Evaluating Coherence in Open Domain Conversational Systems

Ryuichiro Higashinaka[1], *Toyomi Meguro*[2], *Kenji Imamura*[1], *Hiroaki Sugiyama*[2], *Toshiro Makino*[1], *and Yoshihiro Matsuo*[1]

[1]NTT Media Intelligence Laboratories
[2]NTT Communication Science Laboratories

{higashinaka.ryuichiro, meguro.toyomi, imamura.kenji, sugiyama.hiroaki, makino.toshiro, matsuo.yoshihiro}@lab.ntt.co.jp

10.21437/Interspeech.2014-38

Abstract

We propose a method for evaluating coherence between user utterances and those generated from open domain conversational systems. Our aim is to make it possible for such systems to ascertain whether utterances generated from them are appropriate to the context before generation so that possible breakdown in conversation arising from inappropriate utterances can be avoided. In our method, we train a classifier that distinguishes a pair of a user utterance and that generated from a system as coherent or incoherent by using various pieces of information related to dialogue exchange, such as dialogue acts, question types, and predicate-argument structures. Experimental results show that our method significantly outperforms the baseline, confirming its effectiveness.

Index Terms: open domain conversation, dialogue systems, coherence

1. Introduction

Although task-oriented dialogue systems have been actively investigated [26], we have been seeing the emergence of open domain conversational systems or chat systems. Since such systems allow users to talk freely on a wide range of topics, they can be used to entertain users [22], build long-term relationships with users [3], or to enhance task-oriented dialogue systems by making them able to handle frequent out-of-domain utterances [25].

Building an open domain conversational system is difficult because user utterances are diverse. There are currently two major approaches for addressing this issue. One is to manually create a large number of rules to cover a wide variety of user utterances as much as possible; winners of the Loebner prize (a chatbot contest) are mostly based on such rules [27]. The other is to use the Web to retrieve sentences that can be used as responses [19, 2]. This is done in the hope that at least some sentences relevant to user input can be found on the Web. Although these approaches work well in some cases, since manually created rules have an obvious limitation of coverage and Web data are too diverse to control, current systems still produce inappropriate responses.

This paper proposes a method for evaluating coherence between user utterances and those generated from open domain conversational systems. The method allows a system to recognize that the utterance the system is about to generate is not coherent with the previous user utterance, making it possible for the system to change its decision and thereby avoid possible breakdown in conversation. This idea is similar to confidence scoring [4, 7], in which the reliability of system recognition results are evaluated. We apply a similar notion to system utterance candidates.

Although there have been many studies focusing on coherence in texts [8, 15, 1], there have been few studies on evaluating it in open domain conversation. Swanson and Gordon [24] proposed a system similar to ours that interactively generates to collaboratively create stories, but it does not focus on dialogue. In dialogue, utterances can be more diverse (ranging from interjections to long statements) and contain various dialogue acts, such as greetings, questions, expressing sympathy, and providing information. The aim with this paper is to provide a systematic way to evaluate coherence in open domain conversation to cope with such diverse linguistic phenomena. It should be noted that we focus on Japanese text conversation, although we plan to apply our method to other languages and to spoken dialogue in the future.

In the next section, we describe our proposed method. In Section 3, we explain the dialogue data we use for evaluation. In Section 4, we discuss the experiment we conducted to evaluate our method, and in Section 5, we summarize the paper and mention future work.

2. Proposed Method

In this paper, we focus on the coherence of two utterances; that is, local coherence or *cohesion*. Although findings in the work on adjacency pairs can be applied [20], this cannot be done straightforwardly. This is because conversational participants can say anything in open domain conversation and utterances that do not conform to typical adjacency pairs are not necessarily inappropriate. For example, after a question, there can be many ways to respond, such as asking a question in return, changing topics, talking about related topics, showing non-understanding, or just answering the question; they can all be appropriate. The problem is that it is difficult to determine under what condition a pair of utterances is coherent or incoherent.

Our solution is to mine such conditions using pattern-mining techniques. From a pair of utterances, we first create a *single* tree structure containing various types of information related to the two utterances. Having created such trees for many utterance pairs and labeled them as coherent or incoherent, we then mine the trees to obtain patterns (subtrees) that are useful for the classification of coherent/incoherent pairs. Finally, we use such patterns as conditions to classify unseen utterance pairs into *coherent* or *incoherent*.

