Experience-aware Item Recommendation in Evolving Review Communities

Subhabrata Mukherjee*, Hemank Lamba† and Gerhard Weikum*
*Max Planck Institute for Informatics, †Carnegie Mellon University
Email: {smukherjee, weikum}@mpi-inf.mpg.de, hlamba@cs.cmu.edu

***Abstract*—Current recommender systems exploit user and item similarities by collaborative filtering. Some advanced methods also consider the temporal evolution of item ratings as a global background process. However, all prior methods disregard the *individual evolution* of a user's *experience* level and how this is expressed in the user's *writing* in a review community.**

In this paper, we model the *joint evolution* of *user experience*, interest in specific *item facets*, *writing style*, and *rating behavior*. This way we can generate individual recommendations that take into account the user's maturity level (e.g., recommending art movies rather than blockbusters for a cinematography expert). As only item ratings and review texts are observables, we capture the user's experience and interests in a *latent model* learned from her reviews, vocabulary and writing style.

We develop a generative HMM-LDA model to trace user evolution, where the Hidden Markov Model (HMM) traces her latent experience progressing over time — with solely user reviews and ratings as observables over *time*. The facets of a user's interest are drawn from a Latent Dirichlet Allocation (LDA) model derived from her reviews, as a function of her (again latent) experience level. In experiments with four real-world datasets, we show that our model improves the rating prediction over state-of-the-art baselines, by a substantial margin. In addition, our model can also give some *interpretations* for the user experience level.

I. Introduction

Motivation and State-of-the-Art: Collaborative filtering algorithms exploit user-user and item-item similarities, based on latent factor models over user and item features [6], for recommending items to users. All these data evolve over *time* leading to bursts in item popularity. State-of-the-art recommender systems capture these temporal aspects by introducing global bias components that reflect the evolution of the user and community as a whole [5]. What is missing in all the approaches, though, is the awareness of how *experience* and *maturity* levels evolve in *individual users*.

Individual experience is crucial in how users appreciate items, and thus react to recommendations. For example, a mature cinematographer would appreciate tips on art movies much more than recommendations for new blockbusters. Also, the facets of an item that a user focuses on change with experience. For example, a mature user pays more attention to narrative, light effects, and style rather than actors or special effects. Similar observations hold for ratings of wine, beer, food, etc.

Our approach advances state-of-the-art by tapping review texts, modeling their properties as latent factors, using them to explain and predict item ratings as a function of a user's experience evolving over time. Prior works considering review texts (e.g., [7], [10], [13]) did this only to learn topic similarities in a static, snapshot-oriented manner, without considering time at all. The only prior work [8], considering time, ignores the text of user-contributed reviews in harnessing their experience. However, user experience and their interest in specific item facets at different timepoints can often be observed only *indirectly* through their ratings, and more *vividly* through her vocabulary and writing style in reviews.

Use-case: Consider the reviews and ratings by two users on a "Canon DSLR" camera about the facet camera *lens*.

- *User 1: My first DSLR. Excellent camera, takes great pictures in HD, without a doubt it brings honor to its name. [Rating: 5]*
- *User 2: The EF 75-300 mm lens is only good to be used outside. The 2.2X HD lens can only be used for specific items; filters are useless if ISO, AP,... are correct. The short 18-55mm lens is cheap and should have a hood to keep light off lens. [Rating: 3]*

The second user is clearly more experienced than the first one, and more reserved about the lens quality of that camera model. Future recommendations for the second user should take into consideration the user's maturity.

Approach: We model the joint evolution of *user experience*, interests in specific *item facets*, *writing style*, and *rating behavior* in a community. As only item ratings and review texts are directly observed, we capture a user's experience and interests by a latent model learned from her reviews, and vocabulary. All this is conditioned on *time*, considering the *maturing rate* of a user. Intuitively, a user gains experience not only by writing many reviews, but she also needs to continuously improve the quality of her reviews. This varies for different users, as some enter the community being experienced. This allows us to generate individual recommendations that take into account the user's maturity level and interest in specific facets of items, at different timepoints.

