Statistical Models of Music-listening Sessions in Social Media

Elena Zheleva
Dept. of Computer Science
Univ. of Maryland, College Park, USA
elena@cs.umd.edu

John Guiver
Microsoft Research Ltd.
Cambridge, UK
joguiver@microsoft.com

Eduarda Mendes Rodrigues
Dept. of Informatics Engineering
University of Porto, Portugal
eduardamr@acm.org

Nataša Milić-Frayling
Microsoft Research Ltd.
Cambridge, UK
natasamf@microsoft.com

ABSTRACT

User experience in social media involves rich interactions with the media content and other participants in the community. In order to support such communities, it is important to understand the factors that drive the users' engagement. In this paper we show how to define statistical models of different complexity to describe patterns of song listening in an online music community. First, we adapt the LDA model to capture *music taste* from listening activities across users and identify both the groups of songs associated with the specific taste and the groups of listeners who share the same taste. Second, we define a graphical model that takes into account listening sessions and captures the *listening moods* of users in the community. Our *session model* leads to groups of songs and groups of listeners with similar behavior across listening sessions and enables faster inference when compared to the LDA model. Our experiments with the data from an online media site demonstrate that the session model is better in terms of the perplexity compared to two other models: the LDA-based *taste* model that does not incorporate cross-session information and a baseline model that does not use latent groupings of songs.

Categories and Subject Descriptors

H.2.8 [**Information Systems**]: Data Mining

General Terms

Algorithms, Experimentation

Keywords

social media, sessions, music, taste, mood, graphical models, recommendations, collaborative filtering

1. INTRODUCTION

With broad proliferation of online social networks around media content, there is an increased interest in analyzing interactions among users and characterizing their behavior in terms of the individuals' and community preference for specific types of content. Among the popular and ever-growing social media sites centered around music are Last.fm, Zune Social, Flotones, JamNow, Haystack, Midomi, Sellabound, MySpace, Mercora radio, iLike, MusoCity, Sonific, and iJigg. Many of them include features that encourage social interactions by providing personalized recommendations to influence media selection of individuals. Furthermore, they offer community-based recommendations and interfaces for browsing and searching for available content.

For such complex systems, it is important to develop techniques that can be used to describe and study processes that drive the observed user engagement. Such methods need to be able to handle large-scale data logs from social media services and, therefore, produce effective representations of media consumption in order to enable efficient processing. In this paper we use the example of music listening to demonstrate how that objective can be achieved. We illustrate an effective representation of usage data that can be applied to enhance individual user's experience, e.g., by recommending songs for the user's playlist that would be relevant for the current music-listening session. Considering the large number of users and songs, such contextual recommendations require highly compact data representations.

Selecting a suitable song descriptor is an important initial step. We observe that many media services provide a static taxonomy of media types or *genre*. Such taxonomies serve as the means for individuals to express their interests and find adequate media. They provide media categories that are commonly adopted by the user community and, thus, could be used to characterize user's song-listening behavior, e.g., as a probability distribution over clusters of same-genre songs. The genre also captures an essential aspect of the song-listening process: while a person may not necessarily wish to repeat the same song, the person is likely to choose the next song to play from the same or a related genre.

On the other hand, even basic genre taxonomies may have a large number of categories and lead to sparse and ineffective representations of listening patterns. Thus, we aim to create a compact representation of media listening that retain the essential statistical properties and relations among data. For that purpose we choose to derive generative probabilistic models based on the logs of song-listening and control the number of the underlying media clusters.

WWW 2010, April 26–30, 2010, Raleigh, North Carolina, USA.
ACM 978-1-60558-799-8/10/04.

The contributions of our work are:

- A systematic approach to characterizing social media processes that drive music listening patterns
- A novel graphical model which provides a compact representation of the media based on listening sessions
- A model that has better predictive properties and enable faster inference than other known models.

More precisely, we define graphical models with latent variables that are intuitive and appropriate for modeling song listening. The first model captures the collective *music taste* as a set of tastes or media preferences that a particular community develops. We use them to characterize song listening by an individual user as a finite mixture of the underlying tastes. The second model captures the *listening moods* across listening sessions of the users in the community. In such a model, an instance of song listening by a user is described as a finite mixture of the underlying *set of listening moods*. In both cases we can vary the model parameters and explore the effect that different number of derived tastes and moods have on the model quality. In particular, we demonstrate the computational efficacy and compare the perplexity of the two models.

Our work is the first to utilize a hierarchical graphical model to incorporate listening moods based on session information. By applying the models to half a million song-listening instances from the Zune Social[1] music community, we demonstrate a clear advantage of using a more refined model to achieve both better perplexity for the co-occurrence of genres in sessions and higher computational efficiency. Although we introduced and evaluated it in the context of song listening, the same model can be applied to a broad range of scenarios, from browsing sessions on YouTube or Flickr to characterizing the sentiment and topics of blog-posting within given periods of time.

In the following Section 2 we give an overview of the related work and provide background on graphical models. We then discuss the social media context in Section 3. In Section 4 we describe the data and define the hierarchical graphical models. In Section 5 we present experimental results and then reflect on broader implications of our work in Section 6. We conclude with a summary of our contributions and directions for further work.

2. BACKGROUND

Creating a successful, self-sustaining social media service is a challenge because of the complexity of social interactions that ensue once the service is in place. A broad range of issues related to this problem have been addressed in the literature on social networks, e-commerce, recommendations, rating, collaborative filtering, and similar. Here we provide context for our work by discussing research related to our approach and provide background information with prerequisites for the models we explore.

2.1 Related work

2.1.1 User modeling

An individual's taste and mood are two factors that are likely to influence consumption of media and social interactions. Thus, characterizing them in an effective manner would be invaluable for personalizing retrieval, classification, and recommendation of media content. However, the variability and subjective nature of these notions makes it difficult to describe them in a systematic way. Nonetheless, there have been efforts to characterize mood as a property of songs and the effects they may have on listeners.

Feng et al. [4] attempt to detect mood of songs from their acoustical features such as tempo and articulation. Liu et al. [9] use intensity, timbre and rhythm instead. Hu & Downie [8] study the relationship between mood and music genre, and mood and artists. In all these cases, the researchers proposed taxonomies of mood types. Feng et al. [4] define four mood labels: *happiness*, *sadness*, *anger*, and *fear* for training a music classifier. Liu et al. [9] use a mood model that characterizes emotions along two dimensions, *energy* and *stress*. They define four mood quadrants: *contentment*, *depression*, *exuberance*, and *anxious/frantic* and use them as labels for mood detection in music using a framework based on Gaussian mixture models. Hu and Downie [8] derive a set of five mood clusters from the *All Music Guide*[2] mood repository to examine the correlation between music genre and mood and artist and mood.

