INTERSPEECH 2015

An Entropy Minimization Framework for Goal-Driven Dialogue Management

Ji Wu[1], Miao Li[1], Chin-Hui Lee[2]*

[1]Department of Electronic Engineering, Tsinghua University, Beijing, China
[2]School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA. USA
Wuji_ee@tsinghua.edu.cn, chinhui.lee@ece.gatech.edu

10.21437/Interspeech.2015-460

Abstract

We propose an entropy minimization dialog management (D-M) strategy for goal-driven information retrieval (IR). By associating each goal of an IR task with a set of stochastic attributes, reaching a goal can then be accomplished by filling the "attribute slots" corresponding to the goal. Information access can now be cast as a dialog problem that specific information about the attributes is solicited from a user by the system through multiple dialog turns. For a real-world music search task with 38118 songs and 12 attributes corresponding to each song, we demonstrate the concept by designing a simulation game to order song from the above music database. We show that 8.3 dialog turns are needed on the average if random questions are asked by the system, whereas the entropy minimization DM is a very efficient goal seeking method to order a song with the least amount of 3.3 dialog turns among different strategies. Furthermore, the proposed DM techniques can manage the dialog process in a more efficient and flexible manner.

Index Terms: information retrieval, probabilistic dialogue management, entropy minimization, goal-driven strategy

[1]

1. Introduction

With the fast growing availability of a vast amount of useful data on the web, information retrieval [1], search [2] and question answering [3] are now critical parts of human daily activities in the society. Online customer services [4] and call routing [5] at call centers are technology extensions. Designing efficient and effective access systems to such data is therefore becoming very important research topics [6]. A majority of the systems in use today are based on one-shot interactions that the system will give results based a single user query. The responses are often arranged in a list of web links (e.g., Google [7], Bing [8] and Baidu [9] search engines). It is then up to the users to formulate their queries in a smart way in order to reduce the size of the returned lists to have fast and usable search results.

[1] *This work was done during the first authors visiting stay at School of Electrical and Computer Engineering, George Institute of Technology in Atlanta, Georgia, USA, 2013-2015. This work was funded partly by the National Natural Science Funds of China under Grant 61170197, partly by the National High-Tech. R&D Program of China (863 Program) under Grant 2012AA011004, partly by the Planned Science and Technology Project of Tsinghua University under Grant 20111081023, and partly by the Electronic Information Industry Development Fund of China under project "The R&D and Industrialization on Information Retrieval System Based on Man-Machine Interaction with Natural Speech".

Although we have witnessed successful deployments of the abovementioned one-shot interactive systems in real-world applications, they still have two major shortcomings. First of all, "best-match" is still the basic principle of such IR systems, which means the best possible system response will be the result most closely matches the user request [10]. It requires the user to be able to specify all the relevant aspects of the target goal in the users request, and the systems could fully understand the intention from the users input. However this is far from the real situation in which neither the user nor the system had a full knowledge about each others goal and expectations. The second major problem lies in the fact that the user and the system are not working collaboratively together to achieve the users goal in an efficient manner. So these one-shot interactive systems often only work passively not being able to provide intelligent feedback according to the users request. There is therefore no guarantee for an optimal system performance.

Nonetheless if we relax the one-shot constraint and allow multiple turns in the above interactions, then the IR problem can be formulated as a human-machine dialog process that many existing system design principles can be incorporated into implementing multi-turn information access [11]. It is clear that users would like to obtain the desired information exactly within a minimum number of dialog turns [12]. Therefore a good dialogue strategy plays a critical role in information retrieval dialog management.

There are many categories of dialogue management strategies. For example, the strategy can be system-initiative or mixed-initiative [13]. It has been argued that a mixed-initiative conversational agent provides an easier, interactive and natural access to information [14]. However not all experiments support this claim [15].

