WWW 2011 Invited Tutorial Overview Latent Variable Models on the Internet

Amr Ahmed
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213, USA
amahmed@cs.cmu.edu

Alexander J. Smola
Yahoo! Research
4301 Great America Pky
Santa Clara, CA 94043, USA
alex@smola.org

ABSTRACT

Graphical models are an effective tool for analyzing structured and relational data. In particular, they allow us to arrive at insights that are implicit, i.e. latent in the data. Dealing with such data on the internet poses a range of challenges. Firstly, the sheer size renders many well-known inference algorithms infeasible. Secondly, the problems arising on the internet do not always fit well into the known categories for latent variable inference such as Latent Dirichlet Allocation or clustering.

In this tutorial we address a number of aspects. Firstly, we present a variety of applications ranging from general purpose document analysis, ideology detection, clustering of sequential data, and dynamic user profiling to recommender systems and data integration. Secondly we give an overview over a number of popular models such as mixture models, topic models, nonparametric variants of temporal dependence, and an integrated analysis and clustering approach, all of which can be used to solve a range of data analysis problems at hand. Thirdly, we present a range of sampling based algorithms for large scale distributed inference using multicore systems and clusters of workstations.

Categories and Subject Descriptors

G.3 [**Mathematics of Computing**]: Probability and Statistics—*Monte Carlos methods*; I.2.11 [**Computing Methodologies**]: Artificial Intelligence—*Distributed Artificial Intelligence*; I.2.6 [**Computing Methodologies**]: Artificial Intelligence—*Learning*

General Terms

Graphical models

Keywords

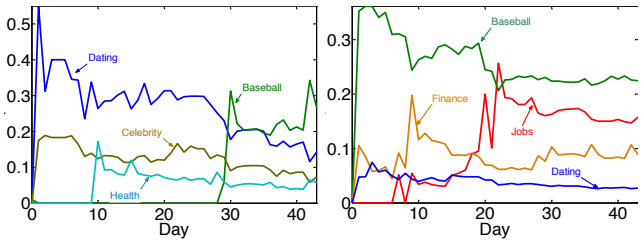
Sampling, latent variables, topic models, clustering

1. DATA ANALYSIS ON THE INTERNET

Sophisticated latent variable models are gaining popularity for data analysis on the World Wide Web. They are particularly useful whenever one is given a huge amount of data without necessarily much annotation in terms of what response is desired given this data, or how the instances are explicitly expected to interact. This is the case, e.g. when we want to assign categories to documents, when we want to group user activities into sets of preferences, or when it comes to recommending relevant users and content on a social network.

One option is to employ a considerable number of editors to define, curate, and update categories/behavioral patterns/interests and then to invoke what is commonly known as supervised learning algorithms to recover the editor's choices. This has the advantage of yielding categories that are formed exactly the way a human would prefer them, alas at the expense of possibly missing key aspects of the data we are modeling (an editor might miss that some categories are almost empty or that some others would benefit from considerably refinement). Furthermore, such a process is quite expensive in terms of human labor and it does not scale well to new domains, languages, or changes in the environment.

An alternative is to use the data more directly and to obtain what is commonly known as a 'generative' model describing the data. While it is a-priori not a given (in fact, there are no theoretical guarantees for it) that such an approach might yield human-understandable categories, clusters, relevant terms, or anything useful at all, it has been shown repeatedly that many data sources are amenable to such a treatment. This works well particularly whenever the statistical model provides a good 'description' of the processes possibly underlying the data generation. The images below provide example of such a situation:



The graphs describe varying user interest for two users over a period of 40 days. While the sets of activities themselves were automatically generated, we used human annotation to *name* the topics. The topics were found to be coherent by human standards. Furthermore, they are valuable in targeting advertisements to users with specific interests.

Unfortunately estimation in latent variable models with huge amounts of data is highly nontrivial. It requires us to address two aspects: firstly we need to choose models that are both reasonably truthful to the observed reality and which can be estimated efficiently.

WWW 2011, March 28–April 1, 2011, Hyderabad, India.
ACM 978-1-4503-0637-9/11/03.