Figure 1 shows our tree structure representing a pair of utterances. At the top, there is a root node. Beneath it, there are

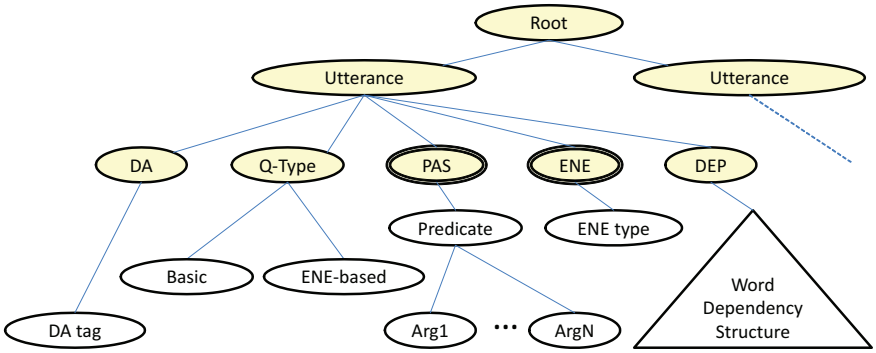

Figure 1: Tree structure representing pair of utterances. Top three layers are static nodes. Double circled nodes (i.e., predicate-argument structures (PASs) and extended named entities (ENEs)) can have multiple daughters if there are multiple PASs and ENEs within an utterance.

two nodes representing the two utterances. Under each utterance node, we have *five nodes*, under each of which there are different pieces of information about each utterance. The five nodes regard dialogue acts, question types, predicate-argument structures (PASs), extended named entities (ENEs), and dependency structures. They are called DA, Q-Type, PAS, ENE, and DEP nodes, respectively. We decided to use these pieces of information because they cover different levels of information, such as the intention and surface levels. In what follows, we explain how the tree is augmented with the above pieces of information and the tree mining process.

2.1. Dialogue-act estimation

The estimated dialogue-act tag for an utterance is added to the DA node as its daughter. The tag is estimated using our support vector machine based classifier. There are 33 dialogue acts in our tag set [16]. The classifier is trained with 1259 listening-oriented dialogues (LoDs) annotated with dialogue acts. Since speakers in an LoD are allowed to speak freely, the tag set covers diverse utterances, making it suitable for open domain conversation. Its estimation accuracy is 45%, which is reasonable when considering that the inter-annotator agreement rate is 59% [23].

2.2. Question-type estimation

The estimated basic and ENE-based question types for an utterance are added to the Q-Type node. The basic question type is based on the question taxonomy that Nagata et al. [17] derived from their analysis of questions posed to their question-answering system from the general public, which suggests that the taxonomy covers diverse questions in open domain conversation. The taxonomy has 13 question types. The ENE-based question type represents an entity requested by a question. We use Sekine's 200 ENE types [21] as our ENE-based question types for their wide coverage of named entities. The classifier for the basic question type was trained with 48K questions and has an accuracy of 92.5%. The classifier for the ENE-based question type was trained with 56K questions with an accuracy of 81.8%. Although it would be ideal to apply question-type estimation only to questions, currently, question-type estimation is applied to all utterances.

2.3. PAS analysis

The obtained PASs for an utterance are added to the PAS node. For this purpose, we use our statistical PAS analyzer [12, 10]. In PAS analysis, which is similar to semantic role labeling [18], predicates and their arguments are detected. A predicate can be a verb, adjective, or copular verb, and the arguments are noun phrases associated with cases in case grammar. We use seven cases, including *ga* (nominative), *wo* (accusative), and *ni* (dative). The PAS analyzer can perform anaphora resolution for zero pronouns. The analyzer has an accuracy of 57-62% for *ga*, *wo*, and *ni* cases. The PAS analysis provides the basic information about the content of an utterance as well as discourse information because zero pronoun resolution provides information about exophora (i.e., unresolved arguments).

2.4. ENE Extraction

The extracted ENEs from an utterance are added to the ENE node. We use an ENE recognizer we developed [6]. The recognizer uses conditional random fields [14] to label ENEs within an utterance. The recognizer has an F-measure of 84.03% for newspaper articles. We detect ENEs because when the previous utterance requests a certain ENE (by the ENE-based question type) and the current utterance has that ENE, coherence could be high.

