Explicit Factor Models for Explainable Recommendation based on Phrase-level Sentiment Analysis

Yongfeng Zhang,
Guokun Lai, Min Zhang, Yiqun Liu, Shaoping Ma, Tsinghua University

Yi Zhang, University of California, Santa Cruz

When I am Choosing a Whitening Cream

Nivea UV Whitening Extra Cell Repair & Protect Body Cream 200ml
by Nivea
13 customer reviews

List Price: ~~$35.00~~
Price: $13.80 & **FREE Shipping**
You Save: $21.20 (61%)

In Stock.
Ships from and sold by Triple_A_.

- Body Cream
- Extra Cell Repair
- UV Protection
- Vitamin C

43 new from $7.15

A whitening cream that my GF used to buy

The recommended items that other customers also viewed

Customers Who Viewed This Item Also Viewed

Nivea Extra Whitening Pore Minimizer Antiperspirant Deodorant Roll-On 50ml
(142)
$8.33

Nivea UV Whitening Extra Cell Repair and Protect Body Lotion 400ml
(14)
$20.80

Nivea Body Extra Whitening Milk Repair 400ml
(6)
$20.00

Nivea UV Whitening Extra Cell Repair Body Lotion 250ml
(1)
$5.95

But I have made a WRONG choice

The recommended item is suitable for DRY skins while she has OIL skins

★★★★☆ **nice scent**

By Y. Xiong on December 16, 2013

Verified Purchase

Unlike other cream I have tried, Nivea has a really great scent. $\boxed{\text{It is greasy and fits for my dry skin well.}}$ I also bought Serum so I used the Serum first before applying the cream. I used it as a daily lotion. It's hard to tell if my skin has lightened since I've used it for over a month and I haven't really seen a difference yet. Maybe a slight change but nothing dramatic to where you can tell my skin has whitened.

Can the recommender system give me more detailed EXPLANATIONS about WHY an item is recommended?

Customers Who Viewed This Item Also Viewed

Challenges in Generating Explanations

- Factorization models are hard to explain
  - The ability to recommend without clear content information
  - High rating prediction accuracy
  - Latent Factor Models (LFM) have achieved significant success

- The latent features make it difficult to explain the recommendation results to users

Can we have a solution that is both highly *accurate* and easily *explainable*?

Backgrounds and Motivations

- However
- One of the underlying reason
  - How users compose the different attributes of a product into a single numerical rating.

Textual Reviews Could be Helpful

Numerical Star Rating

Review Text

Apple iPhone 5 16GB (White) - Unlocked by Apple

By Anand (Chennai) - See all my reviews

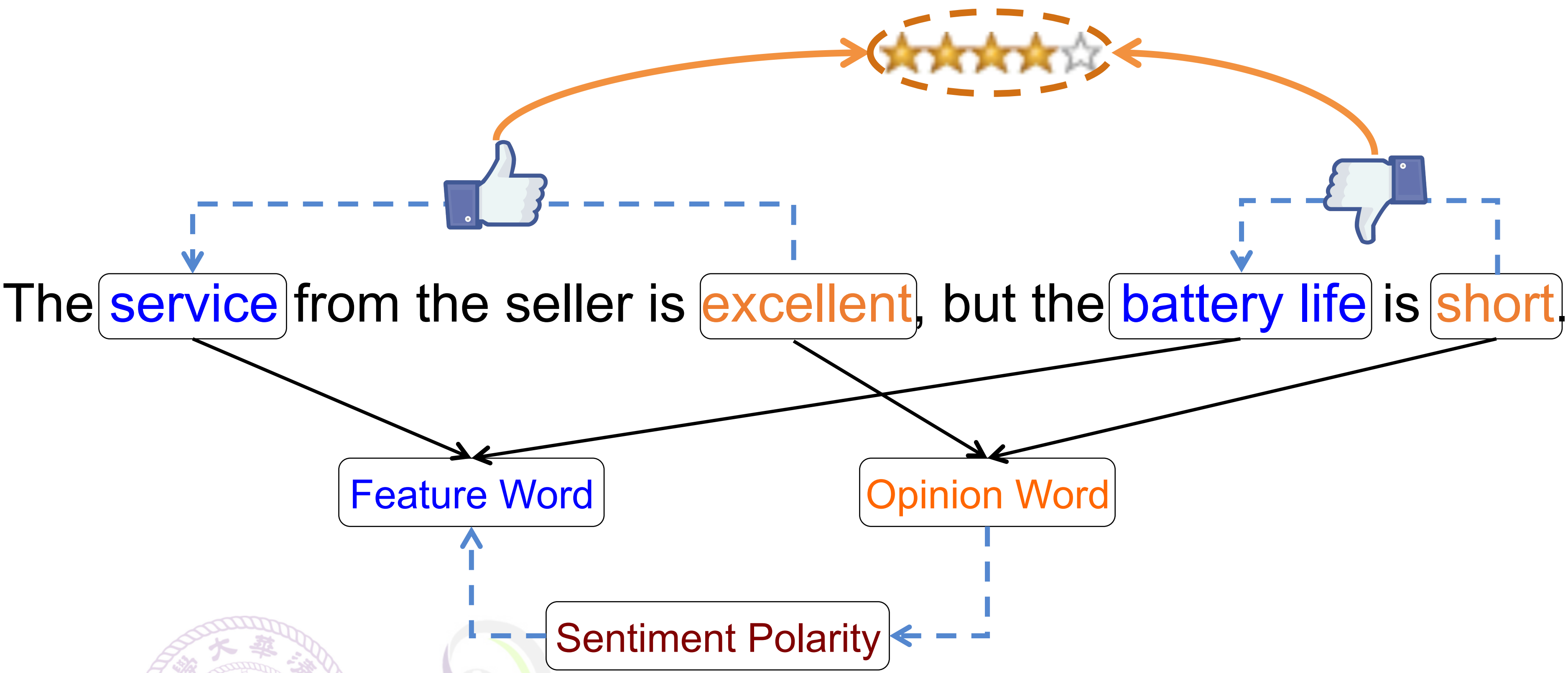
Exceeds the expectation, June 18, 2013

I am very happy to have bought this phone from Amazon and the service rendered from the seller is excellent. Phone quality is perfect as new though I bought an used one. Care to their customers is something a key strategy the seller has followed. I would like to deal again with the same group in near future and recommend to others highly. Thank you.

