# The FLDA Model for Aspect-based Opinion Mining: Addressing the Cold Start Problem

Samaneh Moghaddam
School of Computing Science
Simon Fraser University
Burnaby, BC, Canada
sam39@cs.sfu.ca

Martin Ester
School of Computing Science
Simon Fraser University
Burnaby, BC, Canada
ester@cs.sfu.ca

## ABSTRACT

Aspect-based opinion mining from online reviews has attracted a lot of attention recently. The main goal of all of the proposed methods is extracting aspects and/or estimating aspect ratings. Recent works, which are often based on Latent Dirichlet Allocation (LDA), consider both tasks simultaneously. These models are normally trained at the item level, i.e., a model is learned for each item separately. Learning a model per item is fine when the item has been reviewed extensively and has enough training data. However, in real-life datasets such as those from Epinions.com and Amazon.com more than 90% of items have less than 10 reviews, so-called *cold start items*. State-of-the-art LDA models for aspect-based opinion mining are trained at the item level and therefore perform poorly for cold start items due to the lack of sufficient training data. In this paper, we propose a probabilistic graphical model based on LDA, called *Factorized LDA (FLDA)*, to address the cold start problem. The underlying assumption of FLDA is that aspects and ratings of a review are influenced not only by the item but also by the reviewer. It further assumes that both items and reviewers can be modeled by a set of latent factors which represent their aspect and rating distributions. Different from state-of-the-art LDA models, FLDA is trained at the category level and learns the latent factors using the reviews of all the items of a category, in particular the non cold start items, and uses them as prior for cold start items. Our experiments on three real-life datasets demonstrate the improved effectiveness of the FLDA model in terms of likelihood of the held-out test set. We also evaluate the accuracy of FLDA based on two application-oriented measures.

## Categories and Subject Descriptors

I.7.0 [**Document and Text Processing**]: General; G.3 [**Mathematics of Computing**]: Probability and Statistics—*statistical computing, multivariate statistics*

## General Terms

Algorithms, Design, Experimentation

## Keywords

aspect-based opinion mining, cold start item, latent dirichlet allocation, aspect identification, rating prediction, user modeling


*WWW 2013*, May 13–17, 2013, Rio de Janeiro, Brazil.
ACM 978-1-4503-2035-1/13/05.

## 1. INTRODUCTION

The process of buying a product or selecting a service is usually started with a series of inquires about possible options. Nowadays the Web has become an excellent source of user opinions which answers all of the user's questions by itself. However, the amount of information to be read for decision making is hugely overwhelming. There are now hundreds of Web resources containing user opinions, e.g., reviewing websites, forums, discussion groups, and Blogs, etc. While in most of the reviewing websites, reviewers assigned overall ratings (as stars) to express the overall quality of reviewed items, most of the readers usually need more detailed information than a single rating to make the final decision. For example, in the process of buying a digital camera, one may want to know the quality of zoom, while another may only care about the ease of use. A 4-star rating for a specific camera may convince none of them to purchase it without further review reading.

One of the emerging problems in opinion mining, which attracted a lot of attention recently, is aspect-based opinion mining [17]. Aspect-based opinion mining consists of two main tasks: 1) Extracting major aspects of items from user reviews, and 2) Predicting the rating of each aspect based on the sentiments reviewers used to describe that aspect. Aspects which are attributes or components of items are usually commented on in reviews to give an overview of the item quality. For example, 'zoom', 'LCD' and 'battery life' are some of the aspects of digital cameras. Reviewers express the quality of each aspect using sentiments which are usually adjectives, e.g., 'good zoom', 'blurry LCD' and 'poor battery life'. These sentiments show the level of users' satisfaction regarding the quality of each aspect. To provide a summary of review, aspect-based opinion mining techniques try to interpret sentiments as numerical ratings (usually in the range from 1 to 5) to provide a rated aspect summary of reviews, e.g., 'zoom: 4', 'LCD: 2', and 'battery life: 1'.

In the last decade several latent variable models have been proposed to address the problem of aspect-based opinion mining, e.g., [26, 31, 3, 23, 30, 8, 19, 5, 16, 20]. All of these models are applied at the item level, i.e., they learn one model per item from the reviews of that item. Learning a model per item is logical as the rating of an aspect depends on the aspect quality which usually differs for different items. However, an issue that has been neglected in all of the current works is that latent variable models are not accurate if there is not enough training data. In our recent work [21], we evaluated the impact of the size of the training dataset on models for aspect-based opinion mining. We discussed a series of increasingly sophisticated LDA models representing the essence of the major published methods in the literature. Our comprehensive evaluation of these models on a real-life data set proved that while item level models work well for items with large number of reviews, they perform poorly when the size of the training dataset is

small. In fact, the experimental evaluation showed that the basic LDA model outperforms the more complex models for these items. Borrowing a term from the recommender systems literature, we call such items *cold start items*. In real-life data sets such as those from Epinions.com and Amazon.com more than 90% of items are cold start (less than 10 reviews) which indicates there is a great need for accurate opinion mining models for these items.

In this paper, we introduce the problem of identifying aspects and estimating their ratings for cold start items. To address this problem, we propose a probabilistic graphical model based on LDA, called *Factorized LDA (FLDA)*. The underlying assumption of this model is that the aspects and corresponding ratings of reviews are influenced not only by the items but also by the reviewers. It further assumes that both items and reviewers can be modeled by a set of latent factors. Item factors represent the item's probability distribution over aspects and for each aspect its distribution over ratings. In the same way, reviewer factors represent the reviewer's probability distribution over aspects and for each aspect its distribution over ratings. FLDA generates aspects and ratings of reviews by learning the latent factors of items and reviewers.

Different from state-of-the-art LDA models which are learned per item, FLDA is trained at the category level. Note that, a category of items is a set of items sharing common characteristics, e.g., MP3 players, scanners, Bed and Breakfast Inns, etc. FLDA generates each aspect of a review based on both the aspect distribution of the corresponding item and the aspect distribution of the reviewer. It further generates the rating of an aspect depending on that aspect, the rating distribution of that aspect for that item and the rating distribution of that aspect for the reviewer. These distributions are trained using the reviews of all the items of a category, in particular the non cold start items, and serve as prior for the distributions of cold start items that otherwise could not be learned accurately. In other words, for cold start items the aspect distribution is mainly determined by the prior aspect distribution of the category, and the rating distribution of an aspect is mainly determined by the rating distribution of the reviewer or by the prior rating distribution of all reviewers (if the reviewer is cold start, i.e., has written few reviews). On the other hand, for non-cold start items the aspect and rating distributions are mainly determined by the observed reviews of that item.

We report the results of our extensive experiments on three real-life datasets from Epinions, Amazon, and TripAdvisor. The results demonstrate the improved effectiveness of the FLDA model in terms of likelihood of the held-out test set, in particular for cold start items. We also evaluate the accuracy of FLDA based on two application-oriented measures: item categorization and overall rating prediction for reviews. Both applications are performed based on the learned latent factors. We evaluate these applications by comparing the accuracy of the learned classifiers with the state-of-the-art techniques.

The remainder of the paper is organized as follows. The next section is devoted to related work. Section 3 introduces the problem statement and discusses our contribution. Section 4 presents the proposed model, FLDA. Section 5 describes the inference and estimation techniques for FLDA. In Sections 6 and 7, we report the results of our experimental evaluation and discuss two applications of our model. Finally, Section 8 concludes the paper with a summary and the discussion of future work.

