One-Class Recommendation with Asymmetric Textual Feedback

Mengting Wan[*] Julian McAuley[*]

Abstract

Personalized ranking with implicit feedback (e.g. purchases, views, check-ins) is an important paradigm in recommender systems. Such feedback sometimes comes with textual information (e.g. reviews, comments, tips), which could be a useful signal to reveal item properties, identify users' tastes and interpret their behavior. Although incorporating such information is common in *explicit* feedback settings (such as rating prediction), it is less common when dealing with implicit feedback, as it is often not available for negative instances (e.g. there is no review associated with the item the user *didn't* buy). Thus our goal in this study is to propose a ranking method (**PRAST**) to incorporate such personalized, asymmetric textual signals in implicit feedback settings. We evaluate our model on two real-world datasets. Quantitative and qualitative results indicate that the proposed approach significantly outperforms standard recommendation baselines, alleviates 'cold start' issues, and is able to provide potential textual interpretations for latent feedback dimensions.

1 Introduction

Implicit feedback (e.g. purchases, views, check-ins) is widely available in information systems, where users reveal their preferences through actions rather than expressing them explicitly (e.g. by providing a rating score). In addition to user-item interactions, textual information (e.g. Amazon reviews, Youtube comments, Foursquare tips) may also be available, and provides rich context to better predict or explain users' actions.

Different from item-related textual data (e.g. product description, news content), such textual information is *causal*, *personal*, *asymmetric*, and rarely studied in implicit-feedback settings. Contents could describe users' personal experiences, their favorite properties of an item, or suggestions to other users. By definition such data are only available for positive feedback instances in 'one-class' recommendation settings (e.g. review text is never available for items a user hasn't interacted with). This makes such textual information difficult to incorporate into implicit feedback settings, which typically assume that negative instances (or non-interactions) can be treated in much the same way as positive instances (see Figure 1). Our primary goal in this paper is to build personalized ranking models from implicit feedback, resolving this *asymmetry* issue and appropriately making use of textual signals.

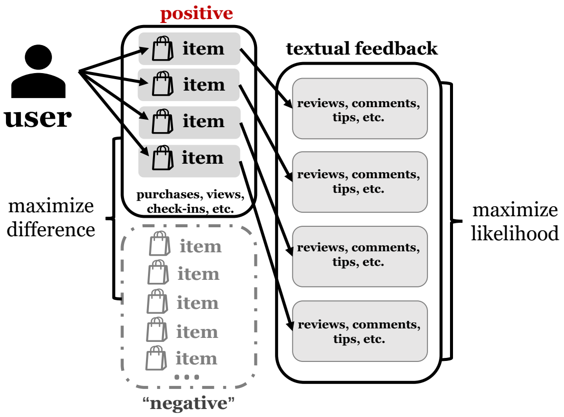

positive
user
item
item
item
item
purchases, views, check-ins, etc.
maximize difference
item
item
item
item
item
"negative"
textual feedback
reviews, comments, tips, etc.
reviews, comments, tips, etc.
reviews, comments, tips, etc.
reviews, comments, tips, etc.
maximize likelihood

Figure 1: Illustration of asymmetric textual information in implicit feedback settings.

Personalized Ranking and Implicit Feedback. In order to provide a personalized ranked list of items to each user, we need to learn users' preferences from their feedback. Explicit feedback interactions (such as star-ratings) directly reflect users' preferences regarding each item, but may be unavailable, or sparse compared to the space of possible interactions. In such cases, we might instead rely on implicit signals describing users' interactions with items. Typically, each time a user interacts with an item in the system, this is regarded as a 'positive' instance; our goal is then to predict (or rank) which items they would be most likely to interact with. However, unobserved user-item pairs cannot simply be treated as 'negative' instances: it could be that a user is not interested in the item, or that they *would be* interested but simply aren't aware of the item yet. Therefore, conventional collaborative filtering methods where the target is to predict a positive or negative signal (e.g. standard Matrix Factorization [12]) may not be appropriate. To address this, one-class algorithms have been proposed [18, 19, 22] where instead of binary prediction accuracy, ranking measures are applied as optimization criteria. For example, Bayesian Personalized

[*] University of California, San Diego, La Jolla, CA, USA; {mwan, jmcauley}@ucsd.edu

Ranking (**BPR**) [18] approximately optimizes a ranking measure, the Area Under the ROC Curve (AUC). The principle of this criterion is to promote observed (positive) items and degrade unobserved (but not necessarily negative) items by maximizing an objective that suggests that positive interactions should simply be *more* compatible with the user than non-interactions.

Incorporating Asymmetric Textual Information. Textual information associated with user-item pairs (e.g. review text) has proven helpful when explaining and predicting explicit feedback (e.g. rating prediction), particularly on 'cold' items [1, 3, 13, 14, 23]. The principle of these approaches relies on factorizing observed ratings and modeling review text by linking latent preference dimensions and topics discovered in text. The success of these methods motivates us to adopt textual information in implicit feedback settings. Specifically, rather than uncovering 'facets' from review text that explain users' ratings, we would like to use these textual signals to learn about the *types* of actions users are likely to perform. For example, we might wish to uncover the aspects of an item from Amazon reviews, or Youtube comments, which may trigger a 'purchase' or 'view' action. However, in addition to the label asymmetry described above, this textual information is also asymmetric. This means that unlike in explicit feedback settings where all responses used for training have the same side-information, this is no longer the case in implicit-feedback settings, where such textual information is only available for positive user-item pairs. We address this asymmetry and describe our goal in this paper as follows:

> **Goal:** *Given (asymmetric) textual information, we seek to understand users' inclinations towards particular kinds of actions, and provide item recommendations guided by these signals.*

In order to incorporate such textual information and overcome the challenge of asymmetry, we propose a new one-class recommendation model – ***P****airwise* ***R****anking with* ***As****ymmetric* ***T****extual Feedback* (**PRAST**), where we assume: 1) positive and negative items differ in terms of their compatibility with a given user, which is consistent with the typical pairwise ranking optimization criterion (e.g. **BPR**); 2) relevant asymmetric textual data are consistent among observed positive items, such that they provide information related to the likelihood of users' actions.

