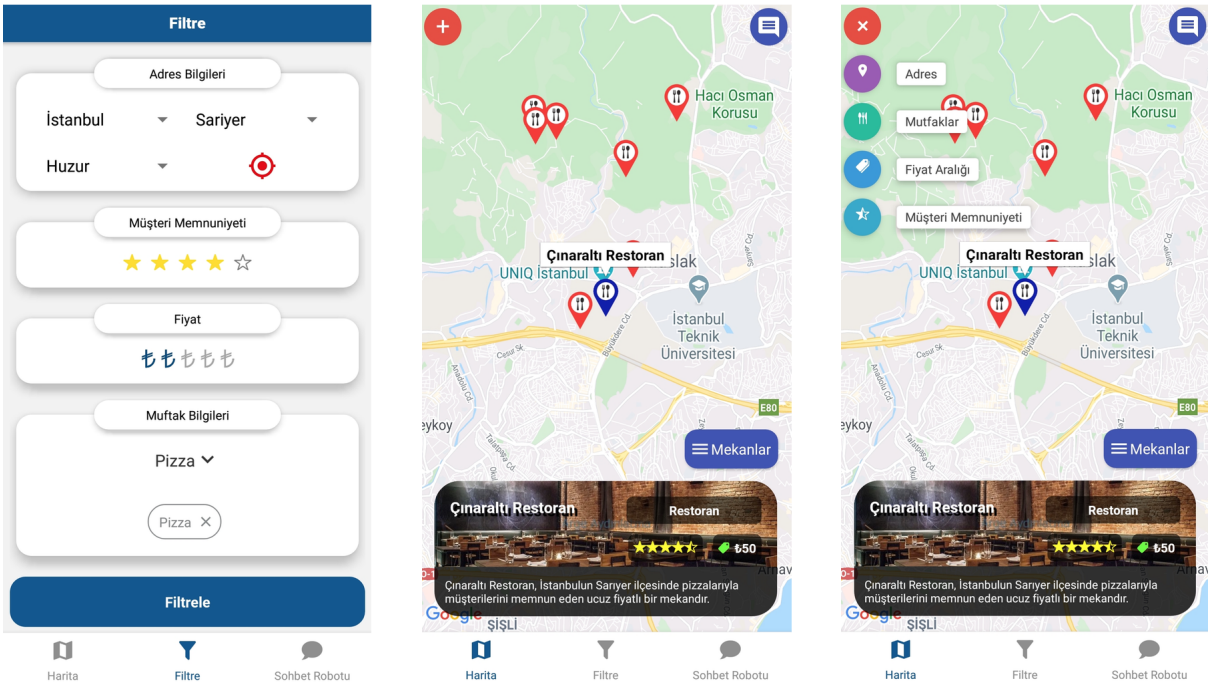

Figure 1: (a) Opening screen view (b) Map listing view (c) Map listing+New search view.

also update venue properties from the menu given on the upper left corner (the red icon) and start a completely new search (Figure 1-c). Although it is not fully implemented yet, the user will engage in a dialogue with the system over this map view (using the blue icon on the upper right corner), and get recommendations/make reservations in the future.

3.2 Natural Language Understanding

This component identifies user's intent from a given utterance by classifying it into predefined classes. Moreover, it extracts information related to that intent and uses them to fill corresponding slots. In the current implementation, we use the RASA NLU framework (Bocklisch et al., 2017) for building our language understanding component. The RASA NLU combines embeddings of word tokens that appear in a sentence in order to obtain a representation of the sentence. An SVM classifier trained on these sentences then classifies a given utterance into one or more intents. For entity extraction, the framework offers different extractors and we train a CRF extractor using our custom entities. To train a Turkish intent classifier and an entity extractor, we use our dataset and some manually translated examples from an English dataset in the restaurant domain (Novikova et al., 2017). For each sentence in our collection, we manually determine the intent and annotate text spans that correspond to different entities with appropriate tags. For instance, Figure 2 shows a sentence and a part of its annotation.

```
{"text": "İstanbul, Karaköy'de yüksek memnuniyete sahip
        kafeterya ürünleri satan mekanları arıyorum.",
 "intent": "request",
 "entities":
  {
   "start": 0,
   "end": 8,
   "value": "İstanbul",
   "entity": "region"
  },
  {
   "start": 21,
   "end": 27,
   "value": "yüksek",
   "entity": "customersatisfaction"
  }, ....
 }
```

Figure 2: An annotated training data example for NLU.