## 2. RELATED WORK

Most of the early works on aspect-based opinion mining are frequency-based approaches [7, 18, 1, 22]. These methods usually mine frequent noun phrases and filter them using certain constraints to identify aspects. These techniques tend to produce too many non-aspects and miss low-frequency aspects [5]. In addition, frequency based approaches require the manual tuning of various parameters which makes them hard to port to another dataset [20]. Addressing these weaknesses, latent variable models automatically learn the model parameters from the data.

While some of the proposed latent variable models are based on Conditional Random Field [14, 4] or Hidden Markov Model [29, 8], most of them are based on Latent Dirichlet Allocation (LDA), e.g., [26, 25, 31, 3, 10, 30, 28, 13, 15, 16, 6]. LDA is a generative probabilistic model of a corpus [2]. The basic idea of this model is that documents are represented as mixtures over latent topics where topics are associated with a distribution over the words of the vocabulary. All of the existing LDA-based opinion mining models are trained at the item level, i.e., from the reviews of a given item. In the following we will discuss the most recent and most important LDA-based models presented in the literature.

The model of [3] assumes that all words in a single sentence are generated from one topic and apply LDA on each sentence to extract topics (as aspects). The authors of [10] further extend the model of [3] to extract sentiments related to each aspect. In this model, each review has a distribution over sentiments and each sentiment has a distribution over aspects. To generate each sentence, a sentiment is first sampled from the review's sentiment distribution and then an aspect is chosen conditioned on the selected sentiment. Each word of the sentence is then generated based on the selected aspect and sentiment.

An LDA-based model for jointly identifying aspects and sentiments is proposed in [31]. This model assumes each review has a distribution over aspects and another distribution over sentiments. The authors further assume there are two types of sentiments in a review: aspect-specific sentiments which are each associated with only a single aspect (e.g., 'tasty' which is associated with 'food'), and general sentiments which are shared across different aspect (e.g., 'great'). They use two indicator variables to distinguish between aspects and sentiments based on their Part-Of-Speech (POS) tags.

The authors of [13] also assume different word distributions for aspects, sentiments, and also background words (other words). The model determines whether the word is an aspect, a sentiment, or a background word, based on the POS tag of that word and the POS tag of the previous word. This model generates each word of a review by first choosing an aspect and a sentiment from the corresponding distributions. Then a word is generated conditioned on both aspect and sentiment. The rating of each sentiment is also computed using a normal linear model learned by the overall rating of review.

In [28] an LDA model is proposed to identify aspects, their ratings, and the weight placed on each aspect by the reviewer. This model takes the overall ratings assigned by reviewers to that product as input. It first samples an aspect from the learned distribution and selects a word conditioned on the that aspect. Then the sampled word and the aspect together generate the rating of the aspect. Finally, the aspect weights are sampled from a normal distribution and the overall rating of review is generated based on the weighted sum of all the aspect ratings.

The model proposed in [26] considers two types of topics for each review: global topics and local topics. Global topics correspond to global properties of the product (e.g., product brand) and local topics are related to the product aspects. The model uses an indicator variable to select the type of topic for generating each word of a review. An existing ranking algorithm is used to estimate the rating of aspects based the learned variables. This model

is further extended in [25] to find the correspondence between the extracted topics and the product aspects.

The authors of [30] propose a model generating opinion phrases (pairs of candidate aspect and related sentiment). They first apply a set of predefined POS patterns on the review text to extract nouns and related adjectives as opinion phrases. Then the basic LDA model is applied on each sentence to cluster opinion phrases into $k$ groups. Similar to [30], in [20] an LDA model, called ILDA, is proposed to learn from opinion phrases. To identify opinion phrases, a set of POS patterns are first mined using a seed set of aspects and related sentiments. These patterns are then applied on reviews to extract opinion phrases. ILDA assumes the dependency between aspects and ratings. To generate each opinion phrase, an aspect is first chosen from a Dirichlet distribution and then a rating is selected conditioned on the chosen aspect. An opinion phrase is finally generated based on the selected aspect and rating.

In our recent work [21], we present a set of guidelines for designing LDA-based models by comparing a series of increasingly sophisticated probabilistic graphical models based on LDA. We start with the basic LDA model and then gradually extend the model by adding latent and observed variables as well as dependencies. We argue that these models represent the essence of the major published methods and allow us to tease apart the impact of various design decisions. In addition to design choices, we further evaluate the impact of the size of the training dataset and the performance of different techniques for extracting opinion phrases. We conduct extensive experiments on a very large real-life dataset from Epinions.com and compare the performance of different models in terms of the likelihood of the held-out test set. Based on our experimental results, we find out while for items with many reviews, the model learning aspects and ratings from opinion phrases with dependency assumption (D-PLDA) performs best, for items with few reviews (cold start items) the basic LDA model outperforms the more complex models. We also conclude that using dependency patterns consistently achieves the best performance for extracting opinion phrases.

# 3. PROBLEM STATEMENT AND CONTRIBUTION

In the opinion mining literature, an *aspect* refers to an attribute or component of an item that has been commented on in a review, e.g., 'sleep quality', 'internet connection', and 'room service' for a hotel. Expressing the *rating* of an aspect by a sentiment, an opinion phrase is a pair of $<$*head term*, *modifier*$>$ where head term refers to an aspect and modifier is the related sentiment, e.g. $<$ room service, great $>$, $<$ screen, inaccurate $>$ [19]. Most of the reviewing websites ask reviewers to express an *overall rating* (as stars) for the reviewed item in addition to the review text.

A *category* of items is defined as a set of items sharing common characteristics. Categorizations can be performed according to different criteria and are typically available in online reviewing websites. For example, hotels may be categorized based on price, location, price & location, etc. Products can be categorized at high level, e.g., electronics, toys, sport, etc. or at more specialized level, e.g., outdoor recreation, team sports, water sports, etc.

Providing aspects and the corresponding ratings does not only help users gain more insight into the quality of items, but also enables them to compare different items. The problem of aspect-based opinion mining addresses this need by performing two main tasks:

- Aspect identification: Identifying and extracting aspects from reviews.
- Rating prediction: Estimating the numerical rating of an aspect (usually in the range from 1 to 5).

As discussed in section 2, all of the current opinion mining models are at the item level. However, learning a model at the item level is not accurate for cold start items, i.e., items that have been reviewed by few reviewers. Since a very large portion of items in real-life reviewing websites are cold start, having a proper model for these items is essential. To address the problem of aspect-based opinion mining for cold start items, we propose a probabilistic model based on LDA, called FLDA. This model assumes that both items and reviewers can be modeled by a set of latent factors. Item's/reviewer's factors represent the item/reviewer distribution over aspects and for each aspect its distribution over ratings. Each review in the FLDA model is generated based on the learned factors of the corresponding item and reviewer. It first samples aspects in a review from the aspect distributions of the corresponding item and reviewer, and then generates the rating of each aspect conditioned on that aspect and the rating distributions of that item and reviewer. For cold start items, the aspect and rating distributions are mainly determined by the prior aspect distribution of the category and the rating distribution of the reviewer (or the prior rating distribution of all reviewers), respectively. For non cold start items, the aspect and rating distributions mainly depend on the observed reviews of that item. In the following section, we will elaborate the proposed FLDA model in detail.