The above suggests some form of joint objective, where our model of users and items should be good at predicting (or 'explaining') observed versus non-interactions, but should also be good at explaining (in terms of likelihood or perplexity) the textual information associated with positive actions. We apply the proposed framework on two large-scale datasets and show that item ranking performance can be significantly improved by appropriately incorporating asymmetric textual data. Our experiments reveal that such side-information not only helps to provide better recommendations, but also can be used to uncover the motivations behind observed user-item interactions.

2 Related Work

Recommendation with explicit and implicit feedback. Traditional models for item recommendation rely on techniques such as Collaborative Filtering (CF) to learn from explicit feedback like star-ratings [11]. Although several paradigms for explicit feedback exist, of most relevance to us are *model-based* methods and in particular Matrix Factorization (MF) methods [12]. Such models have been extended in order to handle implicit feedback data where only positive signals (e.g. purchases, views, clicks) are observed (i.e., the so-called 'one-class' recommendation setting). Most relevant here are pair-wise methods like **BPR-MF** [18] that make an assumption that positive feedback instances are simply 'more preferable' than non-observed feedback.

Textual information with explicit signals. Several models exist that incorporate textual feedback to predict star ratings, including **HFT** ('Hidden Factors and Topics') [14], **JMARS** ('Jointly Modeling Aspects, Ratings, and Sentiments') [3], **RMR** ('Ratings Meet Reviews') [13], **FLAME** ('Factorized Latent Aspect Model') [23] and **SLUM** ('Sentiment Utility Logistic Model') [1]. These models differ from each other in precise formulation, but each essentially assumes that reviews can be used to determine the 'aspects' along which users rate products, using fewer observations than would be required to learn these aspects from ratings alone. This is a natural assumption, as the very purpose of reviews is to explain the different factors that contributed to a user's rating. We rely on a similar assumption, though the 'aspects' we seek to discover should discriminate interactions from non-interactions (e.g. purchases from non-purchases, views from non-views), and thus are quite different.

Symmetric and asymmetric information with implicit signals. Similar to the problem we tackle, several works have attempted to incorporate side-information into implicit feedback settings and have proven helpful when handling 'cold-start' issues. Examples include extensions of BPR, such as *Social BPR* (**SBPR**), which makes use of side information in the

Notation	Description
$\mathcal{U}, \mathcal{I}$	user set, item set
$\mathcal{I}_u^+, \mathcal{I}_u^-$	positive and 'negative' item sets for user u, $\mathcal{I}_u^- = \mathcal{I}_u \backslash \mathcal{I}_u^+$
$i >_u i'$	user u prefers item i over item i'
b_0, b_i, b_u	global, item, user biases
$\boldsymbol{\gamma}_i, \boldsymbol{\gamma}_u$	item and user latent factors
$x_{u,i}$	user u's preference score regarding item i
$\mathcal{R}_{u,i}, \mathcal{R}_{u,i,s}$	text corpus of the action associated document and a sentence s in this document
$\boldsymbol{\theta}_{u,i}$	topic distribution of the review text for item i from user u
$\boldsymbol{\phi}_0, \boldsymbol{\phi}_k$	word distribution for the background model and the topic k
k_s	latent variable of the assigned topic for sentence s

Table 1: Notation.

form of social signals [25], where friends' activities act as a form of implicit signal that guides users' actions; and *Visual BPR* (**VBPR**), where visual attributes are used to estimate item 'facets' that guide users' purchases [7]. In particular, some studies have been proposed to incorporate item-associated textual content (e.g. the content of an article) into this setting where topics in text are used to guide latent item dimensions [21, 24].

Although such information (social networks, images, article texts) have been shown to be effective in such cases, this is different from the setting we study as it does not exhibit the same *asymmetry*: the feedback in question is 'static' (images and article texts are used to extract item features, social networks are used to extract user features), and depends on the user or the item only, not the user-item *interaction.*

3 Background

In order to gradually construct our new framework—Pairwise Ranking with Asymmetric Textual Feedback (**PRAST**), we first introduce the traditional latent factor model and the Bayesian Personalized Ranking (**BPR**) framework as background information. Notation used throughout the paper is provided in Table 1.

Latent Factor Models. A number of modern recommender systems are built on top of latent factor models [12]. In this model, a user u's preference score regarding an item i is defined as

$$x_{u,i} = b_0 + b_i + b_u + \langle \boldsymbol{\gamma}_i, \boldsymbol{\gamma}_u \rangle , \tag{3.1}$$

where b_0 is a global offset, b_i and b_u are item/user biases, and $\boldsymbol{\gamma}_i$ and $\boldsymbol{\gamma}_u$ are K-dimensional vectors, which capture each item's latent 'properties' and users' 'preferences' toward those properties. Here $\langle \cdot, \cdot \rangle$ indicates the inner product such that $\langle \boldsymbol{\gamma}_i, \boldsymbol{\gamma}_u \rangle$ essentially captures the 'compatibility' between user u and item i, i.e., how well the item properties ($\boldsymbol{\gamma}_i$) match the corresponding user's preferences ($\boldsymbol{\gamma}_u$). This preference score $x_{u,i}$ can approximate a rating in explicit-feedback settings or can be correlated to an action probability in implicit-feedback settings, as described below.