From an implementation point of view, different dialog management strategies have also been proposed and no single one clearly stands out [16]. McTear [17] represents the dialog structure as a finite-state transition network. The AT&T HMIHY system [18] introduces a dialog-based call routing system based a hand-coded hierarchy of call types and attempts to use the outcome of the algorithm to select from a set of dialog strategies. Natural language call routing [19] also developed a router and a dialog disambiguation module, both of them can be trained automatically from labeled examples. Markov decision process (MDP) [20][21] based approaches optimize their actions by reinforcement learning [22]. To deal with ASR and NLU errors, an improved framework called partially observable Markov decision process (POMDP) has been proposed [23]. Rather than maintaining one hypothesis for the state, the POMDP framework maintains a distribution over all possible dialogue states [24]. Scalability is assumed to be the major difficulty of the POMDP framework. The number of potential states grows exponentially during a dialogue process and the computational complexity of exact POMDP optimization grows astronomically [24].

Other researchers try to exploit data mining techniques to summarize the back-end database and generate specific queries to users in a cooperative way [25] [26]. The DM module scans the database at every turn in a dialogue process. It filters the database based on the users intentions and preferences, and the resulting items are used to determine the content of the summary and the system actions. These queries tend to ask a user about those attributes with high uncertainty [25], which can reduce the search space at most. Their experiment results show that a dialogue process will become more efficient if the dialogue system can generate suitable questions for the users, which are based on the intentional summaries. One problem of this database summary method is that it implicitly assumes that all attributes are uniform distributed, so those attributes with more distinct values would have higher entropies [25]. However, this assumption prevents to integrate a real prior distribution of database entries into the dialogue management.

In this paper, we consider information retrieval as a multiple-turn, goal-driven task. By associating each goal of an IR task with a set of stochastic attributes, reaching a goal can then be accomplished by filling the "attribute slots" corresponding to the goal. Information access can now be cast as a dialog problem that specific information about the attributes is solicited from a user via multiple dialog turns. We propose a probabilistic framework for that goal-driven information retrieval task and an entropy minimization approach for the corresponding dialogue manager.

We test our proposed framework on a music search task with 38118 songs and 12 attributes corresponding to each song. We demonstrate the entropy minimization concept by designing a simulation game to order song from the above real-world music database. We show that 8.3 dialog turns are needed on the average if random questions are asked by the user. Whereas the entropy minimization is a very efficient goal seeking method to order a song with the least amount of 3.3 dialog turns. Furthermore, if the prior distribution over the database is considered, the proposed method can provide much better performance than other strategies, which shows that it can manage the dialog process in a more efficient and flexible manner.

The rest of the paper is organized as follows. First, a probabilistic framework for dialogue system is described in Section 2. An entropy reduction based strategy for dialogue design is introduced in Section 3. Then in Section 4, an interesting game is designed as testing scenario. And the experiment results and analysis appear in Section 5. Finally, we conclude the paper in Section 6.

2. Probabilistic Framework for Goal-driven Information Retrieval

An information retrieval system should have a knowledge base as the source of information, so the system can provide answer to the requests of users. We note the knowledge base as K, the question from users as q, and the corresponding answer as A. Then we have the joint probability distribution of this question-answer process as $P(A, q, K)$. By the definition of conditional probability, and considering the fact that the question q is independent of knowledge base K, we have:

$$P(A, q, K) = P(A|q, K)P(q)P(K) \tag{1}$$

So the best answer to that question q will be:

$$A = \arg\max_{A} P(A|q, K) \tag{2}$$

In interactive information retrieval system, to achieve a certain goal, the user and the system should work collaboratively together to understand the intention of the user in an efficient manner and find out the corresponding answer. So K is not only stand for the knowledge base, but also extend to the whole background knowledge that the user and the system share.

Now in the multiple-turn dialog process, the initial request is usually initiated by the user, while later, there are also some questions asked by the system to get information from the user. They work collaboratively together to find out the goal of the task. So we note all these questions, both from the user and the system, as a sequence $Q = \{q_1, q_2, ..., q_n\}$. Then the joint probability distribution of multiple-turns dialog process is:

$$\begin{aligned}&P(A, Q, K) = P(A|Q, K)P(Q|K)P(K)\\&= P(A|\{q_1, q_2, ..., q_n\}, K)P(\{q_1, q_2, ..., q_n\}|K)P(K)\end{aligned} \tag{3}$$

While the best result of that dialog process is:

$$A = \arg\max_{A, q_1, q_2, \ldots, q_n} P(A|\{q_1, q_2, ..., q_n\}, K)P(\{q_1, q_2, ..., q_n\}|K) \tag{4}$$

3. Entropy Minimization Approach

In multiple-turns information retrieval systems, the dialogue management plays a critical role to generate the system output, which is crucial to keep the dialogue going forward and achieve the goal efficiently.