The results of this approach are of limited utility because comprehensive, generally accepted, and universally applicable taxonomies for taste and mood do not exist and are difficult to conceive. That would require an in-depth understanding of human emotions, mapping out a wealth of human relations to the external world, and providing a reference scale to measure the intensity of emotions that could be applied in an objective manner.

In our approach, we derive a *latent mood* rather than *a priori* specifying the mood as a property of the music. We use the terms *music taste* and *listening mood* to describe the users' affinity to listen to specific groups of songs as observed from the listening patterns of the whole community. For listening moods we derive the song clusters from the media selection within and across listening sessions, where a session is determined by a threshold of idle time, i.e., a pause between two consecutive songs.

2.1.2 Song recommendations

Ragno et al. [20] address the problem of recommending songs to the user based on a *seed song* that the user has listened to, with the aim to generate a complete playlist that fits the user preferences. It is assumed that the user wishes to listen to songs that are, in some sense, similar to the seed song. In [20] the authors use multiple radio broadcast streams to determine song proximity and define a graph representing the song-similarity. Automatic playlists are generated through random walks of this graph starting on a given seed song. There are many other approaches for automatic playlist generation (e.g., [18, 19]). In [18], Pampalk et al. use audio similarity and feedback from users, in the form of accepting or skipping a song recommendation, to define a set of heuristics for playlist generation. In [19], Platt et al. learn a Gaussian Process kernel to predict user playlists using music metadata such as genre or style as input.

2.1.3 Statistical data modeling

Modeling collections of discrete data has been of growing interest for researchers who study large text corpora. Latent semantic analysis techniques provide a powerful means

[1] `http://social.zune.net/`.

[2] At `http://www.allmusic.com`

of identifying underlying topics as clusters of terms derived from document-word co-occurrences [3, 7].

Recently, the Latent Dirichlet Model (LDA) [2] has been introduced to capture statistical properties of text documents in a collection and provide a compact document representation in terms of underlying topics. More precisely, the method assumes that each document is a mixture of latent topics and uses a three-level hierarchical graphical model to characterize the statistical relations among terms and documents, resulting in topics that are represented as clusters of words. We describe the model in more detail in Section 4.1.

The LDA model has gained popularity due to its simple but powerful structure, and it has been applied to other domains besides topic modeling. Zhang et al. [23] propose an LDA-based model for identifying latent structures in large networks, using topological features as the only input. They apply the model to identify communities in large social networks. A similar model for analyzing graph data is described by Henderson & Eliassi-Rad [5].

There are other generative models that combine topic modeling and social network modeling in a single framework [13, 14]. The Author-Role-Topic (ART) model, proposed by McCallum et al. [13], discovers discussion topics in threaded conversations, conditioned on sender-recipient interactions. The Group-Topic (GT) model [14] discovers latent groups in a network and clusters of associated topics based on text. The recent work on recommender systems by Stern et al. [21] proposes a probabilistic rating model which combines collaborative filtering and item metadata for predicting items that may be of interest to a given user. Marlin et al. [11, 12] also use graphical models for the task of rating prediction. Hoffman et al. [6] propose a probabilistic model which uses audio features to predict song tags.

In our work we use hierarchical graphical models to represent the song-listening activities in terms of *latent tastes* and *latent listening moods* of the community that are derived from the logs of media usage. For the latent tastes characterization we adapted the LDA model to the song-listening activities. Every instance of song listening is modeled as a finite mixture over the underlying set of tastes which, in turn, correspond to the clusters of songs derived from the listening patterns. For listening moods, we increased the complexity of the model by incorporating session information. As a result, we arrive at a novel hierarchical graphical model that exploits additional structure in the data and identifies latent moods as clusters of songs that emerge from the song-listening sessions across the community.

2.2 Preliminary concepts

2.2.1 Graphical models and factor graphs

Factor graphs are a useful way of representing probabilistic graphical models. They consist of two types of nodes representing *variables* and *factors*, respectively. Figure 3 and Figure 4 show examples of factor graphs with standard notation where variables are represented as round nodes and factors as square nodes. In a probabilistic model, the factors refer to probabilistic distributions, deterministic functions, or constraints. Graphically, the factor nodes connect only to variable nodes that are arguments of the factor. The factors are multiplied together to give an overall distribution function. In this sense, a factor graph is a visual representation of the dependency structure among variables in the overall distribution. In case of generative models, for example, we aim to explain the observed data and typically arrive at a rich dependency structure where latent and observed variables are generated from parent variables via a factor. In Section 4 we describe in detail the generative processes inherent in our listening taste and mood models and demonstrate how both the generative process and the joint probability distribution can be directly read off the corresponding factor graphs.

Figure 1: The two-level genre taxonomy of Zune Social. Genres have sub-genres. Examples of sub-genres are shown only for the genres *Rock* and *Classical*.

Factor graphs utilize additional notation that simplifies the visual representation such as *plates* (see for example [1]) which represent replicated parts of the model, and *gates* [17] which represent parts of the model that are switched on or off depending on the value of a random variable. Plates are shown as rectangles with a solid boundary line, and gates are shown as dashed rectangles, with the gating variable attached to the rectangle rather than to the variables inside. The factors inside the gate are switched on or off by the value of the gating variable.

2.2.2 Inference in factor graphs

While useful for visualizing relationships and conditional independence among variables, factor graphs are particularly important as a framework for describing message-passing algorithms for performing inference. In this paper we make use of a message-passing algorithm for approximate inference called variational message passing (VMP) [22]. This is one of a class of algorithms that are given a unified treatment in [15].