Bayesian Personalized Ranking. Suppose $>_u$ is the desired preference ranking for user u, and $\mathcal{I}_u^+$ and $\mathcal{I}_u^-$ are the positive item set and the unobserved (or 'negative') item set. Then our training data for ranking based on implicit feedback consists of a sequence of (*user*, *positive-item*, *negative-item*) triples, i.e.,

$$\mathcal{D}_S = \{(u, i, i') \mid u \in \mathcal{U} \wedge i \in \mathcal{I}_u^+ \wedge i' \in \mathcal{I}_u^-\}. \tag{3.2}$$

In the **BPR** framework [18], the following ranking-based likelihood is optimized:

$$\prod_{u \in \mathcal{U}} P(>_u | \Omega) = \prod_{(u,i,i') \in \mathcal{D}_S} P(i >_u i' | \Omega), \tag{3.3}$$

where Ω is the parameter set. Here $i >_u i'$ indicates that user u prefers item i over item i' and its probability is usually defined via a sigmoid function:

$$P(i >_u i' | \Omega) = \sigma(x_{u,i} - x_{u,i'}) = \frac{1}{1 + e^{-(x_{u,i} - x_{u,i'})}},$$

where the latent factor model (3.1) can be applied for the preference score. $x_{u,i} - x_{u,i'}$ then corresponds to a *pairwise difference* in compatibility between the positive and negative items (note here that when ranking items for each user, the global bias b_0 and user bias b_u cancel out between $x_{u,i}$ and $x_{u,i'}$). By optimizing the log-likelihood (3.3), **BPR** approximately optimizes a ranking measure (the AUC) directly.

4 Pairwise Ranking with Asymmetric Textual Feedback

In this section, we present a new one-class recommendation model—**PRAST**, where an enhanced pairwise ranking optimization criterion is applied to handle evidence such as (asymmetric) textual information, and a relevance-aware topic model is attached to the latent factor model so that text can be incorporated adaptively.

Overview of the Framework. In order to construct a ranking framework with asymmetric textual information, we consider the likelihood of the desired rankings as well as the 'appearance probability' of the observed 'positive-only' text. Then we consider the following training data which consists of a set of (*user*, *positive-item*, *negative-item*, *evidence*) quadruples, i.e.,

$$\mathcal{D}_S = \{(u, i, i', \mathcal{E}_{u,i}) \mid u \in \mathcal{U} \wedge i \in \mathcal{I}_u^+ \wedge i' \in \mathcal{I}_u^-\}. \tag{4.4}$$

Here the evidence $\mathcal{E}_{u,i}$ could be represented either via the 'positive-only' text corpus $\mathcal{R}_{u,i}$, or empty (i.e. no textual information associated with the observed action). Then we wish to maximize (the logarithm of) the following likelihood:

$$\prod_{(u,i,i',\mathcal{E}_{u,i})\in\mathcal{D}_S} \underbrace{P_\Omega(\mathcal{E}_{u,i}|i >_u i')}_{\text{likelihood of the evidence}} \underbrace{P_\Omega(i >_u i')}_{\text{pairwise ranking}} .$$

We use P_Ω as shorthand to denote the probability given the parameter set Ω. The latent factor model (3.1) and the sigmoid transformation in **BPR** (3.3) can be applied to model $P_\Omega(i >_u i')$ as well. Thus we naturally inherit the optimization principle from **BPR**: compatibilities between positive and negative instances can be fairly compared through latent factors and the difference can be maximized. If there is textual information associated with the triple (u, i, i') (e.g. a review was left after user u purchased a product i), we define the likelihood of asymmetric evidence $\mathcal{E}_{u,i}$ as a monotonic function of the likelihood of the interaction-associated text document $\mathcal{R}_{u,i}$, i.e.,

$$P_\Omega(\mathcal{E}_{u,i}|i >_u i') = P_\Omega(\mathcal{R}_{u,i})^\kappa. \tag{4.5}$$

Here κ is a positive hyperparameter which is used to control the confidence of the underlying language model for $\mathcal{R}_{u,i}$. In order to use such textual information to explain and facilitate pairwise ranking, we need to fuse latent dimensions in (3.1) with the language model. Thus in (4.5), larger κ indicates higher confidence that observed textual data are bonded to motivations of the target action. As $\kappa \to 0$, $P_\Omega(\mathcal{R}_{u,i})^\kappa \to 1$ for all $\mathcal{R}_{u,i}$, which implies textual data are ignored and only the pairwise ranking is considered during the training process. Specifically, we define $P_\Omega(\mathcal{E}_{u,i,i'}|i >_u i') = 1$ if there is no textual information provided. Because of the asymmetry of $\mathcal{R}_{u,i}$, we always assume there is no additional information at test time and the predictions we can provide are purely a function of the user and item representations, i.e., the preference scores $x_{u,i}$ as determined by the latent factor model (3.1).

Language Model. Topic models have proven a popular approach to incorporate textual information into latent factor models, by combining Latent Dirichlet Allocation (**LDA**) [2] with latent factor models [3,13,14,23]. Just as Latent Dirichlet Allocation uncovers hidden dimensions in documents, when combined with a latent factor model it can uncover those dimensions that explain variance in people's opinions as represented by rating scores. Based on a similar principle, in implicit feedback settings, we consider distinguishing whether a sentence is relevant to the target behavior and only attach the relevant part to the latent preference dimensions, so that these relevant contents can be consistently and adaptively explained among positive items while others might be explained by a background model. We gradually build the language model as follows.

- **Sentence Relevance.** For each sentence s in a document $\mathcal{R}_{u,i}$, we introduce another binary latent variable l_s to model sentence relevance.[1] We assume that $P_\Omega(l_s = 1) = P_\Omega(l_s = 0) = 0.5$. Then the corpus likelihood in (4.5) can be modeled as

$$P_\Omega(\mathcal{R}_{u,i}) = C \prod_s \left(l_s P_\Omega^{(1)}(\mathcal{R}_{u,i,s}) + (1-l_s) P_\Omega^{(0)}(\mathcal{R}_{u,i,s})\right),$$

  where C is a constant and $P_\Omega^{(l)}(\mathcal{R}_{u,i,s})$ is shorthand for $P_\Omega(\mathcal{R}_{u,i,s}|l_s = l)$.