For goal-driven information retrieval, each goal of an information retrieval task can be represented with a set of stochastic attributes. Reaching a particular goal can then be accomplished by filling the "attribute slots".

Traditional dialogue systems often follow a certain order to get information from the user with system-initiative strategy, or try to extract information from the user freely with a user-initiative strategy, until all attribute slots are filled. So efficiency cannot be guaranteed. While in our proposed framework, we develop an entropy minimization approach to multiple-turn dialogue management.

From Eq.(3), we can decompose question sequence Q with chain rule of probability, in that case, we have:

$$\begin{aligned}&P(A, Q|K) = P(A|Q, K)P(Q|K)\\&= P(A|\{q_1, q_2, ..., q_n\}, K)P(\{q_1, q_2, ..., q_n\}|K)\\&= P(A|\{q_1, q_2, ..., q_n\}P(q_1|K)P(q_2|q_1, K)...P(q_n|q_{n-1}...q_1, K)\end{aligned} \tag{5}$$

Based on entropy theory, we have following formula:

$$
\begin{aligned}
& H(A|\{q_1, q_2, ..., q_n\}, K) \\
& = H(A, Q|K) - H(\{q_1, q_2, ..., q_n\}|K) \\
& = H(A, Q|K) - H(q_1|K) - H(q_2|q_1, K)... \\
& - H(q_n|q_{n-1}...q_1, K)
\end{aligned} \tag{6}
$$

Eq.(6) implies that the multiple-turn dialogue is an entropy reduction process. With each question and its answer, the uncertainty of the result goes down. In the end, we can get the result without any uncertainty. That means if we try to find the question which can reduce the entropy the most, we can reach the final result faster. Based on that idea, we have the entropy minimization approach as follows: in the beginning, when the user initiates the dialog process, with the information extracted from the user input, and fills the attributes slots of the given goal, the system can estimate a probability distribution for the result and corresponding candidate list. In the following dialog process, the system will ask the question which can maximize the entropy reduction in the new turn, or update the distribution for the resulting candidate list with the new input from the user. Thus, the entropy of the candidate list goes down till we get the final result. Entropy minimization based on maximizing entropy reduction provides the most efficient strategy and can be done using a greedy algorithm [27].

4. Testing Scenario

Goal-driven information retrieval usually involves human-machine interactions (HCI) [28]. The process depends on both parties. Its not so easy to test the performance of any dialog strategy. Empirical evaluations are often employed [29] [30], and it is very expensive in both manpower and time. To circumvent this difficult, we design a game to simulate the testing scenario.

Table 1: *The 12 attributes of a song.*

ID	Attributes	Decription	Size
1	Singer	The name of the singer	3021
2	Gender	The gender of the singer	2
3	Region	The region of the singer	19
4	Album	The album of the song	10322
5	Company	The publisher of the song	1193
6	Language	The language of the song	10
7	Lyricist	The lyricist of the song	5603
8	Composer	The composer of the song	5642
9	Live	Live version or not	2
10	Time	Release time of the song	431
11	Style	The style of the song	346
12	Emotion	The emotion of the song	59

The goal of the game is to order a song from a music database consisting of 38118 songs. Corresponding to each entry a set of 12 attributes is used for the user to identify a particular song. Table 1 list all these attributes. Its really difficult to collect all relevant attributes for so many songs. In this particular case, some attribute information is missing for 20.8% of the songs in the music database. The number of entities associated with each attribute is also listed in Table 1. For example, there are 3020 distinct singers issuing a total of 10322 albums.