These algorithms typically make use of a fully factorized approximation of the joint probability distribution; i.e. a factorization of each factor itself into univariate factors. For each factor in the graph, the algorithm will calculate outgoing messages from the factor to each variable; each message is in the form of a univariate distribution over the target variable, and is calculated from the factor itself and all the incoming distribution messages via an update equation

User	Media item	Timestamp
u1	m1	07-11-09 08:12:06
u1	m2	07-11-09 08:15:42
u2	m5	07-11-09 08:16:41
u1	m3	07-11-09 08:20:17
u2	m4	07-11-09 08:20:33
u2	m1	07-11-09 18:56:43
u2	m1	07-12-09 23:54:45
u2	m2	07-12-09 23:58:21
u2	m1	07-13-09 00:01:12
u1	m4	07-15-09 07:54:04
u1	m5	07-15-09 07:56:43

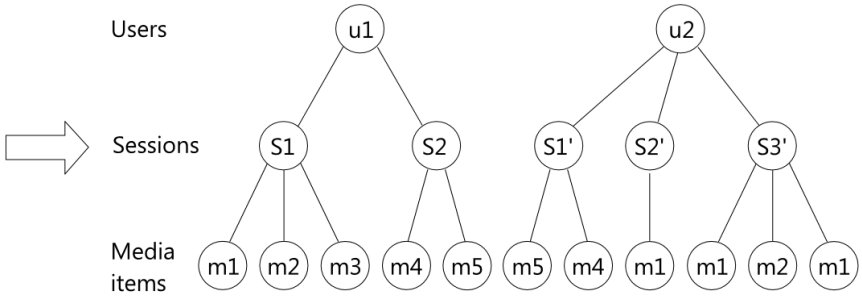

Figure 2: Log data for two users and their corresponding music-listening sessions and media items.

which minimizes a local divergence measure. The factorized approximation to the factor is given by the product of the outgoing messages.

These message-passing algorithms are fast and also have the benefit that calculations are local, so complex models can be pieced together with reusable building blocks — the Dirichlet and Discrete factors in (Figure 3 and Figure 4) are two such building blocks, as are the message update equations to deal with plates and gates. Infer.NET [16], which we use to perform inference in our models, is a framework which makes good use of these considerations to provide a variety of message-passing algorithms for graphical models.

3. SOCIAL MEDIA CONTEXT

In this section we motivate the work through the example of a specific social media service.

3.1 Social media description

For the purposes of our study we consider the Zune Social music community and analyze the data set that comprises 14 weeks of usage logs. For each registered user the Zune Social service maintains a user profile with a list of songs that the user has listened to on the Zune device or via Zune software installed on a personal computer.

The Zune Social community members can rate songs, establish friendship links, and recommend songs to each other. Songs are classified using a two-level genre taxonomy. Figure 1 shows all 17 top level genre categories and the second level categories for two specific genres, *Rock* and *Classical*. The full taxonomy can be found on the Zune Social website.

Our objective is to capture users' listening affinities as reflected in the data logs. Thus, we make a concerted effort to clean the usage logs of accidental data access and playing of songs. For each user we consider only those instances where the user listened to a song and rated it positively. This set could be easily expanded using different heuristics. For example, one could include songs that have no ratings but are listened to multiple times by the user. Analysis of our data shows that, on average, the rated songs are listened to 3.62 times. In comparison, the average/mean across all the songs is only 2.26 times.

We assume that the users listen to songs during listening sessions and we employ a simple segmentation technique to specify the session boundaries. We study the distribution of time intervals between the start times of consecutive songs played by the same user. We identify the peaks and use them as thresholds for determining the start of the new session. We observed a few prominent peaks in the distribution. One of the peaks corresponds to the average song length (3.67 minutes).

3.2 Terminology and data representation

Let $U = \{u_1, ..., u_n\}$ represent a set of users and $M = \{m_1, ..., m_k\}$ represent a set of media items that the users can listen to. A media item can be a song genre, an artist or a particular song. For ease of representation and without loss of generality, we will refer to a media item as a song. Each song-listening instance (u, m, t) represents user u listening to song m at time t. In order to define listening sessions, we define an *interval* as the time difference between the start times of two consecutive songs for the same user. Alternatively, one can define an interval as the time difference between the end time of one song and the start time of the next song but we chose the former definition because we did not have information of the song end times in our data. A *session* $S = (m_1, ..., m_l)$ is then a sequence of l songs that the user u has listened to, such that the interval between every two consecutive songs m_i and m_{i+1} is below a specified threshold $p_{threshold}$. The playlist $\mathbf{S}_u$ of each user includes a sequence of song-listening sessions $\mathbf{S}_u = (S_1, ..., S_{t_u}) = (m_1, ..., m_N)$. Note that, for the same user, a song can be repeated both in the same session, and across sessions. We also assume that there are latent media clusters $C = \{c_1, ...c_n\}$ which explain the co-occurrence patterns of songs that users play, and they provide a soft clustering of the media items M. Thus, for each cluster c_i, there is a distribution ψ_i over the media items M.

Figure 2 shows an example of the data model. The table shows the log of two users $u1$ and $u2$ who have listened to 5 media items at different time points. The log data is visualized as a tree, showing the segmentation into sessions based on the time interval threshold. This threshold can be predefined or learned from data. This example shows some patterns: session $S2$ of user $u1$ is the same as session $S1'$ of user $u2$, and session $S1$ of user $u1$ is similar to session $S3'$

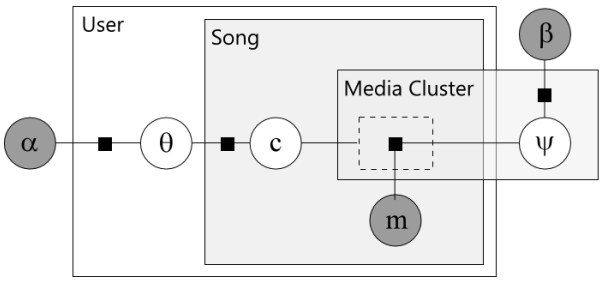

Figure 3: Factor graph of the Taste model.

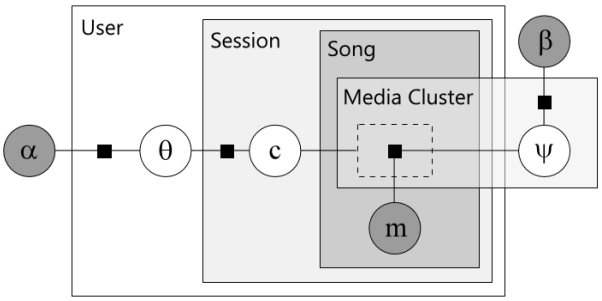

Figure 4: Factor graph of the Session model.

of user $u2$. The goal of our paper is to find and characterize such patterns.

