Actionable Mining of Large, Multi-relational Data using Localized Predictive Models

Joydeep Ghosh and Aayush Sharma

Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, Texas 78712, U.S.A.

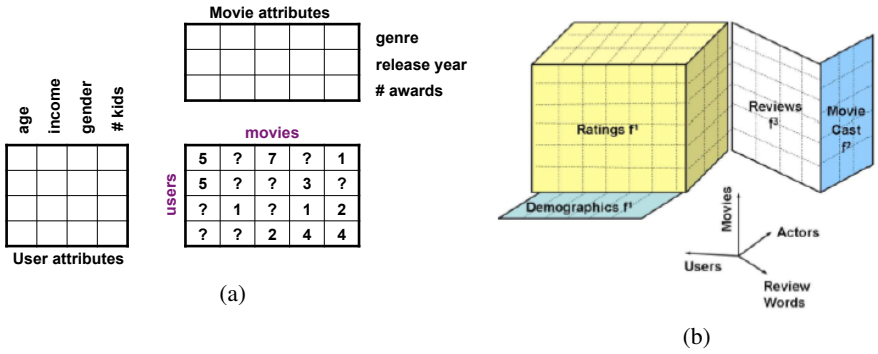
Abstract. Many large datasets associated with modern predictive data mining applications are quite complex and heterogeneous, possibly involving multiple relations, or exhibiting a dyadic nature with associated side-information. For example, one may be interested in predicting the preferences of a large set of customers for a variety of products, given various properties of both customers and products, as well as past purchase history, a social network on the customers, and a conceptual hierarchy on the products. This article provides an overview of recent innovative approaches to predictive modeling for such types of data, and also provides some concrete application scenarios to highlight the issues involved. The common philosophy in all the approaches described is to pursue a simultaneous problem decomposition and modeling strategy that can exploit heterogeneity in behavior, use the wide variety of information available and also yield relatively more interpretable solutions as compared to global "one-shot" approaches. Since both the problem domains and approaches considered are quite new, we also highlight the potential for further investigations on several occasions throughout this article.

1 Introduction

Classical methods for predictive modeling (regression, classification, imputation, etc.) typically assume that the available data is in the form of a single "flat" file that provides, for each record or entity, a list of the values for the independent and (when available) dependent variables associated with it. However many modern data driven applications involve much more *complex* data structures such as multi-modal tensors, sets of interlinked relational tables, networks of objects where both nodes and links have properties and relationships, as well as other dependencies/constraints such as hierarchical or spatial orderings [17]. Drastic problems, including entry duplication and skewing of counts, that can occur when such data forms are forced into a single "flat" format are well known [23, 17]. Therefore, researchers have increasingly concentrated on ways to directly analyze datasets in their natural format, including notable efforts on tensor [49, 29, 30] and multi-relational [19] data mining.

This article focuses on predictive modeling of *dyadic (bi-modal)* data that consist of measurements on dyads, which are pairs of entities from two different sets (modes). The measurements can be represented as the entries of a matrix, whose rows and columns are the two sets of entities. Moreover, independent variables (attributes or *covariates*) are

associated with the entities along the two modes. For concreteness, consider a movie recommendation system to predict user ratings for movies, for which there are additional covariates associated with each user (age, gender, etc) and each movie (genre, release year etc), in addition to ratings data. Attributes may also be associated with a user-movie pair, e.g., whether a user's favorite actor is in the movie. Such data structures, which we shall refer to as "Dyadic data with Covariates" or **DyaC**, can be conceptually visualized as in Fig. 1(a). From the figure, it is clear that the data involves multiple tables and cannot be naturally represented as a single flat file.



(a) (b)

Fig. 1. (a) Conceptual representation of a "Dyadic data with Covariates" (DyaC) dataset on user-movie ratings. (b) Example of multiple relations with some shared modes [8].

The characteristic problem to be solved with such data is one of estimating the affinity (e.g. rating) between the modes (e.g. users and movies) given the values of a small number of such affinities. Indeed, recommender systems, which are a special case of this setting, have proved to be very successful at identifying affinities between users and items. Identifying personalized content of interest can greatly enrich the user experience and help institutions offering the recommendations to effectively target different kinds of users by predicting the propensity of users towards a set of items. Marketing data also lends itself perfectly for an affinity estimation problem wherein effective marketing strategies can be formulated based on the predicted affinities. Additionally, there are useful applications in estimating affinities as clickthrough rates for online ads associated with users, search queries, or web pages.

Many current approaches for affinity estimation have concentrated only on a small number of known affinities to infer the missing ones [41]. However, there are often available, many auxiliary sources of information associated with the entities that can aid the estimation of affinities between them. For example, in a movie recommendation engine, the attributes associated with a user might consist of demographic information such as age, gender, geo-location etc. that are often collected as profile information at the time of user registration. Similarly, movie attributes consist of readily available

features such as genre, release date, running time, MPAA rating etc. The attributes associated with entities can have a strong bearing on the affinity relationships. For example, it may be common for adolescent males to enjoy movies about comic book characters. In this case, it could be very helpful to have the age and gender information of the user when attempting to predict the affinity of that user for such a movie. Another important source of auxiliary information about entities is a neighborhood structure such as a social network represented by a user-user directed graph. The linkage structure can have an impact on a user's affinities, since preferences are often influenced by preferences of one's friends.

Another problem associated with the methods relying only on past affinities is their inability to intelligently cater to affinity estimation for new entities with no prior history of affinities. This is referred to as a *cold-start* problem. The best one can do with these methods is to utilize a global average model, which however, fails to capture subtle correlations that might exist between a few existing and the new entities. Accurate prediction of affinities for new entities is very crucial for many applications. In the recommender system example, predicting affinities for a new product before its launch could help companies to use more targeted marketing techniques, and could help users recently introduced to the system to quickly find products that they will find useful.

A third aspect of large affinity datasets is that they typically exhibit substantial heterogeneity. Here heterogeneity implies that the relationship between the independent and dependent attributes varies substantially in different parts of the problem space. In such scenarios, it is more appropriate to develop multiple predictive models, one for each relatively homogeneous part of this space, rather than building a single global model. A multi-model approach has several advantages in terms of accuracy, interpretability, reliability and computation [37]. It also provides alternate, more effective ways of active learning and incremental learning as well [16]. However such approaches also raise additional issues such as how to best determine the nature and scope of each model as well as the total number of models developed. The issue of hard vs. soft decomposition of the problem space is also intimately related to interpretability and actionability of the overall solution, leading to non-trivial tradeoffs.

The vast majority of the substantial recent literature on recommender systems (many motivated by Netflix!), including collaborative filtering and matrix factorization methods [26, 48, 41] simply ignore the covariate information if present, and concentrate solely on the ratings matrix. This is also true of co-clustering or stochastic blockmodel based approaches [22, 38] that group elements along both modes, and then model the responses to be homogeneous within each (user, movie) group in the cartesian-product space. A few works do incorporate "side-information" provided by covariates indirectly, typically through a kernel, or as a regularizer to matrix factorization [1, 33, 9], but they do not exploit heterogeneity. On the other hand, while the use of multiple predictive models to deal with heterogeneity is encountered in a wide range of disciplines, from statistics to econometrics to control and marketing [37, 39, 32, 21, 24, 40, 35], these approaches typically apply only to single flat-file data representations.

In KDD'07, two related approaches were introduced: SCOAL (Simultaneous Coclustering and Learning) [12] and PDLF (Predictive Discrete Latent Factor Modeling) [5], the first work being nominated for and the second one receiving the Best Re-

search Paper Award. Both papers proposed ways to address DyaC data using localized models that could exploit data heterogeneity, and these approaches have been subsequently refined and expanded. Considering the user-movie recommendation problem again for concreteness, both SCOAL and PDLF will iteratively partition the user-movie matrix into a grid of blocks (co-clusters) of related users and movies, while simultaneously learning a predictive model on each formative co-cluster. The predictors *directly* use covariate information as opposed to the indirect usage of "side-information" through a soft similarity constraint [1, 33, 9]. The organic emergence of predictive models together with the co-clusters that determine their domains, improves interpretability as well as accuracy in modeling several heterogeneous, DyaC datasets, as this mechanism is able to exploit both local neighborhood information and the available attributes effectively [12, 5]. We call the strategies taken by these two methods *Simultaneous Decomposition and Prediction (SDaP)* approaches.

This article first motivates the need for addressing DyaC data through real-world application scenarios outlined in Sec 2. Then it provides a summary of the key ideas behind the SCOAL and PDLF approaches (Sec 3), and sets the context for a novel approach based on a generative (probabilistic) model of DyaC data, which is presented in Sec 4 in its simplest form. More advanced formulations and future work are suggested in the concluding section.

2 Illustrative Applications

In this subsection, we briefly describe three areas that can greatly benefit from SDaP. For all three examples, we show why DyaC based approaches are suitable and also highlight certain unresolved challenges that motivate further work.

