Combination of Example-based and SMT-based Approaches in a Chat-oriented Dialog System

Lasguido Nio, Sakriani Sakti, Graham Neubig, Tomoki Toda, Satoshi Nakamura
Augmented Human Communication Laboratory
Nara Institute of Science and Technology
8916-5 Takayama-cho, Ikoma, Nara 630-0192, Japan

***Abstract*—This paper compares and contrasts example-based and statistical machine translation (SMT) approaches for building a chat-oriented dialog system, and investigates a combined method that addresses the advantages and disadvantages of both approaches. In particular, we compare two example-based (EBDM) techniques: syntactic-semantic similarity retrieval and TF-IDF based cosine similarity retrieval, as well as response generation using phrase-based SMT. Experiments utilize both movie and Twitter scripts, and performance was evaluated based on objective metrics (syntactic-semantic similarity and TF-IDF based cosine similarity evaluation metrics) and human subjective evaluation. Experimental results shows that the combined system provide the best performance. It may indicate that by combining both approaches, we could overcome the shortcomings of each approach.**

I. Introduction

Dialog systems can be broadly divided into two main genres: *goal-oriented* (e.g. ATIS [1], DARPA communicator [2]) and *non-goal-oriented* (e.g. Eliza [3], Alice [4]). Dialog systems can also be described by the amount of human intervention used in their construction, ranging from entirely hand-made to completely data-driven. In this paper, we will focus on data-driven approaches to chat-oriented dialog, a type of non-goal-oriented dialog that aims to provide natural conversation between human and machine without any specific target goals.

Example-based dialog modeling (EBDM) is one of data-driven methods for deploying dialog systems. Instead of using complex rule-based dialog management, EBDM uses dialog examples that are semantically indexed to a database, and proper responses for user input are generated based on these dialog examples. Consequently, to achieve good coverage on various types of natural conversation recording of a large data set of real human-to-human conversation is necessary, which is tedious and time consuming. Common solutions use handmade scripted dialog scenarios which may result in unnatural conversations, as users may respond differently than they would in a real-world situation. Some studies also propose constructing dialog examples from available log databases, such conversation between human subjects and the Wizard of OZ (WOZ) system [5], or human-to-human conversation in Twitter [6].

However, covering all possible patterns that may exist in real human-to-human conversation is still difficult. Therefore, there is always a risk with EBDM technique that the system may not be able to find similar examples to determine the next system output. Currently, most EBDM systems rely on either canned responses by providing error messages [7] or templates for generation which may result in a completely incomprehensible response [8]. On the other hand, [9] have proposed using SMT as an approach for response generation. However, utilizing SMT directly within dialog modeling has not been investigated as of yet. By learning mappings between user-input and system-output pairs in conversation, SMT may have the capability to produce a related response, even if the user input is not similar with training examples.

In this paper, we compare and contrast response generation methods and data sources for data-driven response generation, and propose a novel method to combine both EBDM and SMT-based approaches. To reduce and simplify the work of collecting the real human-to-human conversations, we investigate the use of movie scripts and Twitter. They represent examples of human-to-human conversation that often contain interesting chat conversations, which are easily accessible.

The goal of our work is creating an agent that can interact with the user in as natural a fashion as possible. In this paper we focus on two main challenging issues including (1) filtering and constructing a dialog example database from the drama conversations, and (2) retrieving a proper system response by finding the best dialog example based on the current user query. This paper makes several contributions to attempt to address these problems and inform design decisions for future work on data-driven chat-oriented dialog system design:

- We perform contrastive experiments using two types of easily obtainable data: human-to-human conversation examples from movies and Twitter data. The aim is to gain insights in how to build a conversational agent that can interact with users in as natural a way as possible, while reducing the time requirement for database design and collection.
- We propose a *tri-turn* unit for dialog extraction and semantic similarity analysis from raw movie/drama script files to help ensure that extracted content forms appropriate dialog examples (user input - system output).
- We compare various data-driven approaches to dialog management, including two EBDM techniques (syntactic-semantic similarity retrieval and TF-IDF based cosine similarity retrieval) and using phrase-based statistical machine translation (SMT) to learn a conversational mapping between user-input and system-output dialog pairs.