3.3 Dialogue Management

This component maintains the current dialogue state by keeping user's intents and a dialogue history (dialogue state tracker). Its main responsibility is to estimate the user's goal at each turn of the dialogue. The dialogue history is treated as an abstraction of previous dialogue turns. Moreover, it behaves as the decision maker of the whole system and takes appropriate actions according to a policy by considering the current dialogue state. Due to lack of available Turkish dialogue conversations that we can use for training a dialogue management component, we first analyze the E2E dialogue challenge dataset that consists of English conversations in the restaurant reservation domain (Li et al.,

2018). By processing the provided dialogues and manually filtering intents and entities that are out of our scope, we then compile training data for our dialogue manager. Since our focus here is to mimic natural conversations rather than modeling the language, this data collection approach enables us to train our language-independent dialogue manager with 2800 different representations of actual conversations of varying length. Using an RNN-based approach, the RASA Core dialogue engine learns policies from our training data.

4 Neural Turkish Generation Component

We develop a sequence to sequence (Seq2Seq) model (Liu et al., 2017; Sha et al., 2018) as our generation component. The model utilizes a dialog act as input and produces a single Turkish sentence to preferably convey all the information expressed in that act. Since there is no available data that we can use to train the model, we first conduct human subject experiments in order to collect a small-sized corpus as our starting point.

4.1 Corpus Collection

A dialog act is a logical representation of meaning that might be expressed using single or multiple sentences. Each dialog act contains an action type (i.e., what is intended to be conveyed by the system or user) and a set of slot-value pairs associated with that action (e.g., the properties of a venue in focus). Since our goal is to engage in dialogue with end users, restricting the system to only describe properties of a venue is not adequate. Moreover, the number of slots that might be associated with an action type is too large to be listed in a single sentence with a moderate complexity. In order to determine action types and slots that would be utilized, we explore similar well-studied datasets compiled for other languages (SFRest (Wen et al., 2015b), E2E (Novikova et al., 2017), Bagel (Mairesse et al., 2010)). Nine different action types are incorporated into the current version but these action types and slots will be populated in the future:

- **greeting:** Greet the user
- **goodbye:** Farewell the user
- **inform:** Present all properties of a venue
- **inform_only:** State the uniqueness of a venue with specified properties
- **inform_not:** State the non-existence of a venue with specified properties
- **inform_all:** Present all venues with specified properties
- **request:** Query existence of venues with specified properties
- **compare:** Compare two venues with respect to a property
- **compare_only:** Compare a venue with a number of other venues with respect to a property

One or more slots are defined for each action type as shown in Table 1. For instance, the action type inform might contain up to six slots. The values of some slots are verbatim strings whereas the remaining values are selected from a catalog.

Actions	Slots	Types
greeting, goodbye	Message	String
inform, inform_only, inform_not, inform_all, request, compare, compare_only	Name, Region, Near, Customer Satisf., Price Range, Cuisine	String String String Catalog Catalog Catalog
compare, compare_only	Other Venues' Names, Other Venues' Cust. Satisf., Other Venues' Price Range	String Catalog Catalog

Table 1: Action types and slots.

We conduct a data collection study with 90 participants where each participant is presented with 45-50 dialog acts of different action types. The participants are asked to express a given dialog act in a single sentence and to use all slots given in the act. Moreover, they are told to not rely on their commonsense knowledge or use any information that might be inferred from the given ones. In the study, greeting and goodbye actions are not used. Each dialog act contains two to four randomly chosen slots in addition to the name of the venue in focus. It is guaranteed that a participant receives different sets of slots for the same action type even if the number of slots are the same. We use both real and artificial data in order to fill in slot values. Information about a small set of dining venues is obtained from an online restaurant search service and that information is augmented with artificial information in order to expand the collection. For instance, new dialog acts are produced by adding new neighbour restaurants to existing dialog acts without any neighbourhood information. Each dialog act is presented to four different participants. At the end, 4200 dialog act and reference sentence pairs are collected. Figure 3 shows two dialog acts with three reference sentences from our collection.