4. STATISTICAL MODELS

Here we describe in detail the *taste model*, and also our *session model* which extends the taste model and captures the listening mood across song-listening sessions.

4.1 Taste model

Following the LDA model [2], we define a probabilistic graphical model that represents consumption of media as a distribution over a set of latent media clusters, referred to as 'tastes.' Each taste media cluster is represented as a distribution over the songs. The model generates each song m in the user's playlist $\mathbf{S}_u$ by picking one of the media clusters c, and then picking a song from that media cluster's mixture ψ. We refer to this model as the *taste model* because each media cluster represents a particular taste. It is a direct adaptation of the LDA model.

A factor graph of the model is shown in Figure 3 where the rectangles indicate plates of users, songs of a user, and media clusters. For each user, and each song in the user's playlist, the variable c switches on a particular media cluster, and switches off the others.

The following process describes the generation of a playlist $\mathbf{S}_u$ for each user u:

1. For each media cluster k
   (a) Choose a distribution over songs, $\psi_k \sim Dir(\beta)$
2. For each user u
   (a) Choose a distribution over media clusters, $\theta_u \sim Dir(\alpha)$
   (b) For each song in the user's playlist $\mathbf{S}_u$
      i. Choose a media cluster $c_{uj} \sim Discrete(\theta_u)$
      ii. Observe song $m_{uj} \sim Discrete(\psi(c_{uj}))$

$Dir(\alpha)$ is an exchangeable Dirichlet prior, i.e., all pseudo-counts are identical and given by the parameter α. $\theta(u) \sim Dir(\alpha)$ is the parameter vector for a user-dependent Discrete distribution over media clusters. $Dir(\beta)$ is also an exchangeable Dirichlet prior and $\psi(c) \sim Dir(\beta)$ is the parameter vector for a cluster-dependent Discrete distribution over songs.

The number of media clusters K is fixed in advance but this constraint can be alleviated as discussed by Blei et al. [2]. According to this model, the joint probability distribution of the distributions ψ over songs, the distributions θ over clusters, the cluster choice c for each user and song, and the songs in user u's playlist $\mathbf{S}_u = (S_1, ..., S_t(u)) = (m_1, ..., m_N)$, is:

$$p(m, c, \psi, \theta | \alpha, \beta) = \prod_{u=1}^{n} p(\theta_u|\alpha) \prod_{j=1}^{N} p(m_{uj}|\psi(c_{uj}))p(c_{uj}|\theta_u)) \prod_{k=1}^{K} p(\psi_k|\beta).$$

We then observe $(m_1, ..., m_N)$ and perform Bayesian inference to recover the posterior marginal distributions of ψ and θ.

4.2 Session model

We use the session model to detect music-listening *moods* as exhibited in song-listening sessions. Mood is a latent variable in the session model. The model assumes that each user is represented as a distribution over different moods, and for each session, there is a latent mood which guides the choice of songs. A factor graph of the model is shown in Figure 4. Here, the media cluster c represents the mood as a mixture of songs.

The session model assumes that $\psi(c)$ for each mood c is picked from $Dir(\beta)$. The following process describes the generation of each user's playlist $\mathbf{S}_u$:

1. For each media cluster k
   (a) Choose a distribution over songs $\psi_k \sim Dir(\beta)$
2. For each user u
   (a) Choose a distribution over media clusters $\theta_u \sim Dir(\alpha)$
   (b) For each session $S_i \in \mathbf{S}_u$
      i. Choose a media cluster $c_{ui} \sim Discrete(\theta_u)$
      ii. For each song in the session, observe $m_{uij} \sim Discrete(\psi(c_{ui}))$

The joint distribution is:

$$p(m, c, \psi, \theta | \alpha, \beta) = \prod_{u=1}^{n} p(\theta_u|\alpha) \prod_{i=1}^{t_u} p(c_{ui}|\theta_u) \prod_{j=1}^{l_i} p(m_{uij}|\psi(c_{ui})))) \prod_{k=1}^{K} p(\psi_k|\beta)$$

When there is one song per session (each song in the playlist has its own session), then the session and taste models are equivalent. As the number of songs per session grows,

inference for the session model gets faster than inference on the taste model because it has fewer random variables. In other words, the cluster variable c is picked only once per session and it remains the same for all the songs in the session, whereas in the taste model, c is picked every time a song is generated.

The *session model* embodies the finer level structure in the data. Just as the LDA model, the session model can be applied to a corpus of documents and capture word pattern on the sub-document level. For example, by constraining words within chunks of the document, e.g., paragraphs, to belong to the same topic, we begin to identify topic patterns associated with paragraphs. Again, an important advantage is the simplified inference and, consequently, the ability to process large document collections efficiently.

5. EVALUATION

We present results for the problem of playlist generation and discuss the characteristics of the media clustering approach by visualizing the genres per cluster, comparing the discovered latent clusters with the genre taxonomy, investigating the sensitivity of the clustering to the number of pre-specified clusters, and measuring the time performance of the models. We represent each song-listening instance in terms of the corresponding song genre. Since each song can belong to one or more music genres $g \in G$, for each song-listening instance, there are multiple genre instances. We use this media representation to study the connection between the latent media clusters that correspond to listening mood and taste and the song genres. Furthermore, we can explore the usefulness of our models for generating song playlists of individual users. We do that by predicting the genre of the song that the user may want to hear next during the listening session, considering the few seed songs that the user has already listened to. By identifying the desired genre we provide a good foundation for selecting specific songs to present to the user.

5.1 Data sample

We train and evaluate the models using a sample of $2,014$ users who have listened to songs that belong to 84 different music genres. From the 14 weeks of data, we use the first two months as training data to learn the parameters of each model and the rest as the test data. Considering the song-listening instances in the training data we arrive at $239,425$ genre instances and $14,703$ sessions using a time interval threshold of 30 minutes and no restriction on the number of songs per session. The test data contains $248,631$ genre instances in $5,079$ sessions which contain at least 11 genres. We control the minimum number of genres per session in order to allow testing the session model with 5 and 10 seed songs. The sample includes all users who have joined the Zune Social service in the studied period, and whose playlists include between 120 and 200 different music artists.

5.2 Inference

We implemented the statistical models using Infer.NET, an efficient, general-purpose inference engine for graphical models [16]. Since exact inference is not possible in the taste and session models, we used variational message passing [22] for learning the parameters of each model.