Service – Excellent
Phone quality – Perfect

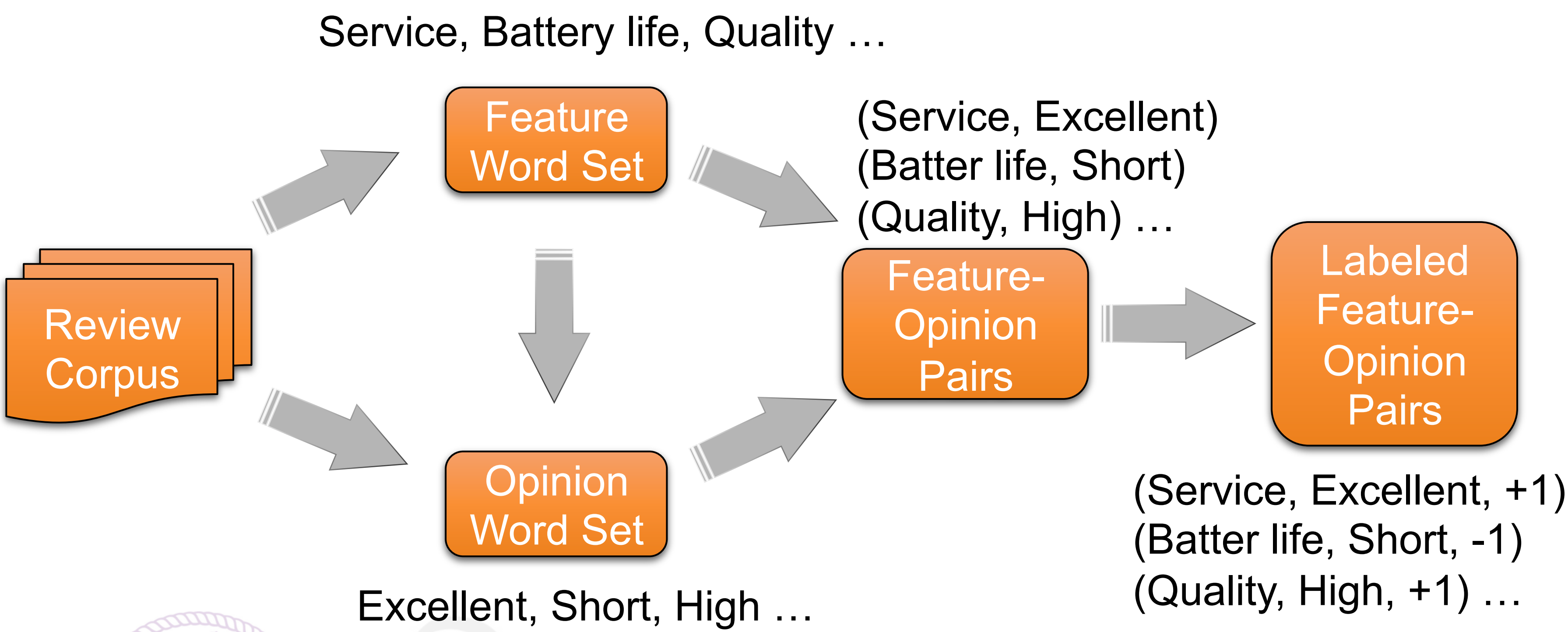
The Role of Textual Reviews

- Phrase-level Sentiment Analysis
  - To extract product features and user opinions from reviews



Sentiment Lexicon Construction

➢ Construct a sentiment lexicon from large amount of textual user reviews.

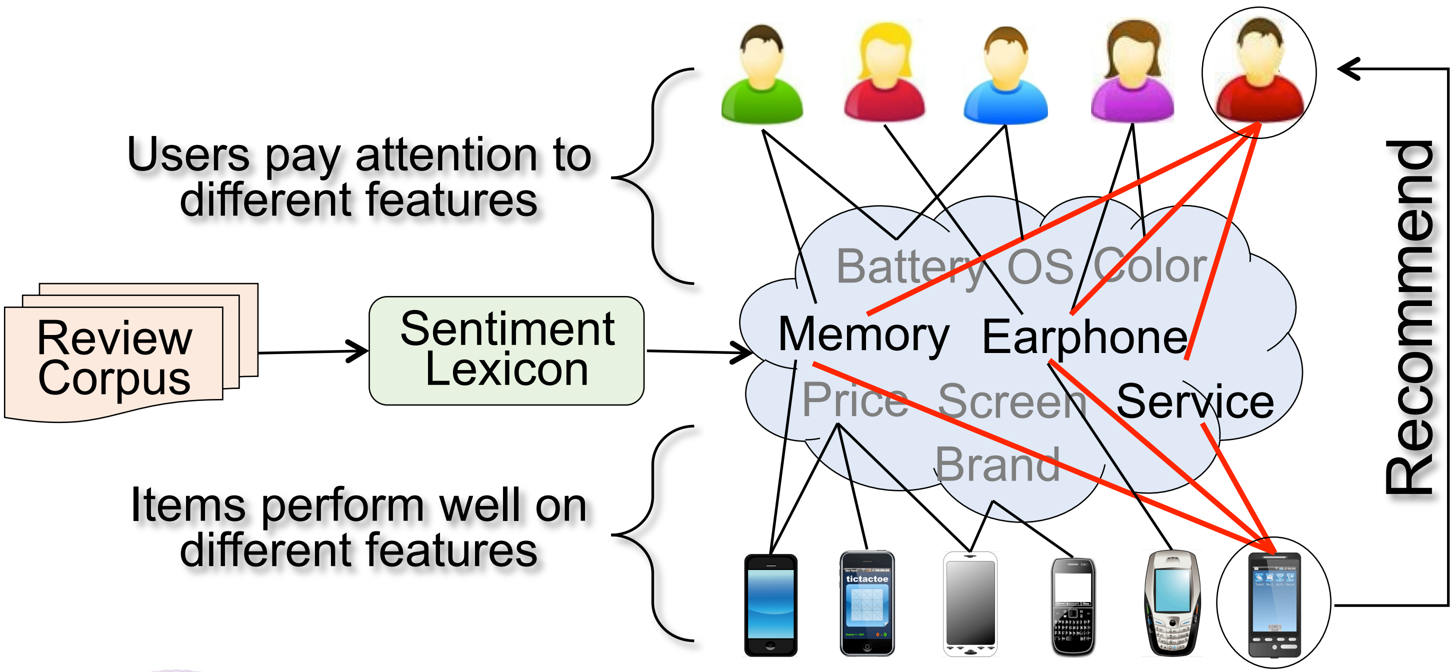


Sentiment Lexicon Construction

- Two basic properties to note
  - The sentiment lexicon is *domain specific*
    - Different product domain may have different product feature words and user opinion words
  - The sentiment lexicon is *contextual*
    - The same opinion word may exhibit different sentiment with different feature word
    - (Quality, High, +1) vs (Noise, High, -1)

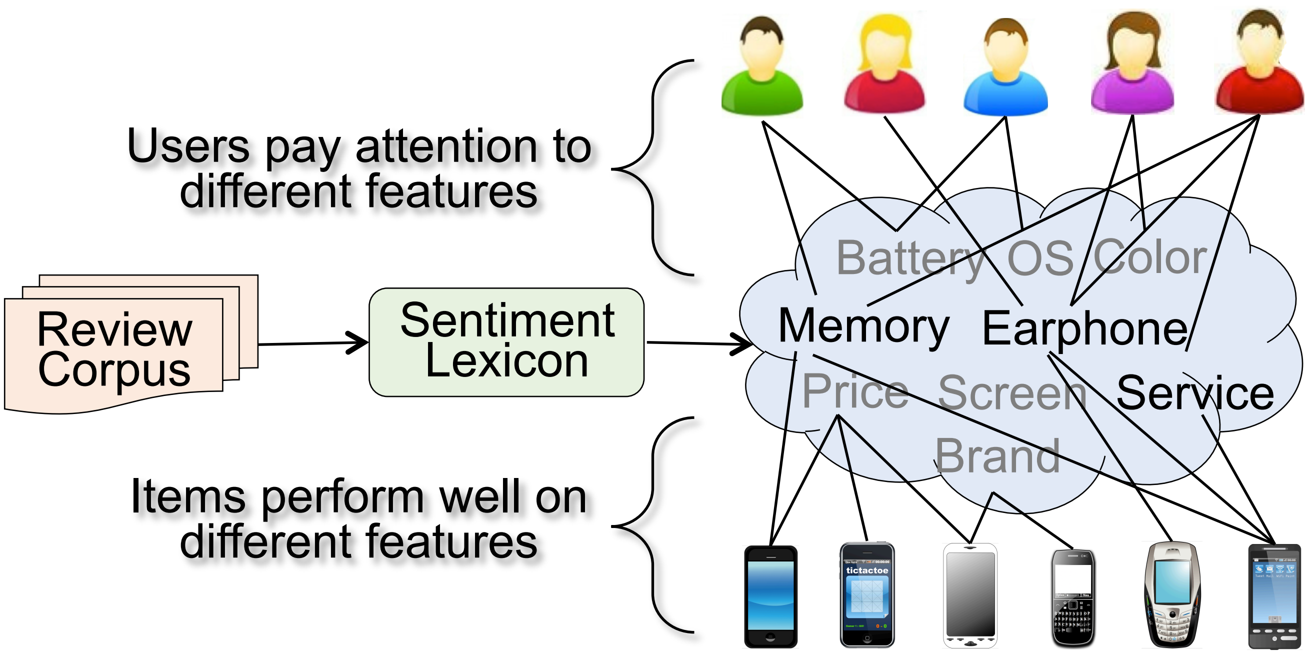
Our Approach: the Intuition

- To recommend a product that performs well on the features that a user concerns.



