Hierarchical Dialogue Management

Francesca Giordaniello[1] (fg350@cam.ac.uk), Thomas Voice[2] (tvoice@apple.com), Milica Gašić[1] (mg436@cam.ac.uk)

[1]Cambridge University Engineering Department, [2]Apple

Spoken Dialogue Systems

Spoken Dialogue Systems (SDSs) offer an easy and intuitive way for the user-machine interaction. The user speech is interpreted through Spoken Language Understanding and mapped to an abstract representation u_t. The Dialogue Manager updates the belief state b of the system and selects an action a_t via a decision rule (*policy*) π, then converting the response into speech through Natural Language Generation (Figure 1).

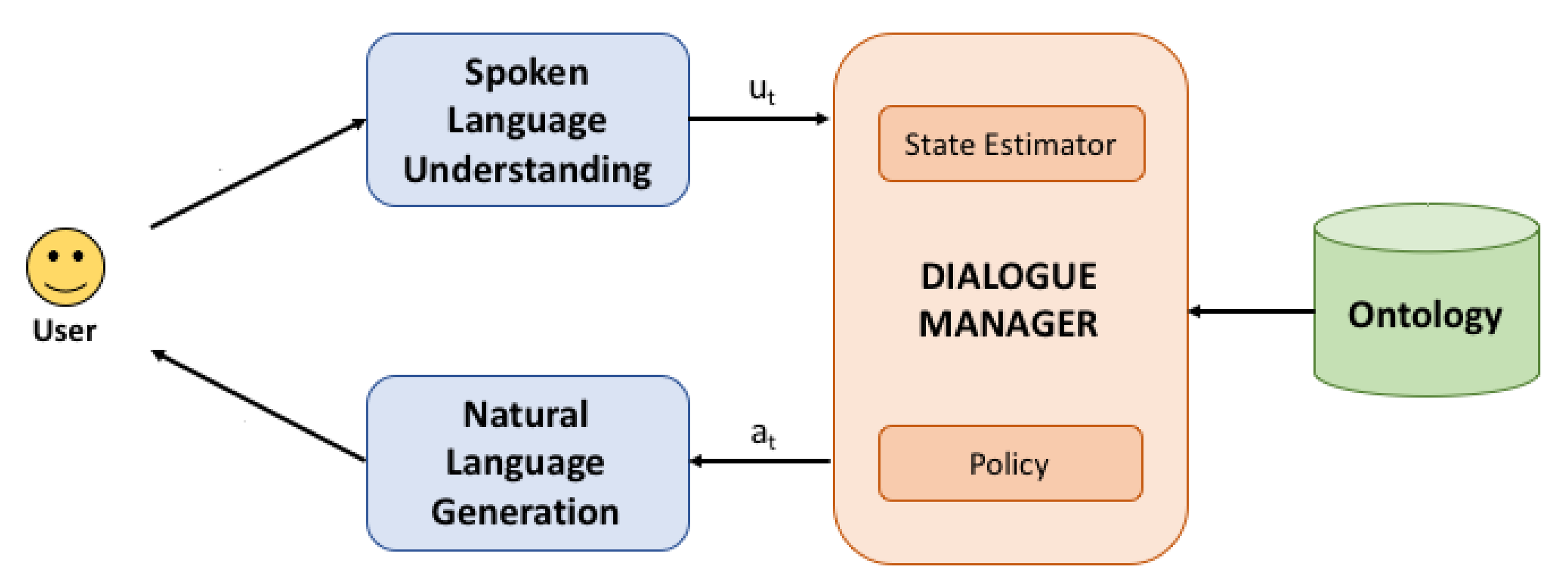

Figure 1: Components of a Spoken Dialogue System.