# 4. PROPOSED MODEL

In this paper, we introduce a probabilistic model based on LDA, called Factorized LDA (FLDA), which models not only items but also reviewers. The FLDA model makes the following assumptions:

- A category has a set of aspects which are shared by all items in that category. For example, {zoom, battery life, shutter lag, etc.} is a set of aspects shared by all products in the category 'digital camera'. Note that, probabilities of occurrence of aspects can differ for different items in the category.
- Each item has a distribution over the aspects representing what aspects of its category are mainly commented on in reviews of that item. Each of these aspects is associated with a distribution of ratings.
- Each reviewer has a distribution over the aspects representing what aspects are more commented on by the reviewer. The reviewer is also associated, for each aspect, with a rating distribution.

Based on the above assumptions, to generate a review, aspects are first sampled conditioned on the aspect distributions of the corresponding item and reviewer. The rating of each aspect is then sampled conditioned on the aspect and the rating distributions of the item and the reviewer. Finally, opinion phrases are sampled based on the chosen aspects and ratings. Figure 1 shows the corresponding graphical model. Following the standard graphical model formalism, nodes represent random variables, edges indicate possible dependency, shaded nodes are observed random variables, and unshaded nodes are latent random variables. A box around

groups of random variables is a plate which denotes replication. The shaded box represents a review written by a reviewer about some item. $P$ and $U$ denote the number of items and reviewers, respectively. $K$ is the number of aspects and $N$ is the number of opinion phrases in a review. Note that, a random variable is a variable that can take on a set of possible different values, each with an associated probability.

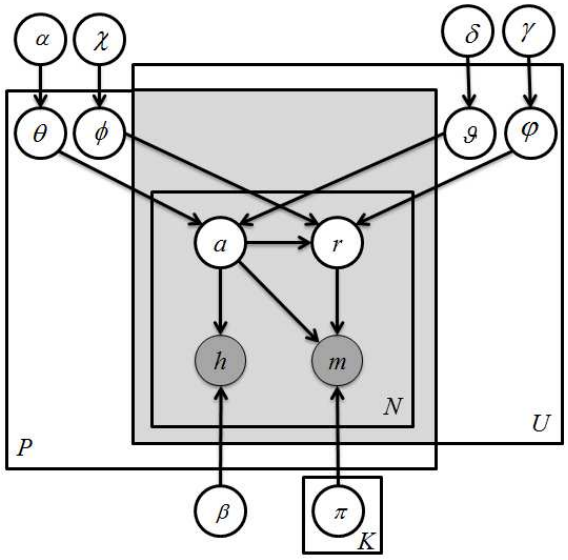



**Figure 1: The graphical model for FLDA**

As shown in Figure 1, $\alpha$ and $\delta$ are the prior aspect distributions and $\chi$ and $\gamma$ are the prior rating distributions for the given category. The basic idea of FLDA is that each item $p$ is represented as random mixtures over latent aspects, $\theta_p$, and latent rating, $\phi_p$, and each reviewer $u$ is represented as random mixtures over latent aspect, $\vartheta_u$, and latent ratings, $\varphi_u$. The FLDA model assumes the following generative process:

1. For each item $p$, $p \in \{1, 2, ..., P\}$
   (a) Sample $\theta_p \sim Dir(\alpha)$
   (b) Sample $\phi_p \sim Dir(\chi)$
2. For each reviewer $u$, $u \in \{1, 2, ..., U\}$
   (a) Sample $\vartheta_u \sim Dir(\delta)$
   (b) Sample $\varphi_u \sim Dir(\gamma)$
3. If there is a review by $u$ about $p$, then for each opinion phrase $< h_{pun}, m_{pun} >, n \in \{1, 2, ..., N\}$
   (a) Sample $a_{pun} \sim P(a_{pun}|\theta_p, \vartheta_u)$ and sample $r_{pun} \sim P(r_{pun}|a_{pun}, \phi_p, \varphi_u)$
   (b) Sample $h_{pun} \sim P(h_{pun}|a_{pun}, \beta)$ and sample $m_{pun} \sim P(m_{pun}|a_{pun}, r_{pun}, \boldsymbol{\pi})$

where $P(h_{pun}|a_{pun}, \beta)$ and $P(m_{pun}|a_{pun}, r_{pun}, \boldsymbol{\pi})$ are multinomial distributions. In the following the resulting joint distribution of the FLDA model is presented:

$$P(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\vartheta}, \boldsymbol{\varphi}|\alpha, \chi, \delta, \gamma, \beta, \boldsymbol{\pi}) = \prod_{p=1}^{P}[P(\theta_p|\alpha)P(\phi_p|\chi)] \prod_{u=1}^{U}[P(\vartheta_u|\delta)P(\varphi_u|\gamma)] \prod_{p=1}^{P}\prod_{u=1}^{U}\epsilon(p,u) \prod_{n=1}^{N}[P(a_{pun}|\theta_p, \vartheta_u)P(r_{pun}|a_{pun}, \phi_p, \varphi_u) P(h_{pun}|a_{pun}, \beta)P(m_{pun}|a_{pun}, r_{pun}, \boldsymbol{\pi})] \quad (1)$$

where $\epsilon(p, u) = 1$ if there is a review written by $u$ about item $p$, otherwise $\epsilon(p, u) = 0$. The goal is to compute the posterior distribution of the latent variables given a review:

$$P(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\vartheta}, \boldsymbol{\varphi}|\boldsymbol{h}, \boldsymbol{m}, \alpha, \chi, \delta, \gamma, \beta, \boldsymbol{\pi}) = \frac{P(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\vartheta}, \boldsymbol{\varphi}|\alpha, \chi, \delta, \gamma, \beta, \boldsymbol{\pi})}{P(\boldsymbol{h}, \boldsymbol{m}|\alpha, \chi, \delta, \gamma, \beta, \boldsymbol{\pi})} \quad (2)$$

Similar to the basic LDA, due to the coupling between $\boldsymbol{\theta}$ and $\boldsymbol{\vartheta}$ with $\beta$ and also between $\boldsymbol{\phi}$ and $\boldsymbol{\varphi}$ with $\boldsymbol{\pi}$, the conditional distribution of latent variables given observed data is intractable to compute. A wide variety of approximate inference algorithms have been proposed for LDA models. In this paper, we use variational inference [2] to compute an approximation for the posterior distribution.

## 5. INFERENCE AND PARAMETER LEARNING

In this section, we describe approximate inference and parameter learning for the FLDA model, adopting a variational method. As computing the posterior distribution of the latent variables for FLDA is intractable, we obtain a tractable lower bound by modifying the graphical model through considering a variational parameters for generating each latent variable. In particular, we simplify FLDA into the graphical model shown in Figure 2.

