Task Complexity measurement in the evaluation of Spoken Dialogue Systems

JAVIER VELAZQUEZ, M.C. and INGRID KIRSCHNING, PHD.
TLATOA Speech Processing Group, CENTIA,
Universidad de las Américas – Puebla,
Ex-Hda. Sta. Catarina Martir, Cholula Puebla,
MÉXICO

Abstract: - Spoken Dialogue Systems (SDS´s) is a technology that provides users with a speech based interaction with machines to carry out several activities in the pursuit of their goals (information retrieval, business transactions, automatic translation, etc.). Development of this kind of systems needs an evaluation process to show the real system performance, to identify implementation errors, etc. The resources to carry out this process are few, and for most of them their functional capabilities are constrained and not so general. This paper presents an approach for measure the Task Complexity on SDS. The description of the approach is carried out to show that this is one of main factors that should be considered in order to develop a consistent and general enough evaluation framework for SDS´s.

Key-Words: - Spoken Dialogue Systems, Evaluation Framework, Task Complexity.

1 Introduction

The Spoken Dialogue Systems are characterized by the fact that they allow users to perform at least part of their tasks through some form of spoken language dialogue [1]. Users can employ these systems for travel planning, business transactions, urban navigation support and other problem-solving tasks. SDS's are becoming systems of common use, although the development of most of these systems is still more like an art rather than a good engineering practice. Successful creation of an SDS depends on more than only a complete and graceful integration of several language technology components such as a speech recognizer with a good performance, or a robust parser. Development of these systems is an iterative process that involves stages of design, implementation and evaluation. But this is not the end of the process, stages of redesign and an implementation of corrections given by results of the evaluation should be included, this is in order to get a sustained progress. However, due to the lack of proper resources to carry out the evaluation stage, this progress is delayed.

In order to approach the previous problem, and as a part of a major effort in the creation of a new SDS´s evaluation framework which will be consistent enough and more general we have defined that Task Complexity, User Expertise and Dialogue Strategy are three important aspects that have to be considered.

In this paper we present a new approach for the treatment of the Task Complexity aspect. The paper is organized as follows: section 2 gives an overview of resources for the evaluation of SDS, mainly focused on task-based evaluations. Section 3 describes the treatment proposed for Task Complexity and the metric to evaluate this factor for SDS's. Conclusions and future work are presented in the last section.

2 SDS´s Evaluation Resources

People involved in the development of SDS are becoming aware that evaluation has demanded the legitimacy to become a research area by itself. It is necessary to count with the proper means to know if a usable system was created. In case that the system is not implemented yet, then it is evaluated to know how much advance has been achieved. Currently available resources for SDS´s evaluation, includes: theoretical distinctions in the SDS´s types of evaluation (adequacy, performance, diagnostic, etc.) given by [3], empirical methods to develop the evaluation as in [4], metrics [5, 6], architectures for evaluation like Galaxy II used in the DARPA Communicator project [7, 8], and several large test data suites as those described in [9].

In evaluation of interactive speech systems little is known about the interaction model evaluation, as well as the evaluation of dialogue components and

integrated interactive speech systems [2]. Models to evaluate speech recognition systems like those for database query, mainly focus on compare the system responses with a set of correct reference answers, but for these kinds of systems the answer reference set is closely related with a particular dialogue strategy, that means that for a dialogue evaluation and with the different styles of user interaction, it could be possible that there exist many different ways to answer, each of them equally correct and valid.

Dialogue strategy independent evaluation methods, develop analyses over user-SDS interaction log-files, focused more in measure the user repair interventions average when detection and correction of system errors occurs. Other methods rely on measure the contextual appropriateness and in the dialogue strategy ability to recover from partial/total misrecognitions, even misunderstandings or inappropriate utterance ratio.

Some methods use the task completion time for evaluation, this like a black-box evaluation metric, in others like [10] employ the transaction success reflected in a single coefficient (*kappa* coefficient in this case [11,12]) to observe how much accuracy the system showed in order to handle correctly the user's request. Last approach looks feasible since it is independent on dialogue strategies and speaker styles, but the evaluation set apart that some solutions might be better than others, depending on user preferences. In other words, while a particular decision criterion is the optimal for one user, for another it is not, affecting the dialogue development in the interaction and missing the quality of the last solution. Moreover, this approach does not consider the true complexity of the task, the comparison of the task success rate for systems developing the same or different tasks is not well founded, (some systems performs faulty when they have to face very difficult tasks, while others performs very well with easier tasks). Moreover it does not give any clue about why the task success rate was higher or lower, and it shows little about the behavior of the information elements present in the interaction.