(type='inform', name='Lezzet Mekanı', customer_satisfaction='Yüksek', cuisines='Tatlı, Dünya Mutfağı Yemekleri', price_range='Pahalı', region='Caddebostan, İstanbul')

i) Lezzet Mekanı, İstanbul Caddebostan'da, tatlı ve dünya mutfağı yemekleri servis eden pahalı fakat lezzetli yemekleriyle müşteri memnuniyetini üst seviyede tutan bir mekandır.
(Lezzet Mekanı is a place in Caddebostan, Istanbul that serves sweet and world cuisine and keeps customer satisfaction at the highest level with its expensive but delicious dishes.)

ii) Dünya mutfağına ait yemekler ve tatlılar bulabileceğiniz, müşteri memnuniyeti konusunda çok başarılı olması rağmen fiyatları pahalı olan Lezzet Mekanı, İstanbul Caddebostan'da bulunmaktadır.
(Lezzet Mekanı, where you can find desserts and dishes from the world cuisine, is very successful in customer satisfaction though it is expensive, and is located in Caddebostan, Istanbul.)

iii) İstanbul Caddebostan'da tatlılar ile dünya mutfağına ait yemekler yenebilecek Lezzet Mekanı, pahalı fiyata yemekler sunan ve müşterilerin çok memnun olduğu bir restorandır.
(Lezzet Mekanı in Istanbul Caddebostan, where you can eat desserts and dishes from the world cuisine, is a restaurant that offers expensive dishes and where customers are very satisfied.)

(type = 'compare', name = 'Cafe Botanica', price_range = 'Ortalama', other_venues_names = 'Mayday Cafe Bar, Mevlana Lokantası, Cafe de Kedi', other_venues_price_range = 'Ucuz')

i) Cafe Botanica; ucuz fiyatlı Mayday Cafe Bar, Mevlana Lokantası, Cafe de Kedi'ye kıyasla ortalama fiyatlı bir mekandır.
(Cafe Botanica is an average-priced venue compared to the cheaply priced Mayday Cafe Bar, Mevlana Lokantası and Cafe de Kedi.)

ii) Cafe Botanica ortalama fiyatlardayken Mayday Cafe Bar, Mevlana Lokantası ve Cafe de Kedi ucuz mekanlardır
(While Cafe Botanica is at average prices, Mayday Cafe Bar, Mevlana Restaurant and Cafe are cheap venues.)

iii) Ortalama fiyatlarıyla bilinen Cafe Botanica, Mayday Cafe Bar, Mevlana Lokantası ve Cafe de Kedi gibi mekanların ucuz menülerine kıyasla pahalı kalmaktadır
(Cafe Botanica which is known with its average prices is expensive compared to the venues with cheap menus Mayday Cafe Bar, Mevlana Lokantası and Cafe de Kedi.)

Figure 3: Examples of dialog acts and reference sentences.

4.2 Input Representation

A dialog act is represented as a sequence of field value pairs (e.g., field$_1$= value$_1$) where the first pair corresponds to the action type and the rest are slot value pairs. The value of a field might contain a single word or a sequence of words. The field name (f_x) and its position in the value sequence (p_x) are used to represent each word (w_x). To represent the position of a word in a sequence, its position from the beginning of the sequence (p_x+) and from the end of the sequence (p_x-) are used. Therefore, a word that appears in a field value is represented as $R_x = (f_x , p_x+ , p_x-)$. All punctuation characters in field values are represented similarly. Table 2 shows the representations of all words in the dialog act *(type = 'inform_only', name = 'Denizaltı Restaurant', cuisine = 'Kafeterya Ürünleri, Türk Yemekleri', region = 'Urla, İzmir', near = 'VVapiano').* In this example, the value of the name field consists of two words, namely Denizaltı and Restaurant. The word Denizaltı is the first word starting from the beginning of value sequence and the second word from the end of the sequence. Therefore, its representation is *(name,1,2).*