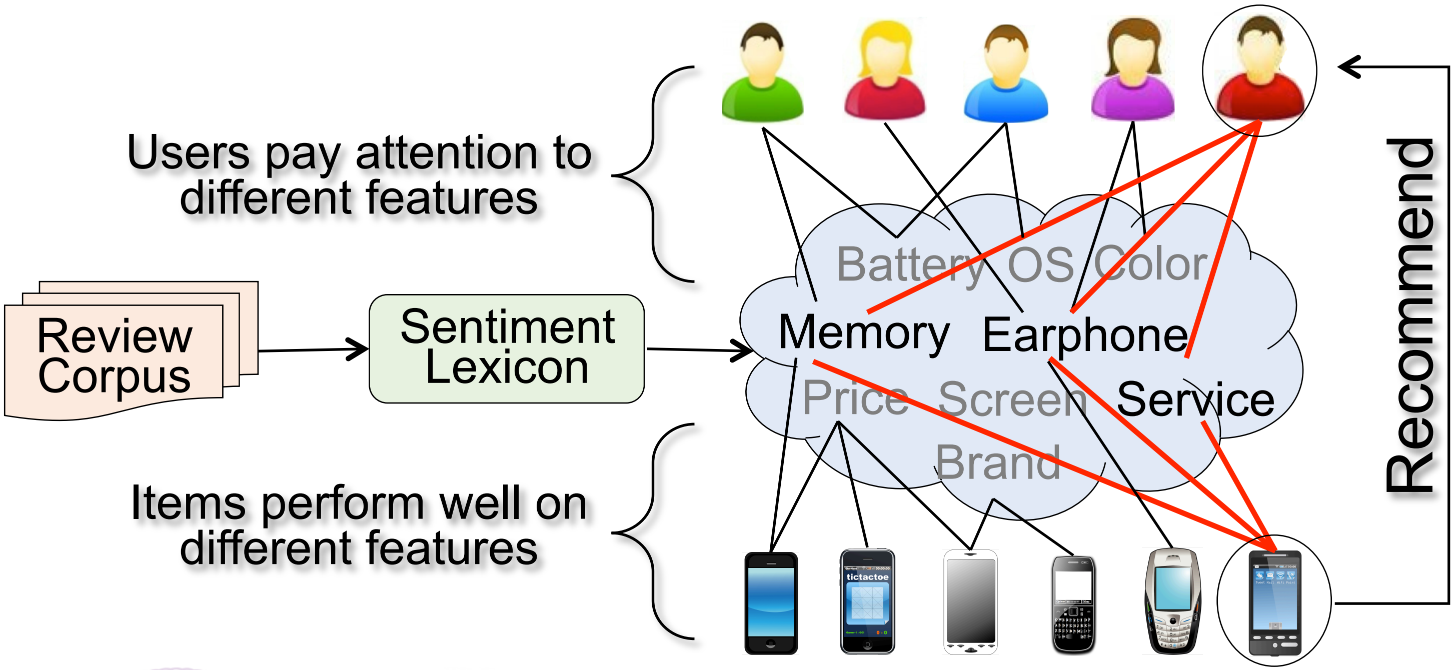
Our Approach: the Intuition

- To recommend a product that performs well on the features that a user concerns.

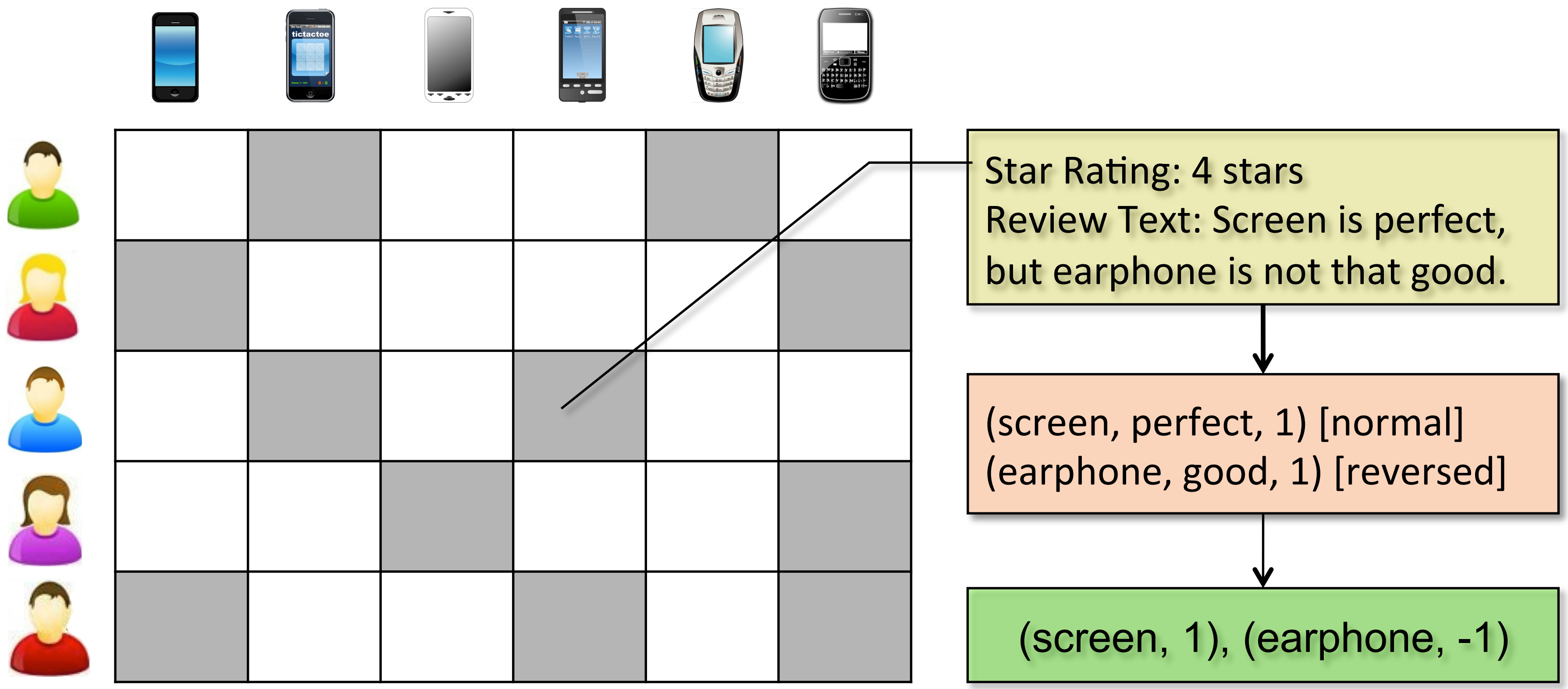


Our Approach: the Intuition

- To recommend a product that performs well on the features that a user concerns.