We develop a generative HMM-LDA model for a user's evolution, where the Hidden Markov Model (HMM) traces her latent experience progressing over time, and the Latent Dirichlet Allocation (LDA) model captures her interests in specific item facets as a function of her (again, latent) experience level. The only explicit input to our model is the ratings and review texts upto a certain timepoint; everything else – especially the user's experience level – is a latent variable. The output is the predicted ratings for the user's reviews following the given timepoint. In addition, we can derive interpretations of a user's experience and interests by salient words in the distributional vectors for latent dimensions. Although it is unsurprising to see users writing sophisticated words with more experience, we observe something more interesting. For instance in specialized communities like `beeradvocate.com` and `ratebeer.com`, experienced users write more descriptive and *fruity* words to depict the beer taste (cf. Table IV).

We apply our model to 12.65 million ratings from 0.88 million users on 0.46 million items in four different communities on movies, food, and beer achieving an improvement of 5% to 15% for the mean squared error for rating predictions over the most competitive baseline. We also show that users at the same (latent) experience level do indeed exhibit similar vocabulary.

II. Overview

A. Model Dimensions

Our approach is based on the intuition that there is a strong coupling between the *facet preferences* of a user, her *experience*, *writing style* in reviews, and *rating behavior*. All of these factors jointly evolve with *time* for a given user.

We model the user experience progression through discrete stages, so a state-transition model is natural. Once this decision is made, a Markovian model is the simplest, and thus natural choice. This is because the experience level of a user at the current instant t depends on her experience level at the previous instant t-1. As experience levels are latent (not directly observable), a Hidden Markov Model is appropriate. Experience progression of a user depends on the following factors:

- *Maturing rate* of the user which is modeled by her *activity* in the community. The more engaged a user is in the community, the higher are the chances that she gains experience and advances in writing sophisticated reviews, and develops taste to appreciate specific facets.
- *Facet preferences* of the user in terms of focusing on particular facets of an item (e.g., narrative structure rather than special effects). With increasing maturity, the taste for particular facets becomes more refined.
- *Writing style* of the user, as expressed by the language model at her current level of experience. More sophisticated vocabulary and writing style indicates higher probability of progressing to a more mature level.
- *Time difference* between writing successive reviews. It is unlikely for the user's experience level to change from that of her last review in a short time span (within a few hours or days).
- *Experience level difference*: Since it is unlikely for a user to directly progress to say level 3 from level 1 without passing through level 2, the model at each instant decides whether the user should stay at current level l, or progress to l+1.

In order to learn the *facet preferences* and *language model* of a user at different levels of experience, we use *Latent Dirichlet Allocation* (LDA). In this work, we assume each review to refer to exactly one item. Therefore, the facet distribution of items is expressed in the facet distribution of the review documents.

We make the following assumptions for the generative process of writing a review by a user at time t at experience level e_t:

Level 1: stupid people supposed wouldnt pass bizarre totally cant
Level 2:storyline acting time problems evil great times didnt money ended simply falls pretty
Level 3: movie plot good young epic rock tale believable acting
Level 4: script direction years amount fast primary attractive sense talent multiple demonstrates establish
Level 5: realism moments filmmaker visual perfect memorable recommended genius finish details defined talented visceral nostalgia
Level 1: film will happy people back supposed good wouldnt cant
Level 2: storyline believable acting time stay laugh entire start funny
Level 3 & 4: narrative cinema resemblance masterpiece crude undeniable admirable renowned seventies unpleasant myth nostalgic
Level 5: incisive delirious personages erudite affective dramatis nucleus cinematographic transcendence unerring peerless fevered

TABLE I: Salient words for two facets at five experience levels in movie reviews.

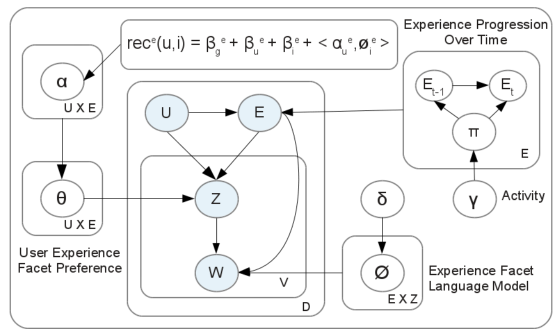

Fig. 1: Supervised model for user experience, facets, and ratings.

- A user has a distribution over *facets*, where the facet preferences of the user depend on her experience level e_t.
- A facet has a distribution over *words* where the words used to describe a facet depend on the user's vocabulary at experience level e_t. Table I shows salient words for two facets of Amazon movie reviews at different levels of user experience, automatically extracted by our latent model. The facets are latent, but we can interpret them as *plot/script* and *narrative style*, respectively.

