SAWDUST: a Semi-Automated Wizard Dialogue Utterance Selection Tool for domain-independent large-domain dialogue

Sudeep Gandhe **David Traum**
University of Southern California, Institute for Creative Technologies
srgandhe@gmail.com, traum@ict.usc.edu

Abstract

We present a tool that allows human wizards to select appropriate response utterances for a given dialogue context from a set of utterances observed in a dialogue corpus. Such a tool can be used in Wizard-of-Oz studies and for collecting data which can be used for training and/or evaluating automatic dialogue models. We also propose to incorporate such automatic dialogue models back into the tool as an aid in selecting utterances from a large dialogue corpus. The tool allows a user to rank candidate utterances for selection according to these automatic models.

1 Motivation

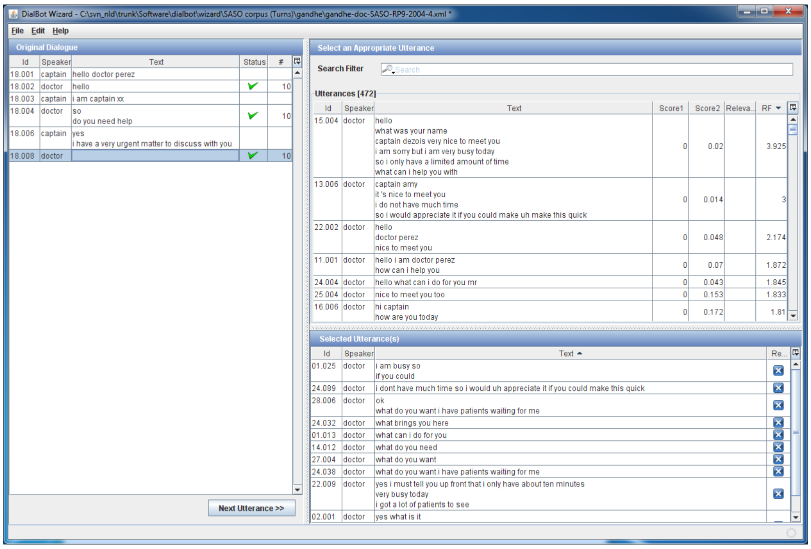
Dialogue corpora play an increasingly important role as a resource for dialogue system creation. In addition to its traditional roles, such as training language models for speech recognition and natural language understanding, the dialogue corpora can be directly used for the *selection approach* to response formation (Gandhe and Traum, 2010). In the *selection* approach, the response is formulated by simply picking the appropriate utterance from a set of previously observed utterances. This approach is used in many wizard of oz systems, where the wizard presses a button to select an utterance, as well as in many automated dialogue systems (Leuski et al., 2006; Zukerman and Marom, 2006; Sellberg and Jönsson, 2008)

The resources required for the *selection* approach are a set of utterances to choose from and optionally, a set of pairs of ⟨context, response utterance⟩ to train automatic dialogue models. A wizard can generate such resources by performing two types of tasks. First is the traditional Wizard-of-Oz dialogue collection, where a wizard interacts with a user of the dialogue system. Here the wizard selects an appropriate response utterance for a context that is being updated in a dynamic fashion as the dialogue proceeds (*dynamic context setting*). The second task is geared towards gathering data for training/evaluating automatic dialogue models, where a wizard is required to select appropriate responses (perhaps more than one) for a context which is extracted from a human-human dialogue. The context does not change based on the wizard's choices (*static context setting*).

A wizard tool should help with the challenges presented by these tasks. A challenge for both of these tasks is that if the number of utterances in the corpus is large (e.g., more than the number of buttons that can be placed on a computer screen), it may be very difficult for a wizard to locate appropriate utterances. For the second task of creating human-verified training/evaluation data, tools like NPCEditor (Leuski and Traum, 2010) have been developed which, allow the tagging of a many to many relationships between contexts (approximated simply as input utterance) and responses. In other cases, a corpus of dialogues is used to acquire the set of selectable utterances, in which each context is followed by a single next utterance, and many utterances appear only once. This sparsity of data makes the selection task hard. Moreover, it may be the case that there are many possible continuations of a context or contexts in which an utterance may be appropriate (DeVault et al., 2011).

We address these needs with a semi-automated wizard tool that allows a wizard to engage in dynamic or static context utterance selection, select multiple responses, and use several kinds of search tools to locate promising utterances from a large set that can't all be displayed or remembered. In the next section we describe the tool and how it can be used. Then we describe how this tool was used to create evaluation data in the static context setting.