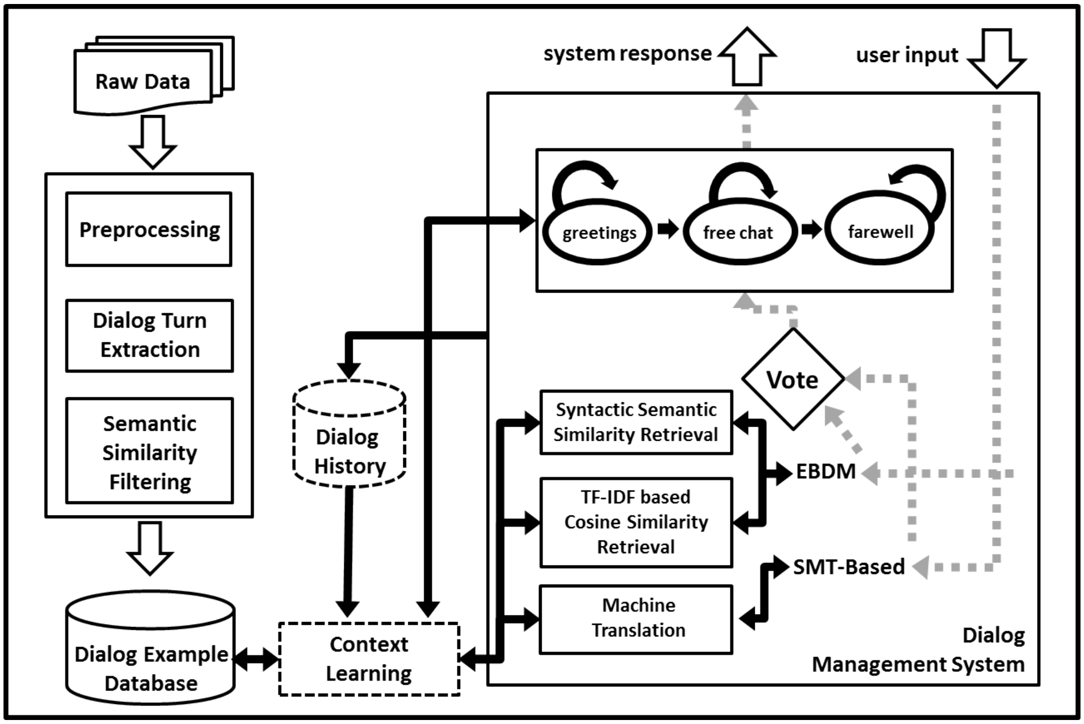


Fig. 1. Overview of the proposed dialog system.

- We propose a simple, but effective way to perform system combination of example-based and SMT-based techniques into one dialog management framework. Experimental results demonstrate that our combined system shows promise for overcoming the shortcomings of each approach.

II. Dialog System Overview

The overview of our proposed dialog system is shown in Figure 1, which mainly focuses on two challenging issues: (1) filtering and constructing a dialog example database from raw movie/Twitter scripts (see Section III), and (2) dialog management for retrieving a proper system response (see Section IV). Note that, as current focus here is to investigate the optimal technique for retrieving a proper system response based on the current user query, the utilizing of user history and dialog context will not be discussed in this paper.

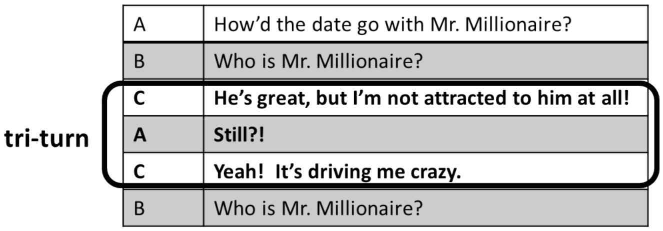
III. Dialog-Pair Extraction

A. Preprocessing

As the movie scripts and Twitter data used in this work contain very different types of text, preprocessing is done in different fashions. For the movie scripts, preprocessing is applied to transform and clean the movie script into a conversation script and further remove unnecessary explanatory information about the movie scenes. For the Twitter data, preprocessing removes information about person identity, hash tags, and URLs. Next, for both data sets all the words in the sentences are labeled with parts of speech (POS) and named entities (NE). Finally, to ensure the integrity of the Twitter data, language filtering[1] and non-standard word (NSW) normalization [10] is also performed.

B. Dialog Turn Extraction

The next task is to construct proper dialog-pair examples (user input - system output) from conversation dialog in movie scripts. The challenge here is that the conversation dialog usually involves many actors and does not necessarily contain only two-way *dialog-pair* sentences. Consequently, the conversation dialog does not have a clear distinction of which utterances are responses to a particular utterance. For example, in Figure 2 there are consecutive utterances where the second utterance is clearly not a response to the first. To ensure that the dialog example database contains only input-response pairs, we propose a simple and intuitive method for selection of the dialog data: trigram turn sequences, or *tri-turns*.



