A Query Weighted-based Method for User Modeling

Hu Juan, Bai Yu, Cai Dongfeng

Knowledge Engineering Research Center
Shenyang Aerospace University
Outlines

- Background
- Query Weighted-based User Modeling
- Experiments and Results
Outlines

➢ Background
➢ Query Weighted-based User Modeling
➢ Experiments and Results
Background

- With the rapid growth of Internet-scale, information overload is an increasing problem for web users.
- Recommendation system is one of the most promising approaches to solve the problem of information overload.
- A personalized recommendation system can divide into three parts:
  
  User interest modeling
  Recommendation object modeling
  Recommendation algorithm
The recommendation system

- Object source by the object modeling methods obtain the object model
- User behavior by the user modeling methods generate the user model
- Combining the user model and object model to obtain recommender list, and then return to user
Background

- **User Modeling**: is a process of obtaining and maintaining the user interest, needs and habits, and generates user model that can reflect the users’ specific interest.

- The purpose of user modeling are:
  1. Mining user interests → Query Weighted-based user Modeling
  2. Representing user model → Set of keywords
Outlines

- Background
- Query Weighted-based User Modeling
- Experiments and Results
Query Weighted-based User Modeling

The framework of user modeling

- We preprocess the query log
- The second step is session division, and we obtain session set for each user
- For each user, we use the query weighted method to get the weight of each query in a session
- The last step is interest voting, and then we get user model
The preprocessing of query log:

- Splitting the query log by user, put the query log of same user together;

- Filtering the users by the number of query log is more than the threshold.
The framework of session division

The principles of session division:

1) The time interval of a session <= session time threshold
2) The time interval between adjacent queries in a session <= query time threshold
3) The cosine similarity between adjacent queries >= query similarity threshold

Based on: Mining user web search activity with layered bayesian network or how to capture a click in its context. (2009)
## Session Division

The session sample of an user:

<table>
<thead>
<tr>
<th>Session</th>
<th>Query</th>
<th>QueryTime</th>
<th>Rank</th>
<th>ClickURL</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Session1</strong></td>
<td>midway online literary journal</td>
<td>2006-04-21 11:24:43</td>
<td>3</td>
<td><a href="http://www.mndaily.com">http://www.mndaily.com</a></td>
</tr>
<tr>
<td></td>
<td>midway online literary journal</td>
<td>2006-04-21 11:24:44</td>
<td>9</td>
<td><a href="http://www.smallspiralnotebook.com">http://www.smallspiralnotebook.com</a></td>
</tr>
<tr>
<td><strong>Session2</strong></td>
<td>mark twain middle school</td>
<td>2006-04-21 14:38:23</td>
<td>2</td>
<td><a href="http://www.fcps.k12.va.us">http://www.fcps.k12.va.us</a></td>
</tr>
<tr>
<td></td>
<td>mark twain middle school</td>
<td>2006-04-21 14:38:27</td>
<td>1</td>
<td><a href="http://www.fcps.k12.va.us">http://www.fcps.k12.va.us</a></td>
</tr>
<tr>
<td></td>
<td>babies are fireproof</td>
<td>2006-04-22 07:31:22</td>
<td>1</td>
<td><a href="http://babiesarefireproof.blogspot.com">http://babiesarefireproof.blogspot.com</a></td>
</tr>
</tbody>
</table>
Query Weighted

We proposed three hypotheses:

The query weight is bigger when:

(1) the query occurs more times in a session;
(2) the query average duration in a session last longer;
(3) the url average rank of the query in a session is higher.
The framework of query weighted