We fixed $\beta = 0.5$ and $\alpha = \frac{1.5}{K}$. β was set to give the best performance for the baseline test model (see Section 5.3.1),

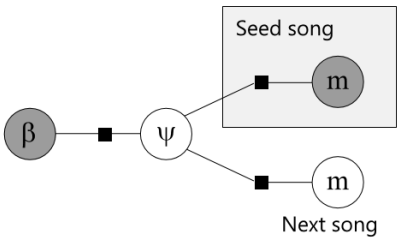

Figure 5: Factor graph of the baseline, unigram model.

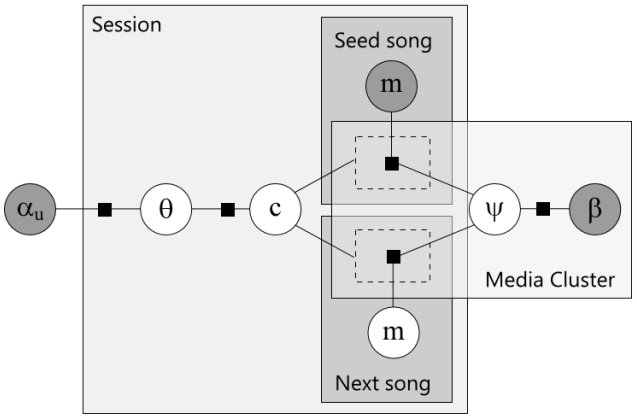

Figure 6: Factor graph of the test model for evaluating the session and taste models.

and the same value was used for the taste and session models. The value of α was set based on limited manual optimization with respect to the taste model and adopted for the session model as well.

5.3 Results for playlist generation

We evaluated the proposed session model by comparing its performance in terms of model perplexity to that of the taste model on the task of playlist generation for a song-listening session. Besides these two models, we consider a *unigram model* as a simple baseline model that does not consider latent media clusters and learns each session distribution over genres independently. First, we present the unigram model in more detail and then we describe the experimental setup and results.

5.3.1 Baseline test model

In the unigram model the genres in each music-listening session are drawn independently from a single discrete distribution that describes the session. A factor graph of the model is shown in Figure 5. More specifically, the generative process is as follows:

1. For each session $S_i \in \mathbf{S}$
   (a) Choose $\psi_i \sim Dir(\beta)$
   (b) For each song in the session, observe $m_{ij} \sim Discrete(\psi_i)$

Here, $Dir(\beta)$ is an exchangeable Dirichlet prior and ψ is the parameter vector for a Discrete distribution over songs. During inference, it learns the distribution over genres based

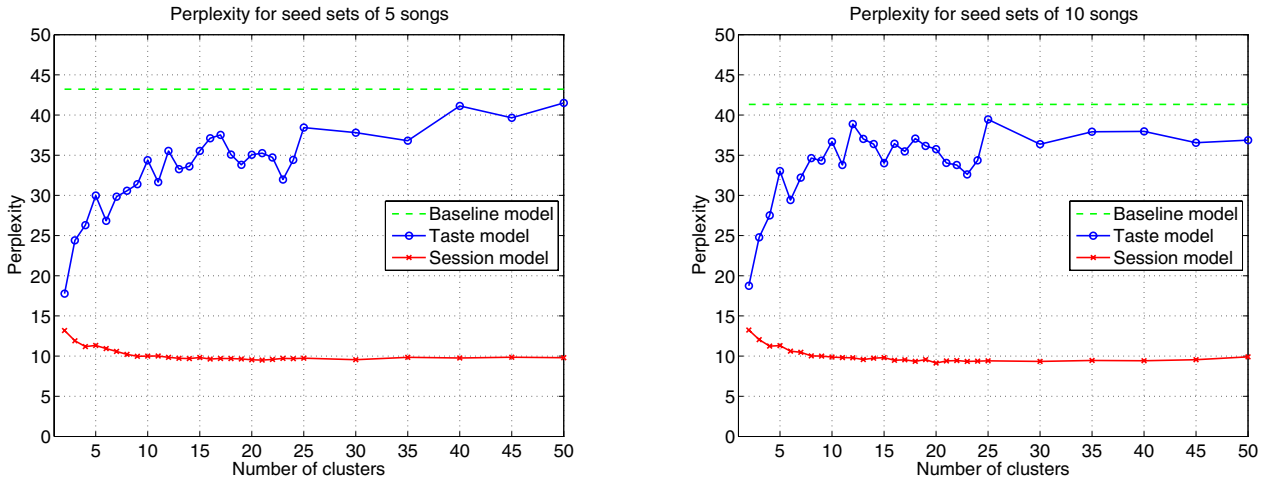

Figure 7: Comparison of the perplexity of each model for session genres after observing a) 5 seed genres and b) 10 seed genres.

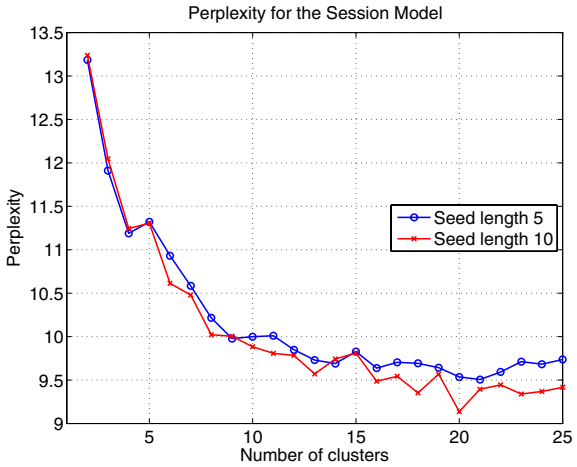

Figure 8: Session model perplexity for session genres after observing 5 or 10 seed genres.

on the seed songs in the session and uses it to predict the genres of the remainder of the songs in the session.

The joint distribution for session S_i is

$$p(\psi_i, m|\beta) = p(\psi_i|\beta) \prod_{j=1}^{l_i} p(m_{ij}|\psi_i)$$

This model assumes that sessions are independent of each other and, unlike the taste and session models, it does not consider latent media clusters.

5.3.2 Test model

The taste and session models learn the posterior distributions for their parameters from the training data. These posteriors are used as priors in the testing phase. In the testing phase, the model "observes" the first few seed songs, in our case 5 or 10 songs in a test session, it infers the posteriors of the model parameters, and then finds the likelihood of the genres for the rest of the session songs.