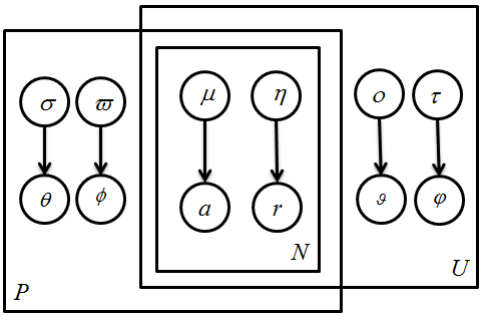



**Figure 2: Graphical model representation of variational distribution for FLDA**

This model specifies the following variational distribution on the latent variables:

**Algorithm 1** E-step of Variational Inference for FLDA

initialize $\mu^0_{puni} = 1/k$ for all $p$, $u$, $n$ and $i$
initialize $\eta^0_{punj} = 1/5$ for all $p$, $u$, $n$ and $j$
initialize $\sigma^0_{pi} = \alpha_i + (N \times U)/k$ for all $p$ and $i$
initialize $\varpi^0_{pij} = \chi_{ij} + (N \times U)/(k \times 5)$ for all $p$, $i$, $j$
initialize $o^0_{ui} = \delta_i + (N \times P)/k$ for all $u$ and $i$
initialize $\tau^0_{uij} = \gamma_{ij} + (N \times P)/(k \times 5)$ for all $u$, $i$, $j$
**repeat**
  **for** $p = 1$ to $P$ **do**
    **for** $u = 1$ to $U$ **do**
      **if** $\epsilon(p, u) == 1$ **then**
        **for** $n = 1$ to $N$ **do**
          **for** $i = 1$ to $k$ **do**
            $\mu^{t+1}_{puni} = \beta_{ix} \prod_j^5 \pi_{ijy}^{\eta^t_{punj}} \exp(\psi(\sigma^t_{pi})\psi(o^t_{ui}) + \sum_j^5 \eta^t_{punj}\psi(\tau^t_{uij})\psi(\varpi^t_{pij}))$
          **end for**
          normalize $\mu^{t+1}_{puni}$ to sum to 1
          **for** $j = 1$ to 5 **do**
            $\eta^{t+1}_{punj} = \prod_i^K \pi_{ijy}^{\mu^t_{puni}} \exp(\sum_i^K \mu^t_{puni} \psi(\tau^t_{uij})\psi(\varpi^t_{pij}))$
          **end for**
          normalize $\eta^{t+1}_{punj}$ to sum to 1
        **end for**
      **end if**
    **end for**
  **end for**
  **for** $p = 1$ to $P$ **do**
    $\sigma^{t+1}_p = \alpha + \sum_u^U \sum_n^N \mu^{t+1}_{pun}\psi(o^{t+1}_u)$
    $\varpi^{t+1}_p = \chi + \sum_u^U \sum_n^N \mu^{t+1}_{pun}\eta^{t+1}_{pun}\psi(\tau^{t+1}_u)$
  **end for**
  **for** $u = 1$ to $U$ **do**
    $o^{t+1}_u = \delta + \sum_p^P \sum_n^N \mu^{t+1}_{pun}\psi(\sigma^{t+1}_p)$
    $\tau^{t+1}_u = \gamma + \sum_p^P \sum_n^N \mu^{t+1}_{pun}\eta^{t+1}_{pun}\psi(\varpi^{t+1}_p)$
  **end for**
**until** convergence

$$Q(\boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\vartheta}, \boldsymbol{\varphi}, \boldsymbol{a}, \boldsymbol{r}|\boldsymbol{\sigma}, \boldsymbol{\varpi}, \boldsymbol{o}, \boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\eta}) = \prod_{p=1}^{P}[Q(\theta_p|\sigma_p)Q(\phi_p|\varpi_p)] \prod_{u=1}^{U}[Q(\vartheta_u|o_u)Q(\varphi_u|\tau_u)] \prod_{p=1}^{P}\prod_{u=1}^{U}\epsilon(p,u) \prod_{n=1}^{N}[Q(a_{pun}|\mu_{pun})Q(r_{pun}|\eta_{pun})] \quad (3)$$

where the Dirichlet parameters $\boldsymbol{\sigma}$, $\boldsymbol{\varpi}$, $\boldsymbol{o}$ and $\boldsymbol{\tau}$, and the multinomial parameters $\boldsymbol{\mu}$ and $\boldsymbol{\eta}$ are free variational parameters. The KL-divergence between the variational distribution and the true posterior should be minimum to have a good approximation. To this end, we set the derivative of the KL-divergence with respect to variational parameters equal to zero, to obtain the update equations. The update equations are invoked repeatedly until the change in KL-divergence is small.

Algorithm 1 presents the pseudo-code of the variational inference procedure where $\beta_{ix}$ is $P(h^x_{pun} = 1|a^i_{pun} = 1)$ for the appropriate $x$ and $\pi_{ijy}$ is $P(m^y_{pun} = 1|a^i_{pun} = 1, r^j_{pun} = 1)$ for the appropriate $y$. Recall that $h_{pun}$ and $m_{pun}$ are vectors with exactly one component equal to one. We can select the unique $x$ and $y$ such that $h^x_{pun} = 1$ and $m^y_{pun} = 1$ [2].

**Algorithm 2** M-Step of Variational Inference for FLDA

$\beta_{ix} = \sum_p^P \sum_u^U \sum_n^N \mu^*_{puni} h^x_{pun}$
$\pi_{ijy} = \sum_p^P \sum_u^U \sum_n^N \mu^*_{puni}\eta^*_{punj} m^y_{pun}$
$\alpha_{new} = \alpha_{old} - H(\alpha_{old})^{-1} g(\alpha_{old})$
$\chi_{new} = \chi_{old} - H(\chi_{old})^{-1} g(\chi_{old})$
$\delta_{new} = \delta_{old} - H(\delta_{old})^{-1} g(\delta_{old})$
$\gamma_{new} = \gamma_{old} - H(\gamma_{old})^{-1} g(\gamma_{old})$

By computing the approximate posterior, we can find a lower bound on the joint probability, $P(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\vartheta}, \boldsymbol{\varphi})$. Using this lower bound we can find approximate estimates for FLDA parameters via an alternative variational EM procedure [2]. The variational EM algorithm alternates between Expectation (E-step) and Maximization (M-step) steps until the bound on the expected log likelihood converges. The variational EM algorithm for FLDA is as follows[1]:

1. (E-step) For each review, find the optimizing values of the variational parameters $\sigma^*$, $\varpi^*$, $o^*$, $\tau^*$, $\mu^*$, and $\eta^*$ (using Algorithm 1).

2. (M-step) Maximize the resulting lower bound on the log likelihood with respect to the model parameters $\alpha$, $\chi$, $\delta$, $\gamma$, $\beta$, and $\boldsymbol{\pi}$ (using Algorithm 2).

The M-step update for the Dirichlet parameters $\alpha$, $\chi$, $\delta$ and $\gamma$ are implemented using the Newton-Raphson optimization technique that finds a stationary point of a function by iterating [2]. In Algorithm 2, $H(x)$ and $g(x)$ are the Hessian matrix and gradient respectively at the point $x$.

Note that, to deal with over fitting, we smooth all the parameters which depend on the observed data by assigning positive probability to all vocabulary terms whether or not they are observed in the training set.

