A domain ontology based metric to evaluate spoken dialog systems

Jan Kleindienst, Jan Cuřín, Martin Labský
IBM Research
Prague, Czech Republic
{jankle, jan_curin, martin.labsky}@cz.ibm.com

Abstract

Current methods and techniques for measuring performance of spoken dialog systems are still very immature. They are either based on subjective evaluation (Wizard of Oz or other usability studies) or they are borrowing automatic measures used in speech recognition, machine translation or action classification, which provide only an incomplete picture of the performance of the system. We introduce a method for quantitative evaluation of spoken dialog systems that utilizes the domain knowledge encoded by a human expert. The evaluation results are described in the form of a comparison metric consisting of domain coverage and dialog efficiency scores allowing to compare relative as well as absolute performance of a system within a given domain. This approach has the advantage of comparing incremental improvements on an individual dialog system that the dialog designer may want to verify along the way. In addition, the method allows to cross-check the performance of third-party dialog systems operating on the same domain and understand the strong and weak points in the dialog design.

1 Introduction

Research in the field of conversational and dialog systems has a long tradition starting in 1966 with Weizenbaum's Eliza (Weizenbaum, 1966). More recently, research in spoken dialog systems has tackled more ambitious domains, such as problem solving (Allen et al., 2007), navigation (Cassell et al., 2002), or tutoring systems (Graesser et al., 2001). This paper is organized as follows: in the introduction we outline our motivation and the principle of the proposed method. Section 2 describes in detail the proposed dialog score and its computation. Section 3 presents a case study in the music management domain and demonstrates the application of the scoring to a real-world task. We discuss the correlation of the proposed metric with subjective evaluation in Section 4, and conclude by Section 5.

1.1 Rationale

Current methods and techniques for measuring performance of speech-enables user interfaces are still very immature. They are either based on subjective evaluation (Wizard of Oz or other usability studies) or they are borrowing automatic measures used in speech recognition, machine translation or action classification, which provide only incomplete picture of the performance of the system. Nowadays, dialog systems are evaluated by action classification error rate (Jurafsky and Martin, 2008), by techniques that measure primarily dialog coherence (Gandhe and Traum, 2008), by methods based on human judgment evaluation, such as PARADISE (Walker et al., 2000; Hajdinjak and Mihelific, 2006), or using reward function values (Rieser and Lemon, 2008; Singh et al., 1999). What is particularly missing in this area are (1) a measurement of performance for a particular domain, (2) possibility to compare one dialog system with others, and (3) evaluation of a progress during the development of dialog system. The score we present attempts to address all three issues.

2 The Proposed Method of Dialog System Evaluation

The proposed dialog score (DS) consists of two ingredients both of which range from 0 to 1:

- Domain Coverage (DC) score,
- Dialog Efficiency (DE) score.

The DC expresses how the evaluated system covers the set of tasks in the ontology for a particular domain, while the DE indicates the performance of the evaluated system on those tasks supported by the system over user test sessions.

We describe both scores in the following subsections. Note that the results of domain coverage and dialog efficiency may be combined into a single compound score to attain a single overall characteristic (the eigen value) of the assessed dialog system.

2.1 Scoring of Domain Coverage

The domain coverage (DC) is a sum of weights of the tasks supported by the system (S) over the sum of weights of all tasks from the ontology (O).

$$DC(S, O) = \frac{\sum_{t \in supported\_tasks(S,O)} w_t}{\sum_{t \in all\_tasks(O)} w_t} \qquad (1)$$

Table 1 shows a sample domain task ontology for the music management domain that shows the raw points assigned by a domain expert and their normalized versions that are used to assess the relative importance of individual tasks. The expert may control the weights of whole task groups (such as Playback control) as well as the weights of individual tasks that comprise these groups. Generally, the ontology can have more than two levels of sub-categorization that are shown in the example. So far our task ontologies have been limited to hierarchical sets of weighted tasks. We are however investigating whether introducing domain concepts, such as “song”, “album” or “playlist”, and relations among them, can help derive possible user tasks and their weights semi-automatically.

2.2 Scoring of Dialog Efficiency

The actual efficiency of a dialog is measured using the number of dialog turns (Le Bigot et al., 2008; Nielsen, 1994) needed to accomplish a chosen task. In spoken dialog systems, a dialog turn corresponds to a pattern of a user speech input followed by the system’s response. We introduce a generalized penalty turn count (PTC) that measures overall dialog efficiency by incorporating other considered factors: number of help requests, number of rejections, and user and system reaction times, and in the future possibly also others.