Ecology. The analysis of population dynamics and their interaction with the environment is a central problem in the field of ecology. A typical data setup for this problem consists of population count data for different species across varying environmental conditions. The objective then is to predict the population counts for species of interest in certain locations/environments at current or future times. Typical datasets of this nature are highly heterogeneous with differing population patterns across different species, environments and seasons, and is also extremely sparse with unavailable counts for many species and environments. For example, the recently compiled and NSF funded *eBird Reference Dataset* [36] (avianknowledge.net, ebird.org) contains over 27 million observations of birds count for over 4000 species collected from more than 250,000 locations. Each location is annotated with over 50 covariates such as habitat, climatic conditions, elevation etc. Bird species are described by about 25 attributes such as preferred habitat, global population size, breeding information etc. The data is collected by human observers over different times (sampling events which are further associated with 15 covariates) adding a time dimension to the dataset as well, which makes the DyaC even more rich and complex.

Current approaches typically model each species separately using poisson regression over the independent environmental variables [11, 43], and thus flatten the data. Generalization ability is inadequate [11]. We note that the problem domain is inherently dyadic in nature, with the species and the environments forming the two modes

of variation, each with associated covariates. Considering time adds a third mode with strong seasonality properties. Current methods fail to leverage the dyadic or tensor nature of the data: each species is treated independently of others and their attributes are ignored. Learning multiple models using SDaP can greatly improve the performance by efficiently capturing the data heterogeneity as well as relations among sites and among species. Equally important, it can significantly enhance interpretability and actionablity as closely related subsets of species (in terms of the influence of environment on their count) and associated locations will automatically emerge from the model. This domain also imposes (soft) spatial constraints and partial orderings via factors such as geographical location, altitude etc. A hierarchy defined on the birds (ebird.org) adds a different type of soft constraint on a different mode! Such constraints can also be exploited to influence the decomposition process, for example via an efficient markov random field (MRF) based latent dyadic model.

Customer Product Preference Analysis. The problem of analyzing customer purchase or preference behavior involves multi-modal, inter-connected relational data and forms a suitable and broad application domain for SDaP. Consider the publicly available ERIM household panel dataset that has been widely studied in the marketing research community [27, 28, 44]. This dataset has purchase information over a period of 3 years and covers 6 product categories (ketchup, tuna, sugar, tissue, margarine and peanut butter) with a total of 121 products. Each household is annotated with demographic information including income, number of residents, male head employed, female head employed, etc. Products are described by attributes such as market share, price and advertising information. Details of each shopping visit of each household over the 3 year period are recorded, adding a third time dimension to the dataset.

SCOAL has been applied to ERIM for predicting the number of units of specified products purchased by households, given household and product covariate information. For this problem, SCOAL substantially improves accuracy over alternative predictive techniques [12, 14], including sophisticated Hierarchical Bayesian approaches on a flattened data representation, thereby pointing to the utility of the dyadic viewpoint. In addition, SCOAL also provides interpretable and actionable results by indicating what factors influence purchases in different household-product groups [13]. However, the current approach needs further extension to cater well to additional related information that is available, including attributes of the shops and of the city of residence. Moreover, the increasing popularity of customer interaction and feedback channels, including social networks and ratings sites, is bound to lead to additional acquired data that add new dimensions to the customer purchase behavior modeling problem, which also need to be leveraged.

Click-Through-Rate Prediction. A key goal of content providers and search engines such as Yahoo! and Google is to get as high a click-through-rate (CTR) as possible by serving users the content and ads they are most likely to click on. The massive scale of the ads targeting problem and its obvious business relevance has started attracting attention from the data mining community [3, 2]. A typical data setup for the problem is as follows: Ads are categorized into a hierarchical taxonomy. Each category in the taxonomy is a specific topic of interest, e.g., loans, travel, parenting and children. The categories are annotated by attributes such as descriptive keywords, historic CTR rates

and volume. Users are also described by features such as demographics, geographical location and metrics computed based on previous browsing behavior. For some (ad category, user) pairs, the target CTR value is known or easily estimated, and these form the training data. Given a user, the objective is then to select the categories to be served based on the highest predicted CTRs, among other criteria.

Once again we have DyaC data, with users and ad categories representing two sets of entities. Also, such data is very large (typically several hundred million users per week and several thousand categories), very sparse and noisy, with little activity in some low traffic categories. Moreover, the data is very heterogeneous, with widely varying patterns of user behavior across different user and category groups.

Initial results on applying SCOAL and PDLF approaches to an internal Yahoo! dataset showed substantial promise in terms of both accuracy and speed as compared to traditional predictive models. Yet, they are wanting in several important aspects: (a) they don't have an effective mechanism for exploiting the taxonomy available on the category mode, (b) user behavior is not static but even differs by the day, hence distinct (though related) models for each day, or for weekdays vs. weekends, are desired; and (c) the cold-start problem of determining what ads to serve to new users and predicting user propensity towards a new category, needs to be robustly addressed. Mechanisms for incorporating constraints among entities, for multitask learning and for cold-start can however be added to the generative approach presented in Section 4 to address these challenges. Finally, the ad targeting problem is essentially dynamic. Ad views and clicks are recorded and CTR values are updated in near real time. So SDaP needs to have a scalable, incremental version that is capable of effectively modeling streaming data.

Other application domains that can benefit from SDaP include (i) microarray data annotated with regulatory network information, gene/condition metadata, colocation of names in medical abstracts, etc., (ii) cross-language information retrieval, and (iii) scene analysis of a large number of images, each containing a variety of objects with geometric and co-locational constraints. It is also suitable for a large class of problems that can be represented as directed graphs where both nodes and edges have associated attributes. Email data (4 mode tensor with sender, recepient, time and content/topic; attributes of the people are also provided), and 3-mode web data with source, destination and anchor text, etc, fall in this category, and corresponding large datasets - Enron email and the substantial TREC WT10g Web Corpus - are already available.

3 Latent Factor Modeling of Dyadic Data with Covariates

In this section, we consider the two aforementioned SDaP approaches [12, 5] in a bit more detail.

Simultaneous Co-clustering and Learning (SCOAL). In complex classification and regression problem domains, a single predictive model is often inadequate to represent the inherent heterogeneity in the problem space. The traditional "divide and conquer" solution that partitions the input space *a priori* and then learns models in each "homogeneous" segment is inherently sub-optimal since the partitioning is done independent of the modeling. The key idea behind SCOAL is to partition the entities along

each mode, thus leading to blocks or co-clusters representing the cartesian-product of such partitions across different modes. If a mode has innate ordering, e.g. time, then the partitions need to be contiguous along that axis [14]. For example, a 3-D block in a user x movie x time tensor would be formed by subset of users, rating a subset of movies over a contiguous time period. For each block, a predictive model that relates the independent variables to the response variable in the co-cluster, is learnt. Note that *within a block, the responses themselves do not need to be similar, as distinct from the blocks formed in partitional co-clustering* [34, 7]. A key property of SCOAL is that the fitting of the "local" predictive models in each block is done simultaneously with block formation. The overall goal is to obtain a partitioning such that each co-cluster can be well characterized by a single predictive model.

Specifically, SCOAL aims at finding a co-cluster assignment and corresponding set of co-cluster specific models that minimize a global objective function, e.g. the prediction error summed over all the the known entries in the data matrix. For instance, with linear regression models, the objective function is a suitably regularized sum squared error. A simple iterative algorithm that alternately updates the co-cluster models and the row and column cluster assignments can be applied to obtain a local minimum of the objective function. A variety of regression models can be used for the predictive learning. For example, the data can be modeled by a collection of neural network models or regression models with the L_1 norm (Lasso [25]). The mathematics also carries through for all generalized linear models (GLMs), thus covering binary (classification) and count responses as well. It also generalizes to tensor data, and for any noise term belonging to the exponential family. Recent advances in the SCOAL approach include a dynamic programming solution to segment ordered modes, an incremental way to increase the number of models, active learning methods that exploit the multiple-model nature, and novel ways of determining the most reliable predictions that also exploit the presence of multiple local models [16, 15, 14].

Predictive Discrete Latent Factor Modeling (PDLF). While SCOAL learns multiple, independent local models, the Predictive Discrete Latent Factor (PDLF) model simultaneously incorporates the effect of the covariates via a *global* model, as well as any local structure that may be present in the data through a block (co-cluster) specific constant. Similar to SCOAL, the dyadic data matrix is partitioned into a grid of co-clusters, each one representing a local region. The mean of the response variable is modeled as a sum of a function of the covariates (representing global structure) and a co-cluster specific constant (representing local structure). The co-cluster specific constant can also be thought of as part of the noise model, teased out of the global model residues. The authors also formulate scalable, generalized EM based algorithms to estimate the parameters of hard or soft versions of the proposed model.