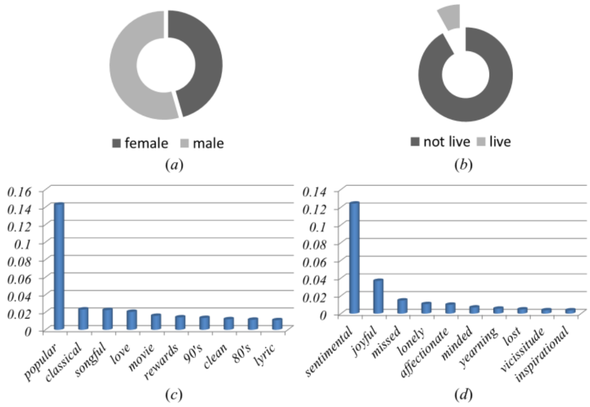

Figure 1: *(a) the distribution of the singer gender(Attri 2); (b) the distribution of the live version and not live version(Attri 10); (c) the part distribution of song style(Attri 11); (d) the part distribution of song emotion. (Attri 12) .*

In Figure 1, we can look into more detail of the music database. There are more songs from male singer than from female singer, but the difference is not much. Only a small part of the songs is from a live version. For top-10 distributions, "Popular" music ranks first in the attribute related to "style", and "Sentimental" ranks first in the attribute associated to "emotion".

To play the simulated music ordering game, two agents, robots RA and RB, are involved. RA selects a song in the database, and RB tries to find out the exact song RA has selected. RB can ask RA any question about these 12 attributes, and RA will answer it to the best of its knowledge. After RB gives a response, RA will update its candidate list and judge if it is good for final result. There are two situations for the correct result, namely: (i) there is only one song in the result and its same as the song that RA has selected in advance; and (ii) RB returns a song list, in which those songs share the same value in all these 12 attributes and the song that RA has selected is included. The absence of an attribute is assumed to be equal to any value in that attribute.

RB attempts to obtain the correct song with as few questions as possible, and RB can use one of the following 4 strategies and compare their performance

- Sequential Strategy: RB asks questions under a fixed order of these attributes;
- Random Strategy: RB randomly chooses a question about one attribute that has not been asked;
- Database Summary Strategy (DSDM): RB scans the candidate set in every turn and chooses a question about the attribute with the highest uncertainty;
- Entropy Minimization Strategy (EMDM): RB calculates the distributions and the corresponding entropies of the remaining attributes that have not been asked, and chooses the one with the maximum entropy reduction as the next question.

In fact, these four strategies can be divided into 2 categories. Sequential and random strategies make no use of any information of the database in the question order. Meanwhile, the DSD-

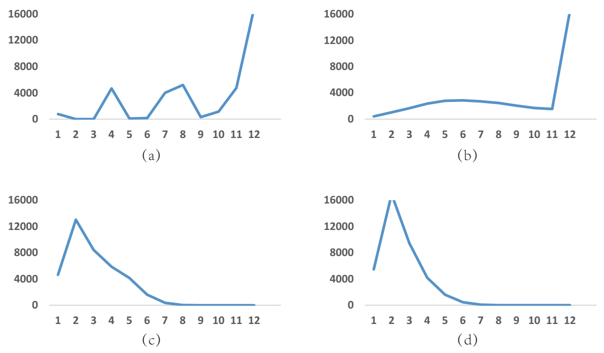

Figure 2: *Histograms of the numbers of turns required by four strategies. (a)sequential strategy; (b)random strategy; (c) DS-DM strategy; (d) EMDM strategy.*

M and EMDM strategies try to find more efficient actions with the remaining candidate set.

We conduct the experiments in two different settings. First, we do not have any knowledge about which song is preferred by the users, and therefore every song in the database shares the same prior probability. In this case, every song works as a goal set all together. Second, we do have a prior distribution over the database, which is obtained from the users dialog history. In this case, we do sampling from the database according to the prior distribution.

5. Experiments and Result Analysis

With the game described in Section 4, it was easy to evaluate the performance of different dialog strategies. For the #uniform setting, every song in the music database was evaluated once with the fours strategies. For #sampling setting, 50000 songs sampled form the database were evaluated. The average number of dialog turns required for each strategy is listed in Table 2.

Table 2: *Average numbers of dialog turns.*

Strategy	#uniform	#sampling
Sequential	9.298	8.306
Random	8.297	7.159
DSDM	3.330	3.223
EMDM	3.309	3.065

With the #uniform setting in the middle column, the sequential and random strategy needed 9.3 and 8.3 dialog turns to achieve the target song on average. With the #sampling setting in the right column, 8.3 and 7.2 dialog turns were needed by these two strategies, respectively. Comparing sequential and random strategies, shown in Figures 2(a) and 2(b), we can see that random ordering is much smoother in the histogram than that for fixed ordering. For these two strategies, unless the situation that only one candidate song left after several turns happened, the dialogue process would not end before getting all 12 attributes inquired. So a large part of dialogues took 12 turns of the interactions.

