MPOWERS: a Multi Points Of VieW Evaluation Refinement Studio

Marianne Laurent, Philippe Bretier
Orange Labs
Lannion, France
{marianne.laurent, philippe.bretier}@orange-ftgroup.com

Abstract

We present our Multi Point Of vieW Evaluation Refinement Studio (MPOWERS), an application framework for Spoken Dialogue System evaluation that implements design conventions in a user-friendly interface. It ensures that all evaluator-users manipulate a unique shared corpus of data with a shared set of parameters to design and retrieve their evaluations. It therefore answers both the need for convergence among the evaluation practices and the consideration of several analytical points of view addressed by the evaluators involved in Spoken Dialogue System projects. After introducing the system architecture, we argue the solution's added value in supporting a both data-driven and goal-driven process. We conclude with future works and perspectives of improvement upheld by human processes.

1 Introduction

The evaluation of Spoken Dialogue Systems (SDS) is a twofold issue. On the one hand, the lack of convention on evaluation criteria and the many different evaluation needs and situations along with SDS projects lead to nomadic evaluation setups and interpretations. We inventoried seven job families contributing to these projects: the marketing people, the business managers, the technical and ergonomics experts, the hosting providers, the contracting owners as well as the actual human operators which integrate SDS in their activity (Laurent et al., 2010). Various experimental protocols for data collection and analytical data processing flourish in the domain. On the other hand, however they may not share evaluation needs and methods, the various potential evaluators need to cooperate inside and across projects. This claims for a convergence of evaluation practices toward standardized methodologies. The domain has put a lot of efforts toward the definition of commensurable metrics (Paek, 2007) for comparative evaluations and improved transparency over communications on systems' performances.

Nonetheless, we believe that no one-size-fits-all solution may cover all evaluation needs (Laurent and Bretier, 2010). We therefore work onto the *rationalization* - not the standardization - of evaluation practices. By rationalization, we refer to the definition of common norms to describe the evaluation protocols; common thinking models and vocabulary, for evaluators to make their procedures explicit. Our *Multi Points Of VieW Evaluation Refinement Studio* (MPOWERS) facilitates the design, from a unique corpus of parameters, of personalized evaluations adapted to the particular contexts. It does not compete with workbenches like MeMo (Möller et al., 2006) or WITcHCRafT (Schmitt et al., 2010) for which the overall evaluation process is predefined within the tool.

The following section details the solution architecture. Then, we present the MPOWERS's purposes, emphasizing on its added value for evaluators. Last, we explain the technical and process-related aspects that must support the system.

2 Architecture of the system

The application is built on a classical Business Intelligence (BI) solution that aims to provide decision makers with personalized information (See Fig. 1). We store, in a single datamart, parameters retrieved from heterogeneous sources: interaction logs, user questionnaires and third-party annotations relative to the evaluation campaigns arranged on the evaluated system(s). Then, data are cleaned, transformed and aggregated into Key Performance Indicators (KPIs). It guarantees that the indicators used across teams and projects are defined, calculated and maintained in the same place.

Proceedings of SIGDIAL 2010: the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue, pages 265–268,
The University of Tokyo, September 24-25, 2010.

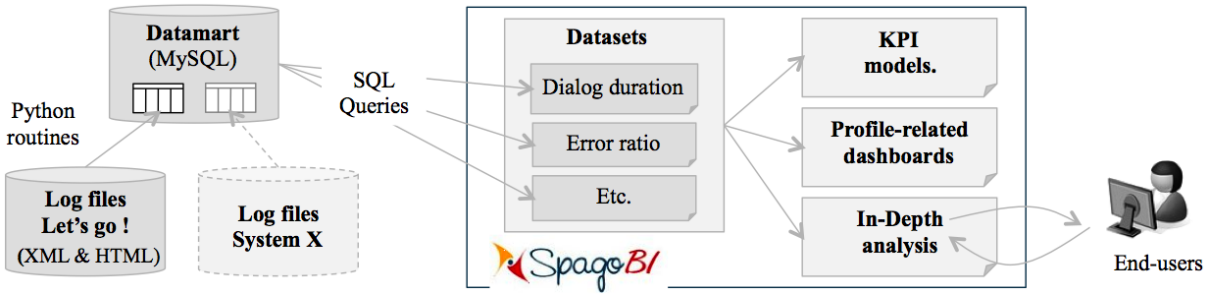

Figure 1: The MPOWERS architecture

On the upper layer, evaluators define and retrieve personalized reports and dashboards.

We use the Let's Go! System corpus shared by the Carnegie Mellon University. It contains log files generated since from 2003 from the Pittsburgh's telephone-based bus information system log files, one per module composing the system, and a summary HTML file. At our stage of the project the html summary allows the calculation of a satisfying number of parameters to support the system development and refinement. We compute the dialogue duration, the number of system and user turns, the number of barge-ins, the ratio between user and system turn number, the number of help requests and of no-matches per call and the ratio of successful interactions.

