Towards Complex User Feedback and Presentation Context in Recommender Systems

Peter Vojtas and Ladislav Peška

Department of Software Engineering,
Charles University in Prague,
Czech Republic

User Implicit Feedback

b)

c)

Results

Dwell time: 16.8 sec

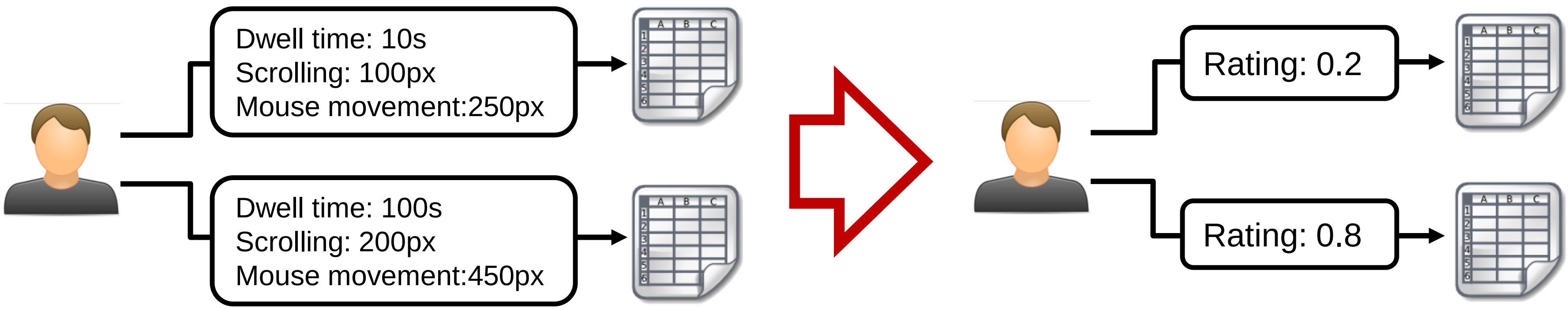
Mouse moving time: 1.8 sec

Travelled distance: 2009px

Software: *Peska, IPIget: The Component for Collecting Implicit User Preference Indicators*

User Feedback – Past Resarch

- Combine multiple implicit feedback features to estimate user rating
    - Standard CB / CF recommender systems can be used afterwards



    - Improvements over the usage of simple implicit feedback

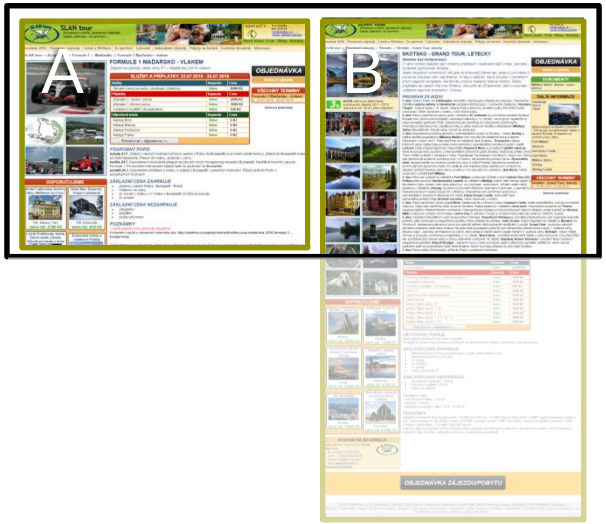
Peska, Vojtas: How to Interpret Implicit User Feedback?
Peska, Eckhardt, Vojtas: Preferential Interpretation of Fuzzy Sets in E-shop Recommendation with Real Data Experiments