Table 1: Speech-enabled reference tasks for the music management domain. (Tasks are divided into groups. Both the group as well as tasks within the group are assigned relative importance points (weights) by an expert. These points are normalized to obtain per-task contribution to the domain’s functionality. ITC shows ideal turn count range for each task.)

Description	Points	Contr	ITC
Volume	2	15.50	-
relative	2	6.20	1
absolute	1	3.10	1
mute	2	6.20	1
Playback	4	31.01	-
play	3	7.75	1
stop	3	7.75	1
pause	1.5	3.88	1
resume	1.5	3.88	1
next, previous track	1	2.58	1
next, previous album	1	2.58	1
media selection	1	2.58	1
Play mode	0.5	3.88	-
shuffle	1	1.94	1
repeat	1	1.94	1
Media library	6	46.51	-
browse by criteria	2	3.93	1..2
play by criteria	4	7.85	1..2
search by genre	2	3.93	1
search by artist name			-
up to 100 artists	1	1.96	1..2
more then 100 artists	2	3.93	1..2
search by album name			-
up to 200 albums	1	1.96	1..2
more than 200 albums	2	3.93	1..2
search by song title			-
up to 250 songs	1	1.96	1..2
more than 2000 songs	2	3.93	1..2
search by partial names			-
words	1	1.96	2
spelled letters	1	1.96	2
ambiguous entries	2	3.93	2
query			-
item counts	0.5	0.98	1
favorites			-
browse and play	0.5	0.98	1..2
add items	0.3	0.59	1
media management			-
refresh from media	0.2	0.39	1
add or remove media	0.2	0.39	1..2
access online content	1	1.96	2..3
Menu	0.4	3.10	-
quit	0.5	1.03	1..2
switch among other apps	1	2.07	1..2
Sum	44.2	100	-

$$\begin{aligned} PTC(t) = \quad & TC(t) + \lambda_{hr} hr(t) + \lambda_{rj} rj(t) \\ & + \lambda_{srt} srt(t) \end{aligned} \qquad (2)$$

where TC is the actual dialog turn count, hr is the number of help requests, rj is the number of rejections, and srt is system response time and the coefficients represent weights of each contributor to the final penalty turn count (PTC)[1]. TC, hr, and rj are averaged over the number of trials. By trial we mean each attempt of the user to perform a specific task. The system response time (srt)

[1] In our experiments, we set $\lambda_{hr} = 0.5$, $\lambda_{rj} = 1$, and $\lambda_{srt} = 0.3$.

is the average of system reaction times (in seconds) exceeding a constant c_{asrt} over the number of turns in trials (t_i). Acceptable systems reaction time constant (c_{asrt}) is set to 0.1, i.e. the acceptable threshold is 100 ms.

$$srt(t) = \frac{\sum\limits_{\text{all turns } t_i \text{ for task } t} \max(st(t_i) - c_{asrt}, 0)}{|t|} \tag{3}$$

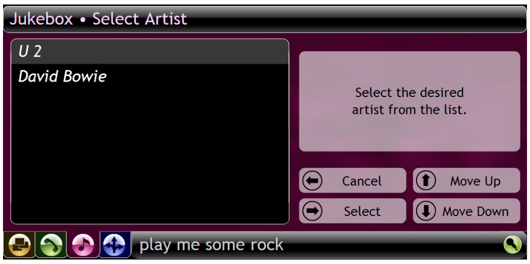
The obtained penalty turn count is then compared to an ideal number of turns for a particular task. The ideal turn count $ITC(t)$ for task t is the number of dialog turns needed to accomplish the task using an ideally efficient dialog system by a native user acquainted with the system.

Currently we determine $ITC(t)$ manually by human judgment. The $ITC(t)$ typically corresponds to the number of *coherent information blocks* that can be identified in the information that needs to be communicated by the user. For example, suppose a "date" value consisting of three information slots (day, month and year) needs to be entered. All slots however comprise a single coherent block of information that is typically communicated at once and thus we would set $ITC(t) = 1$ for this task. Table 2 shows a task in which the user selects a song whose title is ambiguous. The ideal system is expected to disambiguate in one extra turn and therefore we set $ITC(t) = 2$.

The actual score of the dialog efficiency (DE score) for an individual task is then counted as a fraction of the difference between ITC and PTC against current PTC, i.e.:

$$DE(t) = 1 - \max\left(\frac{PTC(t) - ITC(t)}{PTC(t)}, 0\right) \tag{4}$$

To avoid subjective scoring we typically use several human testers as well as several trials per one task. For example for the task "play by artist" the following set of trials can be used: "Play something by Patsy Cline", "Play some song from your favorite interpreter", or "Play some rock album, make the final selection by the artist name". Each of these trials is assigned its ideal number of turns (this is why ITCs for tasks in the ontology are given by ranges in Table 1.) The task dialog efficiency score is then computed as an average over all human testers and dialog efficiency scores for all their trials.