III. Joint Model: User Experience, Facet Preference, Writing Style

We start with a *User-Facet Model (UFM)* (aka. Author-Topic Model [12]) based on *Latent Dirichlet Allocation* (LDA), where users have a distribution over facets and facets have a distribution over words — extending the standard LDA to include authorship information. This is to determine the facets of interest to a user . These facet preferences can be interpreted as latent item factors in the traditional *Latent-Factor Recommendation Model* (LFM) [4]. However, the LFM is supervised as opposed to the UFM. It is not obvious how to incorporate supervision into the UFM to predict ratings. The user-provided ratings of items can take continuous values (in some review communities), so we cannot incorporate them into a UFM with a Multinomial distribution of ratings. We propose an *Expectation-Maximization (EM)* approach to incorporate supervision, where the latent facets are estimated in an *E-Step* using *Gibbs Sampling*, and *Support Vector Regression* (SVR) [2] is used in the *M-Step* to learn the feature weights and predict ratings. Subsequently, we incorporate a layer for *experience* in the UFM-LFM model, where the experience levels are drawn from a *Hidden Markov Model* (HMM) in the *E-Step*. The experience level transitions depend on the evolution of the user's *maturing rate*, *facet preferences*, and *writing style* over *time*. The entire process is a supervised generative process of generating a review based on the experience level of a user hinged on our HMM-LDA model.

A. Generative Process for a Review

Consider a corpus with a set D of review documents denoted by $\{d_1 \ldots d_D\}$. For *each user*, all her documents are ordered by timestamps t when she wrote them, such that $t_{d_i} < t_{d_j}$ for $i < j$. Each document d has a sequence of N_d words denoted by $d = w_1 \ldots . w_{N_d}$. Each word is drawn from a vocabulary V having unique words indexed by $\{1 \ldots V\}$. Consider a set of U users involved in writing the documents in the corpus, where u_d is the author of document d. Consider an ordered set of experience levels $\{e_1, e_2, ... e_E\}$ where each e_i is from a set E, and a set of facets $\{z_1, z_2, ... z_Z\}$ where each z_i is from a set Z of possible facets. Each document d is associated with a rating r and an item i.

At the time t_d of writing the review d, the user u_d has experience level $e_{t_d} \in E$. We assume that her experience level transitions follow a distribution Π with a Markovian assumption and certain constraints. This means the experience level of u_d at time t_d depends on her experience level when writing the previous document at time t_{d-1}.

$\pi_{e_i}(e_j)$ denotes the probability of progressing to experience level e_j from experience level e_i, with the constraint $e_j \in \{e_i, e_i + 1\}$. This means at each instant the user can either stay at her current experience level, or move to the next one.

The experience-level transition probabilities depend on the *rating behavior*, *facet preferences*, and *writing style* of the user. The progression also takes into account the 1) *maturing rate* of u_d modeled by the intensity of her activity in the community, and 2) the *time gaps* between writing consecutive reviews. We incorporate these aspects in a prior for the user's transition rates, γ^{u_d}, defined as:

$$\gamma^{u_d} = \frac{D_{u_d}}{D_{u_d} + D_{avg}} + \lambda(t_d - t_{d-1})$$

D_{u_d} and D_{avg} denote the number of reviews written by u_d and the average number of reviews per user in the community, respectively. Therefore the first term models the user activity with respect to the community average. The second term reflects the time difference between successive reviews. The user experience is unlikely to change from the level when writing the previous review just a few hours or days ago. λ controls the effect of this time difference, and is set to a very small value. Note that if the user writes very infrequently, the second term may go up. But the first term which plays the *dominating* role in this prior will be very small with respect to the community average in an active community, bringing down the influence of the entire prior. Note that the constructed HMM encapsulates all the factors for experience progression outlined in Section II.

At experience level e_{t_d}, user u_d has a Multinomial facet-preference distribution $\theta_{u_d, e_{t_d}}$. From this distribution she draws a facet of interest z_{d_i} for the i^{th} word in her document. For example, a user at a high level of experience may choose to write on the beer "hoppiness" or "story perplexity" in a movie. The word that she writes depends on the facet chosen and the language model for her current experience level. Thus, she draws a word from the Multinomial distribution $\phi_{e_{t_d}, z_{d_i}}$ with a symmetric Dirichlet prior δ. For example, if the facet chosen is beer *taste* or movie *plot*, an experienced user may choose to use the words "coffee roasted vanilla" and "visceral", whereas an inexperienced user may use "bitter" and "emotional" resp.