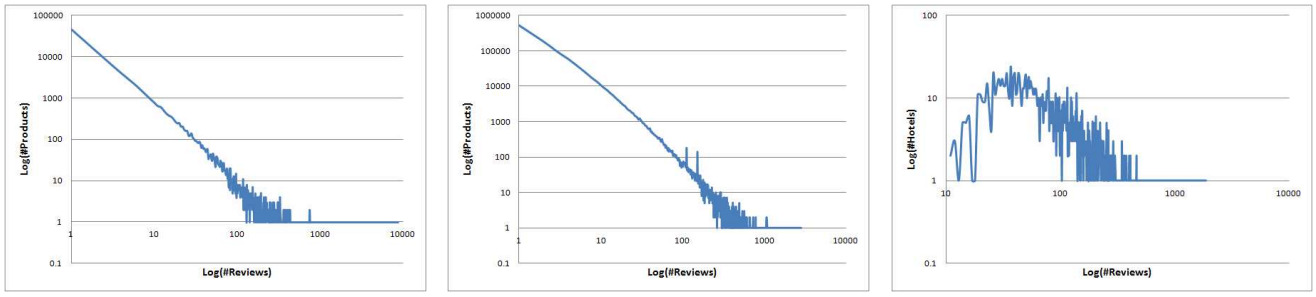
# 6. EXPERIMENTS

In this section, we first briefly describe the real-life datasets we used for our experiments and then present the results of the experimental evaluation of the FLDA model. We evaluate the performance of the model in terms of likelihood of the held-out test set and also based on two application-oriented measures for categorizing items and predicting reviews overall ratings.

## 6.1 Datasets

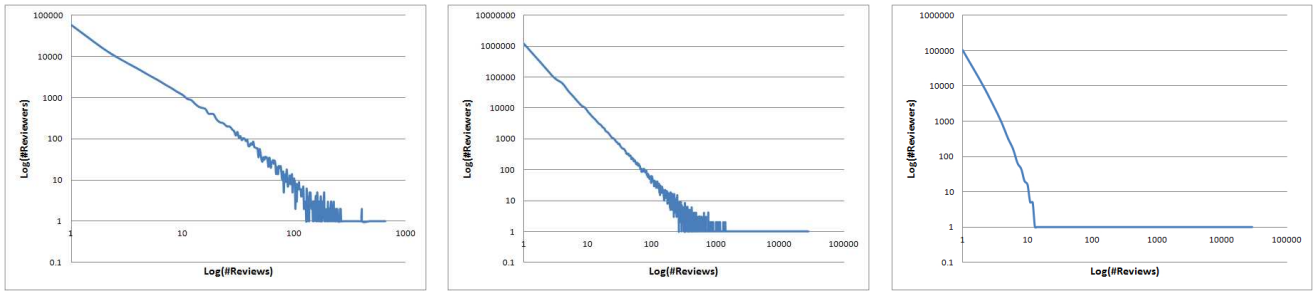
To evaluate the proposed model, we performed experiments on three real-life datasets from Epinions [21], Amazon [9], and TripAdvisor [27]. In each dataset, we select items with at least one review. For preprocessing, we adopt the dependency pattern technique to identify opinion phrases in the form of a pair of head term and modifier. This technique results in the best performance in

[1]The detailed derivation of the variational EM algorithm is available at `http://http://www.sfu.ca/~sam39/FLDA/`




(a) Epinions (b) Amazon (c) TripAdvisor

**Figure 3: Log-log plot of #reviews vs. #items**




(a) Epinions (b) Amazon (c) TripAdvisor

**Figure 4: Log-log plot of #reviews vs. #reviewers**

**Table 1: General statistics of different datasets**

| Dataset | Epinions | Amazon | TripAdvisor |
|---|---|---|---|
| #Categories | 379 | 38 | 5 |
| #Reviews | 541,219 | 5,016,492 | 181,395 |
| #Reviewers | 109,857 | 1,761,879 | 117,976 |
| #Items | 87,633 | 1,108,018 | 1,496 |

**Table 2: Sample categories of each dataset**

| Dataset | Sample Categories |
|---|---|
| Epinions | Accessories, Blazers, Dresses, Outerwear, Pants, Shirts, Skirts, ... |
| Amazon | Apparel, Electronics, Computers, Baby,... |
| TripAdvisor | 1-star, 2-star, 3-star, 4-star, 5-star |

compare to other preprocessing techniques according to [21]. In Table 1, general statistics of these datasets are shown.

Regarding item categories, we used the available categorization in each dataset which were mostly at a high level (5 hotel categories based on their number of stars for TripAdvisor, 38 general categories for Amazon, and 379 product categories for Epinions). Table 2 shows some sample categories for each dataset.

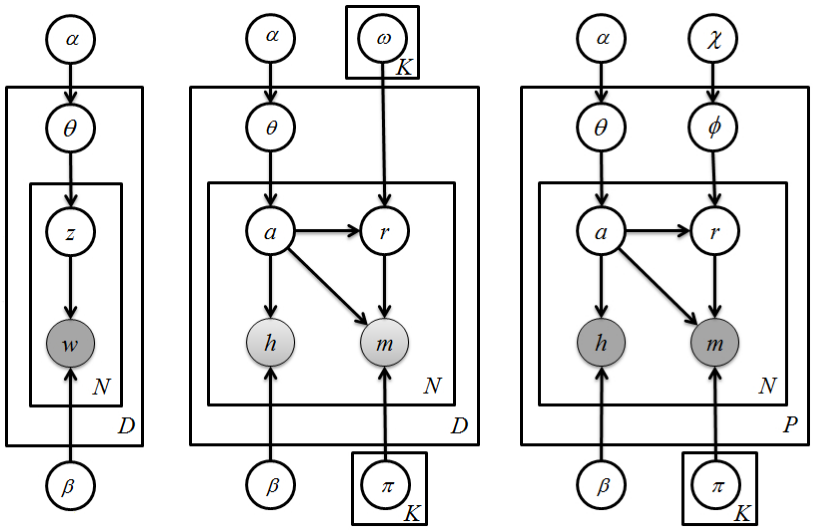
All of the current works report only the average number of reviews per item, somehow masking the large percentage of cold start items in real-life datasets. In fact, cold start items are normally ignored in learning latent variable models. In order to show the variance in the numbers of reviews, Figures 3 and 4 show the distributions of #reviews per item and #reviews per reviewer in different datasets, respectively.

Not surprisingly, in the Epinions and Amazon datasets both distributions follow a power law. We can see that a large number of items has only a few reviews, and a few items have a large number of reviews (Figures 3(a) and 3(b)). A similar property can be seen in the log-log plot of the number of reviews vs. the number of reviewers (Figures 4(a) and 4(b)). In the TripAdvisor dataset, the distribution of the number of reviews per reviewer (Figures 4(c)) also follows a power law. However, the relationship between the number of reviews and the number of hotels (Figure 3(c)) is below an ideal straight line for the first few points, since there are surprisingly few hotels with fewer than 50 reviews.

These power law distributions point out substantial diversity among items in real-life review datasets. To analyze the performance of the comparison partners separately on different types of items, we categorize items of each dataset into 5 groups based on the number of reviews. Table 3 shows the percentage of items in each dataset with the specified number of reviews.