The application relies on the SpagoBI 2.6 open source solution[1]. Once parametrized, it enables non-technical stakeholders to retrieve personalized KPIs reports based on shared resources. For now, it delivers basic dashboards for two user profiles. One focuses on the service monitoring for marketing people and business managers and the other one provides the development team with usability-related performance figures (see fig. 4). The unique datamart guarantees all users to work from similar data. Its population requires parsing routines to identify and extract the relevant data.

3 Evaluation process and added value

By automating tractable tasks, MPOWERS supports the evaluator-users in their evaluation process driven by decision-making objectives. As sketched in figure 2, our application-supported process is slightly modified from the one defined by Stufflebeam (1980): a process through which one defines, obtains and delivers useful pieces of information that enable to settle between the alternative possible decisions.

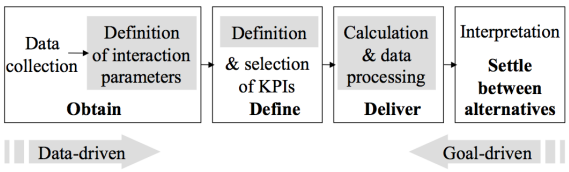

Figure 2: Evaluation process with MPOWERS (grey-tinted stages are supported by the system)

Custom-made Python[2] routines enable to extract relevant data from the log files. They provide CSV[3] formatted files to be converted into SQL scripts. The datamart is designed to be gradually populated from successive evaluation campaigns on one or several SDS. As data may originates from diverse sources, it arrays in different formats and often displays different parameters. Adapted ad hoc routines permit the manipulation into consistent format. We anticipate the use of separate tables in the datamart from comparative evaluations ons distinct systems.

The retrieval of KPIs in SpagoBI requires *datasets* pre-parametrized over SQL-Queries. They describe the SDS's performance and behaviour. We defined the parameters relative to the system performance according to the ITU-T Rec. P.Sup24 (2005). Yet, unless input corpora are defined accordingly not all the recommendation's parameters can be implemented. Three modes to display these datasets are proposed to evaluators:

- *A summary of high-level KPIs* provides a general view on the evaluated system with "red-light indicators" (see fig. 3). Links to more detailed charts or analysis tools are displayed next to each of them.

[1] http://www.spagoworld.org/

[2] http://www.python.org/

[3] Comma-Separated Value

Figure 3: High-level KPIs with link to more detailed documents. Please note that the success ratio is calculated via an ad-hoc query and does not necessarily corresponds to the user being or not satisfied.