2.1 Task-Based evaluation for SDS´s

Task-based evaluations for SDS's focus on assessing whether the speaker's task was achieved, rather than evaluating the impact of the task complexity level on the system performance. Current approaches on Task-based evaluations methodologies for SDS's are focused on using task completion as a black box evaluation metric. Task-based evaluations were focus on whether goals were communicated correctly, and if they were achieved at all. While methods for evaluation does not give any information about how many of the user arrangements over transactions have been conveyed, nor does take into account the complexity of the task, or the priority that they assign to their goals in order to achieve the task. Task-based evaluations approaches should expose the affect of task complexity on system performance by measures such as how long it takes to develop a solution using the system, and the quality of the final solution produced. Accuracy measures are very useful because they give important cues to asses the real component performance for a particular SDS but for the question of how well the system works and the real point to know if the system helps to achieve some task is not really fronted

3 Coping with *Complexity* in SDS's

Regarding the necessity for a perspective of SDS´s evaluation that considers this problem like one closely related to human factors and human computer interfaces, implies know how much efficient the interface is, this by means of assessing how much the interface let users be productive when they were trying to accomplish a well-defined task.

Accomplishment of a complex task with a SDS requires from developer to become aware of the number of necessary information elements and their relation and effect on the future states of the interaction with respect to the user goals. Thus when a user needs to make a decision, he bases it on the available data. Dependence between the task complexity and the amount of information is perceived, where as the number of necessary elements of information increases, more complex the task is. Furthermore, the search process into the representation space of information will be complicated too.

Just taking into account the task completion time, we can distinguish how faster a particular interface design was among others. But focusing more in task is preferable to know how fast any interface design should be. To approach the last observation we appeal to use a measure derived from information theory, in order to establish a reasonable comparing reference point. Thereby is necessary determining a lower bound that reflects the minimal amount of information that the user has to provide in order to accomplish the task. If a particular interface design requires an input of information higher than this lower bound, user is doing unnecessary work and the interface design could be improved. Obviously if the design requires the same amount that the lower bound indicates, interface design doesn't need improving.

3.1 The *Task Complexity* measure

The accomplishment of a task is considered successful if the system was able to handle correctly the user requests, and is very significant to asses if a goal was reached and at what cost. The previous argument is fundamental due with two important aspects: One of them is to know how minimal information elements are required to complete this task and how many operations are necessary for the accomplishment of a domain task. Regarding this observations the creation of the appropriate metric obey to that one which captures the score of the number of operations involved for the achievement of information elements to achieve the user goals. Result of this metric show how much the system enables the users for complete their domain task with his ability to handle user requests correctly in a reasonable amount of time.

Beginning with a theoretic definition of information efficiency, "Efficiency" is calculated by the minimum amount of information necessary to accomplish the task (this minimal amount is independent from the system design), divided by the amount of information supplied by user. In this case efficiency goes from 0 until 1. When no information is needed by user and no information is needed to accomplish the task the efficiency is 1 (This formality is necessary to avoid the case of 0 divided by 0). Efficiency could be 0 when totally unnecessary information from user is needed. Here, due with the main intention for develop systems to help users to solve problems, unusually a particular system can achieve the dubious 0 value.

Taking into account the necessary number of elements that are needed to accomplish the task, we propose the use of a derived metric from entropy measure. Entropy can be interpreted as a measure of the size of the search space consisting of the possible values of a random variable and its associated probabilities. In order to describe how the measure is determined, are useful to provide a representation for the information search space, for this we can employ the Attribute Value Matrix (AVM) developed in [10] and used to represent the information elements (observed as a set of ordered pairs "attribute – value") that must be obtained by the system, corresponding with the information conveyed by users in order to complete a particular scenario. A scenario is a representation of a particular task which the users have to perform by interacting with the system (See Table 1).

The entropy H of a random variable X is described as:

$$H(X) = -\sum_{x \in c} p(x) \log_2 p(x) \qquad (1)$$

The log can be calculated at any base; we use log base 2 because result of this is measured in bits. The expression shows the probability of choosing an X information element from the others, in this case meeting a particular attribute for example. Here we can observe that added to met the exact attribute is necessary to consider the complexity for the exact match to the value for correspondent attribute too, for this reason the entropy measure should be also applied to the possible values set.

Now the expression for the Task complexity measure is expressed like:

$$TC = \frac{H(X) + H(Y)}{\sum user\ turns * IEapt.} \qquad (2)$$

Where *TC* is the task complexity measurement, *H(X)* refers to the complexity for match the correspondent attribute, *H(Y)* to match the correspondent value for the attributes and *IEapt.* is the user conveyed average information elements by turn.