**Table 3: Percentage of items in each item group**

| Item Groups | Epinions | Amazon | TripAdvisor |
|---|---|---|---|
| $1 < \#Rev \leq 10$ | 90% | 91% | 0% |
| $10 < \#Rev \leq 50$ | 8% | 7% | 31% |
| $50 < \#Rev \leq 100$ | 1% | 1% | 30% |
| $100 < \#Rev \leq 200$ | $< 1\%$ | $< 1\%$ | 24% |
| $200 < \#Rev$ | $< 1\%$ | $< 1\%$ | 14% |




(a) LDA (b) D-PLDA (c) I-FLDA

**Figure 5: Comparison partners: LDA and D-PLDA are state-of-the-art models, I-FLDA is a simplified version of FLDA**

In the Epinions and Amazon datasets, more than 90% of products have less than 10 reviews which are considered cold start items. The TripAdvisor dataset has larger numbers of reviews per item. However, as Table 3 shows 31% of hotels have been reviewed by less than 50 reviewers which can be considered cold start in this dataset. These statistics clearly indicate that there is a need for opinion mining models with the focus on cold start items.

Table 4 also presents the average number of reviews per item for the defined item groups. It suggests that the average numbers of reviews for cold start items are indeed very small (2 for Epinions and Amazon, 25 for TripAdvisor) which makes it hard to learn an accurate model for these items.

**Table 4: Average #reviews per item in each item group**

| Item Groups | Epinions | Amazon | TripAdvisor |
|---|---|---|---|
| $1 < \#Rev \leq 10$ | 2 | 2 | 0 |
| $10 < \#Rev \leq 50$ | 16 | 18 | 25 |
| $50 < \#Rev \leq 100$ | 53 | 62 | 54 |
| $100 < \#Rev \leq 200$ | 114 | 122 | 107 |
| $200 < \#Rev$ | 324 | 338 | 297 |

## 6.2 Comparison Partners

We compare FLDA with the basic LDA model that generates all words of reviews [2] (Figure 5(a)) and the D-PLDA model presented in [21] (Figure 5(b)). We selected these two models as comparison partners since experimental evaluation in [21] showed that the basic LDA performs best for cold start items and D-PLDA outperforms other models for non cold start items. Note that, FLDA adopts the same model for generating opinion phrases as D-PLDA, i.e., they have the same inner plate in the graphical model. Both LDA and D-PLDA are trained at the item level and $D$ is the number of reviews for the given item. To tease apart the impact of the two major changes between D-PLDA and FLDA, we also compare a simplified version of FLDA, called I-FLDA, that does not model reviewers and their parameters but is trained at the category level(Figure 5(c)).

## 6.3 Quantitative Evaluation

In this section, we evaluate the generalization performance of all comparison partners based on the likelihood of a held-out test set, which is standard in the absence of ground truth. For comparison, we trained all the latent variable models using EM with exactly the same stopping criteria and for various numbers of aspects, $k = \{5, 10, 15, 20, 25\}$. Since the relative results are similar for different values of $k$, we choose $k = 15$ for our discussion.

In the performance comparison, the goal is achieving high likelihood on a held-out test set. We hold out 10% of the reviews for testing purposes and use the remaining 90% to train models. As is standard for LDA models [2, 30, 20], we computed the perplexity of the held-out test set. A strong correlation of the perplexity and the accuracy (which can be computed only if ground truth is available) of aspect-based opinion mining models is shown in [21]. The perplexity is monotonically decreasing in the likelihood of the test data, and a lower perplexity score indicates better performance. For a test set of $N$ reviews, the perplexity is defined as [2]:

$$perplexity(D_{test}) = exp\{-\frac{\sum_{d=1}^{D} \log P(\boldsymbol{h}_d, \boldsymbol{m}_d)}{\sum_{d=1}^{D} N_d}\} \quad (4)$$

Table 5 and Figure 6 present the perplexity results of FLDA and the comparison partners for different groups of items in different datasets. The first observation, that has already been discussed in [21], is that the D-PLDA model outperforms LDA in all datasets for non cold start items. However, for cold start items it has higher perplexity than LDA, indicating poor performance of the model in the absence of enough training data. We can also observe that I-FLDA, which is trained at the category level but does not model reviewers, achieves lower perplexity than D-PLDA, especially for cold start items. This better performance can be explained by the fact that I-FLDA is trained at the category level and learns the latent factors using the reviews of all the items of a category, in particular the non cold start items, and uses them as prior for cold start items.

Finally, we note that in all datasets and for all item groups, FLDA

**Table 5: Perplexity comparison of different item groups in different datasets**

| Item Groups | LDA | D-PLDA | I-FLDA | FLDA |
|---|---|---|---|---|
| $1 < \#Rev \leq 10$ | 4413.65 | 5413.65 | 4187.98 | 3287.98 |
| $10 < \#Rev \leq 50$ | 2338.67 | 1975.34 | 1903.45 | 1687.67 |
| $50 < \#Rev \leq 100$ | 1671.23 | 592.39 | 588.61 | 468.12 |
| $100 < \#Rev \leq 200$ | 1394.72 | 164.18 | 153.02 | 133.90 |
| $200 < \#Rev$ | 1385.99 | 142.37 | 142.16 | 140.35 |

(a) Epinions

| Item Groups | LDA | D-PLDA | I-FLDA | FLDA |
|---|---|---|---|---|
| $1 < \#Rev \leq 10$ | 5019.79 | 5653.79 | 4302.45 | 3494.01 |
| $10 < \#Rev \leq 50$ | 2434.71 | 2159.02 | 1931.34 | 1833.66 |
| $50 < \#Rev \leq 100$ | 1183.14 | 769.77 | 756.09 | 744.18 |
| $100 < \#Rev \leq 200$ | 993.78 | 339.69 | 331.45 | 318.49 |
| $200 < \#Rev$ | 869.25 | 177.08 | 173.15 | 172.45 |

(b) Amazon

| Item Groups | LDA | D-PLDA | I-FLDA | FLDA |
|---|---|---|---|---|
| $1 < \#Rev \leq 10$ | - | - | - | |
| $10 < \#Rev \leq 50$ | 3446.61 | 3518.95 | 2898.56 | 2725.76 |
| $50 < \#Rev \leq 100$ | 3336.61 | 2673.19 | 2394.09 | 2301.31 |
| $100 < \#Rev \leq 200$ | 2943.46 | 1003.09 | 892.59 | 843.91 |
| $200 < \#Rev$ | 1438.59 | 363.20 | 362.74 | 359.03 |

(c) TripAdvisor

consistently outperforms LDA, D-PLDA and I-FLDA. These findings show that FLDA's assumptions regarding using the category level information for aspect extraction and the user modeling for rating prediction are appropriate. The perplexity gain of FLDA is most notable for cold start items underlining the effectiveness of FLDA in modeling such items. For items with large numbers of reviews, FLDA can slightly improve the performance of I-FLDA by also modeling reviewers. Comparing the results of FLDA, I-FLDA and D-PLDA shows that when there is enough training data (reviews), learning a model at the item level is promising.

# 7. APPLICATIONS

In the following sections we perform two application-oriented evaluations to demonstrate the gains of FLDA in practice.

## 7.1 Item Categorization

One of the applications of category-level models is the ability of categorizing new items based on their reviews, e.g., identifying the class of a hotel (1 to 5 star), or type of a book (e.g., children's books, textbooks, audio books, magazines, etc.) based on their reviews. This feature is especially beneficial when working with uncategorized reviews, e.g., forums, Blogs, discussion groups, etc.