Each word in a field value (w_x) and its representation (R_x) are encoded into four embeddings and then concatenated to form the final input embedding of the encoder ($i_e = w_e \oplus f_e \oplus p_e+ \oplus p_e-$). A reference sentence already has a sequence of word tokens and thus each token is encoded into a word embedding only:

- Word embedding: Vector representation of the word (w_e)
- Field embedding: Vector representation of the field name (f_e)
- Beginning position embedding: Vector representation of the position from the beginning of the field value (p_e+)
- End position embedding: Vector representation of the position from the end of the field value (p_e-)

Field	Value	Word	Represent.
type	inform_only	inform_only	(type,1,1)
name	Denizaltı Restaurant	Denizaltı	(name,1,2)
		Restaurant	(name,2,1)
cuisine	Kafeterya Ürünleri, Türk Yemekleri	Kafeterya	(cuisine,1,5)
		Ürünleri	(cuisine,2,4)
		,	(cuisine,3,3)
		Türk	(cuisine,4,2)
		Yemekleri	(cuisine,5,1)
region	Urla, İzmir	Urla	(region,1,3)
		,	(region,2,2)
		İzmir	(region,3,1)
near	VVapiano	VVapiano	(near,1,1)

Table 2: Word representations.

4.3 Sequence-to-Sequence Generation Model

To capture temporal processing and feedback requirements of sequences in learning, we approach to the generation problem using a recurrent neural network (RNN) based solution. RNN models are of great utility in computing current output with respect to previous computations kept in hidden states and their processing power makes them widely applicable to speech recognition (Hsu et al., 2016; Prabhavalkar et al., 2017) and language processing studies (Socher et al., 2011; Daza and Frank, 2018). In our work, dialog acts and reference sentences are sequences of

variable-length. Thus, we formulate our generation task as sequence-to-sequence (Seq2Seq) learning (Sutskever et al., 2014), a type of an RNN with encoder-decoder. Our model uses a long short-term memory (LSTM) based RNN to encode the input sequence into hidden states. A second LSTM-based RNN is used to decode hidden states and generate the output sequence. Given that x_t and h_t are the input and hidden state at time step t; i, f, and o are input, forget and output gates; and C and $\tilde{C}$ are cell and candidate cell states, the computations used with LSTM units are as follows:

$$\begin{aligned}
\mathbf{i}_t &= \sigma(\mathbf{W}_i.[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_i) \\
\mathbf{f}_t &= \sigma(\mathbf{W}_f.[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_f) \\
\mathbf{o}_t &= \sigma(\mathbf{W}_0.[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_0) \\
\tilde{\mathbf{C}}_t &= \tanh(\mathbf{W}_C.[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_C) \\
\mathbf{C}_t &= \mathbf{f}_t * \mathbf{C}_{t-1} + \mathbf{i}_t * \tilde{\mathbf{C}}_t \\
\mathbf{h}_t &= \mathbf{o}_t * \tanh(\mathbf{C}_t)
\end{aligned} \tag{1}$$

5 Evaluation

Neural models often suffer from rare words while generating text from data since their verbalization cannot be predicted properly. Delexicalization is one of the mostly studied solutions to this issue where words are replaced with placeholders in data before being used for training. Texts produced by the generation model are then processed to replace these placeholders with actual words that appear in the original data. For this study, we delexicalize our input collection (-Del) and obtain a second version of our dataset (+Del). We only replace content words of slots with verbatim strings (e.g., name and region in Table 1) and leave those with categorical values (e.g., cuisine and price range) untouched. We have different dialog acts that differ only in slot values that are not replaced during delexicalization. Therefore, these dialog acts are counted as different acts in the second dataset. The number of placeholders in our delexicalized dataset corresponds to 17.71% of all words in reference sentences. Table 3 presents token-based statistics for both datasets.

We train two models on both original and delexicalized datasets. The first model is the sequence-to-sequence model described in Section 4.3 (Model_Att-) and the same model augmented with an attention mechanism (Model_Att+). We perform experiments to finetune model parameters by optimizing BLEU score on the development set. The models reported here use a single hidden layer and 700 LSTM units in encoder and decoder. Word embeddings of length 400, field embedding of length 50, and position embedding of length 5 are used. The epoch number is set to 10 and Adam optimizer with a learning rate of 0.003 is utilized. We compare our models with a prior Seq2Seq generation model (Liu et al., 2017) (Model_SA) whose primary focus is to generate one sentence biographies from Wikipedia infoboxes where the structure and content of infobox tables are modeled separately. In addition to learning what to convey in the output, the model also learns how to order the selected content. To train this structure-aware generation model with dual attention, we process all dialog acts in our dataset as infobox tables where the action type is considered as infobox table type and remaining slot value pairs as field value pairs of infobox tables. The same model parameters are used in learning.