A	How'd the date go with Mr. Millionaire?
B	Who is Mr. Millionaire?
C	**He's great, but I'm not attracted to him at all!**
A	**Still?!**
C	**Yeah! It's driving me crazy.**
B	Who is Mr. Millionaire?

Fig. 2. Example of dialog and the tri-turn in a conversation with multiple actors.

A tri-turn is defined as three turns in a conversation between two actors A and B that has the pattern A-B-A. In

[1] search.cpan.org/~ambs/Lingua-Identify-0.51/

other words, 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.

We found that when we observed this pattern, in the great majority of the cases this indicated that the first and second utterances, as well as the second and third utterances, formed a proper input-response pair as shown on upper side of Table I. However, noisy cases which contain un-correlated turns still exist (see bottom side of Table I). To address this problem, we further process using the semantic similarity measure described in the following section.

Actor	**Correlated tri-turn**
A	I'm gonna miss you.
B	I'll miss you, too, my friend.
A	I love you.
Actor	**Un-correlated tri-turn**
A	I'm sorry, no one by that name.
B	Hey! You can't go up there!
A	We've got an intruder in the elevator!

TABLE I. EXAMPLE OF AN A-B-A TRI-TURN WITH TWO ACTORS A AND B.

C. Semantic Similarity

Semantic similarity [11], shown in Equation (1), is used to ensure a strong semantic relationship between each dialog turn in the dialog-pair data, by computing the similarity between WordNet[2] synsets in each dialog turn. The dialog pairs with high similarity are then extracted and included into database.

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

IV. DIALOG MANAGEMENT

Dialog management utilizes dialog templates within three states: the *greeting* state, *free-chat* state, and *farewell* state. For every user input, it generates responses according to following strategies.

A. Example-based Dialog Modeling

1) Syntactic-Semantic Similarity Retrieval: The syntactic-semantic similarity retrieval approach scores the response dialog according to WordNet semantic similarity and part-of-speech (POS) tags cosine similarity, in order to measure both semantic and syntactic relations. These two measures are combined using linear interpolation as shown in Equation (2). This value is calculated with respect to real user input (S_1) and the existing input example (S_2) in dialog-pair database according to the following equation.

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

where

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

In this work, we assumed that the semantic factor is more important than syntactic factor, so we set the interpolation coefficient α to be 0.7.

2) TF-IDF based Cosine Similarity Retrieval: Cosine similarity as described in Equation 3 is used to retrieve a proper system response. By constructing the term vector, additional TF-IDF weighting (Equation 4) is used to increase the emphasis on important words [12].

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

We define $F_{t,T}$ as a term frequency 't' in a dialog turn 'T', and DF_t as a total dialog turn that contains term 't'.

B. SMT-based

The second approach we tested was based on statistical machine translation (SMT) [9]. With this approach, the dialog-pair data is treated as a parallel corpus for training an SMT system. 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

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

V. EXPERIMENTS AND EVALUATION

A. Experimental Setting

In this paper, the movie script dialog is collected through Friends TV show scripts[3], The Internet Movie Script Database[4], and The Daily Script[5]. This resulted in a total of 1,786 movie scripts. After performing dialog turn extraction and semantic similarity filtering, the total number of query-response pairs is 86,719. The Twitter data was collected through the Twitter API[6], resulting in a total of 1,076,447 query-response pairs. After performing language filtering and semantic similarity filtering, the total number of query-response pairs was reduced to 7,048. Figure 3 shows the resulting improvements in the evaluation metrics (cosine similarity with TF-IDF weighting) after conducting filtering process. The filtering not only helps to improve the performance but also reduce significantly the response time by reducing dialog examples in training set. Next, we randomly selected 500 and 1000 dialogs from Twitter and movie conversation dialog, respectvely, as test set (The query-response pairs here are denoted as *Qtest* - *Rtest*). Then, the remaining of dialogs

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

[3] http://ufwebsite.tripod.com/scripts/scripts.htm

[4] http://imsdb.com/

[5] http://dailyscript.com/

[6] http://dev.twitter.com

will be used as dialog examples for EBDM, and training data for SMT (The query-response pairs here are denoted as *Qtrain* - *Rtrain*).