SCOAL and PDLF show benefit in complementary situations; SCOAL works well in domains with very high heterogeneity where sufficient data is available to learn multiple models, while PDLF shows better value in situations where the training data is limited and several outliers are present. Also note that the standard approach in the statistics community for such problems would be to develop a semi-parametric hierarchical model, which at first blush is structurally similar to PDLF [21]. However, as discussed in great detail by [5, 2], the assumption made in PDLF that block membership

can be completely specified in terms of row and column memberships, is a key feature that makes it much more scalable. The corresponding factorization of the joint space also leads to vastly simpler and efficient inference. Similarly, the smoothing effect of a soft partitioning within the block-structure is found to be more effective than the widely used stastical approach of using a hierarchical random effects model.

Agarwal and Chen [2] recently generalized the PDLF as well as the Probabilistic Matrix Factorization [41] approach to regression based latent factor models (RLFM), which provides a unified framework to smoothly handle both cold-start and warm-start scenarios. RLFM is a two stage hierarchical model with regression based priors used to regularize the latent factors. A scalable Monte Carlo EM approach or an Iterated Conditional Mode technique can be used for model fitting. RLFM has shown substantially better results than competing techniques in a challenging content recommendation application that arises in the context of Yahoo! Front Page.

4 Latent Dirichlet Attribute Aware Bayesian Affinity Estimation (LD-BAE)

Several Bayesian formulations have been proposed in the context of affinity estimation problems. Mixed Membership stochastic Blockmodels (MMBs) [20] is one such method that utilizes affinity values to group the two entity sets via a soft co-clustering. A weighted average of the pertinent co-cluster means is then used to estimate missing affinities. The model is shown to be quite efficient in scaling to large datasets, however it fails to utilize any available side information. Other efforts include fully Bayesian frameworks for PMF([31], [42]) with differing inference techniques - ranging from Variational approximations to sampling based MCMC methods. However, the stress again is only on utilizing the available affinities. Recently, Bayesian models based on topic models for document clustering [18] have been utilized for estimating affinities between users and News articles [4]. Two sided generalizations of topic models have also been utilized for co-clustering and matrix approximation problems ([46], [45]) without taking into account auxiliary sources of information.

This section introduces a Side Information Aware Bayesian Affinity Estimation approach that is related to Latent Dirichlet Allocation [18], as explained shortly, and is hence called the Latent Dirichlet Attribute Aware Bayesian Affinity Estimation (LD-BAE) model. For simplicity, we consider the available side information to be only a set of attributes (covariates) associated with each entity. Additional sources of side information such as network structures over entities, evolution over time or knowledge that known ratings are not given at random, can be accommodated by extending this basic framework [47].

Notation. Before describing the LD-BAE framework, a quick word on the notation. We use capital script letters for sets, $\{\cdot\}$ denote a collection of variables for unnamed sets and $\dagger$ represents transpose of a matrix. Let $\mathcal{E}_1 = \{e_{1m}\}, [m]_1^M$ and $\mathcal{E}_2 = \{e_{2n}\}, [n]_1^N$ represent the sets of entities between which affinities need to be estimated. $\mathcal{Y} = \{y_{mn}\}$ is a set of $M \times N$ affinities between pairs of entities of the form $(e_{1m}, e_{2n}), e_{1m} \in \mathcal{E}_1$ and $e_{2n} \in \mathcal{E}_2$. The subset $\mathcal{Y}_{\text{obs}} \subseteq \mathcal{Y}$ is a set of observed affinities while $\mathcal{Y}_{\text{unobs}} = \mathcal{Y} \backslash \mathcal{Y}_{obs}$ denotes a set of missing affinities. A weight w_{mn} is associated with each affinity y_{mn}

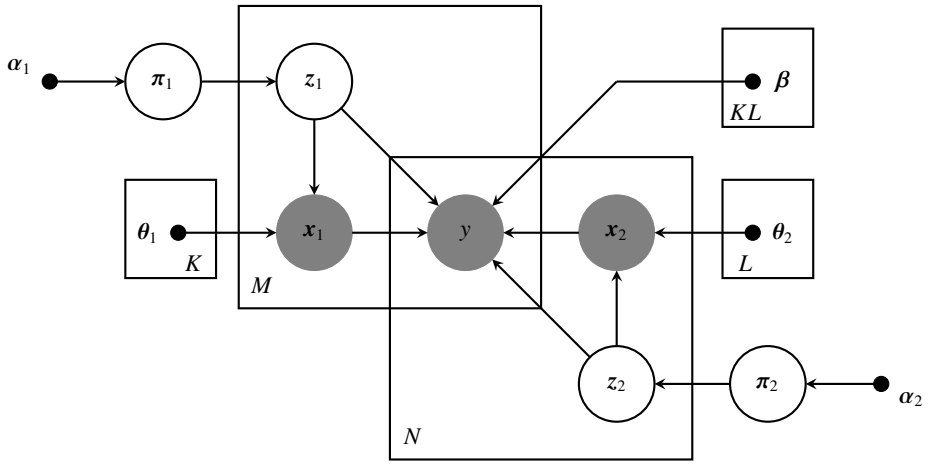


Fig. 2. Graphical model for Latent Dirichlet Attribute Aware Bayesian Affinity Estimation

(affinity between a pair of entities e_{1m} and e_{2n}) such that $w_{mn} = 1$ if $y_{mn} \in \mathcal{Y}_{\text{obs}}$ and $w_{mn} = 0$ if $y_{mn} \in \mathcal{Y}_{\text{unobs}}$. The set of all $M \times N$ weights is denoted by $\mathcal{W}$. The set of entity attributes associated with $\mathcal{E}_1$ and $\mathcal{E}_2$ are respectively described by the sets $\mathcal{X}_1 = \{\boldsymbol{x}_{1m}\}$ and $\mathcal{X}_2 = \{\boldsymbol{x}_{2n}\}$. The notation $\boldsymbol{x}_{mn} = [\boldsymbol{x}_{1m}^\dagger \boldsymbol{x}_{2n}^\dagger]^\dagger$ is used to denote the attributes associated with the entity pair (e_{1m}, e_{2n}).

Figure 2 shows the graphical model for LD-BAE - a mixture model of KL clusters obtained as a cross-product of clustering the two sets of entities into K and L clusters respectively. First, the mixing coefficients $\boldsymbol{\pi}_1(\boldsymbol{\pi}_2)$ are sampled (only once) from the corresponding Dirichlet distributions parameterized by $\text{Dir}(\boldsymbol{\alpha}_1)$ and $\text{Dir}(\boldsymbol{\alpha}_2)$ for entity set $\mathcal{E}_1$ and $(\mathcal{E}_2)$ respectively. Hence, all the entities in a particular set share the same mixing coefficients, thereby inducing statistical dependency between them. Then each entity $e_{1m} \in \mathcal{E}_1$ is assigned to one of K clusters by sampling cluster assignments $z_{1m} \in \{1, \ldots, K\}$ from a discrete distribution $\text{Disc}(\pi_{1m})$. Similarly, the entities $e_{2n} \in \mathcal{E}_2$, are clustered into L clusters by sampling cluster assignments $z_{2n} \in \{1, \ldots, L\}$ from a discrete distribution $\text{Disc}(\boldsymbol{\pi}_{2n})$. $\mathcal{Z}_1$ and $\mathcal{Z}_2$ respectively denote the sets of cluster assignments for the two entity sets. It is easy to see that by sharing mixing coefficients across entities in a set, the model is an attribute sensitive two sided generalization of the Latent Dirichlet Allocation (LDA) [18] model.

The attributes $\boldsymbol{x}_{1m}$ associated with the entity e_{1m} are drawn from one of K possible exponential family distributions of the form $p_{\psi_1}(\boldsymbol{x}_{1m}|\boldsymbol{\theta}_{1z_{1m}})$[1], such that the parameter $\boldsymbol{\theta}_{1z_{1m}}$ of the family, is chosen according the entity cluster assignment z_{1m}. Likewise, attributes $\boldsymbol{x}_{2n}$ for an entity e_{2n} are generated from one of L possible exponential family distributions $p_{\psi_2}(\boldsymbol{x}_{2n}|\boldsymbol{\theta}_{2z_{2n}})$. The cluster assignments z_{1m} and z_{2n} over the two entities together determine a co-cluster (z_{1m}, z_{2n}), which then selects an exponential family dis-

[1] We use the canonical form of exponential family distributions: $p_\psi(x|\theta) = p_0(x)\exp(\langle x, \theta\rangle - \psi(\theta))$

tribution, $p_{\psi_y}(y_{mn}|\boldsymbol{\beta}^{\dagger}_{z_{1m}z_{2n}}\boldsymbol{x}_{mn})$ (out of KL such distributions), to generate the affinity y_{mn} associated with the entity pair (e_{1m}, e_{2n}). The parameters $\boldsymbol{\beta}_{z_{1m}z_{2n}}$ of the distribution are specific to the co-cluster (z_{1m}, z_{2n}). In summary, the generative process for the attributes and the affinities between each pair of entities is as follows:

