A Dialog System with Human-to-Human Conversation Example

Lasguido Nio, Sakriani Sakti, Graham Neubig, Tomoki Toda, Satoshi Nakamura (NAIST)

1 Introduction

Example-based dialog modeling (EBDM) is data-driven approach for deploying dialog systems [1, 2]. Some studies propose constructing dialog examples from available human-to human conversation log databases [3, 4] or movie scripts [5]. However, these works did not filter any uncorrelated consecutive lines in the movie data. As the authors state, this causes failures and diminishes the ability to maintain a consistent conversation.

In this paper, we summarize our work on a dialog agent that utilizes human-to-human conversation examples from movies and Twitter to reduce the time requirement for database design and collection, and allow the agent to interact with the user in as natural as fashion as possible. Next, we investigate various data-driven approaches to dialog management, including two EBDM techniques (syntactic-semantic similarity retrieval and TF-IDF based cosine similarity retrieval) and using statistical machine translation (SMT) to learn a conversational mapping between user-input and system-output dialog pairs.

2 Dialog Data Collection

We collect a dialog corpus of dialog pairs from movie scripts and Twitter data.

2.1 Preprocessing

We remove unnecessary explanatory information from the movie script, and information about personal identity, hash tags, and URLs from the Twitter data. Both data sets are labeled with parts of speech (POS) and named entities (NE). Unifying both data sources into one dialog corpus, we define two basic types of information about each dialog: actor and utterances. The utterances are the actual content of each dialog turn in the movie scripts or tweets. The actor refers to the character name in the movies, or the name of the Twitter user that posted each tweet. This actor and utterance information will be utilized to construct the dialog corpus.

2.2 Turn Extraction and Filtering

To ensure that the dialog example database contains only query-response pairs, we use a simple and intuitive method for selection of the dialog data: trigram turn sequences, or *tri-turns*. A tri-turn is defined as three turns in a conversation between two actors X and Y that has the pattern X-Y-X. Within a tri-turn, the first and last dialog turn are performed by the same actor and the second dialog turn is performed by the other actor. Next the query-response pairs are made by separating the tri-turn pattern X-Y-X into two pairs, X-Y and Y-X.

However we found that even after the tri-turn filtering, noisy cases that contain uncorrelated turns still exist. This happens because the speakers are not actually speaking to each-other. To address this problem, we perform further filtering using the semantic similarity (similar to the approach introduced in [6]). This method ensure a semantic relationship between each dialog turn in the dialog pair data by computing the similarity between WordNet[1] synsets in each dialog turn (see Eq. (1)).

$$sem_{sim}(S_1, S_2) = \frac{2 \times |S_{syn1} \cap S_{syn2}|}{|S_{syn1}| + |S_{syn2}|} \tag{1}$$

S_{syn1} and S_{syn2} respectively are groups of WordNet synsets for each word in the sentences S_1 and S_2, linked by a complex network of lexical relations. The similarity of sentence pair X-Y (S_1 = X; S_2 = Y) can be obtained by calculating the relations between S_{syn1} and S_{syn2}. $|S_{syn1} \cap S_{syn2}|$ is the number of co-occurring WordNet synsets and $|S_{syn1}| + |S_{syn2}|$ is a total number of effective WordNet synsets. Dialog pairs with high similarity are then extracted and included into the database.

3 Dialog Management System

3.1 Syntactic-Semantic Similarity Retrieval

In this approach (sssr), a proper system response is retrieved by measuring both semantic and syntactic relations. These two measures are combined using linear interpolation as shown in Eq. (2). This value is calculated over the user input sentence (S_1) and every input examples on database (S_2). These values are calculated using Eq. (1) as a semantic factor and cosine similarity (Eq. (3)) over part-of-speech (POS) tag vectors as a syntactic factor.

$$sim(S_1, S_2) = \alpha[sem_{sim}(S_1, S_2)] + (1 - \alpha)[cos_{sim}(S_1, S_2)] \tag{2}$$

where

$$cos_{sim}(S_1, S_2) = \frac{S_1 \cdot S_2}{\| S_1 \| \| S_2 \|}. \tag{3}$$

3.2 TF-IDF-based Cosine Similarity Retrieval

Cosine similarity over the term vector (csm) as described in Eq. (3) is used to retrieve a proper system response. To increase the emphasis on important words, additional TF-IDF weighting (Eq. (4)) is performed to construct the term vector [7].

$$TFIDF(t, T) = F_{t,T} \log \left(\frac{|T|}{DF_t} \right) \tag{4}$$

We define $F_{t,T}$ as term frequency t in a sentence T, and DF_t as the total number of sentences in the query-response pairs that contain term t.

[1] http://wordnet.princeton.edu/

3.3 SMT-based Generation

The dialog-pair data is treated as a parallel corpus for training an SMT system [8]. Given the trained SMT system, the user dialog is treated as an input and “translated” into the system response. The system response is chosen to be system output S of maximal probability given the user input T

$$\hat{S} = \arg\max_{S} P(S \mid T). \tag{5}$$

4 Experimental Evaluation

After extracting all the dialog-pairs, we randomly separate our query-response pairs from Twitter and movie conversation dialog, as a test set (the query-response pairs are denoted as $\langle Q_{test}, R_{test}\rangle$), and as dialog examples for EBDM, or training data for SMT (the dialog-pairs here are denoted as $\langle Q_{train}, R_{train}\rangle$).