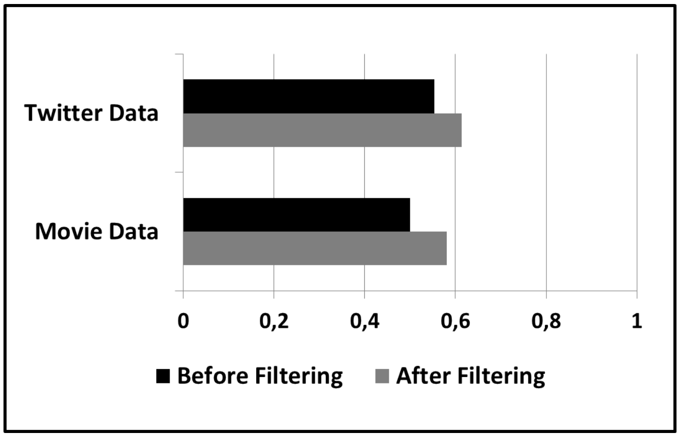


Fig. 3. TF-IDF based cosine similarity evaluation metrics improvement after performing semantic similarity filtering.

The natural language processing tools and Wordnet synsets used were provided by NLTK toolkit[7], and the example-based TF-IDF based cosine similarity retrieval was performed using Apache Lucene[8]. For the SMT approach, Moses[9] was used to build the translation model and perform translation for the dialog system. Here, four-gram language models built with the Kneser-Ney smoothing and the lexicalized distortion model were used.

Input	Shall we eat at my house?
Response 1	Sorry, I eat already.
Response 2	Yes, sure.
Response 3	Of course, will you cook?
Response 4	Great! But, where is your house?

TABLE II. EXAMPLE OF DIFFERENT SYSTEM RESPONSES.

Given a query from the test set (*Qtest*), the EBDM will search the closest query examples using syntactic-semantic similarity retrieval: sim(*Qtest*, *Qtrain*) or TF-IDF based cosine similarity retrieval: cos(*Qtest*, *Qtrain*), and output a response of *R_output*. However, as the system response from a single user query may vary (see the example in Table II), it is not trivial to evaluate the system performance. Here, we attempt to investigate the performance using various objective metrics and subjective evaluation. For objective evaluation, the *R_output* are evaluated by computing similarity with *Rtest*: sim(*R_output*, *Rtest*) and cos(*R_output*, *Rtest*). Also when performing subjective evaluation in user-system interaction, *Qtest* are given, and the users are evalute the naturalness of *R_output* in comparison with *Rtest*.

Approach	Abbreviation
EBDM	
- Syntactic-Semantic Similarity Retrieval	sssr
- TF-IDF based Cosine Similarity Retrieval	csm
SMT	smt
Combination EBDM and SMT	comb

TABLE III. VARIOUS APPROACH ON THE SYSTEM RESPONSE GENERATION.

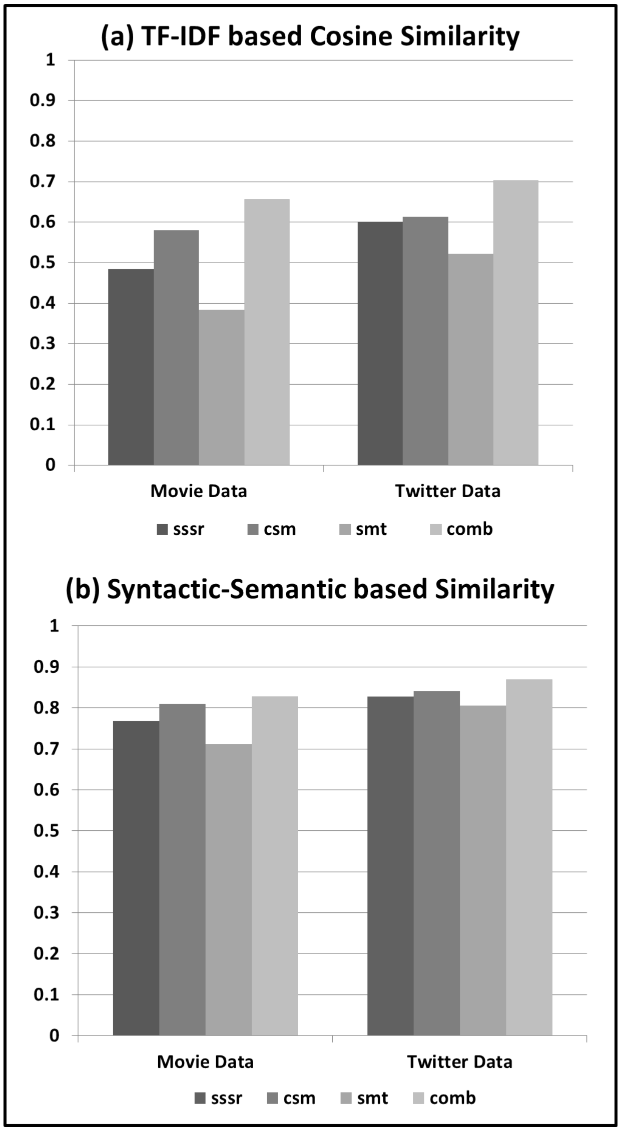


Fig. 4. Objective evaluation result on the movie and Twitter data by various data-driven approaches.