1. Sample mixing coefficients: $\boldsymbol{\pi}_1 \sim \text{Dir}(\boldsymbol{\alpha}_1)$
2. Sample mixing coefficients: $\boldsymbol{\pi}_2 \sim \text{Dir}(\boldsymbol{\alpha}_2)$
3. For each entity $e_{1m} \in \mathcal{E}_1$
   (a) Sample cluster assignment: $z_{1m} \sim \text{Disc}(\boldsymbol{\pi}_1)$
   (b) Sample entity attributes: $\boldsymbol{x}_{1m} \sim p_{\psi_1}(\boldsymbol{x}_{1m}|\boldsymbol{\theta}_{1z_{1m}})$
4. For each entity $e_{2n} \in \mathcal{E}_2$
   (a) Sample cluster assignment: $z_{2n} \sim \text{Disc}(\boldsymbol{\pi}_2)$
   (b) Sample entity attributes: $\boldsymbol{x}_{2n} \sim p_{\psi_2}(\boldsymbol{x}_{2n}|\boldsymbol{\theta}_{2z_{2n}})$
5. For each pair of entities (e_{1m}, e_{2n}) such that $e_{1m} \in \mathcal{E}_1, e_{2n} \in \mathcal{E}_2$
   (a) Sample affinity: $y_{mn} \sim p_{\psi_y}(y_{mn}|\boldsymbol{\beta}^{\dagger}_{z_{1m}z_{2n}}\boldsymbol{x}_{mn})$

The overall joint distribution over all observable and latent variables is then given by:

$$p(\mathcal{Y}, \mathcal{X}_1, \mathcal{X}_2, \mathcal{Z}_1, \mathcal{Z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2|\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2, \boldsymbol{\beta}) = p(\boldsymbol{\pi}_1|\boldsymbol{\alpha}_1)p(\boldsymbol{\pi}_2|\boldsymbol{\alpha}_2)$$
$$\left(\prod_m p(z_{1m}|\boldsymbol{\pi}_1)p_{\psi_1}(\boldsymbol{x}_{1m}|\boldsymbol{\theta}_{1z_{1m}})\right)\left(\prod_n p(z_{2n}|\boldsymbol{\pi}_2)p_{\psi_2}(\boldsymbol{x}_{2n}|\boldsymbol{\theta}_{2z_{2n}})\right)\left(\prod_{m,n} p_{\psi_y}(y_{mn}|\boldsymbol{\beta}^{\dagger}_{z_{1m}z_{2n}}\boldsymbol{x}_{mn})\right)$$

Marginalizing out the latent variables, the probability of observing the known affinities and the attributes is:

$$p(\mathcal{Y}_{\text{obs}}, \mathcal{X}_1, \mathcal{X}_2|\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2, \boldsymbol{\beta}) = \int_{\mathcal{Y}_{\text{unobs}}} \int_{\boldsymbol{\pi}_1} \int_{\boldsymbol{\pi}_2} (p(\boldsymbol{\pi}_1|\boldsymbol{\alpha}_1))\,(p(\boldsymbol{\pi}_2|\boldsymbol{\alpha}_2))$$
$$\sum_{\mathcal{Z}_1}\sum_{\mathcal{Z}_2}\left(\prod_m p(z_{1m}|\boldsymbol{\pi}_1)p_{\psi_1}(\boldsymbol{x}_{1m}|\boldsymbol{\theta}_{1z_{1m}})\right)\left(\prod_n p(z_{2n}|\boldsymbol{\pi}_2)p_{\psi_2}(\boldsymbol{x}_{2n}|\boldsymbol{\theta}_{2z_{2n}})\right)$$
$$\left(\prod_{m,n} p_{\psi_y}(y_{mn}|\boldsymbol{\beta}^{\dagger}_{z_{1m}z_{2n}}\boldsymbol{x}_{mn})\right) d\mathcal{Y}_{\text{unobs}} d\boldsymbol{\pi}_1 d\boldsymbol{\pi}_2$$

Note that even marginalization of only the mixing coefficients $\boldsymbol{\pi}_1$ and $\boldsymbol{\pi}_2$ induces dependencies between the clustering assignments $\mathcal{Z}_1$ and $\mathcal{Z}_2$.

Inference and Learning As a result of the induced dependencies, direct maximization of the observed log-likelihood is intractable using an EM algorithm. One instead needs to resort to Gibbs sampling or related approaches, or use variational methods. In this subsection we take the latter route by constructing tractable lower bounds using a fully factorized mean field approximation to the true posterior distribution over the latent variables. The optimal factorized distribution over the latent variables $(\mathcal{Y}_{\text{unobs}}, \mathcal{Z}_1, \mathcal{Z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2)$ that corresponds to the tightest lower bound on the observed likelihood is then given by:

$$q^*(\mathcal{Y}_{\text{unobs}}, \mathcal{Z}_1, \mathcal{Z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2) \tag{1}$$
$$= q^*(\boldsymbol{\pi}_1|\gamma_1) q^*(\boldsymbol{\pi}_2|\gamma_2) \left(\prod_{\substack{m,n \\ y_{mn} \in \mathcal{Y}_{\text{unobs}}}} q^*(y_{mn}|\phi_{mn}) \right) \left(\prod_m q^*(z_{1m}|r_{1m}) \right) \left(\prod_n q^*(z_{2n}|r_{2n}) \right)$$

Note that, since the mixing coefficients are shared across entities from the same set, we only have two variational factors corresponding to the mixing coefficients $\boldsymbol{\pi}_1$ and $\boldsymbol{\pi}_2$. $q_{\psi_y}(y_{mn}|\phi_{mn})$ is an exponential family distribution with natural parameter ϕ_{mn}, $q(\boldsymbol{\pi}_1|\gamma_1)$ and $q(\boldsymbol{\pi}_2|\gamma_2)$ are K and L dimensional Dirichlet distributions with parameters γ_1 and γ_2 respectively while the cluster assignments z_{1m} and z_{2n} follow discrete distributions over K and L clusters with parameters r_{1m} and r_{2n} respectively. The variational parameters $(\gamma_1, \gamma_2, \phi_{mn}, r_{1m}, r_{2n})$ are then given by (see Appendix for derivation, and [47] for further detail on the variational derivation):

$$\phi_{mn} = \sum_{k=1}^{K} \sum_{l=1}^{L} r_{1mk} r_{2nl} \left(\boldsymbol{\beta}_{kl}^{\dagger} \boldsymbol{x}_{mn} \right) \tag{2}$$

$$\gamma_{1k} = \sum_{m=1}^{M} r_{1mk} + \alpha_{1k} \tag{3}$$

$$\gamma_{2l} = \sum_{n=1}^{N} r_{2nl} + \alpha_{2l} \tag{4}$$

$$\log r_{1mk} \propto \log p_{\psi_1}(\boldsymbol{x}_{1m}|\boldsymbol{\theta}_{1k}) + \Psi(\gamma_{1k}) +$$
$$\sum_{n=1}^{N} \sum_{l=1}^{L} r_{2nl} \left(w_{mn} \log p_{\psi_y}(y_{mn}|\boldsymbol{\beta}_{kl}^{\dagger} \boldsymbol{x}_{mn}) + (1 - w_{mn}) \mathbb{E}_q \left[\log p_{\psi_y}(y_{mn}|\boldsymbol{\beta}_{kl}^{\dagger} \boldsymbol{x}_{mn}) \right] \right) \tag{5}$$

$$\log r_{2nl} \propto \log p_{\psi_2}(\boldsymbol{x}_{2n}|\boldsymbol{\theta}_{2l}) + \Psi(\gamma_{2l}) +$$
$$\sum_{m=1}^{M} \sum_{k=1}^{K} r_{1mk} \left(w_{mn} \log p_{\psi_y}(y_{mn}|\boldsymbol{\beta}_{kl}^{\dagger} \boldsymbol{x}_{mn}) + (1 - w_{mn}) \mathbb{E}_q \left[\log p_{\psi_y}(y_{mn}|\boldsymbol{\beta}_{kl}^{\dagger} \boldsymbol{x}_{mn}) \right] \right) \tag{6}$$

The optimal lower bound on the observed log-likelihood with respect to the variational distribution in (1) is then given by:

$$\log p(\mathcal{Y}_{\text{obs}}, \mathcal{X}_1, \mathcal{X}_2|\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2, \boldsymbol{\beta})$$
$$\geq H[q^*] + \mathbb{E}_{q^*}[\log p(\mathcal{Y}_{\text{obs}}, \mathcal{X}_1, \mathcal{X}_2, \mathcal{Z}_1, \mathcal{Z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2|\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2, \boldsymbol{\beta})]$$

This bound can be maximized with respect to the free model parameters to get their improved estimates. Taking partial derivatives of the bound with respect to the model parameters and setting them to zero, we obtain the following updates (see Appendix for details):