Our input collection of dialog act and reference sentence pairs is splitted into training set of 3360, validation set of 420, and test set of 420 pairs. Table 4 presents the distribution of action types in these sets. In our experiments, we evaluate the efficiency of models in producing utterances from dialog acts and leave an evaluation of fluency and naturalness of these productions to future work. Here, we report performances using three evaluation metrics, BLEU (Papineni et al., 2002), ROUGE-n and ROUGE-L fmeasures (Lin, 2004), and Slot error rate (SER) (Riou et al., 2019). The slot error rate is computed as (M+R)/N where M and R correspond to the number of missing and redundant slots in the generated utterance, and N is the total number of

Property	**Input Data**	**Delexicalized Data**
Input Dictionary Size	2966	1247
Output Dictionary Size	2827	1177
Avg. DA Length	8.23	5.85
Avg. Ref. Text Length	15.13	11.96

Table 3: Properties of input datasets.

Act Type	**Training**	**Validation**	**Test**
inform	1690	220	200
inform_only	448	57	45
inform_not	662	81	93
inform_all	109	14	20
request	217	24	34
compare	120	11	12
compare_only	114	13	16

Table 4: Distribution of action types in datasets.

		BLEU	ROUGE-1	ROUGE-2	ROUGE-3	ROUGE-4	ROUGE-L	SER%
Model_Att-	*-Del*	0.017	0.161	0.033	0.010	0.002	0.129	70.2
	+Del	0.056	0.373	0.142	0.065	0.030	0.280	57.5
Model_SA	*-Del*	0.125	0.417	0.216	0.122	0.063	0.337	53.5
	+Del	0.323	0.849	0.632	0.459	0.328	0.543	35.6
Model_Att+	*-Del*	0.144	0.677	0.490	0.335	0.220	0.439	40.6
	+Del	0.302	0.857	0.634	0.452	0.314	0.524	35.5

Table 5: Performance scores of different models.

slots in the corresponding dialogue act.

For each model, we perform 5 runs with different random initializations on both datasets. Table 5 presents computed average scores. The model without attention (Model_Att-), not surprisingly, fails to learn the generation effectively and receives the lowest performance scores in all metrics. In addition, repetitive slot values and very similar sentence productions for different dialog acts are highly observed in the productions. On the other hand, we observe that our model with attention (Model_Att+) achieves highest BLEU and ROUGE scores on the original dataset (-Del). However, our model is behind the structure-aware model (Model_SA) on the delexicalized dataset (+Del) with respect to the BLEU score and over high order n-grams (ROUGE-3 and ROUGE-4). This less significant difference might be attributed to the fact that structure-aware model performs better in producing longer matching sequences than our model, which is also validated by ROUGE-L scores. Both models exhibit large performance improvements on the delexicalized dataset where BLEU scores are more than doubled. The measured positive impact of delexicalization on structure-aware model is more than what we observe with our model. The contribution of delexicalized dataset to model Model_SA is mainly observed on longer word sequences (e.g., from 0.063 to 0.328 in ROUGE-3).

Although BLEU and ROUGE evaluations validate word-based performances of these models, they do not provide any insights into the content quality, particularly the accuracy of selected content and the slot coverage of these models. On both datasets, our model with attention achieves the best slot error rates where delexicalization improves the performance by approximately 5%. The structure-aware model performs similarly only on delexicalized dataset, but the achieved improvement is more substantial than that seen in our model. These results demonstrate that both models need further improvements to better cover slot values resulting in fewer repeated or omitted information in produced utterances.