B. Objective Evaluation

Objective evaluation presented in Figure 4 is performed using TF-IDF based cosine similarity and syntactic-semantic

[7] http://nltk.org

[8] http://lucene.apache.org/

[9] http://statmt.org/moses/

similarity evaluation metrics. The results reveal that, within EBDM approach, TF-IDF based cosine similarity retrieval (denoted as “csm”) always gives a better response score than syntactic-semantic similarity retrieval (denoted as “sssr”). Comparing the best EBDM approach againsts the SMT approach (denoted as “smt”), “csm” always give a better performance than “smt”. Analyzing the data in more detail, we found that “csm” is better in handling when the closest dialogs with *Qtest* exists in the *Qtrain*, while “smt” can provide a better *R_output* when there is no dialogs in *Qtrain* similar with *Qtest*. Combining both approaches (denoted as “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. The results reveal that the best system is provided by the combined system. The optimum score shown here is achieved by 0.4 and 0.6 for movie and Twitter data respectively.

C. Subjective Evaluation

In the subjective evaluation, 10 human annotators were asked to give a naturalness score between 1-3 of the system output, with higher scores indicating that the system is giving a natural and relevant system response to the user input. Each person assesses 10 randomly selected user inputs. The results of this evaluation are shown in Figure 5. We also a dummy system as a baseline which output a response by simply repeating the user input, i.e. user-input: ”How are you?”, then the system’s output is also: ”How are you?”. For greeting conversations, this simple approach may work. But, for the other cases, the system may result in a completely incomprehensible response.

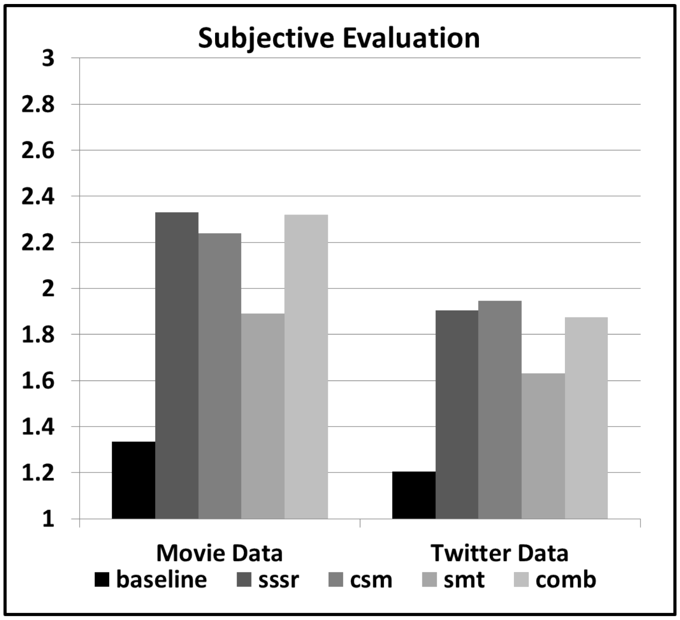


Fig. 5. Subjective evaluation result on the movie and Twitter data by various data-driven approaches.

In contrast with objective evaluation, the results show that both EBDM approaches (sssr and csm) always outperform SMT approach (smt). This may indicate that the smt responses consists of several matching phrases with the reference, but have not yet reached the naturalness of real human responses. This factor seems to affect the system combination as well, where it reduced the score slightly compared with the EBDM approach. Nevertheless, the SMT approach still performed significantly better than the baseline system.