Algorithm 1 Learn LD-BAE

Input: $\mathcal{Y}_{\text{obs}}, \mathcal{X}_1, \mathcal{X}_2, K, L$
Output: $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2, \boldsymbol{\beta}$
$[m]_1^M, [n]_1^N, [k]_1^K, [l]_1^L$

Step 0: Initialize $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2, \boldsymbol{\beta}$
Until Convergence
Step 1: E-Step
Step 1a: Initialize r_{1mk}, r_{2nl}
Until Convergence
Step 1b: Update ϕ_{mn} using equation (2)
Step 1c: Update γ_{1k} using equation (3)
Step 1d: Update γ_{2l} using equation (4)
Step 1e: Update r_{1mk} using equation (5)
Step 1f: Update r_{2nl} using equation (6)
Step 2: M-Step
Step 2a: Update $\boldsymbol{\theta}_{1k}$ using equation (7)
Step 2b: Update $\boldsymbol{\theta}_{2l}$ using equation (8)
Step 2c: Update $\boldsymbol{\beta}_{kl}$ using equation (9)
Step 2d: Update $\boldsymbol{\alpha}_1$ using equation (10)
Step 2e: Update $\boldsymbol{\alpha}_2$ using equation (11)

$$\boldsymbol{\theta}_{1k} = \nabla\psi_1^{-1}\left(\frac{\sum_{m=1}^{M} r_{1mk}\boldsymbol{x}_{1m}}{\sum_{m=1}^{M} r_{1mk}}\right) \tag{7}$$

$$\boldsymbol{\theta}_{2l} = \nabla\psi_2^{-1}\left(\frac{\sum_{n=1}^{N} r_{2nl}\boldsymbol{x}_{2n}}{\sum_{n=1}^{N} r_{2nl}}\right) \tag{8}$$

$$\boldsymbol{\beta}_{kl} = \underset{\beta\in\mathbb{R}^D}{\arg\max} \sum_{m=1}^{M}\sum_{n=1}^{N} r_{1mk}r_{2nl}\left[\left\langle (w_{mn}y_{mn} + (1-w_{mn})\nabla\psi_{\mathcal{Y}}(\phi_{mn})), \boldsymbol{\beta}^\dagger \boldsymbol{x}_{mn}\right\rangle - \psi_{\mathcal{Y}}\left(\boldsymbol{\beta}^\dagger \boldsymbol{x}_{mn}\right)\right] \tag{9}$$

$$\boldsymbol{\alpha}_1 = \underset{\boldsymbol{\alpha}_1\in\mathbb{R}_{++}^K}{\arg\max}\left(\log\frac{\Gamma(\sum_{k=1}^K \alpha_{1k})}{\prod_{k=1}^K \Gamma(\alpha_{1k})} + \sum_{k=1}^{K}\left(\alpha_{1k} + \sum_{m=1}^{M} r_{1mk} - 1\right)\left(\Psi(\gamma_{1k}) - \Psi\left(\sum_{k'=1}^{K}\gamma_{1k'}\right)\right)\right) \tag{10}$$

$$\boldsymbol{\alpha}_2 = \underset{\boldsymbol{\alpha}_2\in\mathbb{R}_{++}^L}{\arg\max}\left(\log\frac{\Gamma(\sum_{l=1}^L \alpha_{2l})}{\prod_{l=1}^L \Gamma(\alpha_{2l})} + \sum_{l=1}^{L}\left(\alpha_{2l} + \sum_{n=1}^{N} r_{2nl} - 1\right)\left(\Psi(\gamma_{2l}) - \Psi\left(\sum_{l'=1}^{L}\gamma_{2l'}\right)\right)\right) \tag{11}$$

The updates for the parameters of the Dirichlet distributions $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$, can be efficiently performed using the Newton-Raphson's method. An EM algorithm for learning the model parameters of LD-AA-BAE is given in algorithm 1.

5 Concluding Remarks and Future Work

The side information aware Bayesian affinity estimation approach introduced in this article is a promising framework that efficiently incorporates attribute information within dyadic data. The approach can be readily generalized to where there are more than two modes, or sets of interacting items. Moreover, the graphical model can be further elaborated on to accommodate other types of side information including temporal information, and/or neighborhood structures. The use of exponential family distributions for modeling entity attributes as well as the affinity relationships renders great flexibility for modeling diverse data types in numerous domains. The approach can also be extended to non-parametric models by replacing the Dirichlet distribution priors with the corresponding process prior, which is useful when the desired number of clusters is not known. But clearly there is much work to be done on this line of models, in terms of both algorithmic development and applications.

A common feature of affinity datasets is sparsity - often only a very small percentage of the affinities are known. In the derivation provided in this paper, one carries around the unobserved affinities as well, which adds to computational demands and does not benefit from sparsity. In several settings however, one can simply ignore these values and just build a model based on the observed values, since conditioning on the unobserved affinity values does not effect any of the posterior distributions. This observation can be exploited to develop more efficient versions of LD-BAE.

Acknowlegments: This research was supported by NSF grants IIS-0713142 and IIS-1016614, and by NHARP. We thank Meghana Deodhar for her collaboration on SCOAL.

References

1. Abernethy, J., Bach, F., Evgeniou, T., Vert, J.P.: A new approach to collaborative filtering: Operator estimation with spectral regularization. In: The Journal of Machine Learning Research. vol. 10, pp. 803–826 (June 2009)
2. Agarwal, D., Chen, B.: Regression-based latent factor models. In: KDD '09. pp. 19–28 (2009)
3. Agarwal, D., Chen, B., Elango, P.: Spatio-temporal models for estimating click-through rate. In: WWW '09: Proceedings of the 18th international conference on World wide web. pp. 21–30 (2009)
4. Agarwal, D., Chen., B.: flda: matrix factorization through latent dirichlet allocation. In: Proc. ACM international conference on Web search and data mining, 2010. pp. 91–100 (2010)
5. Agarwal, D., Merugu, S.: Predictive discrete latent factor models for large scale dyadic data. In: KDD '07. pp. 26–35 (2007)
6. A.P. Dempster, N.L., Rubin, D.: Maximum likelihood from incomplete data via the em algorithm. J. Royal Statistical Society, Series B(Methodological) 39(1), 1–38 (1977)
7. Banerjee, A., Merugu, S., Dhillon, I., Ghosh, J.: Clustering with Bregman divergences. Jl. Machine Learning Research (JMLR) 6, 1705–1749 (October 2005)
8. Banerjee, A., Basu, S., Merugu, S.: Multi-way clustering on relation graphs. In: SDM (2007)
9. Basilico, J., Hofmann, T.: Unifying collaborative and content-based filtering. In: ICML (2004)
10. Bertsekas, D.: Nonlinear Programming. Athena Scientific (1999)

11. Chamberlain, D.E., Gough, S., Vickery, J.A., Firbank, L.G., Petit, S., Pywell, R., Bradbury, R.B.: Rule-based predictive models are not cost-effective alternatives to bird monitoring on farmland. Agriculture, Ecosystems & Environment 101(1), 1 – 8 (2004)
12. Deodhar, M., Ghosh, J.: A framework for simultaneous co-clustering and learning from complex data. In: KDD '07. pp. 250–259 (2007)
13. Deodhar, M., Ghosh, J.: Simultaneous co-clustering and modeling of market data. In: Workshop for Data Mining in Marketing, Industrial Conf. on Data Mining '07. pp. 73–82 (2007)
14. Deodhar, M., Ghosh, J.: Simultaneous co-segmentation and predictive modeling for large, temporal marketing data. In: Data Mining for Marketing Workshop, ICDM/08 (2008)
15. Deodhar, M., Ghosh, J.: Mining for most certain predictions from dyadic data. In: Proc. 15th ACM SIGKDD Conf. on Knowledge Discovery and Data Mining (KDD09) (2009)
16. Deodhar, M., Ghosh, J., Tsar-Tsansky, M.: Active learning for recommender systems with multiple localized models. In: Proc. Fifth Symposium on Statistical Challenges in Electronic Commerce Research (SCECR09) (2009)
17. Dietterich, T.G., Domingos, P., Getoor, L., Muggleton, S., Tadepalli, P.: Structured machine learning: the next ten years. Machine Learning 73(1), 3–23 (2008)
18. D.M. Blei, A.Y.N., Jordan., M.I.: Latent dirichlet allocation. JMLR 3, 993–1022 (2003)
19. Dzeroski, S.: Multi-relational data mining: an introduction. SIGKDD Explorations 5(1), 1–16 (2003)
20. E. Airoldi, D. M. Blei, S.E.F., Xing., E.: Mixed membership stochastic blockmodels. JMLR 9, 1981–2014 (2008)
21. Gelman, A., Hill, J.: Data Analysis Using Regression and Multilevel/Hierarchical Models. Cambridge University Press (2007)
22. George, T., Merugu, S.: A scalable collaborative filtering framework based on co-clustering. Proceedings of the Fifth IEEE International Conference on Data Mining pp. 625 – 628 (2005)
23. Getoor, L., Friedman, N., Koller, D., Taskar, B.: Learning probabilistic models of relational structure. In: Proc. 18th International Conf. on Machine Learning. pp. 170–177. Morgan Kaufmann, San Francisco, CA (2001), citeseer.ist.psu.edu/article/getoor01learning.html
24. Grover, R., Srinivasan, V.: A simultaneous approach to market segmentation and market structuring. Journal of Marketing Research pp. 139 – 153 (1987)
25. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. Springer, (2nd Ed) (2009)
26. Herlocker, J., Konstan, J., Borchers, A., Riedl, J.: An algorithmic framework for performing collaborative filtering. In: SIGIR '99: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. ACM 1999 230–237. Berkeley, CA, USA (August 15-19 1999)
27. Kim, B., Rossi, P.: Purchase frequency, sample selection, and price sensitivity: The heavy-user bias. Marketing Letters pp. 57 – 67 (1994)
28. Kim, B., Sullivan, M.: The effect of parent brand experience on line extension trial and repeat purchase. Marketing Letters pp. 181 – 193 (1998)
29. Kolda, T.: Tensor decompositions and data mining. In: Tutorial at ICDM (2007)
30. Kolda, T.G., Sun, J.: Scalable tensor decompositions for multi-aspect data mining. In: ICDM. pp. 363–372 (2008)
31. Lim, Y., Teh., Y.: Variational bayesian approach to movie rating prediction. In: Proc. KDD Cup and Workshop (2007)
32. Lokmic, L., Smith, K.A.: Cash flow forecasting using supervised and unsupervised neural networks. IJCNN 06, 6343 (2000)
33. Lu, Z., Agarwal, D., Dhillon, I.: A spatio-temporal approach to collaborative filtering. In: RecSys'09 (2009)
34. Madeira, S.C., Oliveira, A.L.: Biclustering algorithms for biological data analysis: A survey. IEEE/ACM Trans. Comput. Biology Bioinform. 1(1), 24–45 (2004)