Proceedings of the SIGDIAL 2014 Conference, pages 251–253,
Philadelphia, U.S.A., 18-20 June 2014. ©2014 Association for Computational Linguistics



DialBot Wizard - C:\svn_nld\trunk\Software\dialbot\wizard\SASO corpus (Turns)\gandhe\gandhe-doc-SASO-RP9-2004-4.xml *

Original Dialogue

Id	Speaker	Text	Status	#
18.001	captain	hello doctor perez		
18.002	doctor	hello	✓	10
18.003	captain	i am captain xx		
18.004	doctor	so do you need help	✓	10
18.006	captain	yes i have a very urgent matter to discuss with you		
18.008	doctor		✓	10

Select an Appropriate Utterance

Search Filter: Search

Utterances [472]

Id	Speaker	Text	Score1	Score2	Releva...	RF
15.004	doctor	hello what was your name captain dezois very nice to meet you i am sorry but i am very busy today so i only have a limited amount of time what can i help you with	0	0.02		3.925
13.006	doctor	captain amy it 's nice to meet you i do not have much time so i would appreciate it if you could make uh make this quick	0	0.014		3
22.002	doctor	hello doctor perez nice to meet you	0	0.048		2.174
11.001	doctor	hello i am doctor perez how can i help you	0	0.07		1.872
24.004	doctor	hello what can i do for you mr	0	0.043		1.845
25.004	doctor	nice to meet you too	0	0.153		1.833
16.006	doctor	hi captain how are you today	0	0.172		1.81

Selected Utterance(s)

Id	Speaker	Text	Re...
01.025	doctor	i am busy so if you could	☒
24.089	doctor	i dont have much time so i would uh appreciate it if you could make this quick	☒
28.006	doctor	ok what do you want i have patients waiting for me	☒
24.032	doctor	what brings you here	☒
01.013	doctor	what can i do for you	☒
14.012	doctor	what do you need	☒
27.004	doctor	what do you want	☒
24.038	doctor	what do you want i have patients waiting for me	☒
22.009	doctor	yes i must tell you up front that i only have about ten minutes very busy today i got a lot of patients to see	☒
02.001	doctor	yes what is it	

Figure 1: A screenshot of the interface for the wizard data collection in static context setting.

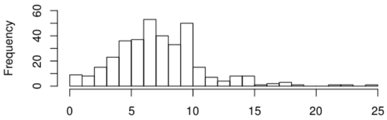


Figure 2: A Histogram for the number of selected appropriate responses.

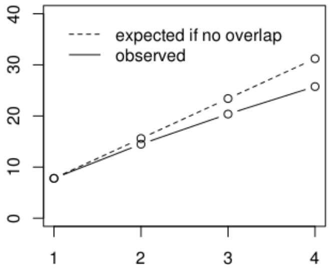


Figure 3: Avg. cardinality of the set for different values of $|R|$.

2 Wizard Tool

Our wizard tool consists of several different views (see figure 1), and is similar in some respects to the IORelator annotation tool (DeVault et al., 2010), but specialized to act as a wizard interface. The first view (left pane) is a dialogue context, that shows the recent history of the dialogue, before the wizard's decision point. The second view (top right pane) shows a list of possible utterances that can be selected from. This view can be ordered in several different ways, as described below. Finally, there is a view of selected utterances (bottom right pane). In the case of dynamic context, the wizard will probably only select one utterance and then a dialogue partner will respond with a new utterance that extends the previous context. In the case of static evaluation, however, used for training and/or evaluating automated selection algorithms, it is often helpful to select multiple utterances if more than one is appropriate.

To help wizards explore the set of all possible utterances, we provide the ability to rank the utterances by various automated scores. Our configuration used in the static context task uses *Score1* as the score calculated using one of the automatic dialogue models, specifically *Nearest Context* model (Gandhe and Traum, 2007) - this model orders candidate utterances from the corpus by the similarity of their previous two utterances to the current dialogue context. *Score2* is surface text similarity, computed as the METEOR score (Lavie and Denkowski, 2009) between the candidate utterance and the actual response utterance present at that location in original human-human dialogue (which is not available to the wizard). Wizards can also search the set of utterances for specific keywords and the third column, *Relevance*, shows the score for the search string entered by the wizards. The last column *RF* stands for relevance feedback and ranks the utterances by similarity to the utterances that have already been chosen by the wizard. This allows wizards to easily find paraphrases of already selected response utterances. Clicking the header of any of these columns will reorder the utterance list by the automated score, by relevance (assuming a search term has been entered) or by relevance feedback (assuming one or more utterances have already been chosen).

3 Evaluation

We evaluated the tool by having four human volunteers (wizards) use it in order to establish an upper baseline for human-level performance in the static context evaluation task described in (Gandhe and Traum, 2013). Wizards were instructed in how to use the search and relevance feedback features. In order to not bias the wizards, they were not told exactly what *score1* and *score2* indicate, but just that the scores can be useful in search.