- **Topic Distribution.** For the relevant textual contents (i.e. $l_s = 1$), similar to **LDA**, we have a K-dimensional topic distribution $\boldsymbol{\theta}_{u,i}$ for each document, which indicates the probability that a particular word in this document discusses a particular topic. We apply the item latent factor $\boldsymbol{\gamma}_i$ in (3.1) and introduce another user-specific non-negative K-dimensional parameter $\boldsymbol{\alpha}_u$ to model this distribution as follows:

$$\theta_{u,i,k} = \frac{\exp(\alpha_{u,k}\gamma_{i,k})}{\sum_{k'=1}^K \exp(\alpha_{u,k'}\gamma_{i,k'})}, \tag{4.6}$$

  where $\alpha_{u,k} \geq 0, \forall k$. Recall that $\boldsymbol{\gamma}_i$ captures item i's 'properties' in the latent factor model, and here we use $\boldsymbol{\alpha}_u$ to capture variation due to user u's writing style (i.e., a user-specific topic weighting determining which topics this user prefers to write about in their reviews). By applying such a transformation, we are assuming that if an item exhibits certain properties which may motivate users to take actions (i.e., high $\gamma_{i,k}$), then these aspects should be reflected in users' reviews, so long as that user has a tendency to discuss them (high $\alpha_{u,k}$). This modeling approach is an enhanced version of that of the **HFT** model [14] proposed for explicit-feedback settings, where $\alpha_{u,k}$ is assumed to be constant among all users and all topics.

- **Topic Assignment.** Furthermore, we assume that words within a sentence s discuss the same aspect k_s (i.e., topic) of the item. Such a sentence-level topic is generated from a multinomial distribution with its corresponding review topic parameter $\boldsymbol{\theta}_{u,i}$. Note here the each sentence-level topic k_s is a latent variable in the probabilistic model, which is usually estimated through sampling approaches [17] or variational inference [20].

[1] $l_s = 1$ indicates this sentence is relevant to motivations of the observed user-item interaction; $l_s = 0$ otherwise.

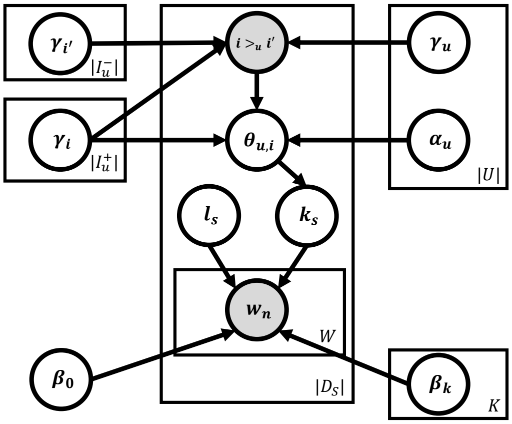


Figure 2: Plate-notation illustration of the proposed **PRAST** model.

- **Word Distribution.** Suppose $\mathcal{W}$ is the dictionary used in the model, $\boldsymbol{\beta}_0, \boldsymbol{\beta}_k, k = 1, \ldots, K$ are W-dimensional vectors where W is the dictionary size. Then given the sentence relevance and topic assignment, we could generate the complete review text $\mathcal{R}_{u,i}$ from word distributions. Specifically, for a relevant sentence, given the topic assignment k_s for a word w_n, its likelihood can be modeled as

  $$\phi_{w_n,k_s} \coloneqq P_\Omega(w_n|k_s, l_s = 1) = \frac{\exp(\beta_{k_s,w_n})}{\sum_{w' \in \mathcal{W}} \exp(\beta_{k_s,w'})}.$$

  For an irrelevant sentence, we have the following background word distribution:

  $$\phi_{w_n,0} \coloneqq P_\Omega(w_n|l_s = 0) = \frac{\exp(\beta_{0,w_n})}{\sum_{w' \in \mathcal{W}} \exp(\beta_{0,w'})}.$$

Therefore, the final likelihood of textual information in each sentence is

$$P_\Omega(\mathcal{R}_{u,i,s}) = l_s\big(\sum_{k_s} \theta_{u,i,k_s} \prod_w \phi_{w,k_s}\big) + (1-l_s)\big(\prod_w \phi_{w,0}\big).$$

The graphical representation of the complete model is included in Figure 2.

Model Inference. We apply an EM-style variational inference method to fit the text term $P_\Omega(\mathcal{R}_{i,u}|i >_u i')$ and the ranking term $P_\Omega(i >_u i')$ jointly, which is similar to the techniques applied in previous textual model studies [4, 23]. To do so we introduce an intermediate parameter τ_s for sentence relevance indicator l_s, which can be easily updated in the E-step: $\tau_s = P_\Omega^{(1)}(\mathcal{R}_{u,i,s})/(P_\Omega^{(1)}(\mathcal{R}_{u,i,s}) + P_\Omega^{(0)}(\mathcal{R}_{u,i,s}))$. Then our target is to maximize the following log-likelihood for each sentence:

$$\tau_s \log P_\Omega^{(1)}(\mathcal{R}_{u,i,s}) + (1-\tau_s) \log P_\Omega^{(0)}(\mathcal{R}_{u,i,s}) \tag{4.7}$$