Jukebox • Select Artist

U 2

David Bowie

Select the desired artist from the list.

play me some rock

Figure 1: GUI of Jukebox application

Samples of trials used in the evaluation of the music management domain are given in Table 2. Figures of ITC and average turn count in this table are further discussed in Section 3.

The final dialog score is then counted as a sum of products of domain coverage and dialog efficiency for each task in the domain ontology, i.e.:

$$DS(S, O) = \frac{\sum_{t \in tasks(S,O)} w_t \, DE(t)}{\sum_{t \in all\_tasks(O)} w_t} \tag{5}$$

3 Example of Dialog Scoring on Music Management Domain

We applied the dialog scoring to our two dialog systems developed at different times and both partially covering the music management dialog domain. Both allow their users to play music by dynamically generating grammars based on meta tags found in users' mp3 files. The first one, named A-player, is simpler and covers a limited part of the music management domain. The second, named Jukebox, covers a larger part of the domain and also allows free-form input using a combination of statistical language models and maximum entropy based action classifiers. Figure 1 shows the GUI of the Jukebox application.

For both applications, we collected input from a group of 15 speakers who were asked to accomplish tasks listed in Table 2. Each of these user tasks corresponded to a task in the domain task ontology and there was at least one user task per each ontology task that was supported by either A-player or Jukebox. The subjects were given general guidance but no sample English phrases were suggested to them that could be used to control the system. In order not to guide users even by the wording of the user tasks, the tasks were described to them in their native language. All subjects were non-native but fluent English speakers.

Table 2: Specific tasks to be accomplished by personas using A-player and Jukebox with ideal number of turns (ITC) and average turn count (TC). Tasks which appeared to be more hard than expected are indicated in bold, easier than expected are in italic.

		Aplayer		Jukebox	
Task	ITC	TC	TC/ITC	TC	TC/ITC
Start playback of arbitrary music	1	1.5	1.5	3.1	**3.1**
Increase the volume	1	-	-	1.4	1.4
Set volume to level 10	1	-	-	1.4	1.4
Mute on	1	-	-	1.2	1.2
Mute off	1	-	-	1.5	1.5
Pause	1	-	-	2.1	**2.1**
Resume	1	-	-	2.5	**2.5**
Next track	1	1.4	1.4	1.1	1.1
Previous track	1	1.5	1.5	1.3	1.3
Shuffle	1	1.0	1.0	1.3	1.3
Play some jazz song	1	-	-	1.4	1.4
Play a song from Patsy Cline	1	1.5	1.5	2.0	**2.0**
Play Iron Man from Black Sabbath	1	1.9	**1.9**	2.8	**2.8**
Play the album The Best of Beethoven	1	1.1	1.1	1.7	**1.7**
Play song Where the Streets Have No Name	1	1.4	1.4	1.3	1.3
Play song Sonata no. 11 (ambiguous)	2	1.1	*0.6*	3.7	**1.8**
Play a rock song by your favorite artist	3	2.6	*0.9*	4.4	1.5
Reload songs from media	1	1.5	1.5	-	-

3.1 Domain Coverage for Music Management Domain

This restricted ontology represents the human expert knowledge of the domain and is encoded as a set of tasks with two kinds of relations between the tasks: task generalization and aggregation. Individual tasks are defined as sequences of parametrized actions. Actions are separable units of domain functionality, such as volume control, song browsing or playback.

Parameters are categories of named entities, such as album or track title, artist name or genre. Tasks are labeled by weights, which express the relative importance of a particular task with respect to other tasks. The ontology may also define task aggregations which explicitly state that a complex task can be realized by sequencing several simpler tasks. Table 1 shows a sample task ontology for the music control domain. For example, the task volume control/relative with weight of 2 (e.g. “louder, please”) is considered more important in evaluation than its absolute sibling (e.g. “set volume to 5”). This may be highly subjective if scored by a single human judge and thus a consensus of domain experts may be required to converge to a generally acceptable ontology for the domain. Once acknowledged by the community, this ontology could be used as the common etalon for scoring third-party dialog systems.