A representative example could be when a user complete a particular scenario from the table 1 with a system who assess a information element by turn, then *H(X)* is 2 equals *H(Y)*, assuming that the system obtains 1 information element by turn, then the system needs 4 turns to complete the scenario, the *TC* measurement showing the efficiency from the system coping with the present task complexity for filling the required ordered pairs is 1.

Attribute	Possible Values
Depart-city	Turin, Rome, Milan, Palermo
Arrival-city	Turin, Rome, Milan, Palermo
Depart-time	9 a.m., 11 a.m., 9 p.m., 11p.m.
Arrival-time	7 a.m., 9 a.m., 8 p.m., 10p.m.

Table1. Example of an Attribute Value Matrix (AVM) with the possible values for any instantiation of a scenario in a Italian travel scheduling.

Attempting to determine how this task complexity factors affects the success rate of an SDS, we see the following possible application options:

a) It can be considered as an inverse dependency, this means to observe that if the complexity augments, the success rate decreases or vice versa.

b) Another is that it can be considered as a distance measure between a real systems performance dealing with a certain complexity factor against an ideal with the same factor.

4 Conclusions

In this paper we presented some issues that should be considered in order to construct a more general evaluation framework for SDS´s. Currently our position is that we need to deal with the task complexity aspect properly to provide a more justified evaluation of a system performance. Here we presented that the task complexity degree can be assessed by using a entropy measure, this is because this type of coefficient provides more information on the amount of information elements needed to carry out the task, as well as the behavior of this data.

To be able to assess the complexity degree of a task, which directly affects the success rate for the accomplishment of the task, will provide important feedback to the developers on the reasons for a poor or good performance.

Additionally, the results of observing the relationship between Task Complexity and success rate will be useful to define the relation to the other aspects, i.e. User Expertise and Dialogue Strategy. It can be used to determine, among other things, the relation of the selection patterns of these information elements depending on the user preferences, useful to posterior correlation with those aspects mentioned earlier.

We strongly believe that the three factors, Task Complexity, User Expertise and Dialogue Strategy are interrelated and critical in evaluation of SDS´s. It is part of this project to define all three aspects and incorporate them into a Framework, so that it will be able to give more detailed information on a systems performance and can be applied in a more general way.

References:

[1] J. Glass, "Challenges for Spoken Dialogue Systems", *Proc. IEEE ASRU Workshop, Keystone CO.*, December 1999.

[2] Bernsen, N.O., Dybkjaer, H., Dybkjaer, L., "Designing Interactive Speech Systems", *Springer-Verlag* 1998, pp 4-5.

[3] L. Hirschman and H.S. Thompson, "Overview of evaluation in speech and natural language processing", *In* "Survey of the State of the Art in Human Language Technology", *Cambridge University Press and Giardini*, 1997.

[4] T. Paek, "Empirical Methods for Evaluating Dialog Systems", *In Proc. 2nd SIGdial Workshop on Discourse and Dialogue*, Aalborg Denmark, September 1- 2, 2001.

[5] J. Glass, J. Polifroni, S. Seneff, V. Zue, "Data collection and Performance Evaluation of Spoken Dialogue Systems: The MIT Experience", *Proc. 6th International Conference on Spoken Language Processing, Beijing*, China October 2000.

[6] K. Hacioglu, W. Ward, "A Figure of Merit for the Analysis of Spoken Dialog Systems", *In Proc. of ICSLP 2002*, Denver CO. September 2002.

[7] J. Polifroni, S. Seneff, "Galaxy-II as an Architecture for Spoken Dialogue Evaluation" *Proc. Second International Conference on Language Resources and Evaluation (LREC)*, Athens, Greece, May 31-June 2, 2000.

[8] M. A. Walker, and L. Hirschman, "Evaluation for DARPA communicator spoken dialogue systems". *In Proc. Second International Conference on Language Resources and Evaluation*, Athens, Greece. 2000.

[9] W. Minker, "Evaluation methodologies for Interactive Speech Systems", *In Proc. of International Conference on Language Resources and Evaluation (LREC)*, Granada Spain, pp. 199 – 206, May 1998.

[10] M. A. Walker, C. A. Kamm, D.J. Litman, A. Bella, "PARADISE: A framework for evaluating Spoken Dialogue Agents", *In proceedings of ACL/EACL 35th. Annual Meeting of the Association for Computational Linguistics*, ed. Morgan Kauffman, pp. 271 – 280, San Francisco 1997.

[11] Jean Carletta, "Assesing the reliability of subjective codings". *Computational Linguistics*, 1996 pages 136 –143.

[12] Sidney Siegel, N.J. Castellan, "Nonparametric Statistics for the Behavioral Sciences". McGraw Hill, 1988.