In [2], Blei et al. proposed to use the basic LDA model for document classification. In particular, LDA is used as a dimensionality reduction method, as it reduces any document to a vector of real-valued features, i.e., the posterior Dirichlet parameter associated with each document. The parameters of an LDA model are learned using all the documents, without reference to their true class label. The topic distribution provides a low-dimensional representation (feature vector) of a document, and a support vector machine (SVM) is trained on these feature vectors to distinguish the classes.

In our scenario, we can adopt the same approach for item categorization. The FLDA model can be used to produce feature vectors for item categorization as follows. We first estimate the parameters of the FLDA model using all the reviews of all items of all categories. The learned topic distribution $\sigma$ of an item is used as the feature vector of that item, and an SVM classifier is trained on these feature vectors to classify items into categories (FLDA-SVM). Note that, the topic distribution of an item in this model cannot be interpreted as the aspect distribution of the item.

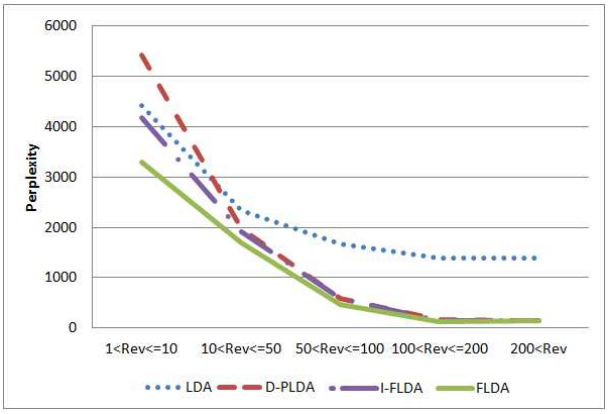
Since the LDA and D-PLDA models learn topic distributions of reviews, not items, they cannot be directly used as comparison partners for item categorization. However, by applying these models at the category level, we can obtain the topic distribution of items as item feature vectors. These models use all reviews of all items of all categories to learn the feature vectors of items (similar to FLDA-SVM). As a baseline, we also train a classifier on simple bag-of-words features (BOW-SVM). Table 6 shows the accuracy of SVM classifiers for cold start and non cold start items trained on different feature spaces.

**Table 6: Average accuracy of SVM classifier trained on different feature sets for item categorization**

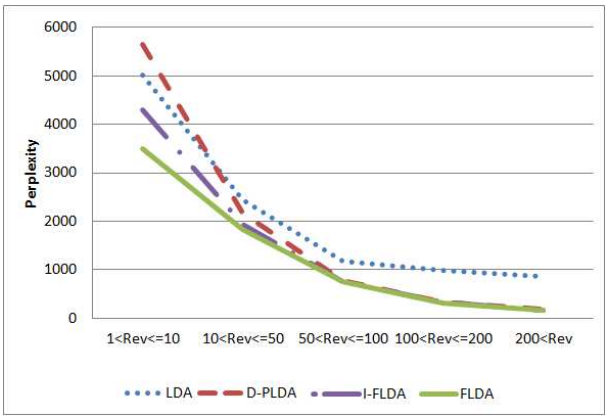
| Dataset | Epinions | | Amazon | | TripAdvisor | |
|---|---|---|---|---|---|---|
| Item Type | cold | non | cold | non | cold | non |
| BOW-SVM | 64% | 88% | 62% | 88% | 68% | 91% |
| LDA-SVM | 71% | 90% | 67% | 91% | 73% | 93% |
| D-PLDA-SVM | 79% | 96% | 75% | 94% | 85% | 97% |
| FLDA-SVM | 83% | 96% | 79% | 95% | 86% | 97% |

The first observation is that for all feature sets the accuracy of item categorization is higher for non cold start items than for cold start items. This was predictable as there is more training data for non cold start items. Comparing BOW-SVM with LDA-SVM and D-PLDA-SVM, we can see an increase in classification accuracy by using the LDA-based features. This suggests that the topic-based representation provided by LDA can be useful for feature selection in item categorization. We also observe that the classifi-

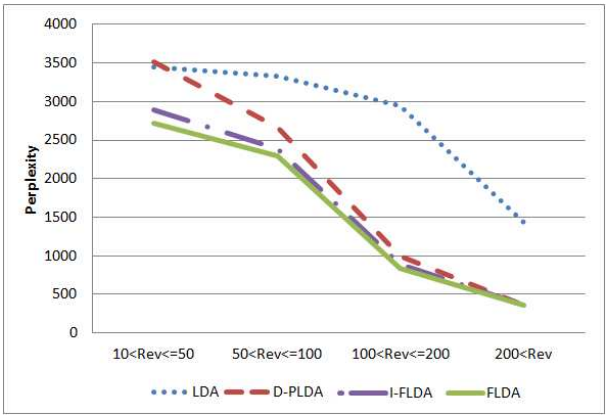
cation accuracy is substantially improved by using FLDA features. This suggests that the learned item factors of FLDA can provide a more accurate low-dimensional feature set for item categorization. Note that the LDA-based models ($k = 15$) reduce the feature space of the Epinions, Amazon, and TripAdvisor datasets by 97%, 99%, and 92%, respectively compared to all word features.




(a) Epinions




(b) Amazon




(c) TripAdvisor

**Figure 6: Perplexity results of all comparison partners for different datasets**

## 7.2 Overall Rating Prediction for Reviews

In most of the reviewing websites, reviewers are asked to assign an overall rating (as a number in some given range) to express their overall level of satisfaction with the reviewed item. However, in other repositories of reviews, such as forums and Blogs, such overall ratings are not normally provided. One of the applications of FLDA is the ability of predicting the overall rating of a review. As each review is written by a reviewer about an item, the overall rating of a review depends on both item and reviewer factors. The aspect and rating distributions of items and reviewers learned by the FLDA model can be used for computing the overall rating of the review as follows.

In recommender systems, Matrix Factorization (MF) is employed to factorize the $user \times item$ rating matrix to predict the rating of a user for an item [24, 11, 12]. Inspired by this model, we can compute the overall rating of an item by a reviewer using the learned item and reviewer factors. In the FLDA model, the latent aspect distribution of review $d_{pu}$ is determined by the aspect distributions of the corresponding item, $p$, and reviewer, $u$, and is denoted by $P(a|\theta_p, \vartheta_u)$. In the same way the latent rating distribution of review $d_{pu}$ is denoted by $P(r|a, \phi_p, \varphi_u)$. According to the probabilistic MF model, the distribution of the overall ratings $o_{up}$ for user $u$ and item $p$, can be computed as follows:

$$P(o_{up} = r) = \sum_{a} P(a|\theta_p, \vartheta_u) P(r|a, \phi_p, \varphi_u) \qquad (5)$$

Since in the review datasets we used, ratings are chosen from the set {1, 2, 3, 4, 5}, we define 5 classes of overall ratings. For each item we train an SVM classifier on the distribution of the overall ratings acquired by Equation (5) to classify the overall rating of a given review (FLDA-SVM). As comparison partners, we train two classifiers on the review feature vectors generated by LDA (LDA-SVM) and D-PLDA (D-PLDA-SVM). The review feature vector of LDA is the topic distribution of the review, and the review feature vector of D-PLDA is the distribution of the overall ratings obtained using the probabilistic MF model (similar to Equation (5)). We also train a classifier on simple bag-of-words features (BOW-SVM) as a baseline. Table 7 shows the accuracy of SVM classifiers for cold start and non cold items trained on different feature spaces.