2. MODELS

In this tutorial we discuss a range of models which can be used for large scale data analysis. In particular we discuss clustering, topic models, and recommendation algorithms.

Clustering: It is an excellent tool for obtaining a coarse grouping of users or documents into basic interest groups. This is useful for computational advertising by associating similar users and for the display of news articles to users. The underlying statistical model is one of the simplest latent variable models (the latent variables in this case being the cluster IDs and the model parameters).

In their simplest form latent variable models for clustering manifest themselves as a mixture of Gaussians or multinomial distributions. In general, the constituent distributions have a much large degree of freedom. For instance when modeling documents we may treat named entities differently from the remainder of the word distribution. Likewise, we may use higher order structures and hierarchical models for this purpose.

Topic Models: Often objects are given by a mix of properties. For instance a webpage might contain information about surfing and California. Likewise a user might be interested in a range of things and we would like to model the *distribution* of interests. Both pieces of information are valuable for the purpose of computational advertising, allowing a better match of ads to users and webpages respectively.

Latent Dirichlet Allocation (LDA) is a suitable tool for uncovering such dependencies. It proved successful on smaller scale datasets for extraction of topics in documents. This simple model can be extended, e.g. by conditioning documents on their context (author, link structure), and by conditioning tokens on their related properties. Furthermore, while LDA is often used as a feature generator, it is possible to integrate LDA with a discriminative model directly.

Recommender Systems: Collaborative filtering in social networks improves when it is based not only on observed properties of a user but rather also on the inherent (latent) properties of the network a user inhabits. The associated latent variable models lead to improved algorithms for social recommendation.

Temporal Structure: Real data has often a significant temporal component. The Recurrent Chinese Restaurant Process extends mixture models in such a manner as to allow for a smooth nonparametric description of these phenomena. In a nutshell, it uses previous data as a (downweighted) prior for the mixture model explaining current data.

3. ALGORITHMS

Exact inference in large scale graphical models is essentially infeasible due to the vast amount of data which makes storing the latent state on a single machine (or small number of machines) and processing all data on one machine infeasible. To address this problem we resort to principled approximations which are applicable even for very large scale problems.

- Variational inference and the Expectation Maximization algorithm are some of the first choices that come to mind. They are efficient whenever there is no additional benefit in a sparse representation of the latent state (e.g. whenever the number of clusters or topics or words is small). One advantage of the variational approach is that estimation can be turned into an optimization problem with (typically) continuous variables, thus admitting dense fast floating point computations. This comes at the expense of parsimony in storage — while storing 100 floating point numbers rather than an integer to encode a single cluster membership is usually not a problem, things become less feasible when the number of clusters increases to 10^3 and beyond since now we are paying a 1000x or worse penalty in terms of memory requirements.
- Gibbs sampling and Markov Chain Monte Carlo methods offer a principled alternative for obtaining samples from the distribution of latent variables and therefore for obtaining reliable estimates. We show how these can be implemented efficiently using a large cluster of computers while relying only on a small number of computational primitives (barriers and distributed key value storage). In particular we show how LDA can be applied to hundreds of millions of documents. Furthermore we show how temporal dependencies can be encoded efficiently in this context. This is achieved by performing sequential estimation akin to what is commonly used in a particle filter.
- Particle filtering advances the state of the art in inference by generating a sample of the joint state incrementally. For instance, in the context of time series we have a natural order to traverse the space of instances. This allows us to obtain estimates even in cases where a full batch sampling approach is computationally infeasible or where not all data is available at the same time. Such situations occur regularly in practical applications where we would like to obtain a best-guess estimate of a user's interest even without knowing what he might be doing in the future. We show that this can be carried out efficiently even for sophisticated clustering and topic models in the multicore setting.

Target Audience

The primary audience are researchers in industry and academia who are interested in applying new graphical modeling techniques in the analysis of large scale structured data. We assume that the audience have some prior knowledge of statistics and probability theory (we assume familiarity with Bayes rule) and some basic notions of large scale computation (MapReduce and inter process communication). In some cases we assume that the readers be familiar with belief propagation and message passing in graphical models (a brief introduction will be provided to keep the tutorial self-contained). Throughout our talk we will be providing examples of how to use the algorithms.