2.5. Dependency Parsing

The word dependency structure tree for an utterance is appended to the DEP node. We use JDEP [11], a dependency parser, to parse an utterance and create its word dependency tree. For the purpose of abstraction, following [5], we use part-of-speech tags as main nodes with word surfaces, base forms, end forms, and semantic categories as their daughters. The semantic categories are similar to synset IDs in WordNet and can be obtained by consulting the Japanese thesaurus Goi-Taikei [9]. See [5] for an example tree structure.

2.6. Pattern Mining and Classification by BACT

For pattern mining, we use the Boosting-based Algorithm for Classifying Trees (BACT) [13]. The BACT enumerates subtrees in the input data (consisting of positive and negative examples) and uses the existence of the subtrees as features for boosting-based classification. Since the BACT mines patterns and the mined patterns are used for classification, this is the ideal algorithm for our purpose. We use the implementation by Kudo (http://chasen.org/˜taku/software/bact/).

Table 1: Statistics of collected dialogues. USR, SYS, and ALL denote user, system, and both, respectively.

System	# Utt	# Word	# Uniq Word	# Word/Utt
Rule USR	1045	6351	986	6.08
SYS	1165	8981	830	7.71
ALL	2210	15332	1381	6.94
Retrieval USR	1106	5473	1049	4.95
SYS	1226	7907	2055	6.45
ALL	2332	13380	2457	5.74

3. Data

For our method to work, we need utterance pairs labeled coherent and incoherent. To this end, we collected dialogue data using automated dialogue systems and annotated the data with coherence labels.

3.1. Systems

We prepared two systems. One is a rule-based system and the other a retrieval-based one. They are the two opposite ends of current standard approaches to open domain conversation, making it possible for us to collect varied data.

3.1.1. Rule-based System

Since there is no off-the-shelf rule-based system in Japanese, we created one. For this purpose, we had a seasoned engineer, who specializes in text analysis, create rules in Artificial Intelligence Markup Language (AIML) for two-and-a-half months. He first created initial rules by referring to the AIML rules of A.L.I.C.E. [27]. He then referred to the dialogue data we separately collected to augment the rules. The dialogue data include open domain conversations between humans, utterance-pairs about certain topics, and question-answer pairs for a persona. In all, he created 149K rules. Our rule-based system uses these rules by loading them with ProgramD (http://aitools.org/Program_D), an AIML interpreter.

3.1.2. Retrieval-based System

We implemented IR-Status by Ritter et al., [19], which is a retrieval-based system that uses Twitter. We first collected about 919M tweets. By extracting tweets connected with an in-reply-to relationship, we then created a Twitter conversation corpus (20M conversations containing 90M tweets). From this corpus, we then created a database for retrieval by extracting two consecutive utterances as input-output pairs and indexing them using the text search engine Lucene (http://lucene.apache.org/core/). For a given utterance as a query, the system retrieves the top-ten utterance pairs on the basis of the similarity (cosine similarity of TF-IDF weighted word vectors) between the query and the input-part of the indexed pairs. Then, one of the retrieved pairs is randomly selected for the system's utterance. Random selection is performed to avoid repeated utterances for the same input.

3.2. Data Collection

We recruited 30 participants to use the systems to collect dialogue data. They were paid for their participation. Each participant used each system four times in a randomized order. Each dialogue lasted two minutes. There was no instruction given about what to say to the systems. They were allowed to say whatever they wanted to say and to enjoy the conversation. We obtained 240 dialogues (120 dialogues for each system). Table 1 lists the statistics of these collected dialogues. Because of the diversity of Twitter, the retrieval-based system contains many unique words.

Table 2: Accuracy and relative accuracy (relative to human accuracy of 0.731; See Section 3.3) when one of the five nodes was enabled. '**' indicates statistical significance over baseline from McNemar's test ($p < 0.01$). Highest accuracies are in bold font.

Nodes	Acc.	Acc. (Relative)
Baseline (majority)	0.527	0.721
Only DA	0.601^{**}	0.822
Only Q-Type	0.572^{**}	0.782
Only PAS	0.587^{**}	0.803
Only ENE	0.521	0.713
Only DEP	$\mathbf{0.635}^{**}$	**0.869**

Table 3: Results of ablation tests. '**' indicates that drop in accuracy was statistically significant from McNemar's test ($p < 0.01$). '+' indicates statistical tendency ($p = 0.05$).