There are two major drawbacks of our model. First, it is learning from a corpus which is relatively small in comparison with many available datasets compiled for other languages. Second, it suffers from semantically similar entities in the dataset (e.g., cuisine or region) and entities that appear more frequently than others in the training data are selected by the model regardless of what is provided in the dialogue act. We argue that with a larger training corpus and more effective attention mechanism, our generation performance would be improved in the future.

6 Conclusion

This work presents our efforts towards developing a Turkish task-oriented dialogue system for venue recommendation and reservation. The current system is implemented using a pipeline approach, and natural language understanding and dialogue management components are built using the RASA open-source framework. In order to generate utterances from dialogue act representations, we develop a sequence-to-sequence neural model with attention. The model is trained with a small-sized Turkish corpus consisting of pairs of dialogue acts and reference sentences. To the best of our knowledge, this work is the first that investigates the use of Turkish neural generation in dialogue systems and measures the effectiveness of conversational generation from structured input on a morphologically rich language. In the future, we plan to collect a larger corpus and improve the performance of our generator. Moreover, enhancing the dialogue capabilities of our overall system and qualitatively evaluating the performance of the generation model are some of our future plans.

Acknowledgments

This work is supported by TUBITAK-ARDEB under the grant number 117E977.

References

Burcu Karagol Ayan. 2000. Morphosyntactic generation of turkish from predicate-argument structure. In *Proceedings of the COLING Student Session*.

Tom Bocklisch, Joey Faulkner, Nick Pawlowski, and Alan Nichol. 2017. Rasa: Open source language understanding and dialogue management. *ArXiv*, abs/1712.05181.

Çagdas Can Birant, Özgün Kosaner, and Özlem Aktas. 2016. A survey to text summarization methods for turkish. *International Journal of Computer Applications*, 144:23–28.

Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. A survey on dialogue systems: Recent advances and new frontiers. *Special Interest Group on Knowledge Discovery in Data Explorations Newsletter*, 19(2):25–35.

Ilyas Cicekli and Turgay Korkmaz. 1998. Generation of simple turkish sentences with systemic-functional grammar. In *Proceedings of the Joint Conferences on New Methods in Language Processing and Computational Natural Language Learning*, page 165–173, USA. Association for Computational Linguistics.

Heriberto Cuayáhuitl. 2017. *SimpleDS: A Simple Deep Reinforcement Learning Dialogue System*. Springer.

Angel Daza and Anette Frank. 2018. A sequence-to-sequence model for semantic role labeling. In *Proceedings of The Third Workshop on Representation Learning for NLP*, pages 207–216, Melbourne, Australia. Association for Computational Linguistics.

li Deng, Gokhan Tur, Xiaodong He, and Dilek Hakkani-Tur. 2012. Use of kernel deep convex networks and end-to-end learning for spoken language understanding. In *2012 IEEE Workshop on Spoken Language Technology, SLT 2012 - Proceedings*, pages 210–215.

Alexandre Denis, Matthieu Quignard, and Guillaume Pitel. 2006. A deep-parsing approach to natural language understanding in dialogue system: Results of a corpus-based evaluation. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*.

Anoop Deoras and Ruhi Sarikaya. 2013. Deep belief network based semantic taggers for spoken language understanding. In *INTERSPEECH*.

Ondřej Dušek and Filip Jurčíček. 2016. Sequence-to-sequence generation for spoken dialogue via deep syntax trees and strings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 45–51.

M. Elifoğlu and T. Güngör. 2018. A restaurant recommendation system for turkish based on user conversations. In *2018 26th Signal Processing and Communications Applications Conference (SIU)*, pages 1–4.

Michael English and Peter Heeman. 2005. Learning mixed initiative dialog strategies by using reinforcement learning on both conversants. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 1011–1018.

Shuyang Gao, Abhishek Sethi, Sanchit Agarwal, Tagyoung Chung, and Dilek Hakkani-Tur. 2019. Dialog state tracking: A neural reading comprehension approach. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 264–273.

Arshit Gupta, Peng Zhang, Garima Lalwani, and Mona Diab. 2019. CASA-NLU: Context-aware self-attentive natural language understanding for task-oriented chatbots. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1285–1290.

Matthew Henderson, Blaise Thomson, and Steve J. Young. 2014. Robust dialog state tracking using delexicalised recurrent neural networks and unsupervised adaptation. *2014 IEEE Spoken Language Technology Workshop (SLT)*, pages 360–365.