VI. Related Work

There have been a number of related works into EBDM and SMT-based approaches for response generation in data-driven chat [13], [14], [15]. Work by [7] proposes a generic dialog modeling framework for a multi-domain dialog systems to simultaneously manage goal-oriented and chat dialogs for information access and entertainment. However, the chat-oriented dialog only include small talk which is limited to 10 topics of daily conversation. Furthermore, if the system cannot find similar examples to determine the next system action, it simply defines “No Example” output error and provides an in-coverage example of what the user could say at the current dialog state. Finally, [16] introduce IRIS (Informal Response Interactive System), a chat oriented dialog system using movie scripts that is based on vector space model. However, the system did not filter any uncorrelated consecutive scripts in the movie data, and as the authors state this causes failures and diminishes the ability to maintain a consistent conversation.

Despite the relatively large interest in data-driven approaches for chat or response generation, there is surprisingly little work comparing and contrasting approaches or data sources. In this work, we attempt make an empirical evaluation that will contrast these approaches and provide a reference for future development in the area. In addition, to the best of our knowledge, our method to combine example-based and SMT-based response generation in dialog modeling is also different from previous work.

VII. Conclusion

In this work, we investigated several approaches to building data-driven chat-oriented dialog systems. We found that the example-based approach is very good in handling the queries which are similar to the examples in the database, but achieves poor performance in handling the queries which are far different from existing examples. On the other hand, SMT-based systems showed the opposite tendency. We also introduced a system that combines example-based and SMT-based approaches to take advantage of the characteristics of both approaches.

As future work, investigating ways to improve the naturalness and cohesion of responses generated by the SMT approach may be necessary. Adding a learning process that considers the context of the conversation could also lead to further improvements. This would allow the system to both remember the context of the conversation and expand the example database. Therefore, including the user history and dialog context to the dialog management system is a promising future direction. Combining other approaches in the chat-oriented dialog system could also demonstrate interesting results.

VIII. Acknowledgment

Part of this work was supported by the collaborative research with I2R[10].

References

[1] E. Seneff, L. Hirschman, and V. Zue, "Interactive problem solving and dialogue in the ATIS domain," in *Proc. of the Fourth DARPA Speech and Natural Language Workshop*, 1991, pp. 354–359.

[2] M. Walker, J. Aberdeen, J. Boland, E. Bratt, J. Garofolo, L. Hirschman, A. Le, S. Lee, S. Narayanan, K. Papineni, B. Pellom, J. Polifroni, A. Potamianos, P. Prabhu, A. Rudnicky, G. Sanders, S. Seneff, D. Stallard, and S. Whittaker, "DARPA communicator dialog travel planning systems: the June 2000 data collection," in *Proc. of EUROSPEECH*, 2000, pp. 1371–1374.

[3] J. Weizenbaum, "Eliza computer program for the study of natural language communication between man and machine," *Commun. ACM*, vol. 9, no. 1, pp. 36–45, Jan. 1966. [Online]. Available: http://doi.acm.org/10.1145/365153.365168

[4] R. Wallace, *Be Your Own Botmaster*. California, USA: A.L.I.C.E A.I. Foundation, 2003.

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

[6] 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.

[7] C. Lee, S. Jung, S. Kim, and G. G. Lee, "Example-based dialog modeling for practical multi-domain dialog system," *Speech Commun.*, vol. 51, no. 5, pp. 466–484, May 2009. [Online]. Available: http://dx.doi.org/10.1016/j.specom.2009.01.008

[8] N. Chambers and J. Allen, "Stochastic language generation in a dialogue system: Toward a domain independent generator." in *Proc. of SIGDIAL*, Cambridge, Massachusetts, USA, 2004, pp. 9–18.

[9] A. Ritter, C. Cherry, and W. B. Dolan, "Data-driven response generation in social media," in *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*. Edinburgh, Scotland, UK.: Association for Computational Linguistics, July 2011, pp. 583–593. [Online]. Available: http://www.aclweb.org/anthology/D11-1054

[10] R. Sproat, A. W. Black, S. F. Chen, S. Kumar, M. Ostendorf, and C. Richards, "Normalization of non-standard words." *Computer Speech and Language*, vol. 15, no. 3, pp. 287–333, 2001. [Online]. Available: http://dblp.uni-trier.de/db/journals/csl/csl15.html

[11] 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.

[12] 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. [Online]. Available: http://doi.acm.org/10.1145/361219.361220

[13] S. Jung, C. Lee, and G. Lee, "Dialog studio: An example based spoken dialog system development workbench," in *Proc. of the Dialogs on dialog: Multidisciplinary Evaluation of Advanced Speech-based Interactive Systems. Interspeech2006-ICSLP satellite workshop,*, Pittsburg, USA, 2006.

[14] 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.

[15] 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.

[16] 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.

[10] http://www.i2r.a-star.edu.sg/