用户奖励模型更新程序

Input Question x_k

k : step/question number in the sequence

v_k : reward received after answering x_k

v_{inc}^k: reward increment after receiving reward v_k

$\overline{v_k} = average\ (\{v_1,\ldots,v_{k-1}\})$

Current reward model R (with parameters ϕ_s, ϕ_t)

1. If feedback is score or engagement then
2. Reward model update w_s, w_t, ϕ_s, ϕ_t
3. Normalize the question and transition descriptor weights, such that they add up to 1:

$$\forall d \in F_s: \varphi_s^{d,l} = \frac{\varphi_s^{d,1}}{\sum_{i=1}^{nbins_s} \varphi_s^{d,i}}$$

$$\forall d \in F_s, \forall l, g=1,2,\ldots,nbins_s: \varphi_t^{d,l,g} = \frac{\varphi_t^{d,l,g}}{\sum_{i=1}^{nbins_s}\sum_{j=1}^{nbins_s} \varphi_s^{d,i,j}}$$

4. Else
5. Run initialization module
6. End if
7. Return: updated ϕ_s and ϕ_t

题目规划策略调整程序

Input Question bank, M

Planning horizon q

Current preference model R (with parameters ϕ_s and ϕ_t)

Currently presented question $question_0 \leftarrow x_k$

B: percent of questions from M to use in planning

1. Select a set of B percent of questions from M, M^*, with the highest reward R_s
2. BestTrajectory=null
3. HighestExpectedPayoff = $-\infty$
4. While computational power not exhausted do:
5. trajectory – []
6. for i = 1 to q do:
7. $question_i \leftarrow$ selected randomly from M^* (avoiding repetitions)
8. add question to trajectory
9. end for
10. expectedPayoffForTrajectory=$R_s(question_i)+\sum_{i=2}^{q}(R_t((question_1,\ldots,question_{i-1}),question_i)+R_s(question_i))$
11. if expectedPayoffForTrajectory> HighestExpectedPayoff then
12. HighestExpectedPayoff= ExpectedPayoffForTrajectory
13. BestTrajectory=trajectory
14. end if
15. end while
16. Return: First question in BestTrajectory

Input M= question bank

q= planning horizon

z_s= number of question for question preference initialization

z_t= number of question for transition preference initialization

$nbins_s$= number of percentile bin descriptors per question feature

$nbins_t$=number of percentile bin descriptors per transition feature

B: percent of top questions to use during planning

1. Obtain GCN-LSTM based engagement prediction
2. Initialization of question weights ϕ_s and transition weights ϕ_t (with $M,z_s,z_t,nbins_s,nbins_t$)
3. k=0
4. While user requesting another question do:
5. $k=k+1$
6. Select the next question: x_k: Run Algorithm2 (with M,q,R, current question playing $question_0 \leftarrow x_k$, B)
7. Obtain question score and engagement trend after x_k: reward v_k, and average reward thus far $\overline{v_k}$.
8. Update reward model R'; s parameters: ϕ_s, $\phi_t \leftarrow$Run Algorithm 1(with x_k, k, v_k, $\overline{v_k}$, ϕ_s, ϕ_t)
9. End while