On the other hand, the DSDM and EMDM strategy were both based on analyzing the candidate sets. So these two methods could choose informative questions to ask the users, which leaded to a significant reduction of the dialogue turns. The DSDM method chose the attribute with the maximum distinct values, but the attribute with the maximum distinct values might not be the most informative one. For example, if a set of songs had many distinct composers, but most of the songs were composed by one of them. In this situation, the composer attribute might not be a good choice. The EMDM strategy could chose the most informative attribute to inquire, so it performed even better than the DSDM method. But the difference was not that much. To explain that results, a more detailed comparison was shown in Table 3.

Table 3: *Comparison of the EMDM with DSDM strategy.*

Setting	#E < #D	#E = #D	#E > #D	total
#uniform	4.09%	93.68%	2.23%	38117
#sampling	15.38%	82.75%	1.87%	500000

#E = EMDM strategy; #D = DSDM strategy
#E < #D: implying that the EMDM strategy can achieve a smaller average number of dialog turns than the DSDM strategy

Because the DSDM and the EMDM strategy are both based on entropy analysis, so in most situations, they make the same choice in the question to be asked next. But with the #uniform setting, we still can see that the chance for the EMDM strategy (4.09%) to perform better about two times than the DSDM (2.23%). With the #sampling setting, the chance for EMDM (15.38%) to be better is more than 8 times than the DSDM (1.87%). That experiment show the EMDM strategy can manage the dialog process in a more efficient and flexible manner.

6. Conclusion and Future Work

In this paper, a probabilistic framework for goal-driven information retrieval is presented. Through a simulation game to order songs from a real-world music database we demonstrate that the collaborative entropy minimization dialogue management strategy gives the best results in terms of the least number of turns required to accomplish a task under the assumption that the user and the system both have a full knowledge about the task being completed and the attributes associated with completing each goal in the task, and work in a collaborative manner. In a real-world dialog scenario, only partial knowledge is available. Automatic speech recognition errors also impose extra efforts for error correction and confirmation in addition to the disambiguation dialog generation needed in the process. More research is needed.

We believe the proposed entropy minimization dialogue management strategy provides an upper bound performance target for system designers to reach. In the meantime a stochastic representation of the dialog goals and the corresponding attributes is needed for designing sub-optimal strategies to approximate the best strategy. Again more research is needed

7. Acknowledgment

The authors thank Tsinghua University and Georgia Institute of Technology for their generous support.

8. References

[1] C. D. Manning, P. Raghavan, H. Schütze *et al.*, *Introduction to information retrieval*. Cambridge university press Cambridge, 2008, vol. 1.

[2] S. Brin and L. Page, "Reprint of: The anatomy of a large-scale hypertextual web search engine," *Computer networks*, vol. 56, no. 18, pp. 3825–3833, 2012.

[3] J. Allan, B. Croft, A. Moffat, and M. Sanderson, "Frontiers, challenges, and opportunities for information retrieval: Report from swirl 2012 the second strategic workshop on information retrieval in lorne," in *ACM SIGIR Forum*, vol. 46, no. 1. ACM, 2012, pp. 2–32.

[4] R. Gibson, "Online customer service," *The economic and social impacts of e-commerce*, p. 95, 2003.

[5] J. Chu-Carroll and B. Carpenter, "Vector-based natural language call routing," *Computational linguistics*, vol. 25, no. 3, pp. 361–388, 1999.

[6] N. J. Belkin, C. Cool, A. Stein, and U. Thiel, "Cases, scripts, and information-seeking strategies: On the design of interactive information retrieval systems," *Expert systems with applications*, vol. 9, no. 3, pp. 379–395, 1995.

[7] L. A. Barroso, J. Dean, and U. Hölzle, "Web search for a planet: The google cluster architecture," *Micro, Ieee*, vol. 23, no. 2, pp. 22–28, 2003.

[8] M. Thelwall and P. Sud, "Webometric research with the bing search api 2.0," *Journal of Informetrics*, vol. 6, no. 1, pp. 44–52, 2012.

[9] S. Zhao, H. Wang, and T. Liu, "Paraphrasing with search engine query logs," in *Proceedings of the 23rd International Conference on Computational Linguistics*. Association for Computational Linguistics, 2010, pp. 1317–1325.