Algorithm 1 describes this generative process for the review; Figure 1 depicts it visually in plate notation for graphical models. We use *MCMC* sampling for inference on this model.

Algorithm 1: Supervised Generative Model for a User's Experience, Facets, and Ratings

```
for each facet z = 1, ...Z and experience level e = 1, ...E do
|   choose φ_{e,z} ~ Dirichlet(β)
end

for each review d = 1, ...D do
|   Given user u_d and timestamp t_d
|   /*Current experience level depends on previous level*/
|   1. Conditioned on u_d and previous experience e_{t_{d-1}}, choose
|   e_{t_d} ~ π_{e_{t_{d-1}}}
|   /*User's facet preferences at current experience level are influenced
|   by supervision via α  –  scaled by hyper-parameter ρ controlling
|   influence of supervision*/
|   2. Conditioned on supervised facet preference α_{u_d,e_{t_d}} of u_d at
|   experience level e_{t_d} scaled by ρ, choose
|   θ_{u_d,e_{t_d}} ~ Dirichlet(ρ × α_{u_d,e_{t_d}})
|   for each word i = 1, ...N_d do
|   |   /*Facet is drawn from user's experience-based facet interests*/
|   |   3. Conditioned on u_d and e_{t_d} choose a facet
|   |   z_{d_i} ~ Multinomial(θ_{u_d,e_{t_d}})
|   |   /*Word is drawn from chosen facet and user's vocabulary at
|   |   her current experience level*/
|   |   4. Conditioned on z_{d_i} and e_{t_d} choose a word
|   |   w_{d_i} ~ Multinomial(φ_{e_{t_d},z_{d_i}})
|   end
|   /*Rating computed via Support Vector Regression with
|   chosen facet proportions as input features to learn α*/
|   5. Choose r_d ~ F(⟨α_{u_d,e_{t_d}}, φ_{e_{t_d},z_d}⟩)
end
```

B. Supervision for Rating Prediction

Assume that we have some estimation of the latent facet distribution $\phi_{e,z}$ of each document after one iteration of MCMC sampling, where e denotes the experience level at which a document is written, and let z denote a latent facet of the document. We also have an estimation of the preference of a user u for facet z at experience level e given by $\theta_{u,e}(z)$.

For each user u, we compute a supervised regression function F_u for the user's numeric ratings with the – currently estimated – experience-based facet distribution $\phi_{e,z}$ of her reviews as input features and the ratings as output.

The learned feature weights $\langle \alpha_{u,e}(z) \rangle$ indicate the user's preference for facet z at experience level e. These feature weights are used to modify $\theta_{u,e}$ to attribute more mass to the facet for which u has a higher preference at level e. This is reflected in the next sampling iteration, when we draw a facet z from the user's facet preference distribution $\theta_{u,e}$ smoothed by $\alpha_{u,e}$, and then draw a word from $\phi_{e,z}$. This sampling process is repeated until convergence.

In any latent facet model, it is difficult to set the hyper-parameters. Therefore, most prior work assume symmetric Dirichlet priors with heuristically chosen concentration parameters. Our approach is to *learn* the concentration parameter α of a *general* (i.e., asymmetric) Dirichlet prior for Multinomial distribution Θ – where we optimize these hyper-parameters to learn user ratings for documents at a given experience level.

C. Inference

We describe the inference algorithm to estimate the distributions Θ, Φ and Π from observed data. For each user, we compute the conditional distribution over the set of hidden variables E and Z for all the words W in a review. The exact computation of this distribution is intractable. We use *Collapsed Gibbs Sampling* to estimate the conditional distribution for each hidden variable, which is computed over the current assignment for all other hidden variables, and integrating out other parameters of the model.

Let U, E, Z and W be the set of all users, experience levels, facets and words in the corpus. In the following, i indexes a document and j indexes a word in it.