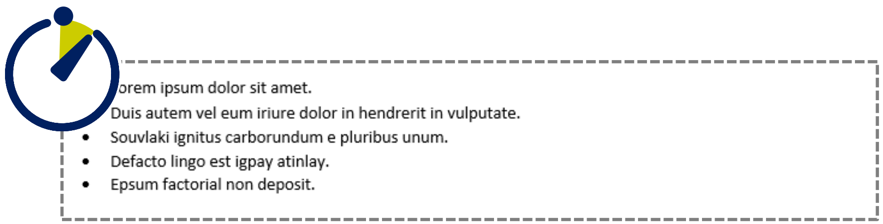
Overview

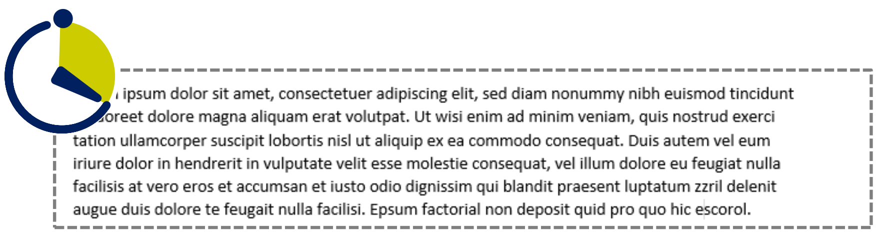
- Context of user feedback – our approach
- Collecting User Behavior
- Estimated Rating from Implicit Feedback
- Employ Context in Rating Estimation
- Evaluation
- Results
- Conclusions, Future Work

Context of User Feedback

- Context of the user
  - Location, Mood, Seasonality...
  - *Can affect user preference*
  - *Out of scope of this paper*
- Context of device and page
  - Page and browser dimensions
  - Page complexity (amount of text, links, images,...)
  - Device type
  - Datetime
  - *Can affect percieved values of the user feedback*







Outline of Our Approach

Traditional recommender

- User rates a sample of objects
  $r_{u,o}: o \in \boldsymbol{S} \subset \boldsymbol{O}$; $r_{u,o} \in [0,1]$
- Preference learning computes expected ratings of all objects
  $R_u \rightarrow \hat{r}_{u,o'} : o' \in \boldsymbol{O}$
- Top-k best rated objects are recommended
  $\hat{R}_u = \{o_1, \dots, o_k\}$

Our approach

- Several implicit feedback and contextual features are collected:
  $F_{u,o} = [f_1, \dots, f_i]$ $C_{u,o} = [c_1, \dots, c_j]$
- Learn estimated rating $\bar{r}_{u,o}$ for visited objects based on feedback and context
  - $F_{u,o}, C_{u,o} \rightarrow \bar{r}_{u,o}: o \in \boldsymbol{S}$
  - „The more the better" heuristics (STD, CDF)
  - Machine learning approach (J48)
- Incorporate context
  - As further feedback features (FB+C)
  - As baseline predictors (AVGBP, CBP)
- Learn rating on all objects as in traditional recommenders
  $\bar{R}_u \rightarrow \hat{r}_{u,o'} : o' \in \boldsymbol{O}$

Collecting User Behavior

- IPIget component for collecting user behavior

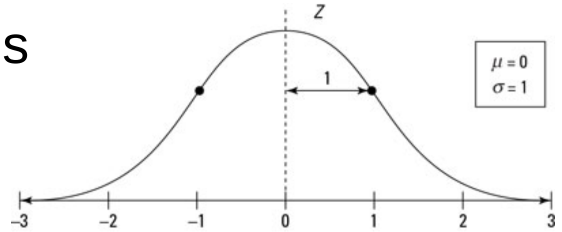
Implicit Feedback Features	
f_1	View Count
f_2	Dwell Time
$f_{3,4}$	Mouse Distance and Time
$f_{5,6}$	Scrolled Distance and Time
f_7	Clicks count
f_8	Hit bottom of the page
r	**Purchase**

Contextual features	
c_1	Number of links
c_2	Number of images
c_3	Text size
c_4	Page dimensions
c_5	Visible area ratio
c_6	Hand-held device