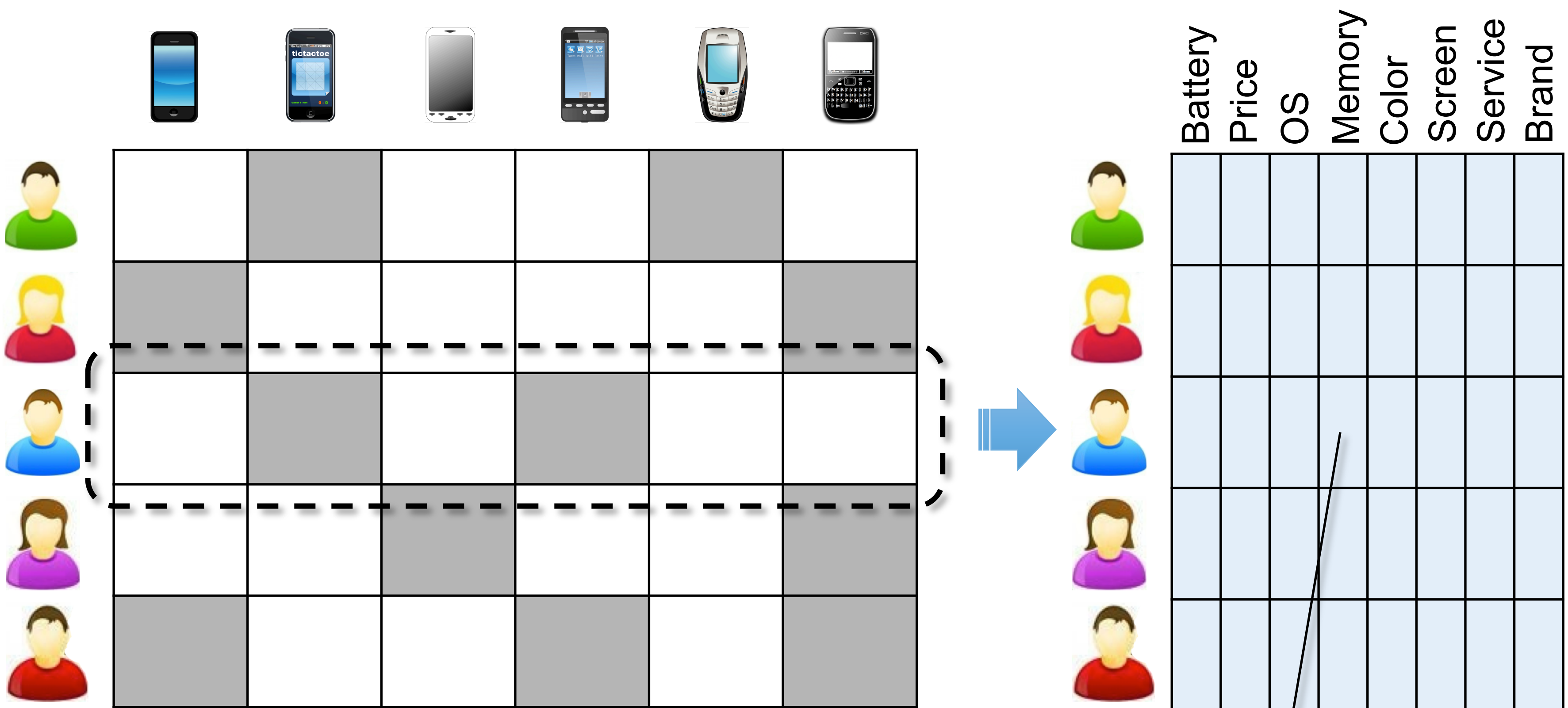


Structure the Textual Reviews



- ➢ Extract the Feature-Opinion pairs contained therein
- ➢ Detect whether the sentiment is reversed by negation words
- ➢ Calculate the real sentiment expressed on each feature

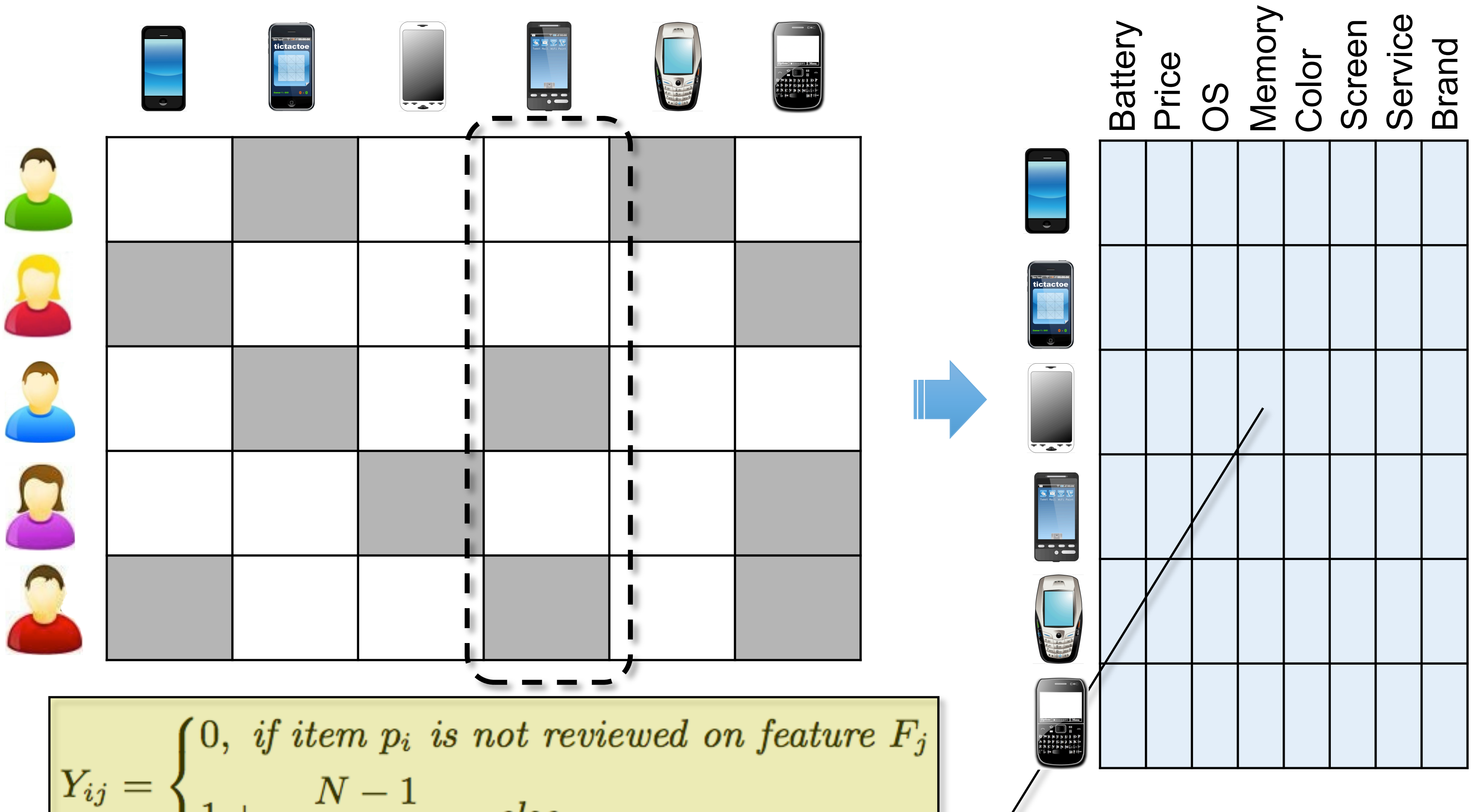
User-Feature Attention Matrix



$$X_{ij} = \begin{cases} 0, & \textit{if user } u_i \textit{ did not mention feature } F_j \\ 1 + (N-1)\left(\dfrac{2}{1+e^{-t_{ij}}} - 1\right), & \textit{else} \end{cases}$$

t_{ij} is the frequency that user i mentions feature j

Item-Feature Quality Matrix

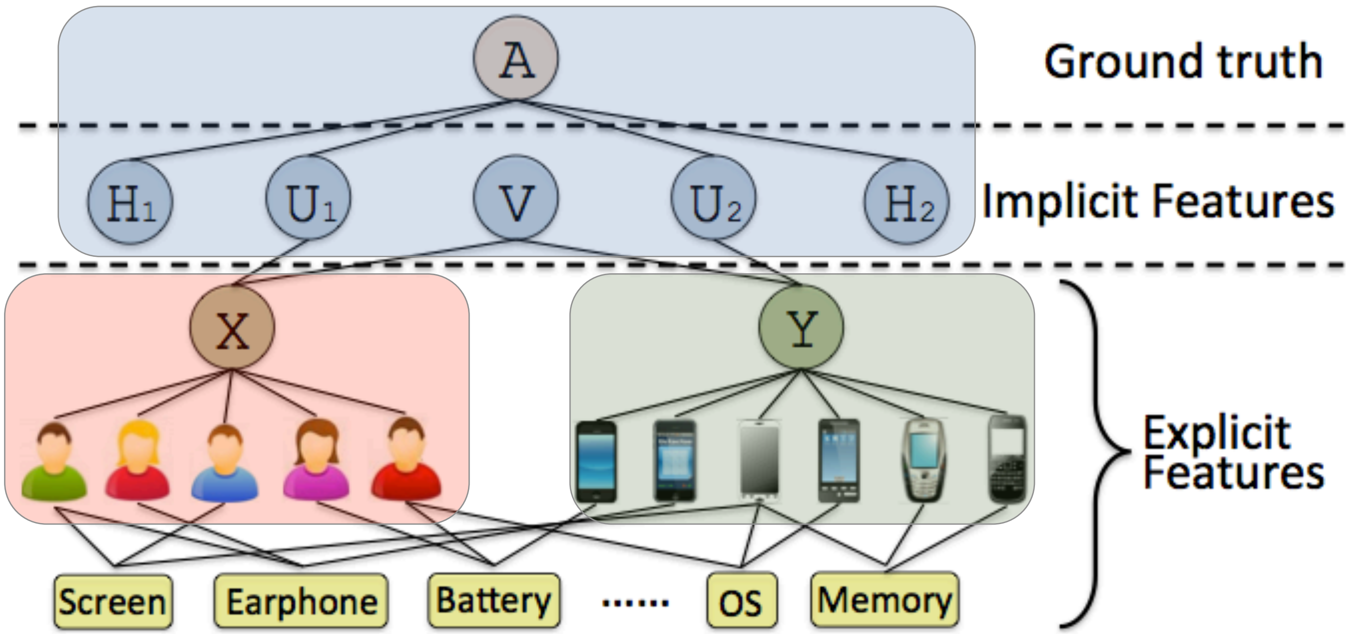


$$Y_{ij} = \begin{cases} 0, & \textit{if item } p_i \textit{ is not reviewed on feature } F_j \\ 1 + \dfrac{N-1}{1+e^{-k \cdot s_{ij}}}, & \textit{else} \end{cases}$$

k is the frequency feature j is mentioned on item i
s_{ij} is the average sentiment of these mentions