The joint probability distribution is given by:

$$P(U,E,Z,W,\theta,\phi,\pi;\alpha,\delta,\gamma) = \prod_{u=1}^{U}\prod_{e=1}^{E}\prod_{i=1}^{D_u}\prod_{z=1}^{Z}\prod_{j=1}^{N_{d_u}}\{ \underbrace{P(\pi_e;\gamma^u)\times P(e_i|\pi_e)}_{\text{experience transition distribution}} \times \underbrace{P(\theta_{u,e};\alpha_{u,e})\times P(z_{i,j}|\theta_{u,e_i})}_{\text{user experience facet distribution}} \times \underbrace{P(\phi_{e,z};\delta)\times P(w_{i,j}|\phi_{e_i,z_{i,j}})}_{\text{experience facet language distribution}} \} \tag{1}$$

Let $n(u,e,d,z,v)$ denote the count of the word w occurring in document d written by user u at experience level e belonging to facet z. In the following equation, $(.)$ at any position in a distribution indicates summation of the above counts for the respective argument.

Exploiting conjugacy of the Multinomial and Dirichlet distributions, we can integrate out Φ from the above distribution to obtain the posterior distribution $P(Z|U,E;\alpha)$ of the latent variable Z given by:

$$\prod_{u=1}^{U}\prod_{e=1}^{E}\frac{\Gamma(\sum_z \alpha_{u,e,z})\prod_z \Gamma(n(u,e,.,z,.)+\alpha_{u,e,z})}{\prod_z \Gamma(\alpha_{u,e,z})\Gamma(\sum_z n(u,e,.,z,.)+\sum_z \alpha_{u,e,z})}$$

where Γ denotes the Gamma function.

Similarly, by integrating out Θ, $P(W|E,Z;\delta)$ is given by

$$\prod_{e=1}^{E}\prod_{z=1}^{Z}\frac{\Gamma(\sum_v \delta_v)\prod_v \Gamma(n(.,e,.,z,v)+\delta_v)}{\prod_v \Gamma(\delta_v)\Gamma(\sum_v n(.,e,.,z,v)+\sum_v \delta_v)}$$

Let $m_{e_i}^{e_{i-1}}$ denote the number of transitions from experience level e_{i-1} to e_i over *all* users in the community, with the constraint $e_i \in \{e_{i-1}, e_{i-1}+1\}$. Note that we allow self-transitions for staying at the same experience level. The counts capture the relative difficulty in progressing between different experience levels. For example, it may be easier to progress to level 2 from level 1 than to level 4 from level 3.

The state transition probability depending on the previous state, factoring in the user-specific activity rate, is given by: $P(e_i|e_{i-1},u,e_{-i}) = \frac{m_{e_i}^{e_{i-1}}+I(e_{i-1}=e_i)+\gamma^u}{m_{(.)}^{e_{i-1}}+I(e_{i-1}=e_i)+E\gamma^u}$

where $I(.)$ is an indicator function taking the value 1 when the argument is true, and 0 otherwise. The subscript $-i$ denotes the value of a variable excluding the data at the i^{th} position. All the *counts* of transitions exclude transitions to and from e_i, when sampling a value for the current experience level e_i during Gibbs sampling. The conditional distribution for the experience level transition is given by:

$$P(E|U,Z,W) \propto P(E|U)\times P(Z|E,U)\times P(W|Z,E) \tag{2}$$

Here the first factor models the rate of experience progression factoring in user activity; the second and third factor models the facet-preferences of user, and language model at a specific level of experience respectively. All three factors combined decide whether the user should stay at the current level of experience, or has matured enough to progress to next level.

In Gibbs sampling, the conditional distribution for each hidden variable is computed based on the current assignment of other hidden variables. The values for the latent variables are sampled repeatedly from this conditional distribution until convergence. In our problem setting we have two sets of latent variables corresponding to E and Z respectively.

We perform Collapsed Gibbs Sampling in which we first sample a value for the experience level e_i of the user for the current document i, keeping all facet assignments Z fixed. In order to do this, we consider two experience levels e_{i-1} and $e_{i-1}+1$. For each of these levels, we go through the current document and all the token positions to compute Equation 2 — and choose the level having the highest conditional probability. Thereafter, we sample a new facet for each word $w_{i,j}$ of the document, keeping the currently sampled experience level of the user for the document fixed.