A session contains Q1Q2……Qk queries, for each query we calculate the FreRate, TimeRate and AverRank, and then, we get the weight of a query.
Query Weighted--*FreRate*

\[ FreRate_{Q_{kj}} : \text{the rate of query } Q_{kj} \text{ occurrence times in session } S_k \]

\[ FreRate_{Q_{kj}} = \frac{Fre_{Q_{kj}}}{Q} \]

\[ Fre_{Q_{j}} : \text{the occurrence times of query } Q_{kj} \text{ in session } S_k \]

\[ Q : \text{the total number of query in session } S_k \]
Query Weighted--**TimeRate**

the query stream, sorted by time: \( \{Q_1, Q_2, \ldots, Q_i, \ldots, Q_K\} \)

\[ \text{the duration of query } Q_i \]

\[ \text{QueryDuration}_{Q_i} = \begin{cases} 
\text{QueryTime}_{Q_{i+1}} - \text{QueryTime}_{Q_i} & 1 \leq i < K \\
\text{EndTime} & i = K 
\end{cases} \]

\[ \text{EndTime} = \begin{cases} 
10s & (Q_K \text{ didn't click url}) \\
60s & (Q_K \text{ clicked url}) 
\end{cases} \]

The time of query \( Q_i \)
Query Weighted--*TimeRate*

*TimeRate* $Q_{kj}$: the rate of the average duration of query $Q_{kj}$ in session $S_k$

\[
\text{TimeRate } Q_{kj} = \frac{\text{QueryDuration } Q_{kj}}{\text{SessionTime } S_k}
\]

\[
\text{QueryDuration } Q_{kj} = \frac{\sum \text{QueryDuration } Q_{kj}}{\text{Fre } Q_{kj}}
\]

*QueryDuration* $Q_{kj}$: the average duration of query $Q_{kj}$ in session $S_k$

*SessionTime* $S_k$: the total duration of session $S_k$
Query Weighted--AverRank

\( \text{AverRank}_{Q_{kj}} \): the reciprocal of the average clicked URL rank of query \( Q_{kj} \) in session \( S_k \)

\[
\text{AverRank}_{Q_{kj}} = \frac{\text{Fre}_{Q_{kj}}}{\sum \text{Rank}_{Q_{kj}}}
\]

\( \text{Rank}_{Q_{kj}} \): each clicked URL rank of query \( Q_{kj} \)
Query Weighted

\( W_{Q_{kj}} \): the weight of query \( Q_{kj} \) in session \( S_k \)

\[
W_{Q_{kj}} = \alpha \times \text{FreRate}_{Q_{kj}} + \beta \times \text{TimeRate}_{Q_{kj}} + \gamma \times \text{AverRank}_{Q_{kj}}
\]

\[
\alpha + \beta + \gamma = 1
\]

\[
0 \leq \alpha \leq 1 \quad 0 \leq \beta \leq 1 \quad 0 \leq \gamma \leq 1
\]
Query Weighted—Interest Voting

Calculating the weight of each word in each user’s query log.

We should preprocess the query as follow:

- Splitting words by white space
- Removing the stop words and the noise words
- Stemming by Porter

\[
W_{T_i} = \text{Vote}(T_i) = \sum_{k}^{K_i} \sum_{j}^{N_{ki}} (W_{Q_{kj}} \ast F_{ij})
\]

\(F_{ij}\): the occurrence times of keyword in query

Keyword \(T_i\) occurred in \(K_i\) sessions, and occurred in \(N_{ki}\) queries in session \(S_k\)

And we can represent the user model:

\[
\text{UserInterest} = \{(T_1, W_{T_1})(T_2, W_{T_2}) \ldots \ldots (T_{T_M}, W_{T_M})\}
\]
Outlines

➤ Background
➤ Query Weighted-based User Modeling
➤ Experiments and Results
Dataset: AOL query log (http://www.datatang.com/data/42724)

- Time: 2006-03-01~2006-05-31
- Form: {UserID, Query, QueryTime, Rank, ClickURL}
- Users: 657,426
- Records: 10,154,742
- We used: 376 users(query number is bigger than 20 both in training set and test set)
- Training set: 2006-03-01~2006-05-15
Evaluation Design:

- The preprocessing work in test set
- Representing the test set as a keyword set to be considered as the real interest word set of user
- Comparing the real interest word set and the user model by the evaluation metrics