Multi-Matrix Factorization

- Integrating the Explicit and Implicit Features



$$\underset{U_1,U_2,V,H_1,H_2}{\text{minimize}} \left\{ \|PQ^T - A\|_F^2 + \lambda_x \|U_1 V^T - X\|_F^2 + \lambda_y \|U_2 V^T - Y\|_F^2 \right.$$

$$\left. + \lambda_u (\|U_1\|_F^2 + \|U_2\|_F^2) + \lambda_h (\|H_1\|_F^2 + \|H_2\|_F^2) + \lambda_v \|V\|_F^2 \right\}$$

$$P = [U_1 \;\, H_1], \;\; Q = [U_2 \;\, H_2]$$

Explicit Factors

Hidden Factors

Algorithm for Model Learning

Algorithm 1: Explicit Factor Model

Input: $A, X, Y, m, n, p, r, r', \lambda_x, \lambda_y, \lambda_u, \lambda_h, \lambda_v, T$

Output: U_1, U_2, V, H_1, H_2

$U_1 \leftarrow \mathbb{R}_+^{m\times r}$, $U_2 \leftarrow \mathbb{R}_+^{n\times r}$, $V \leftarrow \mathbb{R}_+^{p\times r}$;

$H_1 \leftarrow \mathbb{R}_+^{m\times r'}$, $H_2 \leftarrow \mathbb{R}_+^{n\times r'}$; //initialize randomly

$t \leftarrow 0$;

repeat

$t \leftarrow t + 1$;

Update: $V_{ij} \leftarrow V_{ij}\sqrt{\frac{[\lambda_x X^T U_1 + \lambda_y Y^T U_2]_{ij}}{[V(\lambda_x U_1^T U_1 + \lambda_y U_2^T U_2 + \lambda_v I)]_{ij}}}$

Update:

$U_{1ij} \leftarrow U_{1ij}\sqrt{\frac{[AU_2 + \lambda_x XV]_{ij}}{[(U_1 U_2^T + H_1 H_2^T)U_2 + U_1(\lambda_x V^T V + \lambda_u I)]_{ij}}}$

Update:

$U_{2ij} \leftarrow U_{2ij}\sqrt{\frac{[A^T U_1 + \lambda_y YV]_{ij}}{[(U_2 U_1^T + H_2 H_1^T)U_1 + U_2(\lambda_y V^T V + \lambda_u I)]_{ij}}}$

Update: $H_{1ij} \leftarrow H_{1ij}\sqrt{\frac{[AH_2]_{ij}}{[(U_1 U_2^T + H_1 H_2^T)H_2 + \lambda_h H_1]_{ij}}}$

Update: $H_{2ij} \leftarrow H_{2ij}\sqrt{\frac{[A^T H_1]_{ij}}{[(U_2 U_1^T + H_2 H_1^T)H_1 + \lambda_h H_2]_{ij}}}$

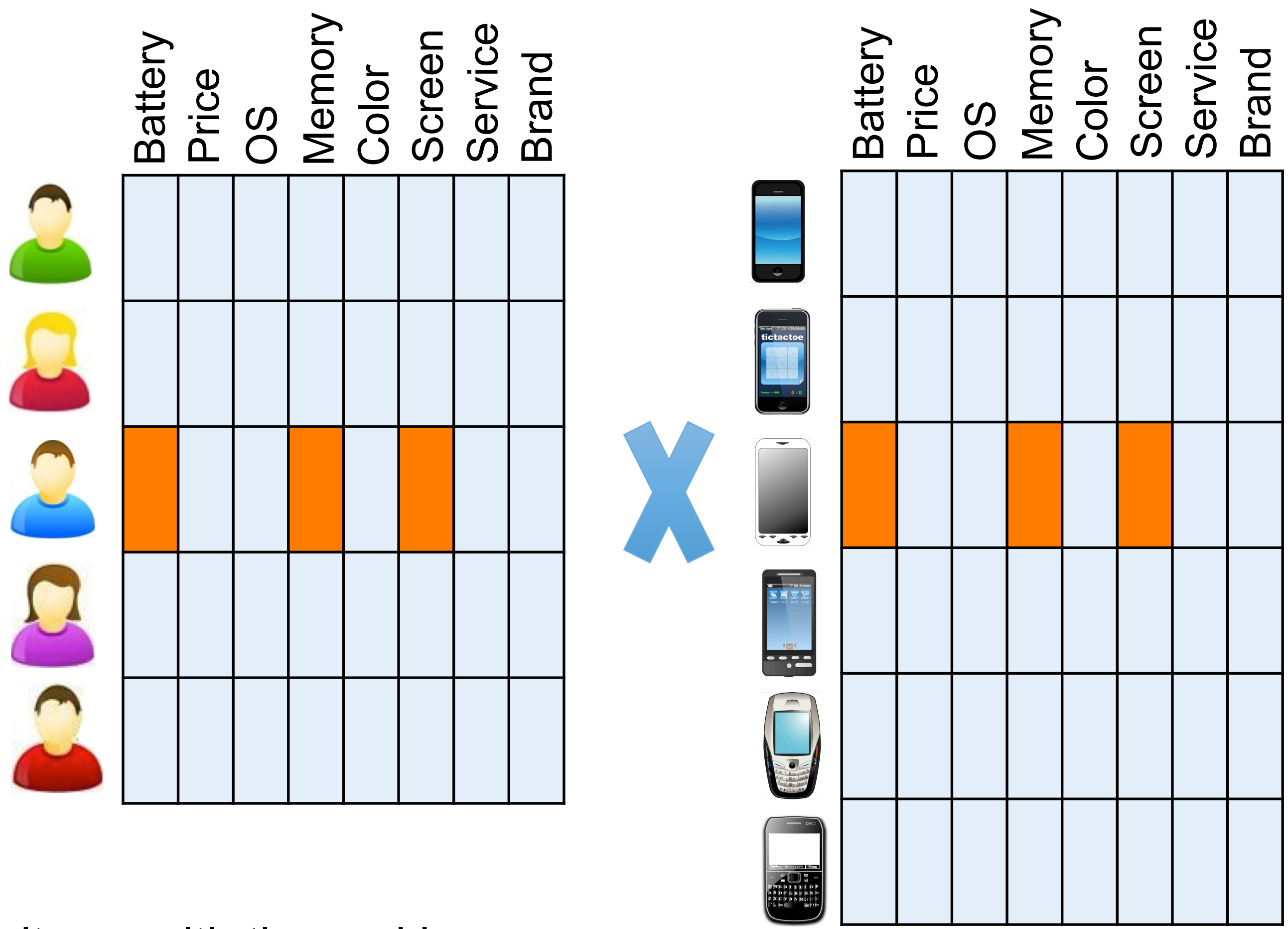
until *Convergence* **or** $t > T$;

return U_1, U_2, V, H_1, H_2;

To select the hyper-parameters, we first randomly initialize the five parameters, and tune them one-by-one with the remaining four fixed. This procedure is conduct several times and we select the best choice.

How to Generate Recommended List

User-based feature selection: select the top-k most cared features (with the highest predicted values) to conduct vector multiplication.