Each wizard is presented with a set of utterances (U_{train}) ($|U_{train}| \approx 500$) and is asked to select a subset from these that will be appropriate as a response for the presented dialogue context. Each wizard was requested to select somewhere between 5 to 10 (at-least one) appropriate responses for each dialogue context extracted from five different human-human dialogues. There are a total of 89 dialogue contexts for the role that the wizards were to play. Figure 2 shows the histogram for the number of utterances selected as appropriate responses by the four wizards. As expected, wizards frequently chose multiple utterances as appropriate responses (mean = 7.80, min = 1, max = 25).

To get an idea about how much the wizards agree among themselves for this task, we calculated the overlap between the utterances selected by a specific wizard and the utterances selected by another wizard or a set of wizards. Let U_c^T be a set of utterances selected by a wizard T for a dialogue context c. Let R be a set of wizards ($T \notin R$) and U_c^R be the union of sets of utterances selected by the set of wizards (R) for the same context c. Then we define the following overlap measures,

$$\text{Precision}_c = \frac{|U_c^T \cap U_c^R|}{|U_c^T|} \quad \text{Recall}_c = \frac{|U_c^T \cap U_c^R|}{|U_c^R|}$$

$$\text{Jaccard}_c = \frac{|U_c^T \cap U_c^R|}{|U_c^T \cup U_c^R|} \quad \text{Dice}_c = \frac{2|U_c^T \cap U_c^R|}{|U_c^T| + |U_c^R|}$$

$$\text{Meteor}_c = \frac{1}{|U_c^T|} \sum_{u_t} \textsc{Meteor}\ (u_t, U_c^R) \quad \forall u_t \in U_c^T$$

We compute the average values of these overlap measures for all contexts and for all possible settings of test wizards and reference wizards. Table 1 shows the results with different values for the number of wizards used as reference.

#ref	Prec.	Rec.	Jacc.	Dice	Meteor
1	0.145	0.145	0.077	0.141	0.290
2	0.244	0.134	0.093	0.170	0.412
3	0.311	0.121	0.094	0.171	0.478

Table 1: Inter-wizard agreement

Precision can be interpreted as the probability that a response utterance selected by a wizard is also considered appropriate by at least one other wizard. Precision rapidly increases along with the number of reference wizards used. This happens because the size of the set U_c^R steadily increases with more reference wizards. Figure 3 shows this observed increase and the expected increase if there were no overlap between the wizards. The near-linear increase in $|U_c^R|$ suggests that selecting appropriate responses is a hard task and may require a lot more than four wizards to achieve convergence.

Subjectively, the wizards reported no major usability problems with the tool, and were able to use all four utterance ordering techniques to find appropriate utterances.

4 Future Work

Future work involves performing some formal evaluations comparing this tool to other tools (that are missing some of the features of this tool) in terms of amount of time taken to make selections and quality of the selections, using the same evaluation techniques as (Gandhe and Traum, 2013).

We also see a promising future for semi-automated selection, which blurs the line between a pure algorithmic response and pure wizard selection. Here the wizard can select appropriate responses, which can be used by algorithms as supervised training data, meanwhile the algorithms can be used to seed the wizard's selection.

References

David DeVault, Susan Robinson, and David Traum. 2010. IORelator: A graphical user interface to enable rapid semantic annotation for data-driven natural language understanding. In *Proc. of the 5th Joint ISO-ACL/SIGSEM Workshop on Interoperable Semantic Annotation (ISA-5)*.

David DeVault, Anton Leuski, and Kenji Sagae. 2011. An evaluation of alternative strategies for implementing dialogue policies using statistical classification and rules. In *Proceedings of the IJCNLP 2011*, Nov.

Sudeep Gandhe and David Traum. 2007. Creating spoken dialogue characters from corpora without annotations. In *Proceedings of Interspeech-07*, Antwerp, Belgium.

Sudeep Gandhe and David Traum. 2010. I've said it before, and I'll say it again: an empirical investigation of the upper bound of the selection approach to dialogue. In *Proceedings of the SIGDIAL '10*, Tokyo, Japan.

Sudeep Gandhe and David Traum. 2013. Surface text based dialogue models for virtual humans. In *Proceedings of the SIGDIAL 2013*, Metz, France.

A. Lavie and M. J. Denkowski. 2009. The meteor metric for automatic evaluation of machine translation. *Machine Translation*, 23:105–115.

Anton Leuski and David R. Traum. 2010. NPCEditor: A tool for building question-answering characters. In *Proceedings of LREC 2010*, Valletta, Malta.

Anton Leuski, Ronakkumar Patel, David Traum, and Brandon Kennedy. 2006. Building effective question answering characters. In *Proc. of SIGDIAL '06*, Australia.

Linus Sellberg and Arne Jönsson. 2008. Using random indexing to improve singular value decomposition for latent semantic analysis. In *Proceedings of LREC'08*, Morocco.

Ingrid Zukerman and Yuval Marom. 2006. A corpus-based approach to help-desk response generation. In *Computational Intelligence for Modelling, Control and Automation (CIMCA 2006), IAWTIC 2006*.