Then we introduce another set of K-dimensional variational parameters $\boldsymbol{\pi}_s$ to approximate the distribution of the sentence topic assignment k_s, i.e., the variational probability is $q(k_s = k|\boldsymbol{\pi}_s) = \pi_{s,k}$, with the constraint $\sum_k \pi_{s,k} = 1$. Instead of optimizing the original text-related log-likelihood $\log P_\Omega^{(1)}(\mathcal{R}_{u,i,s})$, we maximize the lower-bound of this log-likelihood as

$$\begin{aligned} &\mathbb{E}_q \log P_\Omega^{(1)}(\mathcal{R}_{u,i,s}) - \mathbb{E}_q \log q(k_s|\boldsymbol{\pi}_s) \\ =&\sum_k \pi_{s,k}\Big(\log \theta_{u,i,k} + \sum_{w_n \in \mathcal{R}_{u,i,s}} \log \phi_{w_n,k} - \log \pi_{s,k}\Big) \\ =&\sum_k \pi_{s,k}\Big(\log \theta_{u,i,k} + \sum_{w \in \mathcal{W}} \log N_{s,w}\phi_{w,k} - \log \pi_{s,k}\Big), \end{aligned}$$

where $N_{s,w}$ is the frequency of word w in sentence s. In practice, we first fix all other parameters and update τ_s and $\pi_{s,k} \propto \theta_{u,i,k} \prod_w \phi_{w,k}^{N_{s,w}}$. Then we fix $\tau_s, \pi_{s,k}$ and update other parameters to maximize the above lower-bound plus the log-likelihood of the background language model $\log P_\Omega^{(0)}(\mathcal{R}_{u,i,s})$ and the pairwise ranking $\log P_\Omega(i >_u i')$.

Gaussian priors are included for all parameters in Ω, leading to a standard ℓ_2 regularizer. In addition, we apply the **ADAM** optimizer [10], a stochastic gradient-based algorithm. Recall that our primary goal is to produce rankings that are consistent with our training data (i.e., positive instances should be ranked highly). Thus we need to be careful not to overfit too much to side information, which would sacrifice ranking quality. Rather, the textual information is intended to regularize or 'reinforce' the model's latent factors, in order to lead to better ranking performance. Therefore, during stochastic optimization, we periodically compute the ranking measure (i.e., the AUC) on a held-out validation set. We report results on the test set for the model parameters, hyperparameters, and the iteration, leading to the best performance on the validation set.

5 Experiments

We evaluate the proposed **PRAST** model for personalized item ranking on two large-scale datasets where asymmetric textual feedback is available. In particular, we evaluate 1) whether overall item rankings can be estimated more accurately by leveraging such signals; 2) whether *cold start* issues for items can be alleviated; 3) whether latent preference dimensions can be reasonably explained by textual information and motivation-relevant topics can be discovered.

Datasets. We consider two large-scale datasets—`Amazon` [15] and `Google Local` [6], where both review text and ratings are available. Recall that we do not use rating information (instead we are trying to

Amazon	#act.	#users	#items	#act. /item	#sent.	#sent. /act.	Google Local	#act.	#users	#items	#act. /item	#sent.	#sent. /act.
Instant Video	135K	29,756	15,149	8.92	608K	4.50	Colorado	72K	10,512	27,984	2.57	233K	3.24
Office Prod.	287K	59,858	60,641	4.73	1,540K	5.37	North Carolina	73K	10,644	33,071	2.21	214K	2.93
Digital Music	352K	56,814	156,503	2.25	1,821K	5.18	Washington	78K	9,699	29,644	2.63	194K	2.49
Baby	380K	71,826	42,523	8.94	2,071K	5.45	Illinois	135K	17,098	42,329	3.18	377K	2.80
Pet Supplies	478K	93,336	70,105	6.82	2,331K	4.88	Florida	182K	28,898	81,205	2.24	539K	2.96
Grocery	509K	86,400	108,467	4.69	2,390K	4.70	New York	225K	22,199	61,790	3.65	579K	2.57
Health	1,073K	205,704	163,717	6.56	5,126K	4.78	Texas	266K	35,547	96,597	2.75	761K	2.86
Cell Phones	1,079K	245,110	190,089	5.67	4,664K	4.32	California	430K	48,957	145,779	2.95	982K	2.28
Total	4,293K	848,804	807,194	5.32	20,550K	4.79	Total	1,461K	183,554	518,399	2.82	3,879K	2.66

Table 2: Basic dataset statistics: numbers of actions (i.e. reviews), users, items, sentences, actions per item, sentences per document.

predict what items a user would ‘interact’ with, such as what business they would visit), except when adopting explicit-feedback models for comparison.

- **Amazon.** This is a large-scale dataset collected from *Amazon.com* [15]. We consider products in eight top-level categories: `Instant Video`, `Office Products`, `Digital Music`, `Baby`, `Pet Supplies`, `Grocery and Gourmet Food`, `Health and Personal Care` and `Cell Phones and Accessories`. We discard users with fewer than 3 associated actions (i.e., reviews) in total leaving around 4 million actions across 807 thousand items and 849 thousand users. Textual information is available for almost all actions. Models are built independently for different categories and the per-category statistics are included in Table 2.
- **Google Local.** The `Google Local` dataset was introduced in a recent paper [6], which contains reviews about local businesses worldwide. We extract businesses from the following states in the US: `Colorado`, `North Carolina`, `Washington`, `Illinois`, `Florida`, `New York`, `Texas` and `California`. Similarly we discard users with fewer than 3 actions and build models independently for different states. This results in around 1 million actions across 518 thousand items and 184 thousand users, around 71% of which have associated textual information. Compared with `Amazon`, `Google Local` is a relatively sparse dataset in terms of actions associated with items, and contains relatively shorter reviews.

Intuitively we consider ‘review’ actions as positive feedback in our experiments, i.e., we regard all of the reviewed user-item pairs as positive. Appearance of this action indicates that a user bought a product or visited a place. Different forms of implicit feedback could be considered (such as clicks or purchases, if such data were available), but using the presence of reviews is desirable as it allows us to straightforwardly compare against models designed for explicit feedback settings, as described below.