35. Moe, W., Fader, P.: Modeling hedonic portfolio products: A joint segmentation analysis of music compact disc sales. Journal of Marketing Research pp. 376 – 385 (2001)
36. Munson, M.A., et al.: The ebird reference dataset. Tech. Report, Cornell Lab of Ornithology and National Audubon Society (June 2009)
37. Murray-Smith, R., Johansen, T.A.: Multiple Model Approaches to Modelling and Control. Taylor and Francis, UK (1997)
38. Nowicki, K., Snijders, T.A.B.: Estimation and prediction for stochastic blockstructures. Journal of the American Statistical Association 96(455), 1077–1087 (September 2001), http://www.ingentaconnect.com/content/asa/jasa/2001/00000096/00000455/art00025
39. Oh, K., Han, I.: An intelligent clustering forecasting system based on change-point detection and artificial neural networks: Application to financial economics. In: HICSS-34. vol. 3, p. 3011 (2001)
40. Reutterer, T.: Competitive market structure and segmentation analysis with self-organizing feature maps. Proceedings of the 27th EMAC Conference pp. 85 – 115 (1998)
41. Salakhutdinov, R., Mnih, A.: Probabilistic matrix factorization. In: NIPS '07 (2007)
42. Salakhutdinov, R., Mnih, A.: Bayesian probabilistic matrix factorization using markov chain monte carlo. In: Proc. ICML, 2008. pp. 880–887 (2008)
43. Sanderson, F.J., Kloch, A., Sachanowicz, K., Donald, P.F.: Predicting the effects of agricultural change on farmland bird populations in poland. Agriculture, Ecosystems & Environment 129(1-3), 37 – 42 (2009)
44. Seetharaman, P., Ainslie, A., Chintagunta, P.: Investigating household state dependence effects across categories. Journal of Marketing Research pp. 488 – 500 (1999)
45. Shan, H., Banerjee., A.: Residual bayesian co-clustering and matrix approximation. In: Proc. SDM 2010. pp. 223–234 (2010)
46. Shan, H., Banerjee, A.: Bayesian co-clustering. In: ICDM. pp. 530–539 (2008)
47. Sharma, A., Ghosh., J.: Side information aware bayesian affinity estimation. Technical Report TR-11, Department of ECE, UT Austin (2010)
48. TakÅcs, G., PilÅszy, I., NÅÈmeth, B., Tikk, D.: Investigation of various matrix factorization methods for large recommender systems. In: 2nd KDD-Netflix workshop (2008)
49. Vasilescu, M.A.O., Terzopoulos, D.: Multilinear analysis of image ensembles: Tensorfaces. In: Proc. ECCV'02 (2002)
50. Wainwright, M.J., Jordan, M.I.: Graphical models, exponential families, and variational inference. Foundations and Trends in Machine Learning 1(1-2), 1–305 (2008)

6 APPENDIX A: Variational Inference using Mean Field Approximation (MFA)

A maximum likelihood approach to parameter estimation generally involves maximization of the observed log-likelihood $\log p(\mathcal{X}|\boldsymbol{\Theta})$ with respect to the free model parameters, i.e.,

$$\boldsymbol{\Theta}^*_{ML} = \arg\max_{\boldsymbol{\Theta}} \log p(\mathcal{X}|\boldsymbol{\Theta}) \tag{A1}$$

$$= \arg\max_{\boldsymbol{\Theta}} \log \int_{\mathcal{Z}} p(\mathcal{X}, \mathcal{Z}|\boldsymbol{\Theta}) d\mathcal{Z} \tag{A2}$$

where $\mathcal{X}$ and $\mathcal{Z}$ are sets of observed and hidden variables respectively. In the presence of hidden variables, the maximum likelihood estimate is often done using the Expectation-Maximization (EM) algorithm [6]. The following lemma forms the basis of the EM algorithm [50].

Lemma 1. *Let $\mathcal{X}$ denote a set of all the observed variables and $\mathcal{Z}$ a set of the hidden variables in a Bayesian network. Then, the observed log-likelihood can be lower bounded as follows*

$$\log p(\mathcal{X}, \mathcal{Z}|\boldsymbol{\Theta}) \geq \mathcal{F}(Q, \boldsymbol{\Theta})$$

where

$$\mathcal{F}(Q, \boldsymbol{\Theta}) = -\int_{\mathcal{Z}} Q(\mathcal{Z}) \log Q(\mathcal{Z}) d\mathcal{Z} + \int_{\mathcal{Z}} Q(\mathcal{Z}) \log p(\mathcal{X}, \mathcal{Z}|\boldsymbol{\Theta}) d\mathcal{Z} \tag{A3}$$

for some distribution Q and the free model parameters $\boldsymbol{\Theta}$.

Proof. The proof follows from the Jensen's inequality and the concavity of the log function.

$$\begin{aligned}
\log p(\mathcal{X}|\boldsymbol{\Theta}) &= \log \int_{\mathcal{Z}} \frac{Q(\mathcal{Z})}{Q(\mathcal{Z})} p(\mathcal{X}, \mathcal{Z}|\boldsymbol{\Theta}) d\mathcal{Z} \\
&\geq \int_{\mathcal{Z}} Q(\mathcal{Z}) \log \frac{p(\mathcal{X}, \mathcal{Z}|\boldsymbol{\Theta})}{Q(\mathcal{Z})} d\mathcal{Z} \\
&= -\int_{\mathcal{Z}} Q(\mathcal{Z}) \log Q(\mathcal{Z}) d\mathcal{Z} + \int_{\mathcal{Z}} Q(\mathcal{Z}) \log p(\mathcal{X}, \mathcal{Z}|\boldsymbol{\Theta}) d\mathcal{Z} \\
&= \mathcal{F}(Q, \boldsymbol{\Theta})
\end{aligned}$$

Starting from an initial estimate of the parameters, $\boldsymbol{\Theta}_0$, the EM algorithm alternates between maximizing the lower bound $\mathcal{F}$ with respect to Q (E-step) and $\boldsymbol{\Theta}$ (M-step), respectively, holding the other fixed. The following lemma shows that maximization the lower bound with respect to the distribution Q in the E-step makes the bound exact, so that the M-step is guranteed to increase the observed log-likelihood with respect to the parameters.

Lemma 2. *Let $\mathcal{F}(Q, \boldsymbol{\Theta})$ denote a lower bound on the observed log-likelihood of the form in* (A3)*, then*

$$Q^* = p(\mathcal{Z}|\mathcal{X}, \boldsymbol{\Theta}) = \arg\max_{Q} \mathcal{F}(Q, \boldsymbol{\Theta})$$

and $\mathcal{F}(Q^*, \boldsymbol{\Theta}) = \log p(\mathcal{X}|\boldsymbol{\Theta})$.