Policy Optimisation

At each turn, the policy chooses the action that maximises the *expected cumulative reward* $\bar{Q}$:

$$\pi(\boldsymbol{b}) = \text{argmax}_a\{\bar{Q}(\boldsymbol{b}, a) : a \in \mathcal{A}\} \tag{1}$$

The *GPSARSA* algorithm is used, which models the Q-function as a Gaussian Process (GP):

$$Q(\boldsymbol{b}, a) \sim \mathcal{GP}(m(\boldsymbol{b}, a), k((\boldsymbol{b}, a), (\boldsymbol{b}, a))) \tag{2}$$

Bayesian Committee Machine

The Bayesian Committee Machine (BCM) approach combines estimators trained on different datasets [1] (Figure 2), such as multiple estimates of the policy from different domains [2]. In general, it guarantees higher performance with respect to the correspondent in-domain policy (Figure 3).

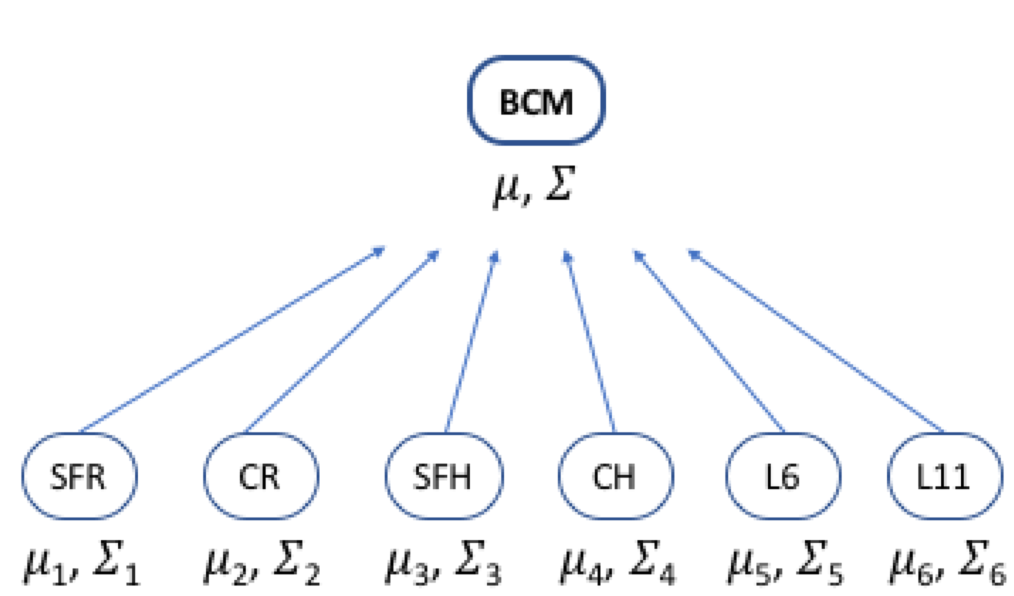

Figure 2: Configuration of the BCM.

