

Developing Spoken Dialogue Systems with the OpenDial toolkit

Pierre Lison

Language Technology Group
University of Oslo (Norway)

plison@ifi.uio.no

Casey Kennington

Dialogue Systems Group, CITEC
Bielefeld University (Germany)

ckennington@cit-ec.uni-bielefeld.de

Abstract

We present OpenDial¹, an open-source toolkit for building and evaluating dialogue systems. The toolkit is centered on a dialogue state expressed as a Bayesian network and acting as a shared memory for the system modules. The domain models are specified via *probabilistic rules* encoded in a simple XML format. The toolkit has been deployed in several applications domains such as human–robot interaction and in-car driver assistants.

1 Introduction

Software frameworks for dialogue systems are often grouped in two categories. Symbolic frameworks rely on finite-state or logical methods to represent and update the dialogue state. While they provide fine-grained control over the dialogue flow, these approaches often assume that the state is fully observable and are thus poor at handling errors and uncertainty. Statistical frameworks, on the other hand, can automatically optimise the dialogue behaviour from data. However, they generally require large amounts of training data, making them difficult to apply in data-scarce domains.

The OpenDial toolkit adopts a hybrid approach that combines the benefits of logical and statistical methods into a single framework. The toolkit relies on *probabilistic rules* to represent the internal models of the domain in a compact and human-readable format. Crucially, the probabilistic rules may include unknown parameters that can be automatically estimated from data using supervised or reinforcement learning.

2 Architecture

The OpenDial toolkit relies on an information-state architecture in which all components work

together on a shared memory that represents the current dialogue state (see Fig. 1). This state is encoded as a Bayesian network, where each variable captures a particular aspect of the interaction.

The toolkit itself is domain-independent. To apply it to a given dialogue domain, the system developer provides an XML-encoded domain specification containing (1) the initial dialogue state and (2) a collection of probabilistic rules for the domain. The probabilistic rules are expressed as *if...then...else* constructions mapping logical conditions to probabilistic effects. The rule conditions are logical formulae using the standard operators and relations from predicate logic. Each condition is associated with a distribution over mutually exclusive effects, where each effect corresponds to an assignment of values to some state variable(s). The parameters of these distributions may be initially unknown, allowing them to be estimated from data via Bayesian learning.

At runtime, the probabilistic rules are instantiated in the Bayesian network representing the dialogue state. Standard algorithms for probabilistic inference are then employed to update the dialogue state and select new system actions. Lison (2015) provides more details about the formalisation of probabilistic rules, their instantiation and the estimation of their unknown parameters.

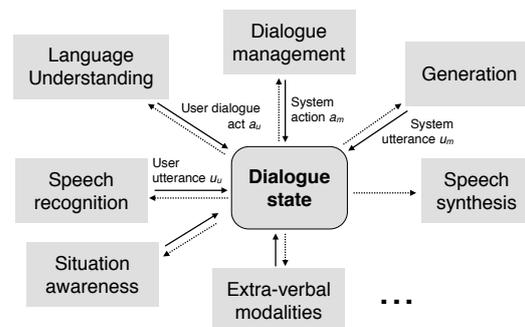
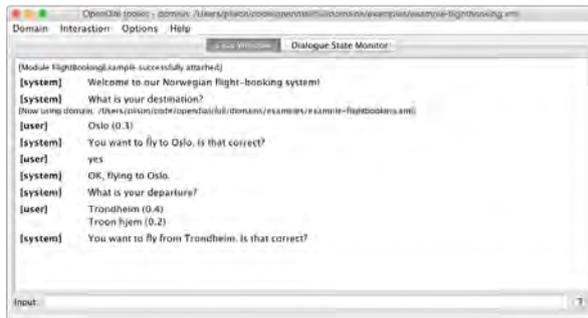
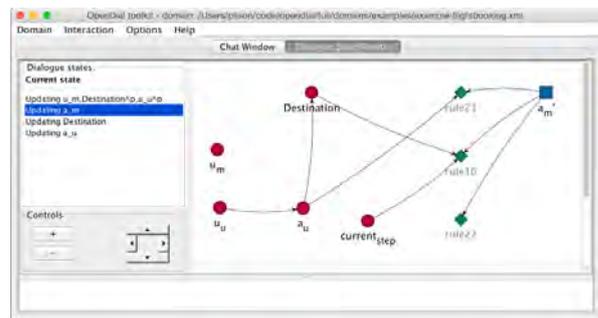


Figure 1: Schematic architecture for the toolkit.

¹<http://opendial-toolkit.net>.



(a) Chat window



(b) Dialogue state monitor

Figure 2: Graphical user interface for the toolkit.

3 Implementation and Applications

The toolkit is implemented in Java and released under an open-source license. The software comes with a graphical user interface allowing system developers to run a given dialogue domain and test its behaviour in an interactive manner (see Fig. 2). A collection of plugins extends the toolkit with external modules for e.g. speech recognition and synthesis or dependency parsing.

The toolkit has been deployed in several application domains, either as an end-to-end system or as a sub-component in a larger software architecture. One application domain is human-robot interaction. Lison (2015) illustrates the use of OpenDial in a domain where a Nao robot is instructed to navigate through a simple environment, fetch an object and bring it to a landmark. The probabilistic rules for the domain relied on parameters estimated from Wizard-of-Oz data. Similarly, Kennington et al. (2014a) describe how the toolkit was used as dialogue manager in a twenty-questions game between a robot and up to three human participants. The parameters of the dialogue policy were also estimated using Wizard-of-Oz data. This domain was not only multi-participant, but also multi-modal: the toolkit tracked the state of each individual participant’s speech, dialogue act, attention (e.g. towards the robot or towards another participant), and participation state (e.g. passive or active) and provided decisions on what to say and where to direct the robot’s attention.

The toolkit has also been deployed as dialogue manager in an in-car dialogue scenario (Kennington et al., 2014b). The system objective was to deliver upcoming calendar entries to the driver via speech. The toolkit was employed to track the state of the dialogue over time (using information from the driving simulator to make the system “sit-

uationally aware”) and decide when the system’s speech should be interrupted, for example when a dangerous driving situation (e.g. a lane change) was detected. This system was later enhanced to allow the driver to signal via speech or a head nod that the interrupted speech should resume.

4 Conclusion

The OpenDial toolkit rests on a hybrid framework combining ideas from both logical and probabilistic approaches to dialogue modelling. The dialogue state is represented as a Bayesian network and the domain models are specified using probabilistic rules. Unknown parameters can be estimated from dialogue data via Bayesian learning.

The toolkit is in our view particularly well-suited to handle dialogue domains that exhibits both a complex state-action space and partially observable environments. Due its hybrid modelling approach, the toolkit is able to capture such dialogue domains in a relatively small set of rules and associated parameters, allowing them to be tuned from modest amounts of training data, which is a critical requirement in many applications.

References

- C. Kennington, K. Funakoshi, Y. Takahashi, and M. Nakano. 2014a. Probabilistic multiparty dialogue management for a game master robot. In *Proceedings of the ACM/IEEE international conference on Human-robot interaction*, pages 200–201.
- C. Kennington, S. Kousidis, T. Baumann, H. Buschmeier, S. Kopp, and D. Schlangen. 2014b. Better Driving and Recall When In-car Information Presentation Uses Situationally-Aware Incremental Speech Output Generation. In *Proceedings of Automotive’UI*, Seattle, USA.
- P. Lison. 2015. A hybrid approach to dialogue management based on probabilistic rules. *Computer Speech & Language*, 34(1):232 – 255.