Policy optimisation of POMDP-based dialogue managers without state-space compression

Milica Gašić, Matthew Henderson, Blaise Thomson, Pirros Tsiakoulis and Steve Young
Cambridge University Engineering Department

1 INTRODUCTION
POMDP-based approach to dialogue management
• Maintains a distribution over possible dialogue states – the belief state
• Enables robustness to speech recognition errors
• Policy optimisation requires:
  – Designer’s effort to specify a state-space compression
  – A large number of dialogues
Gaussian process-based optimisation
• Enables efficient learning by exploiting correlations in nearby parts of the space

2 BAYESIAN UPDATE OF DIALOGUE STATE DIALOGUE MANAGER
The dialogue state is factored into conditionally independent elements.

3 POLICY OPTIMISATION
Policy optimisation is performed via reinforcement learning from interaction with a simulated user.
Natural actor-critic algorithm
• Features of the belief space $B$ important for learning are extracted into summary space $C$
• Summary space $C$ and summary action space $A$ are mapped into feature space $F$
• Policy is parameterised as a linear function of features from $F$ passed through a soft-max nonlinearity

$$\pi(a|c, \theta) = \frac{e^{\theta^T c}}{\sum e^{\theta^T c'}}$$

• Gradient methods are used for policy optimisation
• Gaussian process SARSA algorithm
• The $Q$-function is the expected cumulative reward

$$Q^\pi(b, a) = E_T [\sum_{t=1}^{T} \gamma^{T-t} r_t | b_t = b, a_t = a]$$

• Optimising the $Q$-function $Q^\pi$ is equivalent to optimising the policy $\pi$
• The $Q$-function is modelled as a Gaussian process

$$Q^\pi(b, a) \sim \mathcal{GP}(b, k((b, a), (b', a')))$$

where $k((b, a), (b', a')) = k_b(b, b') k_a(a, a')$

4 KERNEL CHOICE
The optimisation can take place either in $C$ or $B$
• Kernel on the summary space $C$

$$k_C(c, c') = (c, c') + 1$$

• Kernel on the belief space $B$ is the sum of individual kernels on each of the hidden nodes

$$k_B(b, b') = \sum_{i} k_i(b_i, b'_i),$$

• Kernel on the summary action space $A$

$$k_A(a, a') = 1 - \delta(a')$$

5 EXPERIMENTS
Set-up
• TopTable restaurant information domain for Cambridge
• Contains 150 venues and each has 6 attributes
• Summary space $C$: 200 binary features
• Belief space $B$: 25 hidden nodes and each represents a distribution over 3 to 150 values
• Summary action space $A$: 16 summary actions

Training procedure
• Learning from interaction with the simulated user
• Learning curves:

6 CONCLUSION
• Possible to train policy directly on the full belief state
• Discrepancies between real users and simulated users
• Future work includes
  – Improved kernel function design
  – Removing the need for the summary action space

7 ACKNOWLEDGEMENTS
This work was supported by PARLANCE (www.parlance-project.eu), an EU Seventh Framework Programme project (grant number 287615).