**Table 7: Average accuracy of SVM classifier trained on different feature sets for overall rating prediction**

| Dataset | Epinions | | Amazon | | TripAdvisor | |
|---|---|---|---|---|---|---|
| Item Type | cold | non | cold | non | cold | non |
| BOW-SVM | 49% | 83% | 44% | 79% | 47% | 82% |
| LDA-SVM | 56% | 85% | 53% | 80% | 59% | 85% |
| D-PLDA-SVM | 57% | 86% | 54% | 80% | 63% | 87% |
| FLDA-SVM | 72% | 89% | 70% | 83% | 74% | 91% |

Again we can see that for all feature sets the accuracy of overall rating prediction for non cold start items is much higher than that of cold start items, and also the accuracy of all LDA-based models is higher than for bag-of-words features. The accuracy of D-PLDA-SVM is slightly higher than that of LDA-SVM as it uses the rating distribution of the item for generating the feature vectors of reviews. Finally, as shown in Table 7, the accuracy of FLDA-SVM for the task of overall rating prediction is much higher than that of the comparison partners. This suggests that for a given review the learned item and user factors can be used as a low-dimensional feature set for predicting its overall rating.

# 8. CONCLUSION

Aspect-based opinion mining is the problem of automatically extracting aspects and estimating their ratings from reviews. All of the current models are trained at the item level (a model is trained form all reviews of an item) to perform these tasks. In this paper, we argued that while learning a model at the item level is fine for frequently reviewed items, it is ineffective for items with few reviews (cold start items). Note that, more than 90% of products in Epinions and Amazon datasets and 30% of hotels in the TripAdvisor dataset are cold start.

Addressing this need, we introduced the problem of aspect-based opinion mining for cold start items and proposed a probabilistic model based on LDA, called FLDA. Our model assumes that aspects in a review are sampled from the aspect distributions of the corresponding item and reviewer and the rating of each aspect is sampled conditioned on that aspect and the rating distributions of the item and reviewer. For cold start items the aspect distribution is mainly determined by the prior aspect distribution of the category, and the rating distribution of each aspect is mainly determined by the rating distribution of the reviewer (or by the prior rating distribution of all reviewers if the reviewer is cold start). The aspect and rating distributions for non cold start items are mainly determined by the observed reviews of that item.

We conducted extensive experiments on three real-life datasets and compared FLDA against the baseline LDA, the state-of-the-art D-PLDA, and the simplified I-FLDA models. FLDA clearly outperforms all of the comparison partners in terms of likelihood of the test set. For cold start items, the perplexity gain of FLDA is very large. We argued that the major reason for this gain is using the category level information and also modeling reviewers. We further teased apart the impact of modeling reviewers by comparing FLDA with the simplified I-FLDA model showing that modeling reviewers significantly impacts the model performance for cold start items. We also demonstrated the accuracy of FLDA in two applications: categorizing items and predicting the overall rating of reviews based on the learned feature vectors.

This paper suggests several directions for future research. FLDA assumes a given definition of item categories, but there may be alternative options to define them. For example, is it better to categorize hotels based on stars, or location, or price? An item taxonomy is a hierarchical structure of categories and subcategories. For example, the hierarchy for the category 'MP3 Players' could be "Electronics > Audio > Audio Players & Recorders > MP3 Players". In addition, in a scenario with a given item taxonomy, it would be interesting to explore methods to automatically learn the granularity (taxonomy) level that leads to the best model performance.

# 9. REFERENCES


[1] S. Baccianella, A. Esuli, and F. Sebastiani. Multi-facet rating of product reviews. In *ECIR '09*.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 2003.

[3] S. Brody and N. Elhadad. An unsupervised aspect-sentiment model for online reviews. In *HLT '10*.

[4] Y. Choi and C. Cardie. Hierarchical sequential learning for extracting opinions and their attributes. In *ACL '10*.

[5] H. Guo, H. Zhu, Z. Guo, X. Zhang, and Z. Su. Product feature categorization with multilevel latent semantic association. In *CIKM '09*.

[6] Y. He, C. Lin, and H. Alani. Automatically extracting polarity-bearing topics for cross-domain sentiment classification. In *HLT '11*.

[7] M. Hu and B. Liu. Mining and summarizing customer reviews. In *KDD '04*.

[8] W. Jin, H. H. Ho, and R. K. Srihari. Opinionminer: a novel machine learning system for web opinion mining and extraction. In *KDD '09*.

[9] N. Jindal and B. Liu. Opinion spam and analysis. In *WSDM '08*.

[10] Y. Jo and A. H. Oh. Aspect and sentiment unification model for online review analysis. In *WSDM '11*.

[11] Y. Koren. Collaborative filtering with temporal dynamics. In *KDD '09*.

[12] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer Journal*, 42, 2009.

[13] H. Lakkaraju, C. Bhattacharyya, I. Bhattacharya, and S. Merugu. Exploiting coherence for the simultaneous discovery of latent facets and associated sentiments. In *SDM '11*.

[14] F. Li, C. Han, M. Huang, X. Zhu, Y.-J. Xia, S. Zhang, and H. Yu. Structure-aware review mining and summarization. In *COLING '10*.

[15] F. Li, M. Huang, and X. Zhu. Sentiment analysis with global topics and local dependency. In *AAAI '10*.

[16] C. Lin and Y. He. Joint sentiment/topic model for sentiment analysis. In *CIKM '09*.

[17] B. Liu. *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers, 2012.

[18] B. Liu, M. Hu, and J. Cheng. Opinion observer: analyzing and comparing opinions on the web. In *WWW '05*.

[19] Y. Lu, C. Zhai, and N. Sundaresan. Rated aspect summarization of short comments. In *WWW '09*.

[20] S. Moghaddam and M. Ester. ILDA: interdependent LDA model for learning latent aspects and their ratings from online product reviews. In *SIGIR '11*.

[21] S. Moghaddam and M. Ester. On the design of LDA models for aspect-based opinion mining. In *CIKM '12*.

[22] S. Moghaddam and M. Ester. Opinion Digger: an unsupervised opinion miner from unstructured product reviews. In *CIKM '10*.

[23] A. Mukherjee and B. Liu. Aspect extraction through semi-supervised modeling. In *ACL '12*.

[24] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In *NIPS '07*.

[25] I. Titov and R. McDonald. A joint model of text and aspect ratings for sentiment summarization. In *ACL-HLT '08*.

[26] I. Titov and R. McDonald. Modeling online reviews with multi-grain topic models. In *WWW '08*.

[27] H. Wang, Y. Lu, and C. Zhai. Latent aspect rating analysis on review text data: a rating regression approach. In *KDD '10*.

[28] H. Wang, Y. Lu, and C. Zhai. Latent aspect rating analysis without aspect keyword supervision. In *KDD '11*.

[29] T.-L. Wong, L. Bing, and W. Lam. Normalizing web product attributes and discovering domain ontology with minimal effort. In *WSDM '11*.

[30] T.-J. Zhan and C.-H. Li. Semantic dependent word pairs generative model for fine-grained product feature mining. In *PAKDD '11*.

[31] W. X. Zhao, J. Jiang, H. Yan, and X. Li. Jointly modeling aspects and opinions with a maxent-lda hybrid. In *EMNLP '10*.