Baselines and Evaluation Methodology. Besides the proposed **PRAST** model, we consider the following implicit-feedback baselines:

- **itemPop.** As item popularity (i.e., the number of previous actions regarding an item) could be a significant component in item ranking, we simply use the count of positive responses for each item in the training set as its preference score so that items are ranked in terms of their popularity.
- **BPR.** This is a state-of-the-art implicit-feedback pairwise ranking model. As we introduced previously, a latent factor model is applied to generate item preference scores.
- **WARP.** Weighted Approximate-Rank Pairwise [22] is another state-of-the-art loss for Top-K recommendation, which penalizes positive items at lower rank heavily. Specifically, we apply a penalizing scheme similar to [8], where a positive item i based on its rank $w_{i,u} = \log(rank_{i,u} + 1)$.
- **WRMF.** Weighted Regularized Matrix Factorization [9, 16] is another family of implicit-feedback models, where standard matrix factorization is applied and an additional weight is introduced to model unobserved interactions, i.e., the loss function takes the form $\sum_{u,i} c_{u,i}(y_{u,i} - x_{u,i})^2$, where $y_{u,i} \in \{0, 1\}$ is the label of the feedback and $c_{u,i}$ is usually set to be large for positive feedback but small for non-interactions.

Comparing these implicit-feedback methods against **PRAST** allows us to measure the influence of (asymmetric) textual feedback in terms of ranking quality.

In addition, we consider two more alternatives

Dataset	Metric	itemPop	BPR	WARP	WRMF	HFT-b	CAPRF-b	PRAST	improv. vs. BPR	improv. vs. HFT	improv. vs. best
Amazon (overall)	AUC	0.7806	0.7990	0.7917	0.7881	0.7724	0.7846	0.8194	2.55%	6.09%	2.55%
	NDCG	0.1079	0.1052	0.1071	0.1077	0.1010	0.0984	0.1082	2.88%	7.15%	0.29%
Amazon (cold)	AUC	0.5675	0.6000	0.5869	0.5804	0.5588	0.5691	0.6364	6.06%	13.88%	6.06%
	NDCG	0.0713	0.0711	0.0712	0.0713	0.0705	0.0706	0.0715	0.49%	1.32%	0.28%
Google Local (overall)	AUC	0.5458	0.6731	0.5932	0.5730	0.5718	0.5751	0.7068	5.00%	23.60%	5.00%
	NDCG	0.0809	0.0786	0.0766	0.0805	0.0825	0.0813	0.0869	10.51%	5.29%	5.29%
Google Local (cold)	AUC	0.5043	0.6459	0.5625	0.5339	0.5323	0.5346	0.6798	5.25%	27.70%	5.25%
	NDCG	0.0752	0.0761	0.0735	0.0746	0.0762	0.0747	0.0804	5.59%	5.49%	5.49%

Table 3: Results on `Amazon` and `Google Local` (average metric across the complete dataset). The best performance is underlined and the last column shows the percentage improvement of **PRAST** over the strongest baseline.

that make use of the same textual information: 1) a representative probabilistic model from a series of methods where review text is incorporated into rating prediction, and 2) a state-of-the-art model designed for point-of-interest (POI) recommendation where 'tip' texts are included in order to estimate the number of users' check-ins. In particular we consider the following two models:

- **HFT-b.** Hidden Factors as Topics (**HFT**) [14] is an explicit-feedback approach which models both review text and ratings. We still consider the observed reviews only and replace the original Meas Squared Error (MSE) loss by a binary cross-entropy loss: $y_{u,i}\log\sigma(x_{u,i})+(1-y_{u,i})\log(1-\sigma(x_{u,i}))$, where $y_{u,i}=1$ if the rating score is larger or equal to 3 and $y_{u,i}=0$ otherwise.
- **CAPRF-b.** The Context-Aware POI Recommendation Framework **CAPRF** [5] applies a similar loss function to **WRMF** but the number of check-ins is regarded as a label $y_{u,i}$ and all non-interactions are discarded. Here, tip texts are modeled as an additional regularization of item- and user- latent factors through linear embeddings. In our case, similar to **HFT**, we replace the number of check-ins by a binary label based on rating score.

Note that the above two baselines use the same textual information as our method but discard all non-interactions; thus they require minor adaptation to apply them in our implicit-feedback setting: The basic assumption behind our adaptation of these methods is that users are likely to interact with (purchase, visit, or consume) items for which they are predicted to exhibit a high preference score, based on their explicit signals (e.g. high rating scores, multiple check-ins). By comparing these two methods against **PRAST**, we address the difference between explicit-feedback and implicit-feedback objectives and evaluate the influence of taking abundant unobserved interactions into consideration given the same amount of textual information.

As our goal is to provide high-quality personalized item rankings, we adopt the Area Under the ROC Curve (AUC) as the overall evaluation measure (which is also the criterion that **BPR** variants optimize), as well as Normalized Discounted Cumulative Gain (NDCG) as a top-biased ranking measure:

$$\begin{aligned} AUC &= \frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\frac{1}{|\mathcal{I}_u^+||\mathcal{I}_u^-|}\sum_{i\in\mathcal{I}_u^+,i'\in\mathcal{I}_u^-}\delta(i>_u i'), \\ NDCG &= \frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\frac{\sum_{i\in\mathcal{I}_u^+}1/\log_2(rank_{i,u}+1)}{\sum_{0\le k<|\in\mathcal{I}_u^+|}1/\log_2(k+1)} \end{aligned} \tag{5.8}$$

where $\delta(E)$ is an indicator function that takes the value 1 iff E is true.

Results. Following [14] we set $K=10$ for the dimensionality of latent factor vectors and the number of topics. The confidence parameter for the language model κ is set to be 0.1 in all the experiments and the regularization parameter $\lambda\in\{0.01,0.05,0.1,1\}$ is selected based on validation performance. We apply leave-one-out evaluation, where for each dataset, we sample 5000 users and their last action for testing, and their second-to-last action for validation. All other actions in the dataset are used for training. All results are reported on the held-out test data.