Proof. The lower bound on the observed log-likelihood is

$$\begin{aligned}
\mathcal{F}(Q, \boldsymbol{\Theta}) &= -\int_{\mathcal{Z}} Q(\mathcal{Z}) \log Q(\mathcal{Z}) d\mathcal{Z} + \int_{\mathcal{Z}} Q(\mathcal{Z}) \log p(\mathcal{X}, \mathcal{Z}|\boldsymbol{\Theta}) d\mathcal{Z} \\
&= -\int_{\mathcal{Z}} Q(\mathcal{Z}) \log \frac{Q(\mathcal{Z})}{p(\mathcal{Z}|\mathcal{X}, \boldsymbol{\Theta})} d\mathcal{Z} + \int_{\mathcal{Z}} Q(\mathcal{Z}) \log \frac{p(\mathcal{X}, \mathcal{Z}|\boldsymbol{\Theta})}{p(\mathcal{Z}|\mathcal{X}, \boldsymbol{\Theta})} d\mathcal{Z} \\
&= \log p(\mathcal{X}|\boldsymbol{\Theta}) - \mathrm{KL}(Q \parallel p(\mathcal{Z}|\mathcal{X}, \boldsymbol{\Theta}))
\end{aligned}$$

Maximum is attained when the KL-divergence $\mathrm{KL}(Q \parallel p(\mathcal{Z}|\mathcal{X}, \boldsymbol{\Theta}))$ is zero, which is uniquely achieved for $Q^* = p(\mathcal{Z}|\mathcal{X}, \boldsymbol{\Theta})$ at which point the bound becomes an equality for $\log p(\mathcal{X}|\boldsymbol{\Theta})$.

However, in many cases, computation of the true posterior distribution, $p(\mathcal{Z}|\mathcal{X}, \boldsymbol{\Theta})$ is intractable. To overcome this problem, the distribution Q is restricted to a certain family of distributions. The optimal distribution within this restricted class is then obtained by minimizing the KL-divergence to the true posterior distribution. The approximating distribution is known as a *variational distribution* [50].

There are a number of ways in which the family of possible distributions can be restricted. One way of restricting the approximating distributions is to use a parameteric distribution $Q(\mathcal{Z}|\boldsymbol{\Phi})$ determined by a set of parameters $\boldsymbol{\Phi}$, known as *variational parameters*. In the E-step, the lower bound then becomes a function of variational parameters, and standard non-linear optimization methods can be employed to obtain the optimal values of these parameters. Yet another way to restrict the family of approximationg distributions is to assume a certain conditional independence structure over the hidden variables $\mathcal{Z}$. For example, one can assume a family of fully factorized distributions of the following form

$$Q = \prod_i q_i(z_i) \tag{A4}$$

This fully factorized assumption is often known as a *mean field approximation* in statistical mechanics. The following lemma derieves the expression for optimal variational distribution subject to a full factorization assumption.

Lemma 3. *Let $\mathbf{Q} = \{Q\}$ be a family of factorized distributions of the form in* (A4)*. Then the optimal factorized distribution corresponding to the tightest lower bound is given by,*

$$Q^* = \prod_i q_i^*(z_i) = \arg\max_{Q \in \mathbf{Q}} \mathcal{F}(Q, \boldsymbol{\Theta}) \quad \textit{such that} \quad q_i^*(z_i) \propto \exp\left(\mathbb{E}_{-i}[\log p(\mathcal{X}, \mathcal{Z}|\boldsymbol{\Theta})]\right)$$

where $\mathbb{E}_{-i}[\log p(\mathcal{X}, \mathcal{Z}|\boldsymbol{\Theta})]$ denotes a conditional expectation conditioned on z_i.

Proof. Using lemma 2, the optimal distribution $Q \in \mathbf{Q}$ is given by

$$Q^* = \arg\min_{Q \in \mathbf{Q}} \quad \mathrm{KL}(Q \parallel p(\mathcal{Z}|\mathcal{X}, \boldsymbol{\Theta}))$$

where the KL-divergence can be expressed as

$$\begin{aligned}
\mathrm{KL}(Q \parallel p(\mathcal{Z}|\mathcal{X}, \boldsymbol{\Theta})) &= \sum_i \int_{z_i} q_i(z_i) \log q_i(z_i) dz_i - \int_{z_i} q_i(z_i) \left\{ \int_{\mathcal{Z}_{-i}} \log p(\mathcal{Z}|\mathcal{X}, \boldsymbol{\Theta}) \prod_{j \neq i} q_j(z_j) d\mathcal{Z}_{-i} \right\} dz_i \\
&= \sum_{j \neq i} \int_{z_j} q_j(z_j) \log q_j(z_j) dz_j + \int_{z_i} q_i(z_i) \log \frac{q_i(z_i)}{\exp\left(\mathbb{E}_{-i}[\log p(\mathcal{X}, \mathcal{Z}|\boldsymbol{\Theta})]\right)} dz_i
\end{aligned}$$

The second term in the above expression is a KL-divergence. Keeping $\{q_{j \neq i}(z_j)\}$ fixed, the optimum with respect to $q_i(z_i)$ is attained when KL-divergence is zero, i.e. $q_i^*(z_i) \propto \exp\left(\mathbb{E}_{-i}[\log p(\mathcal{X}, \mathcal{Z}|\boldsymbol{\Theta})]\right)$.

The above lemma shows that the optimal variational distribution subject to the factorization constraint is given by a set of consistency conditions over different factors of the hidden variables. These coupled equations are known as *mean field equations* and can be satisfied iteratively. Convergence is guaranteed because the bound $\mathcal{F}$ is convex with respect to each of the factors [10].

7 APPENDIX B: Mean Field Approximation (MFA) for Bayesian Affinity Estimation

This appendix illustrates the derivation of a MFA based expectation maximization algorithm for parameter estimation of a Latent Dirichlet Attribute Aware Bayesian Affinity Estimation framework (LD-AA-BAE). The techniques introduced in this appendix are also used for derieving updates for rest of the models in the paper and the same analysis can be easily extended. For the purpose of exposition, we however, concentrate only on the LD-AA-BAE model.

The joint distribution over all observable and latent variables for the LD-AA-BAE model is given by:

$$p(\mathcal{Y}, \mathcal{X}_1, \mathcal{X}_2, \mathcal{Z}_1, \mathcal{Z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2 | \boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2, \boldsymbol{\beta}) =$$

$$p(\boldsymbol{\pi}_1|\boldsymbol{\alpha}_1) p(\boldsymbol{\pi}_2|\boldsymbol{\alpha}_2) \left(\prod_m p(\boldsymbol{z}_{1m}|\boldsymbol{\pi}_1) p_{\psi_1}(\boldsymbol{x}_{1m}|\boldsymbol{\theta}_{1z_{1m}}) \right) \left(\prod_n p(\boldsymbol{z}_{2n}|\boldsymbol{\pi}_2) p_{\psi_2}(\boldsymbol{x}_{2n}|\boldsymbol{\theta}_{2z_{2n}}) \right) \left(\prod_{m,n} p_{\psi_y}(y_{mn}|\boldsymbol{\beta}^{\dagger}_{z_{1m}z_{2n}} \boldsymbol{x}_{mn}) \right) \tag{B1}$$

The approximate variational distribution Q over the hidden variables is

$$Q(\mathcal{Y}_{\text{unobs}}, \mathcal{Z}_1, \mathcal{Z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2) = q(\boldsymbol{\pi}_1|\gamma_1) q(\boldsymbol{\pi}_{2|}\gamma_2) \left(\prod_{\substack{m,n \\ y_{mn} \in \mathcal{Y}_{\text{unobs}}}} q(y_{mn}|\phi_{mn}) \right) \left(\prod_m q(\boldsymbol{z}_{1m}|r_{1m}) \right) \left(\prod_n q(\boldsymbol{z}_{2n}|r_{2n}) \right) \tag{B2}$$

The updates for factors corresponding to the optimal variational distribution is obtained using lemma 3.