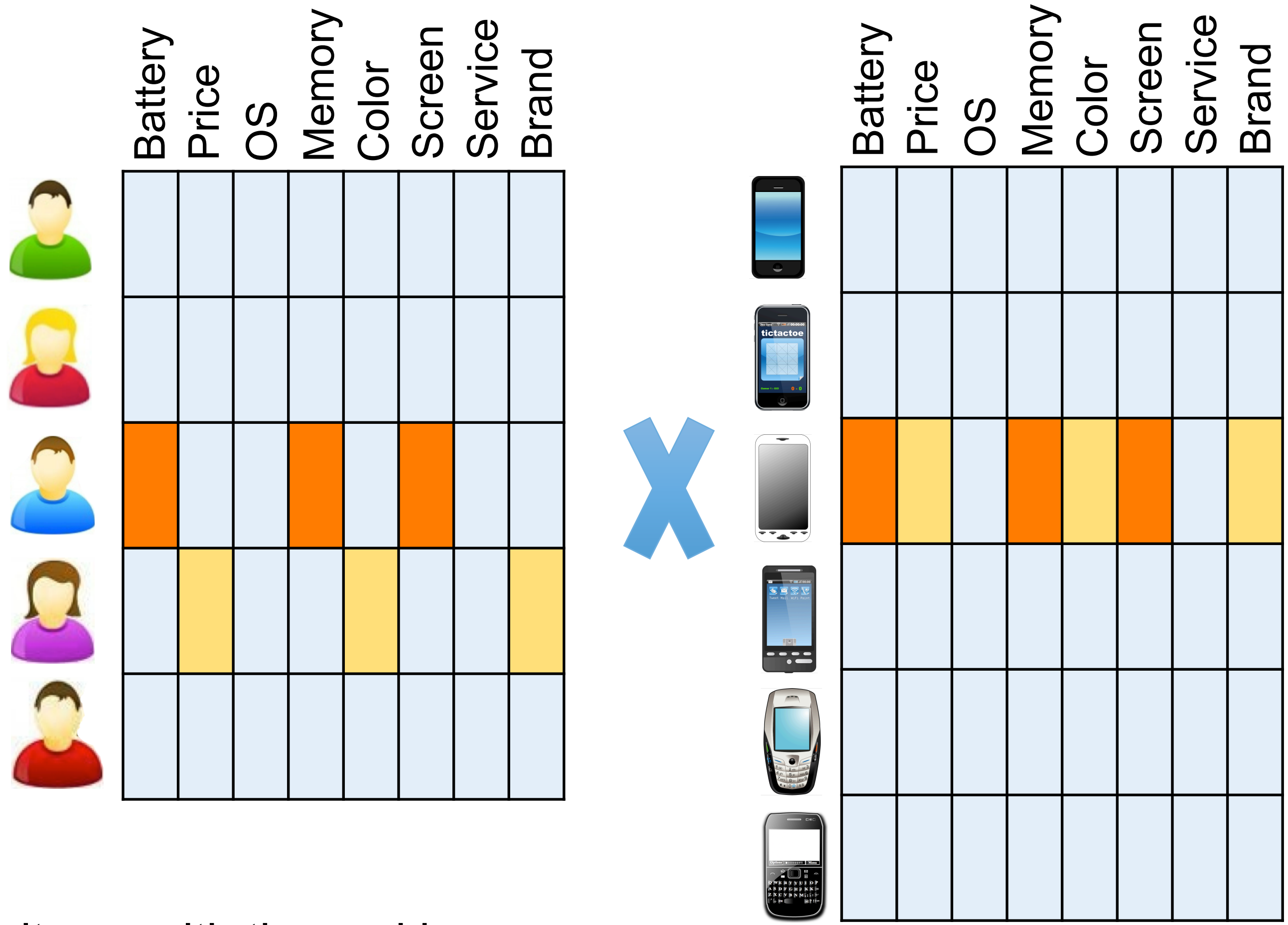


For each user i, rank the items with the ranking score:

$$R_{ij} = \alpha \cdot \frac{\sum_{c \in \mathcal{C}_i} \tilde{X}_{ic} \tilde{Y}_{jc}}{kN} + (1 - \alpha) \tilde{A}_{ij}$$

How to Generate Recommended List

User-based feature selection: select the top-k most cared features (with the highest predicted values) to conduct vector multiplication.



For each user i, rank the items with the ranking score:

$$R_{ij} = \alpha \cdot \frac{\sum_{c \in \mathcal{C}_i} \tilde{X}_{ic} \tilde{Y}_{jc}}{kN} + (1-\alpha)\tilde{A}_{ij}$$

Explanations Could be Very Helpful

- Scrutability: Make the system more transparent and easier to understand
- Effectiveness: Increase users' confidence or trust in the system, help users make better decisions
- Efficiency: Help users to make decisions faster
- Persuasiveness: Convince users to try or buy
- Satisfaction: Increase the ease of the user enjoyment

How to Generate Explanations

Feature-level explanation for a recommended item

```
You might be interested in [feature],
on which this product performs well.
```

For each user u_i and a recommended item p_j, the feature used for explanation construction is F_c, where:

$$c = \operatorname{argmax}_{c \in \mathcal{C}_i} \tilde{Y}_{jc},$$

Provide disrecommendations by telling the user why the current browsing item is disrecommended

```
You might be interested in [feature],
on which this product performs poorly.
```

$$c = \operatorname{argmin}_{c \in \mathcal{C}_i} \tilde{Y}_{jc}$$

Experiments: Setup

- Offline experiment to evaluate recommendation accuracy
  - Rating Prediction & Top-K Recommendation
  - Yelp (English) and Dianping (Chinese) user review datasets

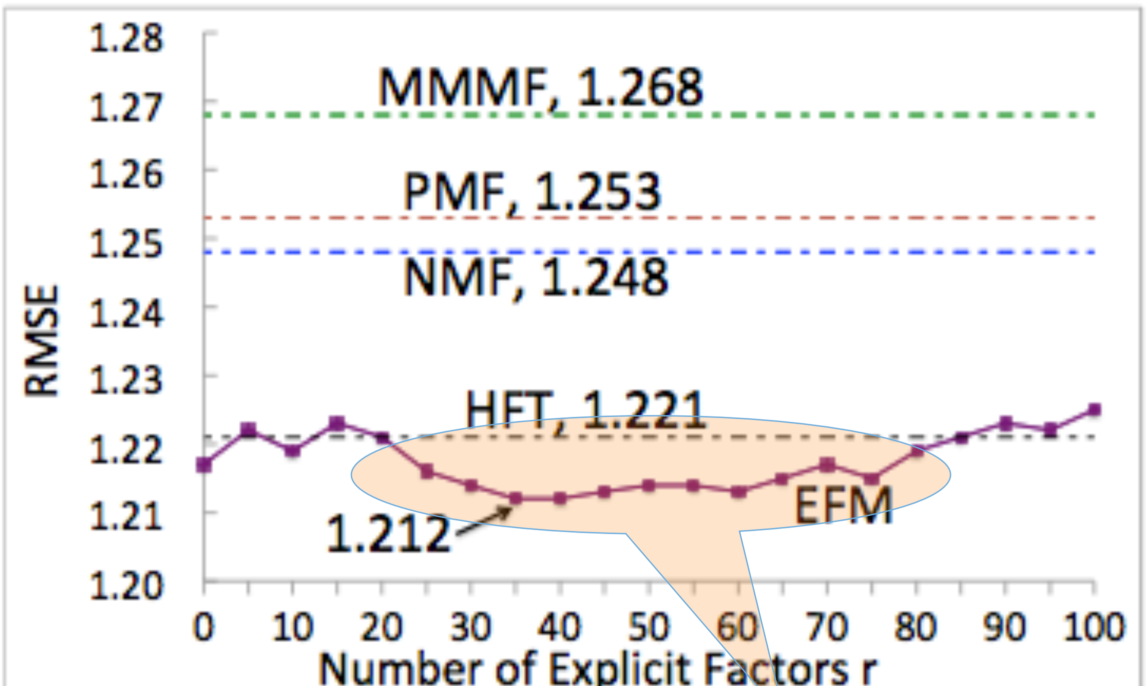
Dataset	#users	#items	#reviews	$\frac{\#reviews}{\#user}$
Yelp	45,981	11,537	229,907	5.00
Dianping	11,857	22,365	510,551	43.06
Yelp10	4,393	10,801	138,301	31.48