Table 3: Computation of domain coverage, dialog efficiency and dialog score for A-player

Task	DC	DE	final DS
play	7.75	0.67	0.052
stop	7.75	1.00	0.078
next, prev. track	2.58	0.73	0.019
play by criteria	7.85	0.71	0.055
search by artist			
$\leq$ 100 artists	1.96	0.60	0.012
$>$ 100 artists	3.93	0.60	0.024
search by album			
$\leq$ 200 albums	1.96	0.89	0.017
$>$ 200 albums	3.93	0.89	0.035
search by song			
$\leq$ 250 songs	1.96	0.86	0.017
$>$ 2000 songs	3.93	0.86	0.04
media refresh	0.39	0.67	0.003
Total (in %)	47.92	71.14	36.11

3.2 Computing Dialog Scores for Music Management Domain

Tables 3 and 4 show the computation of the final dialog system score (DS) and its components: domain coverage (DC) and domain efficiency (DE). For A-player, which is limited in functionality, the weighted domain coverage reached only 47.92%, whereas for Jukebox it was 83.17%. On the other hand, A-player allowed its users to accomplish the tasks it supported faster than Jukebox; this is documented by the weighted dialog efficiency score reaching 71.14% for A-player and 64.62% for Jukebox. This was mainly due to Jukebox being more interactive (e.g. asking questions, presenting choices) and due to a slightly higher error

Table 4: Computation of domain coverage, dialog efficiency and dialog score for Jukebox

Task	DC	DE	final DS
volume relative	6.20	0.74	0.046
volume absolute	3.10	0.74	0.023
mute	6.20	0.82	0.051
play	7.75	0.33	0.025
stop	7.75	0.82	0.064
pause	3.88	0.48	0.019
resume	3.88	0.41	0.016
next, prev. track	2.58	0.93	0.024
next, prev. album	2.58	0.76	0.020
shuffle	1.94	0.76	0.015
browse by criteria	1.97	0.53	0.010
play by criteria	7.85	0.68	0.054
search by genre	3.93	0.74	0.029
search by artist			
$\leq$ 100 artists	1.96	0.50	0.010
$>$ 100 artists	3.93	0.60	0.024
search by album			
$\leq$ 200 albums	1.96	0.35	0.007
$>$ 200 albums	3.93	0.75	0.029
search by song			
$\leq$ 250 songs	1.96	0.65	0.013
$>$ 2000 songs	3.93	0.93	0.036
word part. search	1.96	0.51	0.010
ambiguous entries	3.93	0.54	0.021
Total (in %)	83.17	64.62	54.45

rate of a free-form system (language model-based) as opposed to a grammar-based one. The overall dialog score was higher for Jukebox (54.45%) than it was for A-player (36.11%). This was in accord with the feedback we received from users, who claimed they had better experience with the Jukebox application, see Section 4.

4 Towards Correlation between Proposed Metrics and Subjective Evaluation

The HCI methodology (Nielsen, 1994) advocates several factors that human judges collect in the process of dialog system evaluation. These key indicators include accuracy, intuitiveness, reaction time, and efficiency. When designing the evaluation method we attempted to incorporate the core of these indicators into the scoring method to ensure good correlation of the proposed metric with human judgment.

After performing the case study for DE scoring, we asked the evaluators to fill in a questionnaire with their subjective feedback. There were three sets of questions: (1) speech suitability, (2) application-specific evaluation, and (3) question about location where they would be willing to use such applications.

The human evaluators were asked to rate each question (listed in Table 5), for both applications, with a score of 0 points (worst) to 5 points (best). The meaning of the points is shown below:

0 . . . worst, the system is not usable at all by anyone
1 . . . not sufficient for real usage, only good as a toy
2 . . . reasonable, but I would not consider using it
3 . . . reasonable, I would consider using it
4 . . . good understanding and behavior, I would use it
5 . . . excellent understanding and behavior

Generally, the evaluators were pretty positive in scoring speech suitability for music management domain in Question 1. In the application evaluation group of questions, the more advanced Jukebox application was perceived better (63.2% vs. 50.7% for A-player). Support of free-form commands by the Jukebox application and its broader functionality was reflected in Jukebox's score of 72.9% for Question 4 (vs. 54.3% for A-player) and influenced also answers to Questions 2 and 3. A-player's slightly higher score for Question 5 (65.7% vs. 62.9% for Jukebox) corresponds to the fact that the restricted set of commands and functionality makes the speech recognition task easier and therefore the users feel the system obeys their commands better. Results for the last two questions about location, where the evaluator would be willing to use the voice driven system, are less positive for home usage (54.3% and 57.1%) but the evaluators foresee an added value in using speech modality in environments when other input devices (such as keyboard, buttons, or touch screens) can be disturbing, i.e. in cars.

Statistically speaking, the average correlation between the vector of dialog scores, assembled for each individual speaker, and the vector of averaged points received from his/her subjective evaluation, was 0.67.