Figure 6 shows a factor graph of the test setup for the session and taste models. In the test setup, α_u is the pseudo-count vector of the posterior Dirichlet distribution for θ_u from the training model, where u is the user whose listening session is used as a test. Similarly, for each cluster, β is the pseudo-count vector of the posterior Dirichlet distribution for the ψ of that cluster, derived from the training model. Performing inference on the test model then finds the posterior Dirichlet distributions for θ, the session's distribution over clusters, and ψ, the cluster's distribution over genres, based on a few seed songs (*Seed song* plate in Figure 6). Then the log-likelihood is calculated for the genres of the remaining session songs.

5.3.3 Performance metric

In order to assess which model explains the co-occurrence of song genres in listening sessions better, we compare the perplexities of the three models. Perplexity is an entropy-based score assigned to a probabilistic model and commonly used to evaluate topic models such as LDA [2]. It captures how well a model trained on observed data would predict unobserved data. The lower the perplexity of a model, the better its predictive power. We report on the perplexity of each model on the test data:

$$Perplexity = \exp(\sum_{u=1}^{n} \sum_{S \in S_u} \sum_{i=seed+1}^{size(S)} \frac{1}{G} \ln(p(m_i|\psi(c_{ui}))))$$

Computing the perplexity involves finding the log-probabilities of genres in each test session, excluding the seed song genres, and averaging over the number of genre instances G.

5.3.4 Results

Figure 7 shows the perplexity scores for the three models: baseline, taste and session models. The session model has consistently lower perplexity than both the baseline and the taste model for the number of clusters between 2 and 50. That means it models better than the other two the patterns of co-occurring genres within the same music-listening session. The lowest perplexity of the session model occurs at 21 clusters for 5 seed songs (9.51), and at 20 clusters

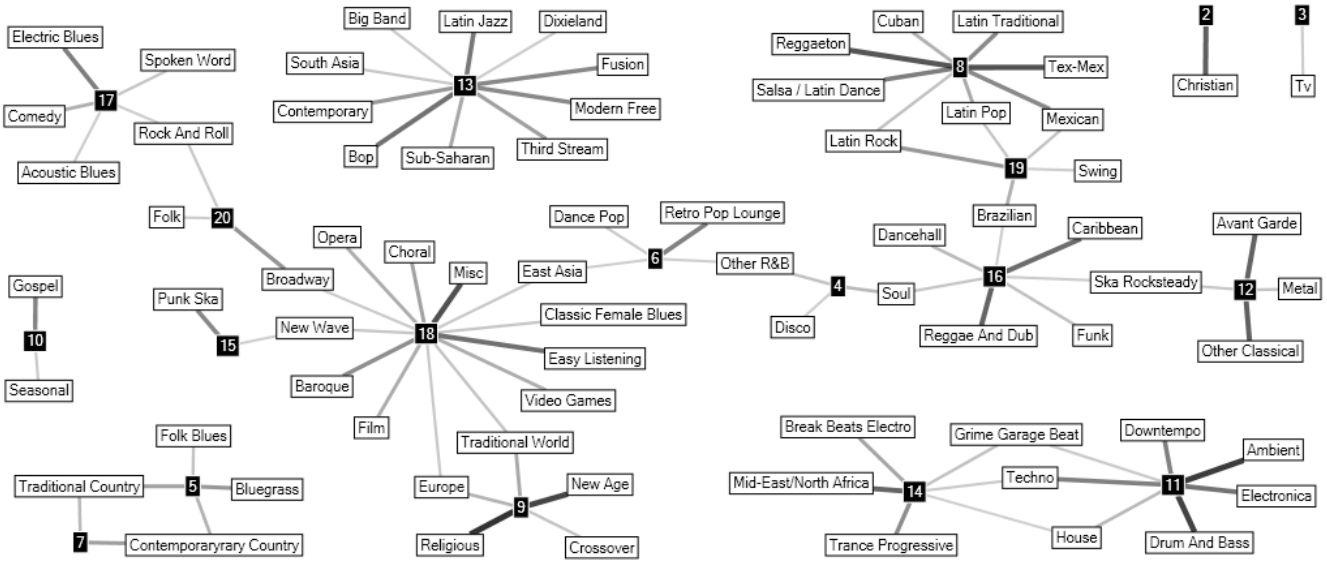

Figure 9: Resulting media clusters for the session model. Line thickness signifies cluster affiliation strength.

for 10 seed songs (9.14), while the lowest perplexity of the taste model occurs at 2 clusters (with perplexity of 18.74 for 5 seed songs, and 17.77 for 10 seed songs). The baseline model perplexity is 43.22 and 41.32 for 5 and 10 seed songs, respectively, and it is constant since it does not assume any latent clusters. These results imply that for the problem of playlist generation, it is better to consider the local patterns across sessions, as captured by the session model, rather than global patterns characterized by the taste model.

Figure 8 shows the results for the session model in more detail. It shows that the predictive power of the model increases as we increase the number of clusters up to $20-21$ clusters, depending on the number of seed songs.

5.4 Characterizing latent media clusters

We can visualize the affinity of genres to clusters by looking at the distribution of each media cluster over the genre categories. Figure 9 shows how genres are associated with listening mood clusters produced by the session model. In the graph we show connecting edges only if the normalized Dirichlet posterior of a genre in the media cluster is more than 0.25. The thickness of the edge reflects the strength of the genre affiliation with the cluster.

We observe that some latent clusters of genre resemble the groupings of genre in the taxonomy shown in Figure 1. Indeed, media clusters 8 and 11 have similar genre grouping as the top genre categories *Latin* and *Electronic/Dance*, respectively. On the other hand, the media cluster 6 comprises a mixture of high-level genres: *Electronic/Dance*, *R&B*, *Pop* and *World*.

5.4.1 Comparing latent clusters with taxonomy

In Section 5.4 we showed that, in some cases, the collection of genres associated with a listening mood corresponds to one of the top-level genres from the Zune Social taxonomy. For other moods that is not the case. Here, we examine how close a media clustering is to the genre taxonomy, i.e., we estimate how well the static genre taxonomy reflects the listening patterns that emerge from the users' behavior in the social media. The taxonomy itself can be considered as a collection of clusters where two sub-genres are in the same cluster if and only if they have the same parent genre.