Matthew Henderson, Ivan Vulic, Inigo Casanueva, Paweł Budzianowski, Daniela Gerz, Sam Coope, Georgios Spithourakis, Tsung Hsien Wen, Nikola Mrksic, and Pei-Hao Su. 2019. Polyresponse: A rank-based approach to task-oriented dialogue with application in restaurant search and booking. In *Proceedings of the 2019 EMNLP and the 9th IJCNLP*, pages 181–186.

Wei-Ning Hsu, Yu Zhang, and James R. Glass. 2016. A prioritized grid long short-term memory rnn for speech recognition. *2016 IEEE Spoken Language Technology Workshop (SLT)*, pages 467–473.

Aaron Jaech, Larry Heck, and Mari Ostendorf. 2016. Domain adaptation of recurrent neural networks for natural language understanding. *arXiv preprint arXiv:1604.00117*.

Filip Jurčíček, Ondřej Dušek, Ondřej Plátek, and Lukáš Žilka. 2014. Alex: A statistical dialogue systems framework. In *Text, Speech and Dialogue*, pages 587–594. Springer International Publishing.

Sungjin Lee and Amanda Stent. 2016. Task lineages: Dialog state tracking for flexible interaction. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 11–21.

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

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81.

Bing Liu and Ian Lane. 2018. End-to-end learning of task-oriented dialogs. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 67–73.

Tianyu Liu, Kexiang Wang, Lei Sha, Baobao Chang, and Zhifang Sui. 2017. Table-to-text generation by structure-aware seq2seq learning. In *CoRR*.

François Mairesse, Milica Gašić, Filip Jurčíček, Simon Keizer, Blaise Thomson, Kai Yu, and Steve Young. 2010. Phrase-based statistical language generation using graphical models and active learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1552–1561.

François Mairesse and Marilyn A. Walker. 2011. Controlling user perceptions of linguistic style: Trainable generation of personality traits. *Computational Linguistics*, 37(3):455–488.

Margaret Mitchell, Dan Bohus, and Ece Kamar. 2014. Crowdsourcing language generation templates for dialogue systems. In *Proceedings of the INLG and SIGDIAL 2014 Joint Session*, pages 172–180.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Blaise Thomson, Milica Gašić, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2015. Multi-domain dialog state tracking using recurrent neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 794–799.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2017. The E2E dataset: New challenges for end-to-end generation. In *Proc. of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 201–206.

Muhammed Yavuz Nuzumlalı and Arzucan Özgür. 2014. Analyzing stemming approaches for Turkish multi-document summarization. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 702–706.

Kemal Oflazer and Murat Saraclar. 2018. *Turkish Natural Language Processing*, 1st. edition. Springer.

Alice H. Oh and Alexander I. Rudnicky. 2000. Stochastic language generation for spoken dialogue systems. In *Proceedings of the 2000 ANLP/NAACL Workshop on Conversational Systems - Volume 3*, page 27–32, USA. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Baolin Peng, Chenguang Zhu, Chunyuan Li, Xiujun Li, Jinchao Li, Michael Zeng, and Jianfeng Gao. 2020. Few-shot natural language generation for task-oriented dialog.

Rohit Prabhavalkar, Kanishka Rao, Tara N. Sainath, Bo Li, Leif Johnson, and Navdeep Jaitly. 2017. A comparison of sequence-to-sequence models for speech recognition. In *Proceedings of the 18th International Speech Communication Association (Interspeech)*.

Antoine Raux, Brian Langner, Dan Bohus, Alan W Black, and Maxine Eskenazi. 2005. Let's go public! taking a spoken dialog system to the real world. In *Proceedings of Interspeech 2005*.

Christian Raymond and Giuseppe Riccardi. 2007. Generative and discriminative algorithms for spoken language understanding. In *Proceedings of the Eighth Annual Conference of the International Speech Communication Association*, pages 1605–1608.

Hang Ren, Weiqun Xu, Yan Zhang, and Yonghong Yan. 2013. Dialog state tracking using conditional random fields. In *Proceedings of the SIGDIAL 2013 Conference*, pages 457–461.