Evaluation Metrics:

- **MeanP**: the mean of user prediction precision
  
  The value of MeanP is higher, the prediction precision of user model is higher

- **MAP**: the mean of the average of each user precision
  
  The value of MAP is higher, the higher ranks of the successful prediction interests

\[
MeanP = \frac{1}{|U|} \sum_{u} \frac{Pr eNum_u}{M_u} \quad MAP = \frac{1}{|U|} \sum_{u} \frac{1}{N_u} \sum_{m} \text{Precision}(R_{um})
\]
Parameter Estimation

The purpose is to confirm the value of $\alpha, \beta, \gamma$

- Set the step value is 0.1, obtain each corresponding values of MeanP and MAP

The figure is the 65 experiments of different of $\alpha, \beta, \gamma$, and at the experiment 36, the value of MeanP is the highest
Parameter Estimation

In order to verify the effects of the three features:

- The 1, 2, 3 are the effects of only one feature, the 4, 5, 6 are the effects of two features, the 7 is the effect of all three features

<table>
<thead>
<tr>
<th>Experiment</th>
<th>Feature Selection</th>
<th>MeanP</th>
<th>MAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>FreRate ((\alpha = 1.0, \beta = 0.0, \gamma = 0.0))</td>
<td>0.23237</td>
<td>0.26883</td>
</tr>
<tr>
<td>2</td>
<td>TimeRate ((\alpha = 0.0, \beta = 1.0, \gamma = 0.0))</td>
<td>0.21619</td>
<td>0.25081</td>
</tr>
<tr>
<td>3</td>
<td>AveRank ((\alpha = 0.0, \beta = 0.0, \gamma = 1.0))</td>
<td>0.23432</td>
<td>0.27096</td>
</tr>
<tr>
<td>4</td>
<td>TimeRate &amp; AveRank ((\alpha = 0.0, \beta = 0.5, \gamma = 0.5))</td>
<td>0.23742</td>
<td>0.27569</td>
</tr>
<tr>
<td>5</td>
<td>FreRate &amp; AveRank ((\alpha = 0.5, \beta = 0.0, \gamma = 0.5))</td>
<td>0.24714</td>
<td>0.28657</td>
</tr>
<tr>
<td>6</td>
<td>FreRate &amp; TimeRate ((\alpha = 0.7, \beta = 0.3, \gamma = 0.0))</td>
<td>0.23892</td>
<td>0.27581</td>
</tr>
<tr>
<td>7</td>
<td>FreRate &amp; TimeRate &amp; AveRank ((\alpha = 0.4, \beta = 0.3, \gamma = 0.3))</td>
<td><strong>0.24873</strong></td>
<td><strong>0.28844</strong></td>
</tr>
</tbody>
</table>

In this table, we can get the result: when three features were used at the same time, the values of MeanP and MAP are both the highest;

So we considered the result of \(\alpha = 0.4, \beta = 0.3, \gamma = 0.3\) as user model.
Experiments

Method 1: considering the user query log as documents, and calculating the TF-IDF value of each word. *(TF-IDF)*

Method 2: Weighted the bipartite graph, imported the diffusion theory, and then, recourses were allocated to realize the prediction of user behavior and generate user model. *(Diffusion-based)*

Method 3: the query weighted-based user modeling *(Query-weighted)*
The figure shows the values of MeanP and MAP of each method. It shows that our method is better than the TF-IDF Method and Diffusion Method.
This figure shows that the different return number of interest of an user correspond with the value of $MeanP$, and the Query_weighted Method is the best all the time.
Conclusions

- We proposed a query weighted-based method for user modeling;

- The experiments show the effectiveness of the three hypotheses;

- The results show that user behavior reflected user interests, user modeling are not only the user contents modeling, but also the user behavior modeling;

- The method just considered the single user information, the information between the user and the user were not included;

- The future work is taking the information between the user and the user into account, and to obtain better prediction.
Thank you!