5.4.2 Similarity metric

To compare two media clusterings, we employ a similarity metric based on the *Mallows distance* [10, 24]. This measure is well-suited for comparing clusterings in which the clusters are soft and exchangeable, i.e., it is not known beforehand which pairs of clusters to compare. Zhou et al. [24] discuss the advantages of this measure over other measures for clustering similarity, such as *pair counting*, *set matching* and *variation of information*. The Mallows distance measures the difference between two multivariable probability distributions, and it can be interpreted as an optimal cluster matching scheme between two clusterings C_1 and C_2:

$$Mallows(C_1, C_2) = \min_{w_{k,j}} \sum_{k=1}^{K} \sum_{j=1}^{J} w_{k,j} \sum_{i=1}^{N} |p_{i,k} - q_{i,j}|$$

with the constraints that $w_{k,j} \geq 0$, $\sum_{k=1}^{K} w_{k,j} = \beta_j$, $\sum_{j=1}^{J} w_{k,j} = \alpha_k$ for all k, j. To compute the Mallows distance, one has to solve an optimization problem using linear programming. It yields a global optimum which is unique.

In our case, the computation involves the pseudo-counts for the media cluster posteriors. For each genre, we normalize across clusters to get $p_{i,k}$ where i is a genre index and k is a cluster index. Similarly for $q_{i,j}$. Then, we find the total count for each Dirichlet and normalize across clusters to get the α_k and β_j. For the optimization part, we apply linear programming using Microsoft Solver Foundation[3].

5.4.3 Cluster comparison results

Figure 10 shows that, as the number of clusters increases, the similarity between the genre clusters derived by the ses-

[3] `http://code.msdn.microsoft.com/solverfoundation`.

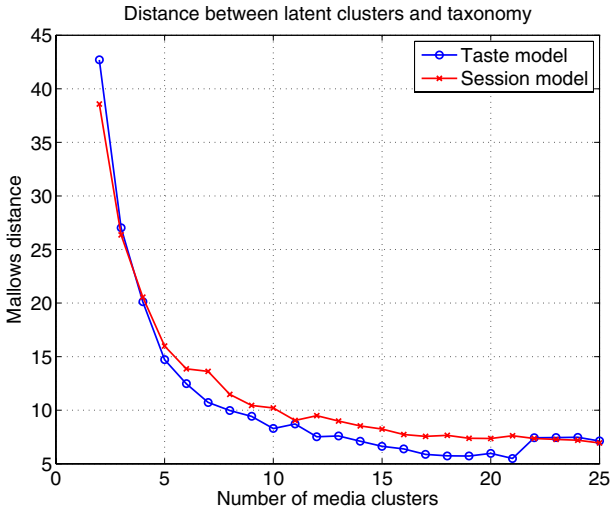

Figure 10: Mallows distance between the genre taxonomy and the clusterings found by the taste and session models.

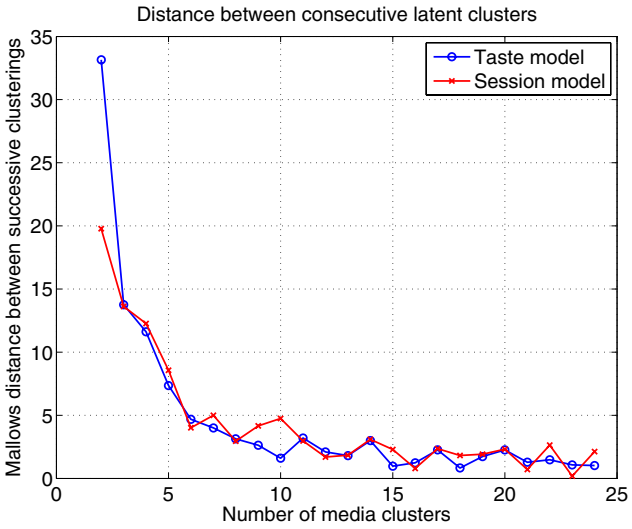

Figure 11: Sensitivity of the models to the pre-specified number of clusters.

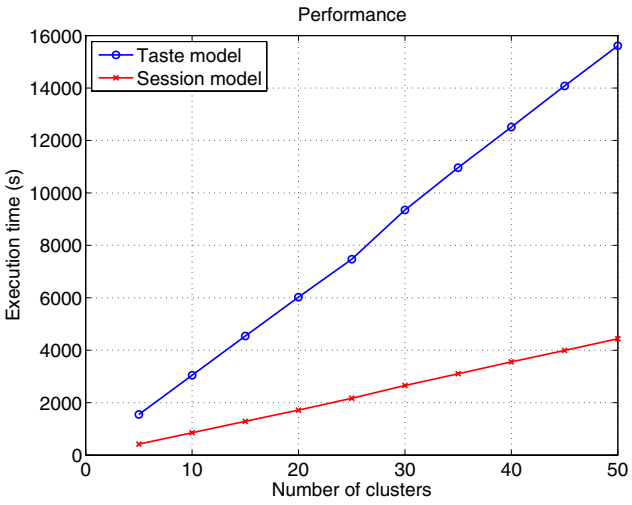

Figure 12: Model training time.

sion or taste model and the Zune genre taxonomy increases as well. For a range of cluster numbers, the Zune genre taxonomy is slightly more similar to clusters resulting from the taste model than from the session model. However, for both models the resulting genre clusters are different from the original genre taxonomy. Thus, the clusters provide alternative groupings of genre categories that reflect the usage of mobile media and the preferences of the community, as confirmed by the perplexity results in Section 5.3.4.

5.5 Sensitivity to number of clusters

In this section, we conduct a simple experiment to investigate how sensitive the models are to the pre-specified number of clusters. For that, we look at the similarity between clusterings that correspond to successive numbers of clusters. For example, we measure whether a clustering with 15 media clusters is very different from a clustering with 16 clusters. It is of interest to know how the similarity between them changes and whether the clusterings converge. We use the Mallows distance as the similarity score. The larger the Mallows distance between two successive clusterings, the more sensitive the clustering model is to increasing the pre-specified number of clusters.

Figure 11 shows that when we increase the number of clusters, the sensitivities of both the taste and session models decrease, i.e. the clusterings become more similar to each other. However, for low numbers of clusters, the clusterings are very different from each other. For example, the distance between the clusterings produced for 2 and 3 clusters is 33.2 for the taste model, and 19.5 for the session model.

