Adapting to Drifting User Preferences in Recommendation

Introduction

Machine learning is often used to acquire knowledge in domains that undergo frequent changes. These frequent changes pose a challenge for adaptation issues. We formulate the recommendation problem as a sequence prediction problem and mainly studied the adaptation properties of using the non-parametric context tree method.

Context (Suffix) Tree Structure

Properties of Context Tree

Adaptation:
- parameter learning process and recommendations generated are online
- The structure of the CT helps new items or patterns to be recognized quickly in new contexts, whereas old items can still be accessed in their old contexts.

Efficiency:
- Recursive computation and update
- For trees of depth $D$, the time complexity of updating and recommending for an observation are both $O(D)$
- For input sequence of length $T$, the updating and recommending complexity are $O(M^2)$, where $M = \min(D, T)$
- Space complexity is $O(N)$, where $N$ is the total number of observations
- Pruning old items (branches)

Experimental Analysis

Sequential forum threads recommendation for three MOOCs:
- CT recommender outperforms other sequential methods.
- CT is strong at recommending fresh items (adaptive to new items).

Prediction Model and Update

Prediction Model:
- A local predictor (expert) to each context
- Dirichlet-multinomial distribution
  \[ P_x(n_{i+1} = x | s_i) = \frac{\alpha_x + \alpha_0}{\sum_{x \in \Sigma} \alpha_x + \alpha_0} \]
- Context activation [Red path above, $A(s_i)$]
- Mixture of predictions by activated experts
  \[ P(n_{i+1} = x | s_i) = \sum_{a \in A(s_i)} u(a(s_i)) P_x(n_{i+1} = x | s_i) \]
- Recursive Update:
  \[ q_k = w_k P_x(n_{i+1} = x | s_i) + (1 - w_k) q_{k-1} \]
  \[ w_k = \frac{w_k P_x(n_{i+1} = x | s_f)}{q_k(x)} \]