Nodes	Acc.	Acc. (Relative)
Proposed (w/o ENE)	**0.661**	**0.904**
w/o ENE, DA	0.650	0.889
w/o ENE, Q-Type	0.645^{+}	0.882
w/o ENE, PAS	0.649	0.888
w/o ENE, DEP	0.617^{**}	0.844

3.3. Coherence Annotation

We had two independent annotators label utterance pairs in our collected data. Since our aim was to detect inappropriate system utterances, we selected pairs in which the first utterance was that of a user and the second utterance was that generated from a system. We derived 2027 utterance pairs. The number was *not* 2391 (1165+1226 from Table 1) because the systems' first generated utterances (initial prompts) were not taken into account and we only used unique pairs for annotation.

The two annotators gave a coherent or an incoherent label to each utterance pair. We instructed them to assess the connectivity of an utterance pair and label it as coherent when the pair was connected to the level where they could easily come up with the next utterance. Otherwise, it was labeled as incoherent. Although there may be contextual cues and referring expressions in the utterance pairs, the annotators did not utilize the context for annotation. This was done to limit our problem to assessing surface-level cohesion.

The inter-annotator agreement in Cohen's κ was 0.468. Since the value is over 0.4, moderate agreement, we regard this annotation as a feasible task. We used the data of one of the annotators as our gold standard; the number of coherent and incoherent utterance pairs was 959 and 1068, respectively. When we calculated how one annotator's labels matched those of the other, the match rate was 0.731. We used this value as an upper bound because it represents human level accuracy.

4. Experiment

Using our method (See Section 2), we created trees for the 2027 utterance pairs. We then applied the BACT to obtain a classi-

Table 4: Top-ten subtrees mined by the BACT.

	Weight	Subtree
1	0.020	(Utterance (PAS (Predicate_*iu*[say])) (PAS))
2	0.019	(Root (Utterance (DA ([S-Disc Pref. Pos.]))) (Utterance (DA ([Q-Plan]))))
3	0.018	(Root (Utterance) (Utterance (Q-Type (ENE_Dish))))
4	0.018	(Utterance (Q-Type (Basic_Yes-No)) (DEP ([Sentence final particle] ([Adjectival suffix (end form)] (*i*[Adjectival end form]) ([Adjective stem]))))
5	0.016	(Root (Utterance (DEP ([Conjugational suffix (end form)]))) (Utterance (Q-Type (Basic_Yes-No)) (DEP ([Sentence final particle]))))
6	0.016	(Root (Utterance (Q-Type (ENE_Person))) (Utterance (Q-Type (ENE_Person))))
7	0.016	(Root (Utterance (DA ([Q-Fact]))) (Utterance (DA ([S-Disc Fact]))))
8	0.016	(Utterance (Q-Type (Basic_Yes-No)) (PAS) (DEP ([Exclamation mark] (*!*) (*!*))))
9	0.016	([Verb stem] ([Semantic category N-1906]) ([Case marker particle]))
10	0.014	(DEP ([Exclamation mark] (*!*) (*!*) ([Sentence final particle])))

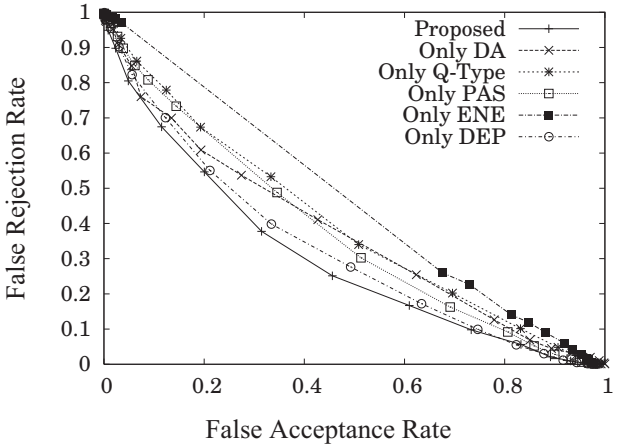

Figure 2: False acceptance rate and false rejection rate curves for proposed method (w/o ENE) and those that used one of the five nodes.

fication model. A model in this context means a set of mined subtrees with their weights for classification. We set the maximum number of nodes within a subtree to nine to reduce computational cost. The number of boosting iterations was set to 1000. We used ten-fold cross validation for evaluation and used accuracy (the number of correctly classified samples over all samples) as our evaluation measure.