The conditional distributions for Gibbs sampling for the joint update of the latent variables E and Z are given by:

E-Step 1: $P(e_i=e|e_{i-1},u_i=u,\{z_{i,j}=z_j\},\{w_{i,j}=w_j\},e_{-i}) \propto$

$$P(e_i|u,e_{i-1},e_{-i})\times\prod_j P(z_j|e_i,u,e_{-i})\times P(w_j|z_j,e_i,e_{-i}) \propto$$

$$\frac{m_{e_i}^{e_{i-1}}+I(e_{i-1}=e_i)+\gamma^u}{m^{e_{i-1}}+I(e_{i-1}=e_i)+E\gamma^u}\times$$

$$\prod_j \frac{n(u,e,.,z_j,.)+\alpha_{u,e,z_j}}{\sum_{z_j} n(u,e,.,z_j,.)+\sum_{z_j}\alpha_{u,e,z_j}}\times\frac{n(.,e,.,z_j,w_j)+\delta}{\sum_{w_j} n(.,e,.,z_j,w_j)+V\delta}$$

E-Step 2: $P(z_j=z|u_d=u,e_d=e,w_j=w,z_{-j}) \propto$

$$\frac{n(u,e,.,z,.)+\alpha_{u,e,z}}{\sum_z n(u,e,.,z,.)+\sum_z \alpha_{u,e,z}}\times\frac{n(.,e,.,z,w)+\delta}{\sum_w n(.,e,.,z,w)+V\delta} \tag{3}$$

The proportion of the z^{th} facet in document d with words $\{w_j\}$ written at experience level e is given by:

$$\phi_{e,z}(d) = \frac{\sum_{j=1}^{N_d}\phi_{e,z}(w_j)}{N_d}$$

For each user u, we learn a regression model F_u using these facet proportions in each document as features, along with the user and item biases, with the user's item rating r_d as the response variable. Consider $\beta_g(e)$ to be the average rating of all items by users at experience level e, and $\beta_u(e)$ to be the offset of the average rating given by user u from the global rating at experience level e. Likewise $\beta_i(e)$ is the rating bias for item i for users at experience level e.

We formulate the function F_u as Support Vector Regression [2], which forms the $M\text{-}Step$ in our problem:

$$\textbf{M-Step: } \min_{\alpha_{u,e}} \frac{1}{2}{\alpha_{u,e}}^T \alpha_{u,e} + C \times$$

$$\sum_{d=1}^{D_u} (max(0, |r_d - {\alpha_{u,e}}^T < \beta_g(e), \beta_u(e), \beta_i(e), \phi_{e,z}(d) > | - \epsilon))^2$$

where $\langle ., . \rangle$ denotes a scalar product.

The total number of parameters learned is $[E \times Z + E \times 3] \times U$. Our solution may generate a mix of positive and negative real numbered weights. In order to ensure that the concentration parameters of the Dirichlet distribution are positive reals, we take $exp(\alpha_{u,e})$. The learned α's are typically very small, whereas the value of $n(u, e, ., z, .)$ in Equation 3 is very large. Therefore we scale the α's by a hyper-parameter ρ to control the influence of supervision. ρ is tuned using a validation set by varying it from $\{10^0, 10^1 ... 10^5\}$. In the *E-Step* of the next iteration, we choose $\theta_{u,e} \sim Dirichlet(\rho \times \alpha_{u,e})$. We use the LibLinear[1] package for Support Vector Regression.

IV. Experiments

Setup: We perform experiments with data from four communities in different domains: BeerAdvocate (`beeradvocate.com`) and RateBeer (`ratebeer.com`) for beer reviews, Amazon (`amazon.com`) for movie reviews, and Yelp (`yelp.com`) for food and restaurant reviews. Table II gives the dataset statistics[2]. We have a total of 12.65 million reviews from 0.88 million users from all of the four communities combined from where we extract the following quintuple for our model $< userId, itemId, timestamp, rating, review >$.

For all models, we used the three most recent reviews of each user as withheld test data. All experience-based models consider the *last* experience level reached by each user, and corresponding learned parameters for rating prediction. In all the models, we group *light* users with less than 50 reviews in *training* data into a background model, treated as a single user, to avoid modeling from sparse observations. We do not ignore any user. During the *test* phase for a light user, we take her parameters from the background model. We set $Z = 20$ for BeerAdvocate, RateBeer and Yelp facets; and $Z = 100$ for Amazon movies which have much richer latent dimensions. For experience levels, we set $E = 5$ for all.