[10] N. J. Belkin, R. N. Oddy, and H. M. Brooks, "Ask for information retrieval: Part i. background and theory," *Journal of documentation*, vol. 38, no. 2, pp. 61–71, 1982.

[11] T. Saracevic, "The stratified model of information retrieval interaction: Extension and applications," in *Proceedings of the Annual Meeting-American Society for Information Science*, vol. 34. LEARNED INFORMATION (EUROPE) LTD, 1997, pp. 313–327.

[12] I. Kopecek, "Modeling of the information retrieval dialogue systems," *Lecture notes in computer science*, pp. 302–307, 1999.

[13] E. Levin, S. Narayanan, R. Pieraccini, K. Biatov, E. Bocchieri, G. Di Fabbrizio, W. Eckert, S. Lee, A. Pokrovsky, M. G. Rahim *et al.*, "The at&t-darpa communicator mixed-initiative spoken dialog system." in *INTERSPEECH*, 2000, pp. 122–125.

[14] I. Androutsopoulos, G. D. Ritchie, and P. Thanisch, "Natural language interfaces to databases–an introduction," *Natural language engineering*, vol. 1, no. 01, pp. 29–81, 1995.

[15] M. Walker, D. Hindle, J. Fromer, G. Di Fabbrizio, and C. Mestel, "Evaluating competing agent strategies for a voice email agent," *arXiv preprint cmp-lg/9706019*, 1997.

[16] A. Loisel, G. D. Duplessis, N. Chaignaud, J.-P. Kotowicz, and A. Pauchet, "A conversational agent for information retrieval based on a study of human dialogues," in *International Conference on Agent and Artificial Intelligence*, 2012, pp. 312–317.

[17] M. McTear, *Spoken dialogue technology: Towards the conversational user interface*. Springer Science & Business Media, 2004.

[18] A. L. Gorin, G. Riccardi, and J. H. Wright, "How may i help you?" *Speech communication*, vol. 23, no. 1, pp. 113–127, 1997.

[19] C.-H. Lee, B. Carpenter, W. Chou, J. Chu-Carroll, W. Reichl, A. Saad, and Q. Zhou, "On natural language call routing," *Speech Communication*, vol. 31, no. 4, pp. 309–320, 2000.

[20] E. Levin, R. Pieraccini, and W. Eckert, "A stochastic model of human-machine interaction for learning dialog strategies," *Speech and Audio Processing, IEEE Transactions on*, vol. 8, no. 1, pp. 11–23, 2000.

[21] S. J. Young, "Probabilistic methods in spoken–dialogue systems," *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 358, no. 1769, pp. 1389–1402, 2000.

[22] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press Cambridge, 1998, vol. 1, no. 1.

[23] N. Roy, J. Pineau, and S. Thrun, "Spoken dialogue management using probabilistic reasoning," in *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 2000, pp. 93–100.

[24] J. D. Williams and S. Young, "Scaling pomdps for spoken dialog management," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, no. 7, pp. 2116–2129, 2007.

[25] J. Polifroni and M. Walker, "Learning database content for spoken dialogue system design," in *5th International Conference on Language Resources and Evaluation (LREC)*, 2006.

[26] J. Polifroni and M. A. Walker, "Intensional summaries as cooperative responses in dialogue: Automation and evaluation." in *ACL*, 2008, pp. 479–487.

[27] Z. Zhang, S. Schwartz, L. Wagner, and W. Miller, "A greedy algorithm for aligning dna sequences," *Journal of Computational biology*, vol. 7, no. 1-2, pp. 203–214, 2000.

[28] M. Maybury, "Intelligent user interfaces: an introduction," in *Proceedings of the 4th international conference on Intelligent user interfaces*. ACM, 1998, pp. 3–4.

[29] T. Paek, "Empirical methods for evaluating dialog systems," in *Proceedings of the workshop on Evaluation for Language and Dialogue Systems-Volume 9*. Association for Computational Linguistics, 2001, p. 2.

[30] M. A. Walker, D. J. Litman, C. A. Kamm, and A. Abella, "Paradise: A framework for evaluating spoken dialogue agents," in *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 1997, pp. 271–280.