4.1. Results

Table 2 lists the results when only one of the five nodes (i.e., DA, Q-Type, PAS, ENE, and DEP nodes in Table 1) was used. The aim was to determine how each piece of information contributes to performance. We use a majority baseline that always classifies an utterance pair as incoherent. As the table shows, each node contributed significantly to classification accuracy except for ENE. It is clear that DEP contributed greatly to performance. This indicates that there are many utterance pairs that can be classified on the basis of surface-level information. This may be because we used system utterances from Twitter, which are quite noisy and can be classified as incoherent on the basis of their not-well-formed structure and unknown words.

Table 3 lists the results when all the nodes (except for ENE, which was found ineffective and actually decreased performance) were used. The best accuracy was achieved when we used all the nodes; we obtained the best accuracy of 0.661, with relative accuracy of 0.904. The table also lists the results of an ablation test, showing how performance changed when one of the nodes was removed. Since the removal of any node leads to performance degradation, we can say that using a combination of information is effective. When DEP was not used, performance dropped significantly, suggesting again the importance of DEP. Note that the performance of 'only DEP' in Table 2 was significantly worse than that of the proposed method (w/o ENE) (McNemar's test, $p < 0.01$), confirming the necessity of using multiple pieces of information. Figure 2 presents another view of the results, with curves showing the false acceptance rate and false rejection rate. The graph shows that the DA, Q-Type, and PAS contributed at the same level, indicating their usefulness for determining coherence. It is also clear that using only one of them cannot guarantee coherence.

Table 4 lists the top-ten subtrees mined using the BACT for the proposed method (w/o ENE). The subtrees were those found for one of the folds in our ten-fold cross validation. The table shows that the most useful subtree is about a PAS containing a predicate *iu* ("say" or "talk" in English) followed by another PAS, which corresponds to the phrase "X *to ieba* (when we talk about X)". This suggests that the continuation or expansion on the current topic is a good indication of coherence. The second useful subtree is about a pair of dialogue acts. This subtree indicates that after self-disclosure about a user's positive preference (e.g., "I like movies"), the system's question about future plans (e.g., "Are you going to the movies today?") is useful. Although we cannot describe all subtrees in the table in detail, the table shows that various pieces of information have to be taken into account to judge coherence in open domain conversation.

5. Summary and Future Work

We proposed a method for evaluating coherence between user utterances and those generated by open domain conversational systems. We used various pieces of information related to utterance pairs, such as dialogue acts, question types, and predicate-argument structures, to train a classifier that classifies the pairs as coherent or incoherent. Experimental results show that our proposed method outperforms the baseline, confirming its effectiveness. It was also confirmed that using a combination of various pieces of information is important for evaluating coherence in open domain conversation. For future work, we plan to apply our method to a longer sequence of utterances and integrate our method into rule-based and retrieval-based systems to make their dialogues more coherent. Other methods for evaluating coherence need to be tested, for example, entity grids [1] and co-reference structures [24]. Last but not least, we also plan to apply our method to other types of dialogue, spoken data, and other modalities.

6. References

[1] R. Barzilay and M. Lapata, "Modeling local coherence: An entity-based approach," *Computational Linguistics*, vol. 34, no. 1, pp. 1–34, 2008.

[2] F. Bessho, T. Harada, and Y. Kuniyoshi, "Dialog system using real-time crowdsourcing and Twitter large-scale corpus," in *Proc. SIGDIAL*, 2012, pp. 227–231.

[3] T. W. Bickmore and R. W. Picard, "Establishing and maintaining long-term human-computer relationships," *ACM Transactions on Computer-Human Interaction*, vol. 12, no. 2, pp. 293–327, 2005.

[4] T. J. Hazen, S. Seneff, and J. Polifroni, "Recognition confidence scoring and its use in speech understanding systems," *Computer Speech & Language*, vol. 16, no. 1, pp. 49–67, 2002.

[5] R. Higashinaka, N. Kobayashi, T. Hirano, C. Miyazaki, T. Meguro, T. Makino, and Y. Matsuo, "Syntactic filtering and content-based retrieval of Twitter sentences for the generation of system utterances in dialogue systems," in *Proc. IWSDS*, 2014, pp. 113–123.