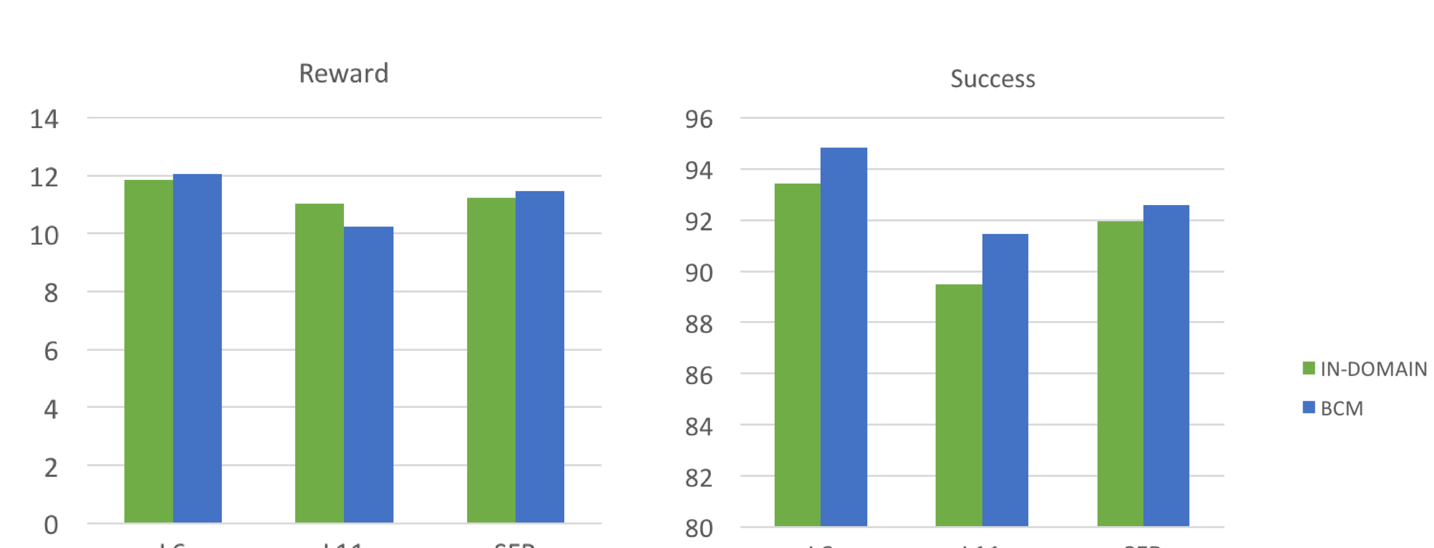

Figure 3: Performance of in-domain and BCM-based policies.

Hierarchical BCM

The same approach can be performed in a *hierarchical* fashion (Figure 4) by specifying a BCM for each of the n subsets of M domains, which compose the upper-level committee [3].

$$\begin{aligned}\bar{Q}_n(\boldsymbol{b}, a) &= \Sigma_n^Q(\boldsymbol{b}, a) \sum_{i=1}^{M} \Sigma_{n,i}^Q(\boldsymbol{b}, a)^{-1} \bar{Q}_{n,i}(\boldsymbol{b}, a) \\ \Sigma_n^Q(\boldsymbol{b}, a)^{-1} &= (1 - M) \cdot k((\boldsymbol{b}, a), (\boldsymbol{b}, a))^{-1} + \sum_{i=1}^{M} \Sigma_{n,i}^Q(\boldsymbol{b}, a)^{-1}\end{aligned} \tag{3}$$

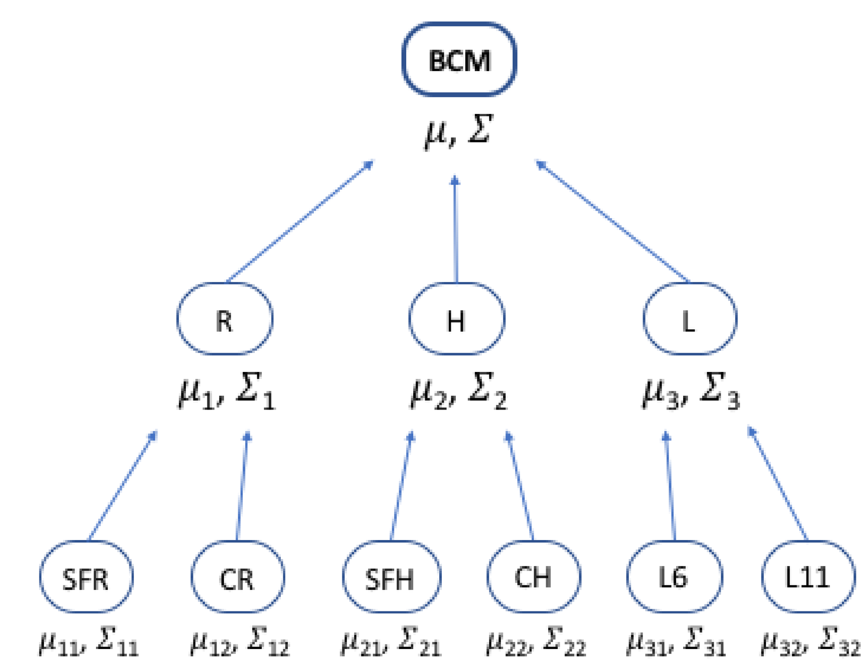

Figure 4: Configuration of the HBCM.

Preliminary Results

The hierarchical configuration allows more efficient scaling and guarantees a simpler parallelisation comparing to the BCM, especially when a larger number of domain is used. (Figure 5).