- Online A/B test to evaluate explanation effectiveness
  - Recommendation explanation on a major e-commerce web site
  - Focus on the persuasiveness of explanation

Results: Rating Prediction is Improved

- Ratio of Explicit (U_1 U_2) and Hidden Factors (H_1 H_2)
  - Fix r+r'=100 and tune their ratio
  - Set r=100 in comparable algorithms for equal model complexity

Yelp Dataset



When an appropriate number of explicit factors is used, our EFM algorithm is better

Results: Top-K Recommendation is Improved

- Comparative Algorithms
  - **MostPopular**: Rank items by popularity
  - **SlopeOne**: Neighborhood-based algorithm [Lemire 2005]
  - **NMF**: Non-negative Matrix Factorization [Ding and Lee 2001]
  - **BPRMF**: Bayesian Personalized Raking (BPR) optimization for Matrix Factorization (MF) [Rendle 2009]
  - **HFT**: Hidden Factors as Topics [McAuley 2013, Recsys]

Results: Top-K Recommendation is Improved

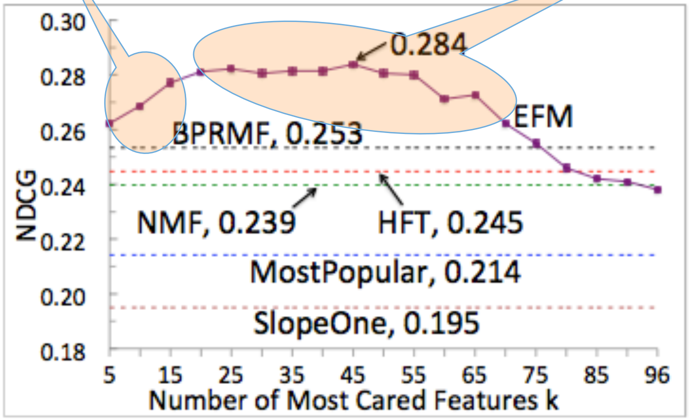
- Number of Most Cared Features k

$$R_{ij} = \alpha \cdot \frac{\sum_{c \in \mathcal{C}_i} \tilde{X}_{ic} \tilde{Y}_{jc}}{kN} + (1-\alpha)\tilde{A}_{ij}$$

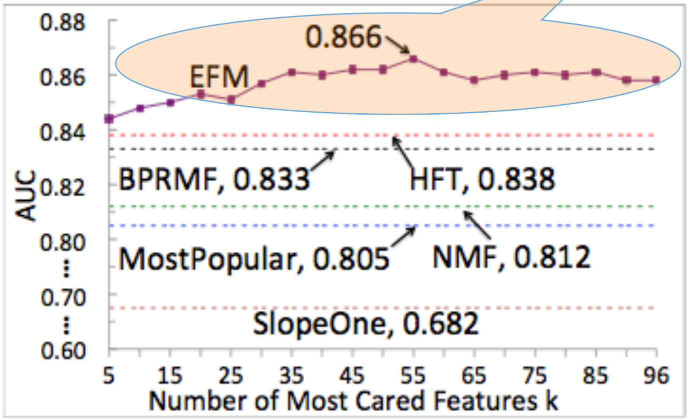
NDCG of EFM rises with the increase of k until about 15

Tends to be stable before it begins to drop when k = 45

However, results on AUC is better consistently



(a) NDCG vs k



(b) AUC vs k

AUC evaluates only the pairwise rankings rather than the positions

Results: Top-K Recommendation is Improved

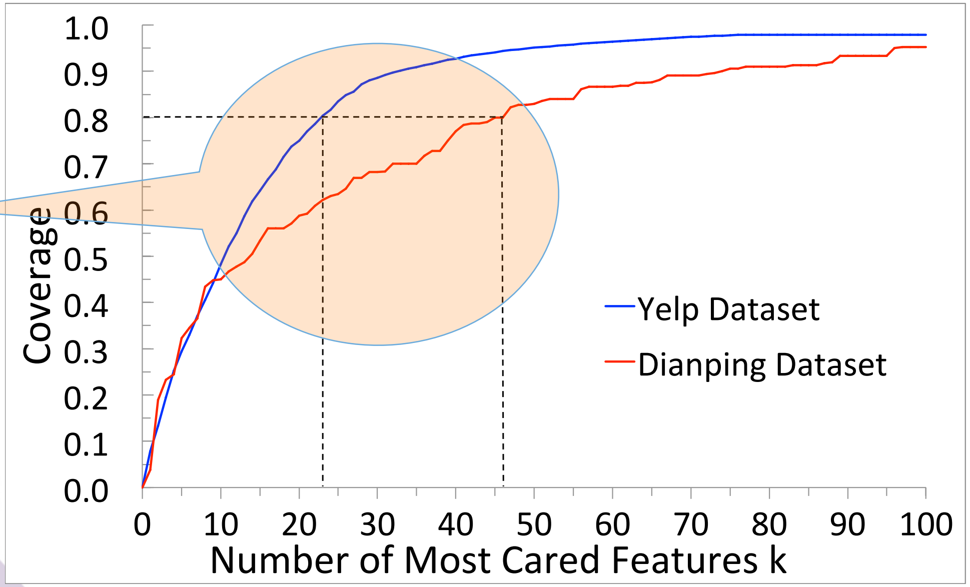
- Further Analysis of Explicit Features
  - It's beyond expectation that a user considers tens of features

- Coverage in term frequency of the top-k most cared features

$$\text{Coverage@}k = \frac{1}{|U|}\sum_{i=1}^{|U|}\frac{\sum_{j\in\mathcal{C}_i} t_{ij}}{\sum_{j=1}^{p} t_{ij}}$$

A small number of explicit features could dominate the term frequency in textual reviews.

This verifies our assumption to use the most cared features for recommendation.



Results: Top-K Recommendation is Improved

- Further Analysis of Explicit Features
  - Why users consider tens of explicit features?

- We group the explicit features into synonym clusters
  - WordNet for English and HowNet for Dianping

Dataset	#Feature	#Cluster	#F/#C
Yelp10	96	31	3.10
Dianping	113	26	4.35

Each synonym cluster has 3~4 explicit features on average.

Users just use different words to express similar concepts!