We include the overall results in terms of the AUC and NDCG on `Amazon` and `Google Local` datasets in Table 3. To address the 'cold-start' problem, in addition to the complete dataset, we report the performance on 'cold' items where the number of associated actions is less than 5. For brevity, we include only the average AUC/NDCG (across all categories and states) for `Amazon` and `Google Local`, and provide barplots of AUC (on the complete dataset) for each product category and each state in Figure 3.

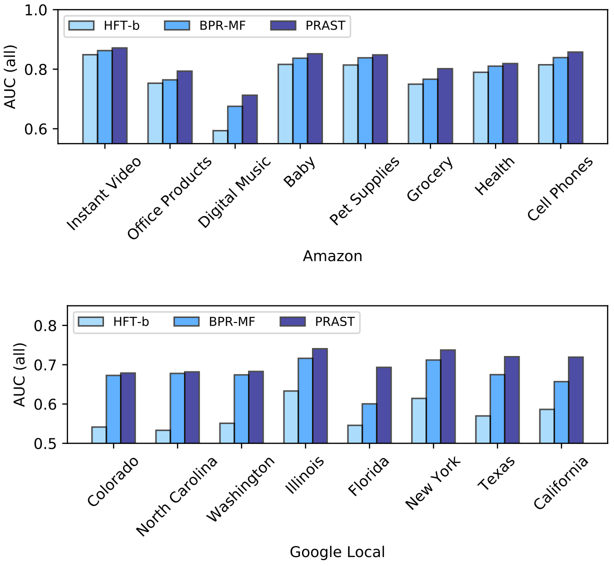
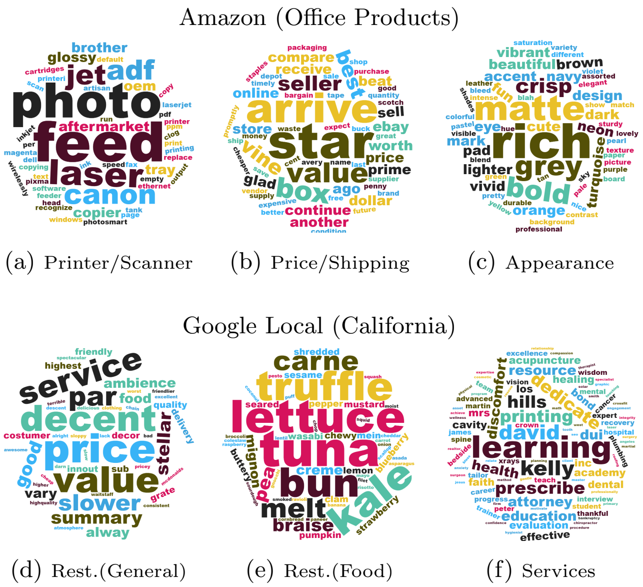


Figure 3: Results for each category in `Amazon` and for each state in `Google Local` in terms of the AUC.



(a) Printer/Scanner (b) Price/Shipping (c) Appearance

(d) Rest.(General) (e) Rest.(Food) (f) Services

Figure 4: Word clouds from three selected topics addressed in textual feedback from `Amazon (Office Products)` and `Google Local (California)`.

From Table 3 we notice that **PRAST** significantly outperforms standard implicit-feedback baselines and adjusted explicit-feedback baselines in terms of overall ranking (AUC), especially when recommending 'cold' items. This indicates that appropriately incorporating textual information into a ranking loss can improve personalized item recommendations. Based on Figure 3 and Table 2, this improvement is substantial on 'sparse' datasets in terms of the number of actions per item (e.g. `Amazon (Digital Music)`) but less significant on relatively 'dense' datasets (e.g. `Amazon (Instant Video)`). For `Google Local`, we observe limited impact of textual information on small datasets (e.g. `Colorado`). One possible reason for this could be the lack of data to fit high-dimensional language models.

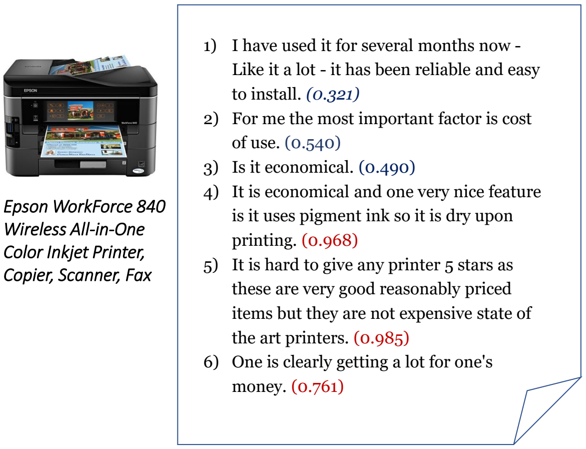
For top-biased evaluation (NDCG), **PRAST** outperforms baselines on both `Amazon` and `Google Local` datasets, though the improvement is relatively limited on `Amazon`. This is possibly because the number of items is very large and item popularity often dominates user preferences (especially on some `Amazon` categories) so that improving a top-biased ranking metric is relatively difficult for 'cold' items.

Qualitative Analysis. We examine the top words for the topics discovered from **PRAST** based on the normalized topic-specific word likelihood $\frac{\phi_{w,k}}{\sum_{k'} \phi_{w,k'}}$. Such topics can be used to explain latent preference/motivation dimensions. Word clouds of three topics discussed in `Amazon (Office Products)` and `Google Local (California)` are shown in Figure 4. In general, topics uncovered from `Amazon (Office Products)` are a mixture of genres (e.g. printer/scanner in Figure 4a) and aspects (e.g. price/shipping in Figure 4b or product appearance Figure 4c), which reveal what product a user wants to buy and what aspect a user cares about when making a purchase. Similarly, we see categories (e.g. education/medical/law services in Figure 4f) from topics in `Google Local (California)` and even different viewpoints for a particular category (e.g. general aspects like price and service for restaurants in Figure 4d, and particular foods in Figure 4e).