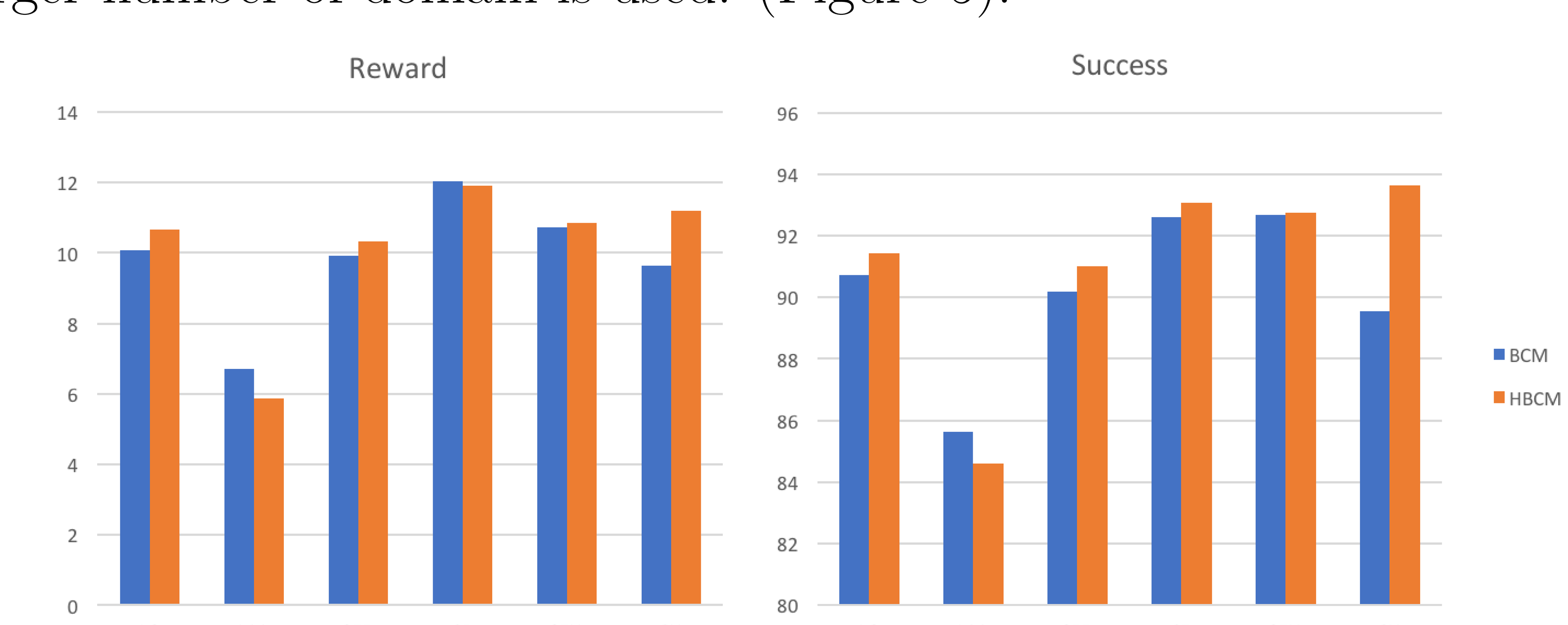

Figure 5: Comparison of the BCM and HBCM for a set of six domains.

Further Experiments

The *generalisation* capability of the policy optimisation algorithm in a HBCM setup could be highlighted by:

1. using a *larger domain database*
2. exploring different *hierarchical configurations*
3. evaluating the performance of a policy on an *unseen domain*

References

[1] Tresp, Volker. "A Bayesian committee machine." Neural computation 12.11 (2000): 2719-2741.

[2] Gašić, Milica, et al. "Dialogue manager domain adaptation using Gaussian process reinforcement learning." Computer Speech & Language 45 (2017): 552-569.

[3] Ng, Jun Wei, and Marc Peter Deisenroth. "Hierarchical mixture-of-experts model for large-scale Gaussian process regression." arXiv preprint arXiv:1412.3078 (2014).