Baselines: We consider the following baselines for our work, and use the available code[3] for experimentation.

a) *LFM*: A standard latent factor recommendation model [4].

[1] http://www.csie.ntu.edu.tw/ cjlin/liblinear

[2] http://snap.stanford.edu/data/, http://www.yelp.com/dataset_challenge/

[3] http://cseweb.ucsd.edu/ jmcauley/code/

Dataset	#Users	#Items	#Ratings
Beer (BeerAdvocate)	33,387	66,051	1,586,259
Beer (RateBeer)	40,213	110,419	2,924,127
Movies (Amazon)	759,899	267,320	7,911,684
Food (Yelp)	45,981	11,537	229,907
TOTAL	879,480	455,327	12,651,977

TABLE II: Dataset statistics.

Models	Beer Advocate	Rate Beer	Amazon	Yelp
Our model (most recent experience level)	0.363	0.309	1.174	1.469
f) Our model (past experience level)	0.375	0.362	1.200	1.642
e) User at learned rate	0.379	0.336	1.293	1.732
c) Community at learned rate	0.383	0.334	1.203	1.534
b) Community at uniform rate	0.391	0.347	1.203	1.526
d) User at uniform rate	0.394	0.349	1.206	1.613
a) Latent factor model	0.409	0.377	1.248	1.560

TABLE III: MSE comparison of our model versus baselines.

b) *Community at uniform rate*: Users and products in a community evolve using a single "global clock" [5][15][14], where the different stages of the community evolution appear at uniform time intervals. So the community prefers different products at different times.

c) *Community at learned rate*: This extends (b) by learning the rate at which the community evolves with time, eliminating the uniform rate assumption.

d) *User at uniform rate*: This extends (b) to consider individual users, by modeling the different stages of a user's progression based on preferences and experience levels evolving over time. The model assumes a uniform rate for experience progression.

e) *User at learned rate*: This extends (d) by allowing each user to evolve on a "personal clock", so that the time to reach certain experience levels depends on the user [8].

f) *Our model with past experience level*: In order to determine how well our model captures *evolution of user experience over time*, we consider another baseline where we *randomly sample* the experience level reached by users at some timepoint *previously* in their lifecycle, who may have evolved thereafter. We learn our model parameters from the data up to this time, and again predict the user's most recent three item ratings. Note that this baseline considers textual content of user contributed reviews, unlike other baselines that ignore them. Therefore it is better than vanilla content-based methods, with the notion of past evolution, and is the strongest baseline for our model.

Discussions: Table III compares the *mean squared error (MSE)* for rating predictions, generated by our model versus the six baselines. Our model consistently outperforms all baselines, reducing the MSE by ca. 5 to 15% over the most competitive one. Improvements of our model over baselines are statistically significant at p-value < 0.0001.

The lowest improvement (over the best performing baseline in any dataset) is achieved for Amazon movie reviews. A

Experience Level 1: drank, bad, maybe, terrible, dull, shit	
Experience Level 2: bottle, sweet, nice hops, bitter, strong light, head, smooth, good, brew, better, good	
Expertise Level 3: sweet alcohol, palate down, thin glass, malts, poured thick, pleasant hint, bitterness, copper hard	
Experience Level 4: smells sweet, thin bitter, fresh hint, honey end, sticky yellow, slight bit good, faint bitter beer, red brown, good malty, deep smooth bubbly, damn weak	
Experience Level 5: golden head lacing, floral dark fruits, citrus sweet, light spice, hops, caramel finish, acquired taste, hazy body, lacing chocolate, coffee roasted vanilla, creamy bitterness, copper malts, spicy honey	

TABLE IV: Experience-based facet words for the *illustrative* beer facet *taste*.

possible reason is that the community is very diverse with a very wide range of movies and that review texts heavily mix statements about movie plots with the actual review aspects like praising or criticizing certain facets of a movie. The situation is similar for the food and restaurants case. Nevertheless, our model always wins over the best baseline from *other* works, which is typically the "user at learned rate" model.

Evolution effects: We observe in Table III that our model's predictions degrade when applied to the users' *past* experience level, compared to their *most recent* level. This signals that the model captures user evolution past the previous timepoint. Therefore the last (i.e., most recent) experience level attained by a user is most informative for generating new recommendations.

