History-Guided Conversational Recommendation

Yasser Salem, Jun Hong, Weiru Liu

WI&C 2014
Overview

Conversational recommendation

Critiquing as a form of feedback

Reusing past critiquing sessions as experience cases

History-Guided Conversational Recommendation (HGR)

Evaluation
Introduction

Recommender Systems
Helping users to navigate complex product spaces, e.g. NetFlix (movies), Amazon (e-commerce), iTunes (Music)

Collaborative filtering vs. Content-based
Collaborative filtering (relying on ratings-based profiles) vs. content-based (relying on item descriptions)

Single-Shot vs. Conversational
Towards engaging the user in an interactive multi-cycle recommendation dialogue
Conversational Recommendation

Conversational Session
In each Cycle of conversation
- Recommend an Item
- Obtain User Feedback
- Update Query/Model
Critiquing

- User provides feedback in the form of feature-value critiques; e.g. cheaper (price < x)

In the standard form of critiquing, the user’s critiques provide an effective means to navigate through a complex product space on a feature-by-feature basis.
Compound Critiquing

- Multiple critiques at the same time
- Help to clarify the different trade-offs between features
Compound critiquing enables the user to navigate a product space more efficiently by allowing them to critique multiple features at the same time.
During the course of conversational recommendation, different users may have followed different paths in a product space, resulting some common sequences of recommendation-critique pairs.
Can we reuse these past common critiquing sessions in order to deliver more effective recommendation?
Each recommendation session is composed of a sequence of recommendation-critique pairs.

e.g. $p_i = \{\text{Item-27, (Price, $200, >)}\}$
History-Guided Conversational Recommendation (HGR)

- Capturing Experience Cases from Past Critiquing Sessions

- Identifying Relevant Experiences
  
  Compare current session’s steps to the steps in past sessions (experience cases) to identify maximally similar sessions.

- Ranking Recommendation Candidates
  
  Score and rank the experience cases and recommend most suitable final (terminal) case.
History-Guided Conversational Recommendation (HGR)

- Capturing Experience Cases from Past Critiquing Sessions
- Identifying Relevant Experience Cases
  Match the sequence of recommendation-critique pairs in the current session to the ones in past sessions (experience cases) to identify similar experience cases as relevant sessions.
- Ranking Recommendation Candidates
  Score and rank the experience cases and recommend the most suitable final (terminal) case.
History-Guided Conversational Recommendation (HGR)

- Capturing Experiences from Past Critiquing Sessions

- Identifying Relevant Experience Cases
  Match the sequence of recommendation-critique pairs in the current session to the ones in experience cases) to identify experience relevant sessions.

- Ranking Candidate Recommendations
  Score and rank the relevant sessions and select the final recommendation in the top-ranked session.
History-Guided Conversational Recommendation (HGR)
Identify Relevant Sessions

- Match current critique session to experience cases
- Compute Overlap

\[
\text{OverlapScore}(q_T, s_i) = \sum_{(r_i, c_i) \in q_T} \sum_{(r_j, c_j) \in s_i} \text{match}((r_i, c_i), (r_j, c_j))
\]

\[
\text{OverlapScore}(q_T, s_i) = \sum_{c_i \in q_T} \sum_{c_j \in s_i} \text{match}(c_i, c_j)
\]
Identify Relevant Sessions

- Match current critique session to experience cases

- Compute Overlap Score
  \[ \text{OverlapScore}(q_T, s_i) = \sum_{(r_i, c_i) \in q_T} \sum_{(r_j, c_j) \in s_i} \text{match}((r_i, c_i), (r_j, c_j)) \]

- Identify Relevant Sessions
  \[ S^{REL} = \text{RelevantSessions}(q_T, S) = \{ s_i \in S : \text{OverlapScore}(q_T, s_i) > t \} \]
Rank Relevant Sessions

- Rank relevant sessions based on overlap scores

\[
\text{RecScore}(r_F, q_T, S^{REL}) = \sum_{\{\forall s_i \in S^{REL} : r_F = \text{terminal}(s_i)\}} \text{OverlapScore}(q_T, s_i)
\]

- Select Next Recommendation
  - Terminal recommendation with the highest recommendation score
  - Compatible with the critiques so far in the current session
  - Usually not most similar to previous recommendation – allows for larger steps through a product space
Evaluation

- **Dataset**
  Recent crawl of 10,000 restaurants that were crawled from an online restaurant database. Each restaurant is represented by 39 different features (e.g. price, quality, etc.)

- **Artificial User Sessions Recorded as Experience Cases**
  Generated a total of a million distinct critiquing sessions based on the methodology of Reilly et al (2004)

- **Algorithms**
  HGR vs. IC and HAC
Methodology

- Test Problems
  Generated a test set of 500 unique ‘problems’ (initial-final cases).

- The Artificial user
  Each test problem is solved by simulating a ‘rational user’, using IC, HAC and HGR to navigate from the initial case to the final case.

- Criterion – average session length
  Compare average session length of IC, HAC and HGR algorithms.
The average session lengths of HGR compared to IC and HAC

(using 6,000 restaurants)
The average session lengths of HGR compared to IC and HAC

(using 8,000 restaurants)
The average session lengths of HGR compared to IC and HAC (using 10,000 restaurants)
Average Session Length

Percentage of reduction of session length of HGR over IC and HAC with different complexities of product spaces

Percentage Benefit

Number of restaurants

HGR over IC

HGR over HAC
Conclusions

■ History-Guided Conversational Recommendation
  Captures and reuses past critiquing sessions to improve the efficiency of future sessions

■ Positive Results
  Initial results demonstrate significant improvement on session lengths (subject to sizes of case bases/richness).