E-step Update for $q^*(y_{mn}|\phi_{mn})$: Collecting terms containing the affinities y_{mn} in the conditional expectation of the complete log-likelihood, we obtain

$$q^*(y_{mn}) \propto p_0(y_{mn}) \exp \left(\sum_{K,L=1}^{K,L} r_{1mk} r_{2nl} \langle y_{mn}, \boldsymbol{\beta}^{\dagger}_{kl} \boldsymbol{x}_{mn} \rangle \right)$$

which shows that variational distribution for the missing affinities is an exponential family distribution having the same form as the one assumed for the affinities with the natural parameter given by:

$$\phi_{mn} = \sum_{k,l=1}^{K,L} r_{1mk} r_{2nl} \left(\boldsymbol{\beta}^{\dagger}_{kl} \boldsymbol{x}_{mn} \right) \tag{B3}$$

E-step Updates for $q^*(\boldsymbol{\pi}_1|\gamma_1)$ and $q^*(\boldsymbol{\pi}_{2|}\gamma_2)$: Conditional expectation with respect to the mixing coefficients $\boldsymbol{\pi}_1$ yields,

$$\begin{aligned} q^*(\boldsymbol{\pi}_1) &\propto \exp \left(\sum_{k=1}^{K} \left(\alpha_{1k} + \sum_{m=1}^{M} r_{1mk} \right) \log \pi_{1k} \right) \\ &= \prod_{k=1}^{K} (\pi_{1k})^{(\alpha_{1k} + \sum_{m=1}^{M} r_{1mk})} \end{aligned}$$

Easy to see that, the optimal variational distribution $q^*(\boldsymbol{\pi}_1|\boldsymbol{\gamma}_1)$ is a Dirichlet distribution over a K-simplex with parameters given by:

$$\gamma_{1k} = \alpha_{1k} + \sum_{m=1}^{M} r_{1mk} \tag{B4}$$

Similarly, $q^*(\boldsymbol{\pi}_2|\boldsymbol{\gamma}_2)$ is a Dirichlet distribution over a L-simplex with parameters:

$$\gamma_{2l} = \alpha_{2l} + \sum_{n=1}^{N} r_{2nl} \tag{B5}$$

E-step Updates for $q(z_{1m}|r_{1m})$ and $q(z_{2n}|r_{2n})$: Conditional expectation with respect to discrete cluster assignment variable z_{1mk} for the cluster k results in the following update:

$$q^*(z_{1mk} = 1) = r_{1mk} \propto \exp\Bigg(\log p_{\psi_1}(\boldsymbol{x}_{1m}|\boldsymbol{\theta}_{1k}) + \Psi(\gamma_{1k}) - \Psi\left(\sum_{k'=1}^{K} \gamma_{1k'}\right) +$$

$$\sum_{n=1}^{N}\sum_{l=1}^{L} r_{2nl}\left(w_{mn}\log p_{\psi_y}(y_{mn}|\boldsymbol{\beta}_{kl}^{\dagger}\boldsymbol{x}_{mn}) + (1-w_{mn})\mathbb{E}_q\left[\log p_{\psi_y}(y_{mn}|\boldsymbol{\beta}_{kl}^{\dagger}\boldsymbol{x}_{mn})\right]\right)\Bigg) \tag{B6}$$

The first term is the log-likelihood of the entity attributes, the second term is the expectation of $\log \pi_{1k}$ with respect to the variational Dirichlet distribution while the last term involves the log-likelihood of all the affinities associated with the entity e_{1m}. The known log-likelihood is used if the affinity is observed ($w_{mn} = 0$), while the log-likelihood for the missing affinities is replaced by the corresponding expecations under the variational distribution $q^*(y_{mn}|\phi_{mn})$. Analogously, the update equation for the cluster assignment variable $q^*(z_{2nl} = 1)$ is given by:

$$q^*(z_{2nl} = 1) = r_{2nl} \propto \exp\Bigg(\log p_{\psi_2}(\boldsymbol{x}_{2n}|\boldsymbol{\theta}_{2l}) + \Psi(\gamma_{2l}) - \Psi\left(\sum_{l'=1}^{L} \gamma_{2l'}\right) +$$

$$\sum_{m=1}^{M}\sum_{k=1}^{K} r_{1mk}\left(w_{mn}\log p_{\psi_y}(y_{mn}|\boldsymbol{\beta}_{kl}^{\dagger}\boldsymbol{x}_{mn}) + (1-w_{mn})\mathbb{E}_q\left[\log p_{\psi_y}(y_{mn}|\boldsymbol{\beta}_{kl}^{\dagger}\boldsymbol{x}_{mn})\right]\right)\Bigg) \tag{B7}$$

M-step Updates for $\boldsymbol{\theta}_{1k}$ and $\boldsymbol{\theta}_{2l}$: Taking expectation of the complete log-likelihood with respect to the variational distribution, we obtain the following expression for the lower bound $\mathcal{F}$ as a function of the entity attributes parameters:

$$\mathcal{F}(\boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2) = \sum_{m=1}^{M}\sum_{k=1}^{K} r_{1mk}\log p_{\psi_1}(\boldsymbol{x}_{1m}|\boldsymbol{\theta}_{1k}) + \sum_{n=1}^{N}\sum_{l=1}^{L} r_{2nl}\log p_{\psi_2}(\boldsymbol{x}_{2n}|\boldsymbol{\theta}_{2l})$$

Taking partial derivatives with respect to $\boldsymbol{\theta}_{1k}$ and $\boldsymbol{\theta}_{2l}$, we obtain the following updates:

$$\boldsymbol{\theta}_{1k} = \nabla\psi_1^{-1}\left(\frac{\sum_{m=1}^{M} r_{1mk}\boldsymbol{x}_{1m}}{\sum_{m=1}^{M} r_{1mk}}\right) \tag{B8}$$

$$\boldsymbol{\theta}_{2l} = \nabla\psi_2^{-1}\left(\frac{\sum_{n=1}^{N} r_{2nl}\boldsymbol{x}_{2n}}{\sum_{n=1}^{N} r_{2nl}}\right) \tag{B9}$$

M-step Updates for $\boldsymbol{\beta}_{kl}$: Collecting terms containing the GLM coefficients in the lower bound, we obtain:

$$\mathcal{F}(\boldsymbol{\beta}_{kl}) = \sum_{m=1}^{M}\sum_{n=1}^{N} r_{1mk} r_{2nl} \left[\left\langle (w_{mn} y_{mn} + (1 - w_{mn})\nabla\psi_{\mathcal{Y}}(\phi_{mn})), \boldsymbol{\beta}^{\dagger}\boldsymbol{x}_{mn}\right\rangle - \psi_{\mathcal{Y}}\left(\boldsymbol{\beta}^{\dagger}\boldsymbol{x}_{mn}\right)\right]$$

As earlier, the missing affinities are replaced by corresponding expected values under the variational exponential family distribution. The lower bound can be maximized using a gradient ascent method. The expressions for the gradient and the gradient-ascent updates are obtained as follows:

$$\nabla\mathcal{F}(\boldsymbol{\beta}_{kl}) = \sum_{m=1}^{M}\sum_{n=1}^{N} r_{1mk} r_{2nl} \left[(w_{mn} y_{mn} + (1 - w_{mn})\nabla\psi_{\mathcal{Y}}(\phi_{mn})) - \nabla\psi_{\mathcal{Y}}\left(\boldsymbol{\beta}^{\dagger}\boldsymbol{x}_{mn}\right)\right]\boldsymbol{x}_{mn} \tag{B10}$$

$$\boldsymbol{\beta}_{kl}^{t+1} = \boldsymbol{\beta}_{kl}^{t} + \eta\nabla\mathcal{F}(\boldsymbol{\beta}_{kl}) \tag{B11}$$

where η is the step-size for the update.

M-step Updates for $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$: The expression for the lower bound as a function of the Dirichlet parameters $\boldsymbol{\alpha}_1$ is:

$$\mathcal{F}(\boldsymbol{\alpha}_1) = \log\frac{\Gamma(\sum_{k=1}^{K}\alpha_{1k})}{\prod_{k=1}^{K}\Gamma(\alpha_{1k})} + \sum_{k=1}^{K}\left(\alpha_{1k} + \sum_{m=1}^{M} r_{1mk} - 1\right)\left(\Psi(\gamma_{1k}) - \Psi\left(\sum_{k'=1}^{K}\gamma_{1k'}\right)\right)$$

Taking derivative with respect to α_{1k} yield:

$$\frac{\partial\mathcal{F}}{\partial\alpha_{1k}} = \Psi\left(\sum_{k'=1}^{K}\alpha_{1k}\right) - \Psi(\alpha_{1k}) + \Psi\left(\sum_{k'=1}^{K}\gamma_{1k}\right) - \Psi(\gamma_{1k})$$

Note that the update for α_{1k} depends on $\{\alpha_{1k'}, [k']_1^K, k' \neq k\}$, so a closed form solution cannot be obtained. Following [18], Newton-Raphson's method can then be used to update the parameters. The Hessian H is given by

$$H(k,k) = \frac{\partial^2\mathcal{F}}{\partial\alpha_{1k}^2} = \Psi'\left(\sum_{k'=1}^{K}\alpha_{1k}\right) - \Psi'(\alpha_{1k})$$

$$H(k,k') = \frac{\partial^2\mathcal{F}}{\partial\alpha_{1k}\partial\alpha_{1k'}} = \Psi'\left(\sum_{k'=1}^{K}\alpha_{1k}\right) \quad (k' \neq k)$$

The update can then be obtained as follows:

$$\boldsymbol{\alpha}_1^{t+1} = \boldsymbol{\alpha}_1^{t} + \eta H^{-1}\nabla(\boldsymbol{\alpha}_1) \tag{B12}$$

The step-size η can be adapted to satisfy the positivity constraint for the Dirichlet parameters. A Similar method is followed for update of $\boldsymbol{\alpha}_2$.