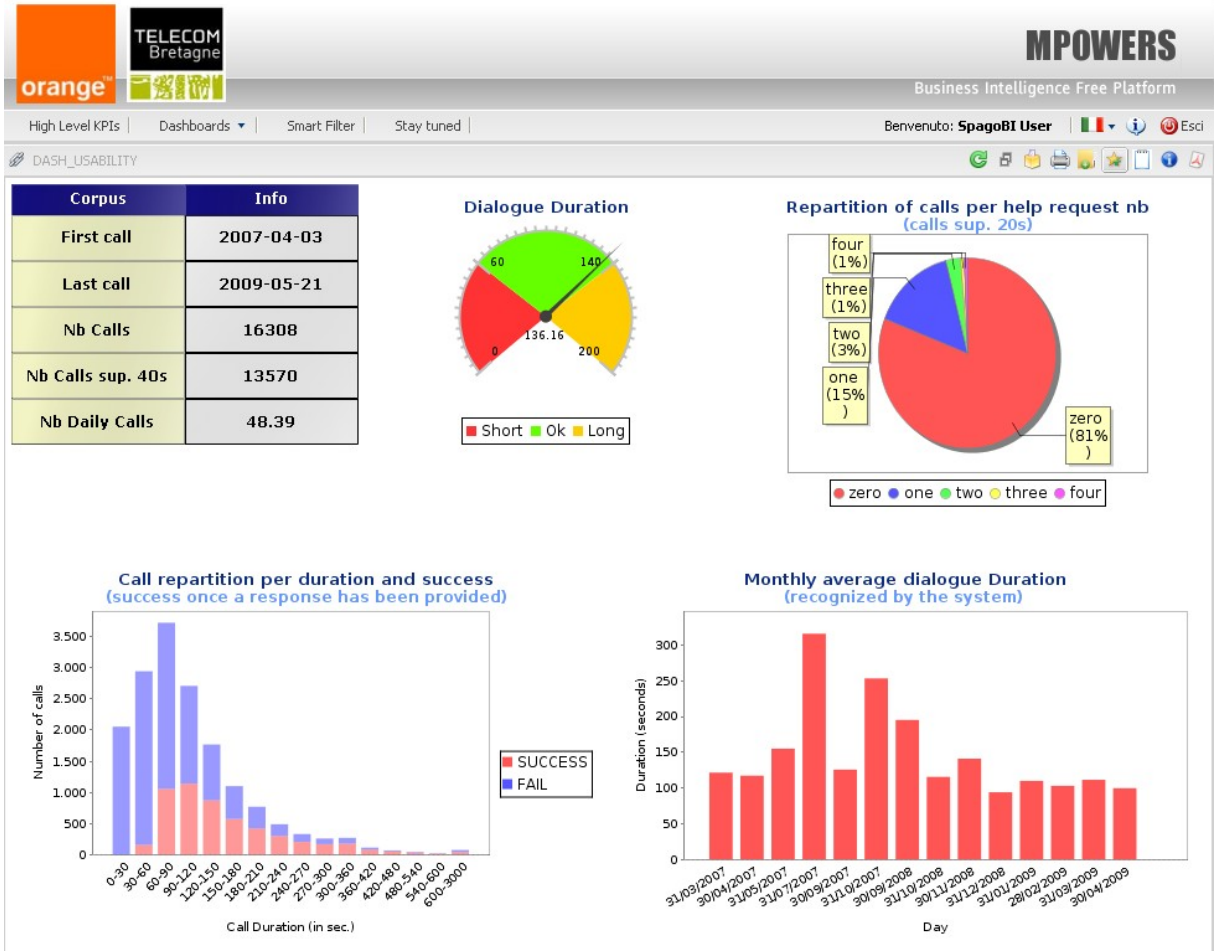

Figure 4: Dashboard dedicated to a high-level view on usability performance.

- *Visual dashboards* display pre-processed data according to pre-defined evaluation profiles (see fig. 4).
- *Tools for in-depth individual analysis* Filtered queries permit evaluators to individually adjust their analysis according to local evaluation objectives. Queries can be stored for later use or saved in PDF documents for distribution to non-MPOWERS users.

End-users, i.e. the evaluators, are limited to display the results and proceed to in-depth queries. An administrator access allows for prior data processing and the configuration of datasets, KPIs and dashboards. With collaborative enhancement purposes, the application supports communication between users with built-in discussion threads information feeds and shared to-do-lists to suggest and negotiate future configurations.

These distinct outlooks on the corpus are complementary. They combine a high-level view on the service's behaviour and performance with detailed personalised analysis. Whatever their layouts, every information displayed to the evaluators-users is retrieved from a unique corpus and from the same SQL-queries. Therefore, even if all evaluators consider distinct features on the evaluated service, our framework brings consistency to their evaluation practices.

4 Future work

MPOWERS is on its first development stages. Several perspectives of enhancement are planned. First, it requires to be augmented with more KPIs and in-depth analytical features. Second, as it only manipulates automated log files, user questionnaires and third-party annotations are expected to enrich its evaluation possibilities. Third, we intent MPOWERS to perform comparative evaluations between distinct services in the future. And last, the framework would benefit from being employed within real evaluators' daily activity.

5 Conclusion

The paper presents a platform that supports the SDS project stakeholders in their evaluation task. While advocating for a rationalization of evaluation practices among project teams and across organizations, it promotes the existence of different cohabiting points of view instead of disregarding them. When most evaluation contributions cover the overall evaluation process, from experimental data collection set-ups to guidance for interpretation, we limit to a user-centric framework, where evaluators remain in charge of the evaluation design. We actually provide them with an operational framework and unified tools to design and process their evaluations. This may help initiate individual, as well as community-wide, gradual refinements of methodologies.

Acknowledgments

The demo makes use of the *Let's Go!* log files provided by the Carnegie Mellon University. We thank Telecom Bretagne, Q. Jin, X. Chen, S. Zarrad, F. Agez and A. Bolze for their contribution in the platform deployment.

References

M. Eskenazi, A. W. Black, A. Raux, and B. Langner. 2008. Let's Go Lab: a platform for evaluation of spoken dialog systems with real world users. In *Interspeech 2008*, Brisbane, Australia.

M. Laurent and P. Bretier. 2010. Ad-hoc evaluations along the lifecycle of industrial spoken dialogue systems: heading to harmonisation? In *LREC 2010*, Malta.

M. Laurent, I. Kanellos, and P. Bretier. 2010. Considering the subjectivity to rationalise evaluation approaches: the example of Spoken Dialogue Systems. In *QoMEx'10*, Trondheim, Norway.

S. Möller, R. Englert, K.-P. Engelbrecht, V. Hafner, A. Jameson, A. Oulasvirta, A. Raake, and N. Reithinger. 2006. MeMo: towards automatic usability evaluation of spoken dialogue services by user error. *9th International Conference on Spoken Language*.

T. Paek. 2007. Toward evaluation that leads to best practices: reconciling dialog evaluation in research and industry. In *Workshop on Bridging the Gap: Academic and Industrial Research in Dialog Technologies*, pages 40–47, New York. ACL, Rochester.

ITU-T Rec. P.Sup24. 2005. Parameters describing the interaction with spoken dialogue systems.

A. Schmitt, G. Bertrand, T. Heinroth, W. Minker, and J. Liscombe. 2010. Witchcraft: A workbench for intelligent exploration of human computer conversations. In *LREC 2010*, Malta.

D. L. Stufflebeam. 1980. *L'évaluation en éducation et la prise de décision*. Ottawa.