Results: Top-K Recommendation is Improved

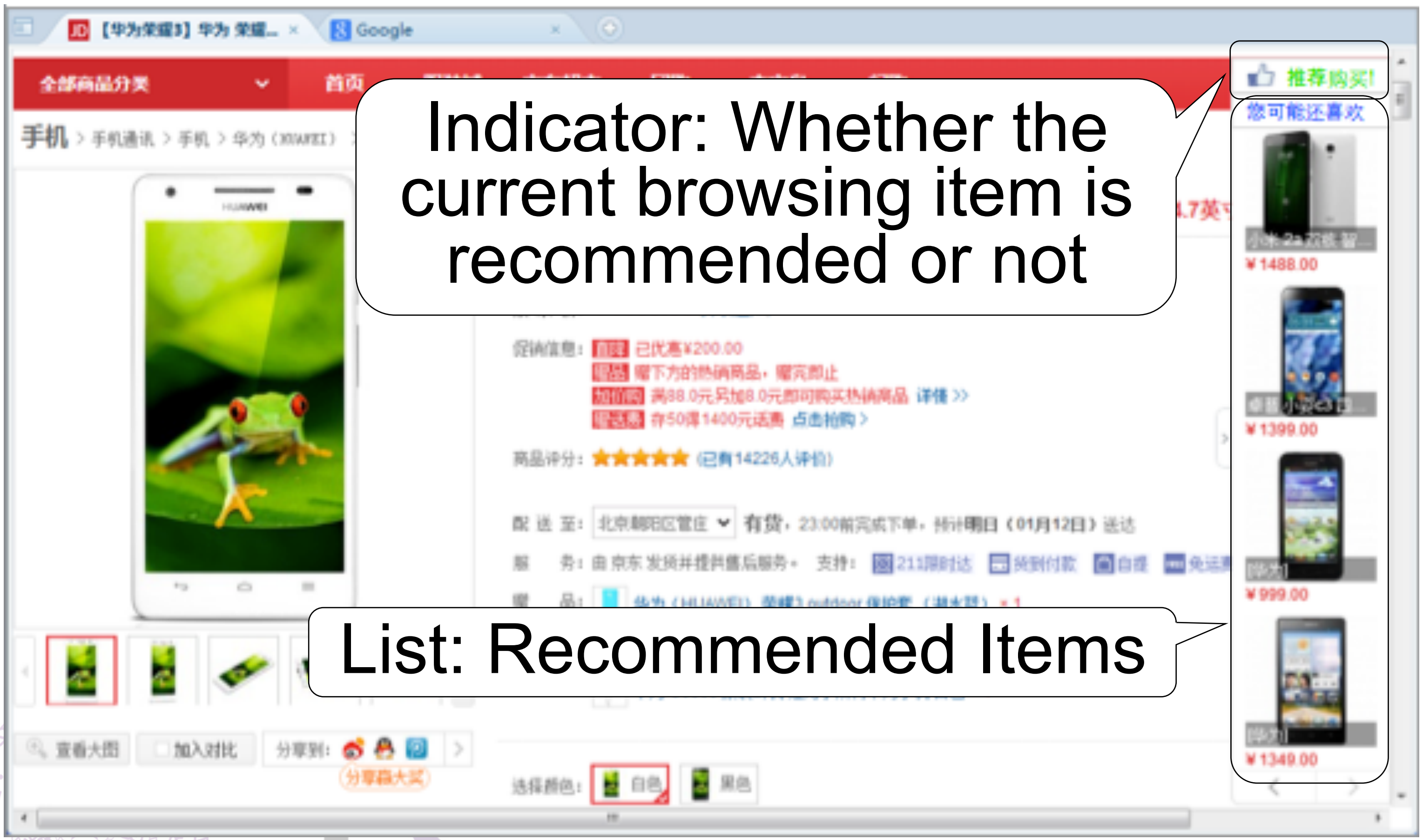
- Further Analysis of Explicit Features
  - The top 15 features can be fully included in the top 7 clusters

1	**place**, **restaurant**, **location**, area, way	2	**food**, **menu**, **lunch**, **pizza**, dinner
3	**service**, **time**, **staff**, order	4	**experience**, quality
5	**room**, **atmosphere**, decor	6	**price**, cost
7	**beer**, wine, drink, water, coffee		

- Relations with previous work
  - Consistent with the Hidden Factors as Topics (HFT) model [McAuley 2013, Recsys]
  - Where the authors find that the performance would not improve with more than 10 topics.
  - They could be 'long tail', redundant topics.

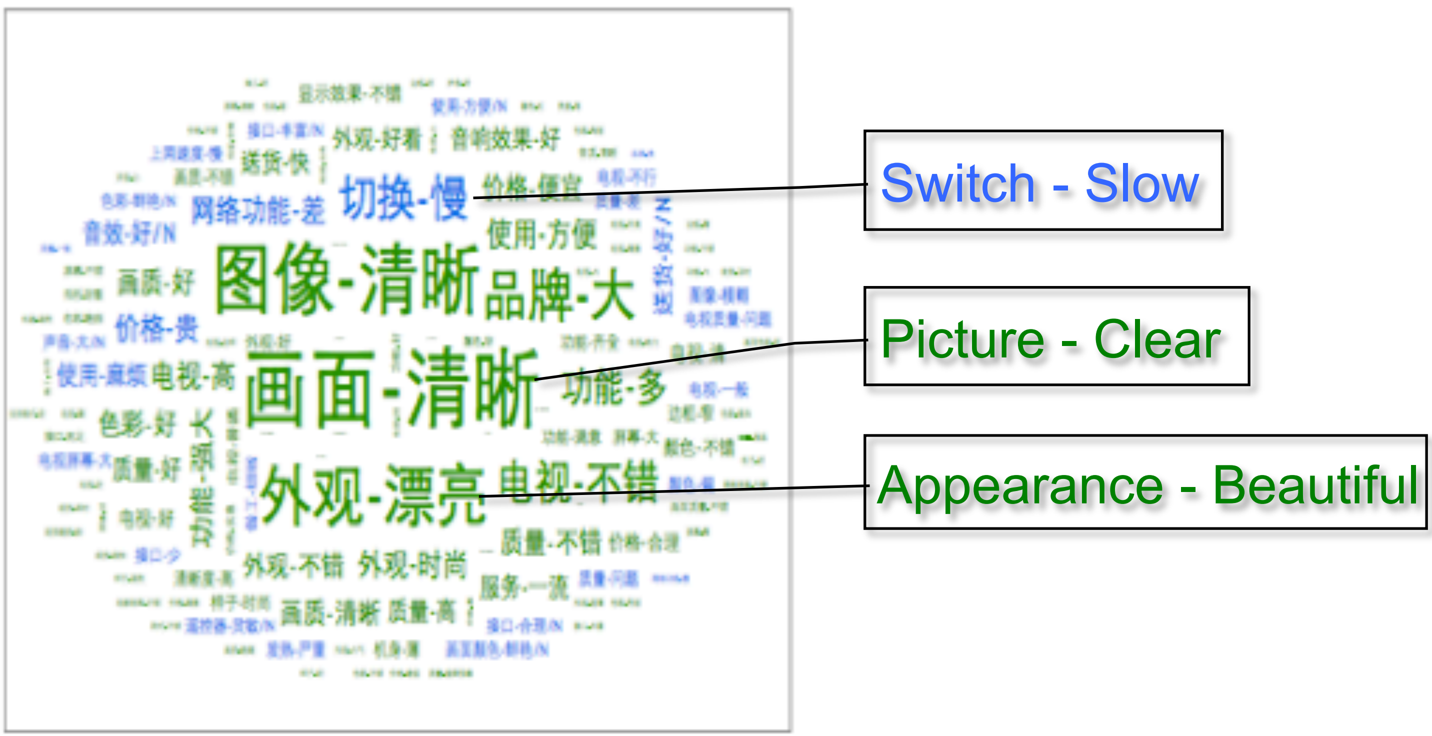
Online Experiment for Explanations: Setup

- Provide mobile phone recommendation by a popular commercial web browser in an e-commerce website.



Online Experiment for Explanations: Setup

- The explanations are displayed when user hover the mouse on an recommended item.
  - To ensure that the users examined the explanations
  - Word cloud to show the detailed performance on features



Click Through Rate on Recommendation List

- Design 3 user groups
  - A (experimental group): Receive our feature-level explanations
  - B (comparison group): Receive the 'people also viewed' explanation
  - C (control group): Receive no explanation

- Only consider the records that hovered the mouse on the recommendations
  - As an indication of examining the explanations.

User Set	A		B		C	
Records	#Record	#Click	#Record	#Click	#Record	#Click
	15,933	691	11,483	370	17,265	552
CTR	4.34%		3.22%		3.20%	

  - Click through rate is significantly higher in group A than B and C.

(Dis)Recommendation with Additional Explanation is More Influential on User Buying Decision

- A group: receives the feature-level explanations
- B group: receives no explanation
  - We didn't assign other comparison groups because these is no previous work presenting disrecommendation explanations

Confusion Matrix	A				B			
	Add		¬Add		Add		¬Add	
Recommend	x_{11}	1,261	x_{12}	16,572	x_{11}	1,129	x_{12}	28,218
¬Recommnned	x_{21}	72	x_{22}	2,830	x_{21}	541	x_{22}	2,749
AddToCart%	6.43%				5.12%			
Agreement%	19.73%				11.89%			

$$\text{AddToCart\%}=\frac{x_{11}+x_{21}}{x_{11}+x_{12}+x_{21}+x_{22}}$$

$$\text{Agreement\%}=\frac{x_{11}+x_{22}}{x_{11}+x_{12}+x_{21}+x_{22}}$$

Explanations help persuade a user to add a recommended product to shopping cart or to ignore a disrecommended product

Conclusions

Attempt to bring new insights into the problem of recommendation explanation

Incorporate phrase-level sentiment analysis into recommender systems

Propose the Explicit Factor Models for both accurate recommendation and intuitional explanations

Good performance on both offline and online A/B tests


THANKS