[6] R. Higashinaka, K. Sadamitsu, K. Saito, and N. Kobayashi, "Question answering technology for pinpointing answers to a wide range of questions," *NTT Technical Review*, vol. 11, no. 7, 2013.

[7] R. Higashinaka, K. Sudoh, and M. Nakano, "Incorporating discourse features into confidence scoring of intention recognition results in spoken dialogue systems," *Speech Communication*, vol. 48, no. 3, pp. 417–436, 2006.

[8] E. H. Hovy, *Approaches to the planning of coherent text*. Springer, 1991.

[9] S. Ikehara, M. Miyazaki, S. Shirai, A. Yokoo, H. Nakaiwa, K. Ogura, Y. Ooyama, and Y. Hayashi, "Goi-Taikei—A Japanese lexicon," 1997.

[10] K. Imamura, R. Higashinaka, and T. Izumi, "Predicate-argument structure analysis with zero-anaphora resolution for dialogue systems," in *Proc. COLING*, 2014.

[11] K. Imamura, G. Kikui, and N. Yasuda, "Japanese dependency parsing using sequential labeling for semi-spoken language," in *Proc. ACL*, 2007, pp. 225–228.

[12] K. Imamura, K. Saito, and T. Izumi, "Discriminative approach to predicate-argument structure analysis with zero-anaphora resolution," in *Proc. ACL-IJCNLP (Short Papers)*, 2009, pp. 85–88.

[13] T. Kudo and Y. Matsumoto, "A boosting algorithm for classification of semi-structured text," in *Proc. EMNLP*, 2004, pp. 301–308.

[14] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. ICML*, 2001, pp. 282–289.

[15] M. Lapata, "Probabilistic text structuring: Experiments with sentence ordering," in *Proc. ACL*, vol. 1, 2003, pp. 545–552.

[16] T. Meguro, Y. Minami, R. Higashinaka, and K. Dohsaka, "Learning to control listening-oriented dialogue using partially observable Markov decision processes," *ACM Transactions on Speech and Language Processing (TSLP)*, vol. 10, no. 4, p. 15, 2013.

[17] M. Nagata, K. Saito, and Y. Matsuo, "Japanese natural language search system: Web Answers," in *Proc. Annual Meeting of the Association for Natural Language Processing*, 2006, pp. 320–323, (In Japanese).

[18] M. Palmer, D. Gildea, and N. Xue, "Semantic role labeling," *Synthesis Lectures on Human Language Technologies*, vol. 3, no. 1, pp. 1–103, 2010.

[19] A. Ritter, C. Cherry, and W. B. Dolan, "Data-driven response generation in social media," in *Proc. EMNLP*, 2011, pp. 583–593.

[20] E. A. Schegloff and H. Sacks, "Opening up closings," *Semiotica*, vol. 8, no. 4, pp. 289–327, 1973.

[21] S. Sekine, K. Sudo, and C. Nobata, "Extended named entity hierarchy," in *Proc. LREC*, 2002.

[22] C. Sidner, T. Bickmore, C. Rich, B. Barry, L. Ring, M. Behrooz, and M. Shayganfar, "Demonstration of an always-on companion for isolated older adults," in *Proc. SIGDIAL*, 2013, pp. 148–150.

[23] H. Sugiyama, T. Meguro, R. Higashinaka, and Y. Minami, "Open-domain utterance generation for conversational dialogue systems using web-scale dependency structures," in *Proc. SIGDIAL*, 2013, pp. 334–338.

[24] R. Swanson and A. S. Gordon, "Say anything: Using textual case-based reasoning to enable open-domain interactive storytelling," *ACM Transactions on Interactive Intelligent Systems (TiiS)*, vol. 2, no. 3, p. 16, 2012.

[25] S. Takeuchi, T. Cincarek, H. Kawanami, H. Saruwatari, and K. Shikano, "Construction and optimization of a question and answer database for a real-environment speech-oriented guidance system," in *Proc. COCOSDA*, 2007.

[26] M. Walker, R. Passonneau, and J. E. Boland, "Quantitative and qualitative evaluation of DARPA Communicator spoken dialogue systems," in *Proc. ACL*, 2001, pp. 515–522.

[27] R. S. Wallace, *The Anatomy of A.L.I.C.E.* A.L.I.C.E. Artificial Intelligence Foundation, Inc., 2004.