CS786: Computational Cognitive Science Assignment 2

AMRIT SINGHAL
150092
amrits@iitk.ac.in

May 2, 2019

Note: Implemented in Python.

Solution 1

The question required us to model the Temporal context model of memory retrieval. For this, we first sample a world state $\mathbf{t}_i^{IN}$ from the currently updated Gaussian distribution. The Gaussian distributions initially have means $\boldsymbol{\mu}_0$. At each time step, the means are updated by δ,

$$\boldsymbol{\mu}_i = \boldsymbol{\mu}_{i-1} + \Delta \quad \forall i$$

where $\Delta = [\delta, \delta, \delta, \delta, \delta]^T$ was a fixed constant (for this question). Thus, parameter δ controls how much the world changes between two time steps. Higher value of δ corresponds to a higher amount of change.
At every time i, $\mathbf{t}_i^{IN}$ is sampled from a Gaussian distribution having mean $\boldsymbol{\mu}_i$ and having a fixed variance. This is then added to the existing context state $\mathbf{t}_{i-1}$ using parameter β, to obtain the new context state $\mathbf{t}_i$ using the equation:

$$\mathbf{t}_i = \rho \mathbf{t}_{i-1} + \beta \mathbf{t}_i^{IN}$$

where $\rho : \|t_i\| = 1$

$$\implies \rho = \sqrt{1 + \beta^2([\mathbf{t}_{i-1} \cdot \mathbf{t}_i^{IN}]^2 - 1)} - \beta(\mathbf{t}_{i-1} \cdot \mathbf{t}_i^{IN})$$

This state $\mathbf{t}_i$ is concatenated with the item f to get $(||\mathbf{t}_i|| + 1)$ sized vectors which are stored as the encoding e_i for that item. Note that the parameter β controls how much new context is added to the current context, and how much old context is retained. During retrieval, we obtain the updated context of that time $\mathbf{t}_{t'}$. The association score $a_{t',m}$ of an item m at the current time t' is obtained as

$$a_{t',m} = \sum_{i:f=m} \mathbf{t}_i \cdot \mathbf{t}_{t'}$$

Using this, we obtain the assocaition score for all items m, and that probability distribution is used to obtain the success score at each time t'.
This method was run for a 1000 times, which resulted in an **average success score of 7.45**.

Solution 2

Part (1)

For the second question, the updation of Gaussian means is not done by adding a fixed δ to it, but rather this δ is sampled from a bi-modal Gaussian mixture model.

$$p(\delta) = \pi_1 \mathcal{N}(\delta|\mu_s, \sigma_s) + \pi_2 \mathcal{N}(\delta|\mu_l, \sigma_l)$$

where, π_1, π_2 are the mixing weights. The paremeter values were set to be

Parameter	value
π_1	0.7
π_2	0.3
μ_s	0.01
μ_l	0.25
σ_s	0.01
σ_l	0.05

Part (2)

The next step was to obtain the encoding schedule that led to the minimum schedule loss, but with a acceptable (> 7) success score on average.

1. For this, the first schedule tried was the trivial one mentioned where we encode all at the start. This led to a very high schedule loss of 500.
2. The next schedule tried was the one where all the items were equi-spaced in time. This lead to a much lower scheduling loss of around 9.
3. The scheduling loss was obtained from 1000 random schedules, and the minimum schedule loss obtained was ~ 6.4. This tells that the optimal should be atleast lower than this.
4. Looking at the expression of the scheduling loss, we see that to minimize the loss, we need to maximise the median of the time interval between two items. Optimising on this, we obtain another scheduling loss (as obtained from the function `get_optimal_schedule()`. For this, the scheduling loss was obtained to be 5.05, which is very low.
5. Further, for the same interval differences in the schedule, there order needs to be fixed. Different orders lead to different success scores. The one chosen finally gave an average success score of ~ 8.16.

So, finally, the optimal encoding schedule was obtained as

$$[[1, 1], [100, 2], [199, 3], [298, 4], [397, 5], [496, 6], [497, 7], [498, 8], [499, 9], [500, 10]]$$

which lead to a **scheduling loss of 5.05** and an **average success score of 8.8**.

Solution 3

For this part, the optimal schedule remains the same as in part (2). Rather than the delta GMM parameters used in encoding phase being known during retrieval, the agent first need to obtain them using EM algorithm. So, we need to estimate the parameters $\pi_1, \pi_2, \mu_s, \mu_l, \sigma_s$ and σ_l from the values of delta that were observed during the encoding phase.
All of these parameters are learned using the EM algorithm, and these learned values are then used to proceed with the retrieval process. Running this methods for 1000 times, we get an **average success score of 8.77**.