In Figure 5, we provide an example review for an item with relatively high scores for the item latent factors in γ_i associated with the topic 'print/scanner' and the topic 'price/shipping'. We split the review text into sentences and provide the estimated relevance score τ_s for each sentence in parentheses. We notice that **PRAST** can automatically highlight the sentences containing genre-specific keywords (e.g. ink, printer) and price-related contents (e.g. cost, price, money, economical) by assigning high relevance scores. This suggests that in addition to improving ranking quality, the **PRAST** model is capable of explaining latent feedback dimensions and detecting the most relevant textual content.

6 Conclusions and Future Work

We presented **PRAST**, a one-class recommendation framework that allows us to make use of asymmetric textual information in implicit feedback settings. In order to overcome the challenge of asymmetry (i.e., side


Epson WorkForce 840 Wireless All-in-One Color Inkjet Printer, Copier, Scanner, Fax

1) I have used it for several months now - Like it a lot - it has been reliable and easy to install. *(0.321)*
2) For me the most important factor is cost of use. (0.540)
3) Is it economical. (0.490)
4) It is economical and one very nice feature is it uses pigment ink so it is dry upon printing. (0.968)
5) It is hard to give any printer 5 stars as these are very good reasonably priced items but they are not expensive state of the art printers. (0.985)
6) One is clearly getting a lot for one's money. (0.761)

Figure 5: An example review selected from an item with large scores on the 'printer/scanner' and the 'price/shipping' dimensions, where the estimated sentence relevance scores τ_s are provided in parentheses.

information that is only available for *positive* instances), we introduced a new optimization criterion incorporating a language model where preference factors and textual topics are matched in a relevance-aware way. Our experiments on two large datasets revealed that asymmetric textual information (like review text) leads to substantial performance improvements, and that such performance cannot be obtained by naïvely adapting existing explicit feedback models.

The principle of **PRAST** can be extended to incorporate other types of 'positive-only' side information (e.g. transaction timestamps and geo-tags, review helpfulness, product prices in transaction logs, etc.). As future work, these asymmetric signals can be modeled with the proposed pairwise ranking criterion, and potentially serve as informative context for better recommendations.

References

[1] K. Bauman, B. Liu, and A. Tuzhilin. Aspect based recommendations: Recommending items with the most valuable aspects based on user reviews. In *SIGKDD*, 2017.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *JMLR*, 2003.

[3] Q. Diao, M. Qiu, C.-Y. Wu, A. J. Smola, J. Jiang, and C. Wang. Jointly modeling aspects, ratings and sentiments for movie recommendation (jmars). In *SIGKDD*, 2014.

[4] J. Eisenstein, A. Ahmed, and E. P. Xing. Sparse additive generative models of text. In *ICML*, 2011.

[5] H. Gao, J. Tang, X. Hu, and H. Liu. Content-aware point of interest recommendation on location-based social networks. In *AAAI*, 2015.

[6] R. He, W.-C. Kang, and J. McAuley. Translation-based recommendation. In *RecSys*, 2017.

[7] R. He and J. McAuley. Vbpr: Visual bayesian personalized ranking from implicit feedback. In *AAAI*, 2016.

[8] C.-K. Hsieh, L. Yang, Y. Cui, T.-Y. Lin, S. Belongie, and D. Estrin. Collaborative metric learning. In *WWW*, 2017.

[9] Y. Hu, Y. Koren, and C. Volinsky. Collaborative filtering for implicit feedback datasets. In *ICDM*, 2008.

[10] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[11] Y. Koren and R. Bell. Advances in collaborative filtering. In *Recommender Systems Handbook*. Springer, 2011.

[12] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 2009.

[13] G. Ling, M. R. Lyu, and I. King. Ratings meet reviews, a combined approach to recommend. In *RecSys*, 2014.

[14] J. McAuley and J. Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. In *RecSys*, 2013.

[15] J. McAuley, C. Targett, Q. Shi, and A. Van Den Hengel. Image-based recommendations on styles and substitutes. In *SIGIR*, 2015.

[16] R. Pan, Y. Zhou, B. Cao, N. N. Liu, R. Lukose, M. Scholz, and Q. Yang. One-class collaborative filtering. In *ICDM*, 2008.

[17] I. Porteous, D. Newman, A. Ihler, A. Asuncion, P. Smyth, and M. Welling. Fast collapsed gibbs sampling for latent dirichlet allocation. In *SIGKDD*, 2008.

[18] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. BPR: bayesian personalized ranking from implicit feedback. In *UAI*, 2009.

[19] Y. Shi, A. Karatzoglou, L. Baltrunas, M. Larson, N. Oliver, and A. Hanjalic. Climf: learning to maximize reciprocal rank with collaborative less-is-more filtering. In *RecSys*, 2012.

[20] Y. W. Teh, D. Newman, and M. Welling. A collapsed variational bayesian inference algorithm for latent dirichlet allocation. In *NIPS*, 2006.

[21] C. Wang and D. M. Blei. Collaborative topic modeling for recommending scientific articles. In *SIGKDD*, 2011.

[22] J. Weston, S. Bengio, and N. Usunier. Large scale image annotation: learning to rank with joint word-image embeddings. *Machine learning*, 2010.

[23] Y. Wu and M. Ester. Flame: A probabilistic model combining aspect based opinion mining and collaborative filtering. In *WSDM*, 2015.

[24] W. Yao, J. He, H. Wang, Y. Zhang, and J. Cao. Collaborative topic ranking: Leveraging item metadata for sparsity reduction. In *AAAI*, 2015.

[25] T. Zhao, J. McAuley, and I. King. Leveraging social connections to improve personalized ranking for collaborative filtering. In *CIKM*, 2014.