Matthieu Riou, Bassam Jabaian, Stéphane Huet, and Fabrice Lefèvre. 2019. Reinforcement adaptation of an attention-based neural natural language generator for spoken dialogue systems. *Dialogue & Discourse*, 10:1–19.

R. Sarikaya, P. A. Crook, A. Marin, M. Jeong, J. P. Robichaud, A. Celikyilmaz, Y. B. Kim, A. Rochette, O. Z. Khan, X. Liu, D. Boies, T. Anastasakos, Z. Feizollahi, N. Ramesh, H. Suzuki, R. Holenstein, E. Krawczyk, and V. Radostev. 2016. An overview of end-to-end language understanding and dialog management for personal digital assistants. In *2016 IEEE Spoken Language Technology Workshop (SLT)*, pages 391–397.

Lei Sha, Lili Mou, Tianyu Liu, Pascal Poupart, Sujian Li, Baobao Chang, and Zhifang Sui. 2018. Order-planning neural text generation from structured data. In *Proc. of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18)*, pages 5414–5421.

Pararth Shah, Dilek Hakkani-Tur, and Larry Heck. 2016. Interactive reinforcement learning for task-oriented dialogue management. In *Workshop on Deep Learning for Action and Interaction (NIPS)*.

Richard Socher, Cliff Chiung-Yu Lin, Andrew Y. Ng, and Christopher D. Manning. 2011. Parsing natural scenes and natural language with recursive neural networks. In *Proceedings of the 28th International Conference on International Conference on*

Machine Learning, page 129–136, Madison, WI, USA. Omnipress.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, page 3104–3112, Cambridge, MA, USA. MIT Press.

G. Tur, L. Deng, D. Hakkani-Tür, and X. He. 2012. Towards deeper understanding: Deep convex networks for semantic utterance classification. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5045–5048.

Zhuoran Wang and Oliver Lemon. 2013. A simple and generic belief tracking mechanism for the dialog state tracking challenge: On the believability of observed information. In *Proceedings of the SIGDIAL 2013 Conference*, pages 423–432.

Gellert Weisz, Pawel Budzianowski, Pei-Hao Su, and Milica Gasic. 2018. Sample efficient deep reinforcement learning for dialogue systems with large action spaces. *IEEE/ACM Transactions Audio, Speech and Language Processing*, 26(11):2083–2097.

Tsung-Hsien Wen, Milica Gašić, Dongho Kim, Nikola Mrkšić, Pei-Hao Su, David Vandyke, and Steve Young. 2015a. Stochastic language generation in dialogue using recurrent neural networks with convolutional sentence reranking. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 275–284.

Tsung-Hsien Wen, Milica Gašić, Nikola Mrkšić, Pei-Hao Su, David Vandyke, and Steve Young. 2015b. Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. In *Proc. of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1711–1721.

Tsung-Hsien Wen, David Vandyke, Nikola Mrksic, Milica Gasic, Lina Maria Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve J. 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*, pages 438–449. Association for Computational Linguistics.

Chien-Sheng Wu, Steven Hoi, Richard Socher, and Caiming Xiong. 2020. Tod-bert: Pre-trained natural language understanding for task-oriented dialogues.

Zhao Yan, Nan Duan, Peng Chen, Ming Zhou, Jianshe Zhou, and Zhoujun Li. 2017. Building task-oriented dialogue systems for online shopping. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 4618–4625. AAAI Press.

Kaisheng Yao, Geoffrey Zweig, Mei-Yuh Hwang, Yangyang Shi, and Dong Yu. 2013. Recurrent neural networks for language understanding. In *Proceedings of Interspeech*, pages 2524–2528.

Tianyu Zhao and Tatsuya Kawahara. 2019. Joint dialog act segmentation and recognition in human conversations using attention to dialog context. *Computer Speech & Language*, 57:108 – 127.

Yin Jiang Zhao, Yan Ling Li, and Min Lin. 2019. A review of the research on dialogue management of task-oriented systems. *Journal of Physics: Conference Series*, 1267:012025.

Chenguang Zhu, Michael Zeng, and Xuedong Huang. 2019. Multi-task learning for natural language generation in task-oriented dialogue. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1261–1266.