5 Conclusion

The objective of our approach is to evaluate spoken and multi-modal dialog systems within a predefined, well-known (and typically narrow) domain. In our labs we have used heterogeneous technologies such as grammars, language models and natural language understanding techniques to develop many speech and multimodal applications for various domains, such as music selection, TV remote control, in-car navigation and phone control. In order to compare two spoken dialog systems that deal with the same domain, we first describe the domain using a task ontology which defines user tasks relevant for the chosen domain as

Table 5: Questionnaire filled by the human evaluators after the test. The figures are given in percentage of "satisfaction" calculated from averaged points (between 0 and 5) given by the human evaluators.

Question	Aplayer	Jukebox
A. Speech suitability		
1. Do you think the concept of voice control makes sense for the jukebox domain?	71.4	
B. Application evaluation		
2. Would you use the system?	37.1	**55.7**
3. Do you think someone else could use the system?	45.7	**61.4**
4. Did you know what to say at each point of interaction?	54.3	**72.9**
5. Did the system obey your commands?	**65.7**	62.9
Application evaluation results (questions 2-5 averaged)	*50.7*	*63.2*
C. Where to use the application		
6. Would you use the system at home?	54.3	**57.1**
7. Would you use the system in car?	62.9	**71.4**

well as their relative importance. This enables us to compare two dialog systems against each other (1) by comparing their coverage of the ontology tasks, and (2) by contrasting their dialog efficiency over the supported tasks. A single dialog score statistic can be produced by combining the dialog coverage and dialog efficiency components.

The presented approach is suitable for comparing different dialog systems of third parties as well as successive versions of a single system being developed. Human evaluations are currently conducted to estimate the correlation between the dialog score and human judgment. The subjectivity of human scoring and consensus on the ontology coverage are subject of further investigation.

Acknowledgments

We thank to the Jukebox application creators, esp. to Luboš Ureš, Ladislav Seredi and Mark Epstein. We would like to acknowledge support of this work by the European Commission under IST FP6 integrated project Netcarity, contract number IST-2006-045508.

References

Allen, J., Chambers, N., Ferguson, G., Galescu, L., Jung, H., Swift, M., Taysom, W. 2007. PLOW: A Collaborative Task Learning Agent. *Twenty-Second Conference on Artificial Intelligence (AAAI-07).*

Le Bigot, L., Bretier, P., Terrier, P. 2008. Detecting and exploiting user familiarity in natural language human-computer dialogue. *Human Computer Interaction: New Developments. Kikuo Asai (Eds), InTech Education and Publishing*, ISBN: 978-953-7619-14-5, 269-382.

Carroll, J. 2001. Human Computer Interaction in the New Millennium. *New York: ACM Press.*

Cassell, J., Stocky, T., Bickmore, T., Gao, Y., Nakano, Y., Ryokai, K. 2002. Mack: Media lab autonomous conversational kiosk. *Imagina02.*

Gandhe, S., Traum, D. 2008. Evaluation understudy for dialogue coherence models. *Proceedings of the 9th SIGdial Workshop on Discourse and Dialogue, Columbus, Ohio, Association for Computational Linguistics*, 172-181.

Graesser, A.C., VanLehn, K., Rosfie, C.P., Jordan, P.W., Harter, D. 2001. Intelligent tutoring systems with conversational dialogue. *AI Mag. 22(4)*, 39-51.

Hajdinjak, M., Mihelific, F. 2006. The paradise evaluation framework: Issues and findings. *Comput. Linguist. 32(2)*, 263-272.

Jurafsky, D., Martin, J.H. 2008. Speech and Language Processing (2nd Edition): An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition. *(Prentice Hall Series in Artificial Intelligence)*

Nielsen, J. 1994. Heuristic evaluation. *Usability Inspection Methods, J. Nielsen and R.L. Mack, R.L. (Eds), John Wiley and Sons: New York*, ISBN: 0-471-01877-5, 25-64.

Rieser, V., Lemon, O. 2008. Learning Effective Multimodal Dialogue Strategies from Wizard-of-Oz data: Bootstrapping and Evaluation. *Proceedings of ACL-08: HLT*, pages 638-646, Columbus, Ohio, USA, June 2008.

Singh, S., Kearns, M. S., Litman, D. J., Walker, M. A. 1999. Reinforcement learning for spoken dialogue systems. *In Proc. NIPS99.*

Walker, M., Kamm, C., Litman, D. 2000. Towards developing general models of usability with paradise. *Nat. Lang. Eng. 6(3-4)*, 363-377.

Weizenbaum, J. 1972. ELIZA - A Computer Program for the Study of Natural Language Communication between Man and Machine. *Communications of the Association for Computing Machinery 9*, 36-45.