Given a query from the test set (Q_{test}), EBDM will search the closest query examples using syntactic-semantic similarity retrieval: sim(Q_{test}, Q_{train}) or TF-IDF based cosine similarity retrieval: cos(Q_{test}, Q_{train}), and output a response of R_{output}.

To evaluate a number of valid system response, we assess each output response with semantic, syntactic, and manual evaluation criteria. We utilize TF-IDF-based cosine similarity as a semantic criterion, syntactic-semantic similarity as a syntactic criterion, and subjective evaluation as a manual opinion score. For objective evaluation, the R_{output} are evaluated by computing similarity with R_{test}: sim(R_{output}, R_{test}) and cos(R_{output}, R_{test}). During subjective evaluation, users evaluate the naturalness of dialog-pair Q_{test} and R_{output} by giving them a score between 1-5.

		sssr	csm	smt	comb
TF-IDF based Cosine Similarity					
		sssr	csm	smt	comb
Movie data	no-filter	55.86%	52.20%	38.29%	**60.53%**
	triturn	58.27%	53.86%	37.85%	**62.05%**
	semantic	62.33%	65.58%	48.43%	**69.64%**
	triturn+semantic	61.85%	66.52%	49.26%	**70.55%**
Twitter data	no-filter	51.73%	52.65%	30.44%	**55.68%**
	semantic	55.74%	66.12%	49.95%	**71.44%**
Syntactic-Semantic base Similarity					
		sssr	csm	smt	comb
Movie data	no-filter	55.93%	54.24%	43.11%	**64.49%**
	triturn	57.95%	55.44%	41.77%	**64.80%**
	semantic	72.18%	75.44%	62.35%	**80.03%**
	triturn+semantic	71.36%	76.81%	62.58%	**80.78%**
Twitter data	no-filter	51.52%	52.47%	36.06%	**58.18%**
	semantic	82.95%	83.64%	73.86%	**85.73%**

Table 1 Objective evaluation result.

To demonstrate the effect of semantic similarity and tri-turn filtering in our data, we compare our system performance with and without the tri-turn and semantic filtering. Table 1 shows the result of objective evaluation given various filter and response retrieval techniques. The tri-turn and semantic filtering manage to increase the evaluation score. Comparing the csm and smt approaches, csm always give a better performance than smt. Analyzing the data in more detail, we found that csm is better in handling when dialogues close to Q_{test} exists in Q_{train}, while smt can provide a better output when there is no dialogs in Q_{train} similar with Q_{test}. Combining both approaches (comb) in which the system uses EBDM if the similarity between user input and dialog examples exceeds given threshold, and responds with SMT output otherwise, could overcome the shortcomings of each approach.

		sssr	csm	smt	comb
Movie data	no-filter	2.5	2.9	2.2	2.3
	triturn+semantic	2.7	**3.5**	2.9	3.0
Twitter data	no-filter	2.7	2.4	2.1	2.1
	triturn+semantic	2.8	**3.1**	2.3	2.7

Table 2 Subjective evaluation result.

The subjective evaluation result (see Table 2) also demonstrate slightly higher scores on filtered data. This shows that the tri-turn and semantic similarity filtering methods manage to increase the naturalness of the response. Furthermore, because response sentences from the smt system are sometimes incomprehensible, more people prefer the csm responses. This also affected the comb performance, where the csm response was slightly better than the comb approach.

5 Conclusion

We investigated several approaches to build a data-driven chat-oriented dialog systems. The proposed tri-turn extraction and semantic similarity filtering are able to extract dialog pair examples from multi-speaker dialog of raw movie scripts and Twitter data. Experimental results also reveal that that the tri-turn and semantic filtering improve the objective evaluation metrics. We also introduced a system that combines example-based and SMT-based approaches to take advantage of the characteristics of both approaches.

参考文献

[1] C. Lee, S. Lee, S. Jung, K. Kim, D. Lee, and G. Lee, “Correlation-based query relaxation for example-based dialog modeling,” in *Proc. of ASRU*, Merano, Italy, 2009, pp. 474–478.

[2] K. Kim, C. Lee, D. Lee, J. Choi, S. Jung, and G. Lee, “Modeling confirmations for example-based dialog management,” in *Proc. of SLT*, Berkeley, California, USA, 2010, pp. 324–329.

[3] H. Murao, N. Kawaguchi., S. Matsubara, Y. Yamaguchi, and Y. Inagaki, “Example-based spoken dialogue system using WOZ system log,” in *Proc. of SIGDIAL*, Sapporo, Japan, 2003, pp. 140–148.

[4] F. Bessho, T. Harada, and Y. Kuniyoshi, “Dialog system using real-time crowdsourcing and twitter large-scale corpus,” in *Proc. of SIGDIAL*, Seoul, South Korea, 2012, pp. 227–231.

[5] R. E. Banchs and H. Li, “IRIS: a chat-oriented dialogue system based on the vector space model,” in *ACL (System Demonstrations)*, 2012, pp. 37–42.

[6] D. Liu, Z. Liu, and Q. Dong, “A dependency grammar and wordnet based sentence similarity measure,” *Journal of Computational Information Systems*, vol. 8, no. 3, pp. 1027–1035, 2012.

[7] G. Salton, A. Wong, and C. S. Yang, “A vector space model for automatic indexing,” *Commun. ACM*, vol. 18, no. 11, pp. 613–620, Nov. 1975.

[8] A. Ritter, C. Cherry, and W. B. Dolan, “Data-driven response generation in social media,” in *Proc. of EMNLP*, Edinburgh, Scotland, UK, 2011, pp. 583–593.