Salient words for facets and experience levels: We point out typical word clusters, with *illustrative* labels, to show the variation of language for users of different experience levels and different facets. Tables I and IV show salient words to describe the beer facet *taste* and movie facets *plot* and *narrative style*, respectively – at different experience levels. Note that the facets being latent, their labels are merely our interpretation.

BeerAdvocate and RateBeer are very focused communities; so it is easier for our model to characterize the user experience evolution by vocabulary and writing style in user reviews. We observe in Table IV that users write more descriptive and *fruity* words to depict the beer taste as they become more experienced. For movies, the wording in reviews is much more diverse and harder to track. Especially for blockbuster movies, which tend to dominate this data, the reviews mix all kinds of aspects. A better approach here could be to focus on specific kinds of movies (e.g., by genre or production studios) that may better distinguish experienced users from amateurs or novices in terms of their refined taste and writing style.

V. Related Work

State-of-the-art recommenders based on collaborative filtering [4][6] exploit user-user and item-item similarities by latent factors. The temporal aspects leading to bursts in item popularity, bias in ratings, and evolution of the entire community as a whole is studied in [5][15][14]. Other papers have studied temporal issues for anomaly detection [3]. However, none of these prior work has considered the evolving experience and behavior of individual users. The recent work [8], which is one of our baselines, modeled the influence of rating behavior on evolving user experience. However, it ignores the vocabulary and writing style of users in reviews, and their natural *smooth* temporal progression. In contrast, our work considers the review texts for additional insight into facet preferences and *smooth* experience progression.

[7][13][10] unified various approaches to generate user-specific ratings of reviews considering only the review text. However, all of these prior approaches operate in a static, snapshot-oriented manner, without considering time at all.

From the modeling perspective, most approaches learn document-specific discrete label [10], [11]. [1] proposed a complex and computationally expensive Variational Inference algorithm to incorporate continuous ratings, and [9] developed a simpler approach using Multinomial-Dirichlet Regression. The latter inspired our technique for incorporating supervision.

VI. Conclusion

Current recommender systems do not consider user experience when generating recommendations. In this paper, we have proposed an experience-aware recommendation model that can adapt to the changing preferences and maturity of users in a community. We model the *personal evolution* of a user in rating items that she will appreciate at her current maturity level. We exploit the coupling between the *facet preferences* of a user, her *experience*, *writing style* in reviews, and *rating behavior* to capture the user's temporal evolution. Our model is the first work that considers the progression of user experience as expressed in the text of item reviews.

References

[1] D. M. Blei and J. D. McAuliffe. Supervised topic models. NIPS, 2007.
[2] H. Drucker, C. J. C. Burges, L. Kaufman, A. Smola, and V. Vapnik. Support vector regression machines. NIPS, 1997.
[3] S. Günnemann, N. Günnemann, and C. Faloutsos. Detecting anomalies in dynamic rating data: A robust probabilistic model for rating evolution. KDD, 2014.
[4] Y. Koren. Factorization meets the neighborhood: A multifaceted collaborative filtering model. KDD, 2008.
[5] Y. Koren. Collaborative filtering with temporal dynamics. *Commun. ACM*, 53(4), 2010.
[6] Y. Koren and R. Bell. Advances in collaborative filtering. In *Recommender systems handbook*. 2011.
[7] J. McAuley and J. Leskovec. Hidden factors and hidden topics: Understanding rating dimensions with review text. RecSys, 2013.
[8] J. J. McAuley and J. Leskovec. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. WWW, 2013.
[9] D. Mimno and A. McCallum. Topic models conditioned on arbitrary features with dirichlet-multinomial regression. UAI, 2008.
[10] S. Mukherjee, G. Basu, and S. Joshi. Joint author sentiment topic model. SDM, 2014.
[11] D. Ramage, C. D. Manning, and S. Dumais. Partially labeled topic models for interpretable text mining. KDD, 2011.
[12] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. UAI, 2004.
[13] C. Wang and D. M. Blei. Collaborative topic modeling for recommending scientific articles. KDD, 2011.
[14] L. Xiang, Q. Yuan, S. Zhao, L. Chen, X. Zhang, Q. Yang, and J. Sun. Temporal recommendation on graphs via long- and short-term preference fusion. KDD, 2010.
[15] L. Xiong, X. Chen, T. K. Huang, J. Schneider, and J. G. Carbonell. Temporal collaborative filtering with bayesian probabilistic tensor factorization. SDM, 2010.