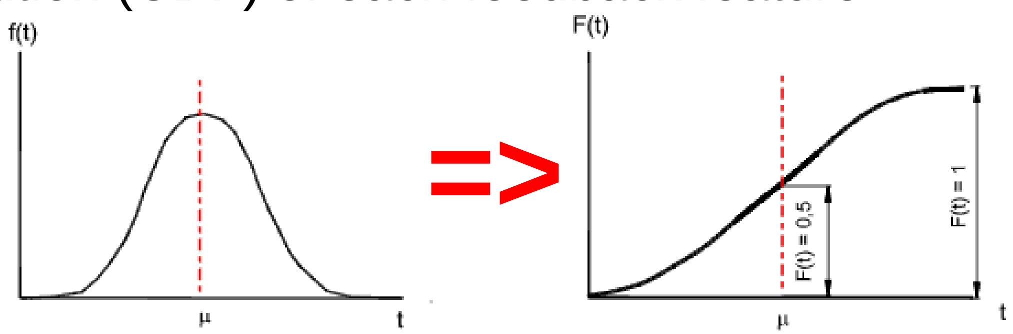
IPIget component download: http://ksi.mff.cuni.cz/~peska/ipiget.zip

Estimated Rating from Implicit Feedback

- „The more the better" heuristics
  - Various feedback features are not comparable in general
    - Dwell time (sec) vs. Distance travelled by mouse (pixels)
    - Transform feedback features on comparable scale and average
  - Use standardization (STD) of feedback features



  - Use cummulative distribution (CDF) of each feedback feature



Estimated Rating from Implicit Feedback

- Machine learning approach
  - J48 decision tree
  - *Purchases* are golden standard
    - The only feedback which is a true indicator of positive preference
  - Predict *purchases* based on other feedback features
  - Use probability of purchase as estimated rating $\bar{r}_{u,o}$

Employ Context in Rating Estimation

- Use context in the same way as feedback *(FB+C)*
  - Leave the decision about usage of context on the underlined model
  - Plausible strategy for e.g. decision trees or rule mining learning approaches
- Use context as a baseline predictor of feedback
  - Calculate estimated value of feedback feature for particular context value $\overline{f_i}(c_j)$
  - Subtract the estimation from the actual value $f_{i.u.o}^{bp} = f_{i,u,o} - \overline{f_i}(c_j)$
    - Use feedback with baseline estimators instead of the original one
  - Either employ average baseline predictor over all context features (AVGBP)
  - Or use Cartesian product of feedback features and baseline predictors based on each context feature (CBP)

Preference Learning and Recommendations

- Collaborative filtering not applicable
  - Continuous cold-start problem
- Use combination of content-based and non-personalized
  - VSM (vector space model) content-based recommendation
    - Vector of object features (TF-IDF)
    - User is represented as weighted sum of visited object's features
    - Resulting score is a cosine similarity of user and object vectors
  - Most popular non-personalized algorithm
    - Based on estimated ratings $\bar{r}_{u,o}$

  - Final score $\hat{r}_{u,o}$, is a multiplication of VSM score and most popular score

Evaluation

- Czech travel agency dataset
  - 3 variants of rating estimation (STD, CDF, J48)
  - 3 variants of context incorporation (FB+C, AVGBP, CBP)
  - 2 baselines (use raw feedback, use binary visits)

- Leave-one-out on purchased objects
  - Ranking prediction
  - nDCG, recall@top-10

Results

Processing method	Feedback and Context composition				
	Binary	*FB*	FB+C	AVGBP	CBP
STD + popVSM	*0.255**	*0.174**	0.197*	0.161*	0.158*
CDF + popVSM	*0.255**	*0.257**	0.253*	0.258*	0.257
J48 + popVSM	*0.255**	*0.256**	**0.274**	0.240*	0.247*
J48 + objects popularity	*0.180***	*0.205**	*0.211**	*0.168**	*0.186**
J48 + VSM	*0.222**	*0.224**	*0.233*	*0.225**	*0.224**

- Results of nDCG, (*) = significant improvement of the best method
- J48 decision tree with both feedback and context on its input performs the best
- Using „the more the better“ heuristics (CDF) with properly processed feedback (AVGBP, CBP) also performs quite well

Conclusions, Future Work

Key outcomes

- Implicit feedback could be more than just a binary variable
- Observed feedback should be considered with respect to the context of page and device
    - Doing so could improve the quality of the recommended objects

Future work, Open Problems

- Better models of context employment and purchase prediction methods
- Further evaluation scenarios
    - Recommending on the beginning of a new session
- More refined feedback?
    - E.g. feedback on object’s attributes?
- On-line deployment and evaluation

Thank you!

Questions, comments?

Supplementary materials: http://bit.ly/2g79VVO