5.6 Time performance of the models

One of the important aspects of statistical models is the computational time required to train the models. Our comparison of the taste and session models confirms that training of the session model is faster. As expected, for both models the training time increases linearly with the number of clusters. However, the rate of increase differs. On our data sample, inference using the session model is 3.7 times faster than for the taste model as Figure 12 shows.

6. DISCUSSION

Reflecting on the experimental results, we consider possible application scenarios. In music communities such as Zune Social or Last.fm, our approach can be used to enrich user experience. Through media clustering, the service can provide song recommendations based on the collective community tastes and listening moods. As we have shown, the session model can facilitate the playlist completion based on previous listening sessions or several songs that the user has just listened to. Indeed, this can be presented as an improved *shuffle feature* offering a selection of song snippets as short previews during a listening session. The shuffle could adapt based on the user's mood. Furthermore, as an added benefit to identifying media clusters, our models produce groupings of individuals with shared tastes and moods. This information can be leveraged to suggest new friendship ties between users in the social media community.

From the perspective of the service architecture and optimization, clustering media content can contribute to im-

proved load balancing and more efficient content access. Since social media services can involve millions of users on a daily basis, it can be beneficial to distribute service requests across several servers based on appropriate media clusters.

From research point of view, it would be interesting to study user interpretations of the discovered media clusters. It would be valuable to investigate whether latent media clusters, representing for example moods, correspond to different experiences that the users may be able to articulate.

7. CONCLUSION

In our paper we presented a novel and improved statistical model for characterizing user preferences in consuming social media content. By taking into account information about the listening sessions of individual users, we arrive at a new, session-based hierarchical graphical model that has lower perplexity and a shorter training time than alternative approach based on the standard LDA model.

Using the data from the Zune Social music community, we show how generative probabilistic models enable us to capture latent variables that drive the consumption of media. In particular, we adapted the LDA model to capture the taste in music and we define a session based model that captures the user mood in listening sessions. Thus, an instance of song listening can be represented as a finite mixture of the underlying tastes that have been discovered through statistical modeling. Similarly, a song listening within a session can be modeled with respect to the latent moods that the session model generates. Both taste and mood are essentially media clusters that are identified from the statistical analysis of the media usage.

In Zune Social the songs are classified using a fixed two-level taxonomy of music genres. We use genre to characterize the individual songs, and the resulting taste and mood media clusters are represented as genre distributions. In our analysis we conclude that both the taste and mood-based clusterings derived from usage data differ from the static taxonomy. Thus, they offer alternative genre taxonomies, informed by the community listening patterns. Furthermore, we show that the resulting clusters can be used for playlist generation. The service can thus recommend songs based on a few songs that the user has already listened to.

Our future work will focus on refinements of the session model to capture additional aspects of song listening. One such aspect is listening 'saturation' that would require extending the model to include a 'decay factor.' We also intend to explore application and evaluation of the session model in contexts other than online media consumption.

8. ACKNOWLEDGMENTS

The authors would like to thank Gavin Smyth for sharing his database expertise, Tom Minka and John Winn for discussions on the models, and Jordan Boyd-Graber for providing feedback on an earlier draft of the paper.

9. REFERENCES

[1] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[2] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet allocation. *JMLR*, 3:993–1022, January 2003.

[3] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6), 1990.

[4] Y. Feng, Y. Zhuang, and Y. Pan. Music information retrieval by detecting mood via computational media aesthetics. In *WI*, 2003.

[5] K. Henderson and T. Eliassi-Rad. Applying latent Dirichlet allocation to group discovery in large graphs. In *SAC*, pages 1456–1461, 2009.

[6] M. Hoffman, D. Blei, and P. Cook. Easy as CBA: a simple probabilistic model for tagging music. In *ISMIR*, 2009.

[7] T. Hofmann. Probabilistic latent semantic analysis. In *UAI*, 1999.

[8] X. Hu and J. S. Downie. Exploring mood metadata: Relationships with genre, artist and usage. In *ISMIR*, pages 67–72, 2007.

[9] D. Liu, L. Lu, and H.-J. Zhang. Automatic mood detection from acoustic music data. In *ISMIR*, 2003.

[10] C. L. Mallows. A note on asymptotic joint normality. *The Annals of Mathematical Statistics*, 43(2):508–515, 1972.

[11] B. Marlin. Modeling user rating profiles for collaborative filtering. In *NIPS*, 2003.

[12] B. Marlin and R. Zemel. Collaborative prediction and ranking with non-random missing data. In *RecSys*, 2009.

[13] A. McCallum, X. Wang, and A. Corrada-Emmanuel. Topic and role discovery in social networks with experiments on Enron and academic email. *JAIR*, 30:249–272, 2007.

[14] A. McCallum, X. Wang, and N. Mohanty. Joint group and topic discovery from relations and text. In *Statistical Network Analysis: Models, Issues and New Directions*, volume 4503, pages 28–44. LNCS, 2007.

[15] T. Minka. Divergence measures and message passing. Technical report, Microsoft Research, 2005.

[16] T. Minka, J. Winn, J. Guiver, and A. Kannan. Infer.NET 2.3, 2009. Microsoft Research Cambridge. http://research.microsoft.com/infernet.

[17] T. Minka and J. M. Winn. Gates. In *NIPS*, 2008.

[18] E. Pampalk, T. Pohle, and G. Widmer. Dynamic playlist generation based on skipping behavior. In *ISMIR*, 2005.

[19] J. C. Platt, C. J. C. Burges, S. Swenson, C. Weare, and A. Zheng. Learning a Gaussian process prior for automatically generating music playlists. In *NIPS*, 2001.

[20] R. Ragno, C. J. C. Burges, and C. Herley. Inferring similarity between music objects with application to playlist generation. In *MIR*, 2005.

[21] D. H. Stern, R. Herbrich, and T. Graepel. Matchbox: large scale online Bayesian recommendations. In *WWW 2010*, pages 111–120, 2009.

[22] J. Winn and C. Bishop. Variational message passing. *JMLR*, 6:661–694, 2005.

[23] H. Zhang, B. Qiu, C. L. Giles, H. C. Foley, and J. Yen. An LDA-based community structure discovery approach for large-scale social networks. In *ISI*, 2007.

[24] D. Zhou, J. Li, and H. Zha. A new Mallows distance based metric for comparing clusterings. In *ICML*, 2005.