A Bayesian Network for Facebook Comment Volumes Prediction

Anurag Dixit and Pavan Gururaj Joshi
Department of Computer Science
University at Buffalo, The State University of New York
Buffalo, NY 14260
anuragdi@buffalo.edu, a.dixit91@gmail.com, pavangur@buffalo.edu

Abstract

This paper involves discovering some interesting relationships between multiple random variables to predict the volume of comments received on a facebook post. A **Bayesian Network** is very crucial approach to determine the reasoning between the variables. The variables involved in the network could be discrete, continuous or mixture of both. This is where **Hybrid** model of **Bayesian Network** comes into play. The hybrid model of a Bayesian Network facilitates handling the data from different distributions and perform inference over the network.

1 Introduction

1.1 Graphical Model

The graphical language exploits structure that appears present in many distributions that we want to encode in practice: the property that variables tend to interact directly only with very few others. Distributions that exhibit this type of structure can generally be encoded naturally and compactly using a graphical model.[3]

A Bayesian Network is a probabilistic directed acyclic graphical model which illustrates the relationship between random variables via edges between them. A node in the graph represents a random variable. A graph with no edges between random variables is considered to be independent[7]. Facebook is a vast network of data and information flow in distributed fashion and comprises of millions of users. In order to determine the relationship between the variables, a sample subset of Facebook's total dataset is considered for this project. The inference performed over this dataset can account for representing to the over all facebook's dataset volume considering a lot of factors such as data set may vary from any geographical location to another, also the connectivity of users may vary across locations for a given post. While the representation of probabilistic graphical models applies, to a great extent, to mtodels including both discrete and continuous-valued random variables, inference in models involving continuous variables is significantly more challenging than the purely discrete case.[3] Some of the readers may have a doubt as why probability theory comes into play with Graphical models.

By Kevin Murphy, 1998 "Graphical models are a marriage between probability theory and graph theory. They provide a natural tool for dealing with two problems that occur throughout applied mathematics and engineering – uncertainty and complexity – and in particular they are playing an increasingly important role in the design and analysis of machine learning algorithms. Fundamental to the idea of a graphical model is the notion of modularity – a complex system is built by combining simpler parts. Probability theory provides the glue whereby the parts are combined, ensuring that the system as a whole is consistent, and providing ways to interface models to data. The graph theoretic side of graphical models provides both an intuitively appealing interface by which humans can model

30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

highly-interacting sets of variables as well as a data structure that lends itself naturally to the design of efficient general-purpose algorithms.

Many of the classical multivariate probabalistic systems studied in fields such as statistics, systems engineering, information theory, pattern recognition and statistical mechanics are special cases of the general graphical model formalism – examples include mixture models, factor analysis, hidden Markov models, Kalman filters and Ising models. The graphical model framework provides a way to view all of these systems as instances of a common underlying formalism. This view has many advantages – in particular, specialized techniques that have been developed in one field can be transferred between research communities and exploited more widely. Moreover, the graphical model formalism provides a natural framework for the design of new systems." — Michael Jordan, 1998.

2 Problem Domain

The main focus of this paper is to study the relationships between random variables for a given post on facebook and predicting the volume of comments a post can receive. Although the relationship need not be causal but preferred in most of the cases since graph network gives a intuition about the real world problems. The "Facebook comment volume dataset" considered for this paper comprising of 54 multivariate random variables with a mixture of discrete and continuous data. For a hybrid network of such combination, the conditional dependencies calculation becomes more interesting. Since, there can be multiple combinations such as Discrete Parent with Continuous child node, Constinuous parent with Continuous child, Continuous children with both discrete and continuous parents. Bayesian Network graph learned for this project involves all such cases.

Joint Probability of a Bayesian Network can be formulated using Sum Rule and Product Rule of probability.

$$p(a, b) = p(b \mid a)\ p(a)$$

Using these two rules all probabilistic inference and learning can be achieved for a network. Bayesian Network provides intuition for the interfacing the variables. Therefore, it is easy to visualize and get some insight just by looking the graph. The scope of this paper is limited to Bayesian Network so undirected graphs such as Markov Models are not within the scope of this paper.

Joint probability distribution for a graph can be defined as a product of conditional probabilities of the nodes conditioned on variables corresponding to parents of that node in the graph.

$$p(x) = \Pi_{k=1}^{K} p(x_k|pa_k)$$

where x denotes the variable and $\mathrm{pa_k}$ denotes the parents of $\mathrm{a_k}$

Given above basics, we propose the Bayesian Network model approach for determining the relationship between the variables and let the reader know what causes the higher or lower volume of comments over their post on facebook. The content of this paper is mainly summarized as follows:

- DataSet - Source and preprocessing required
- Structure Learning - about the Bayesian Network and data
- Algorithms Used - approach used
- Performance Evaluation Metrics - how to determine whether the model works better or not
- Application - future scope of such models

3 Dataset

The dataset has been referenced from UCI Machine Learning Repository[6]. It has 54 multivariate random variates. It has both continuous and discrete distributions of data. There are more than 600,000 inputs in this dataset. As the hybrid model of Bayesian Network requires explicit compatible data input to perform the sampling for approximate inference, it is very important to preprocess the data into a compatible format[4]. The preprocessing involves Linear Regression over parent node's data and determine the variance, mean_scal and mean_base. It also involves using marginal probabilities as a conditional probabilty for nodes with no parents. This stage requires a lot of effort to filter out the irrelevant data to our models and reduce dimensionality along with parsing the data as per hybrid model. libpgm[4] package has been used for modeling the hybrid network. Conditional Probability distribution of variables such as postDay and baseDay are calculated as deterministic CPDs for optimization.

4 Structure Learning

The dataset consists of 54 multivariate variables[6] and the Bayesian Network defined has both Continuous and Discrete type of data thereby making it a Hybrid Bayesian Network model. Variables such as postDay, baseDay, pageCategory, postPromotionStatus are discrete data where as the rest are Continuous data.

The network model for this paper is proposed as Figure 1

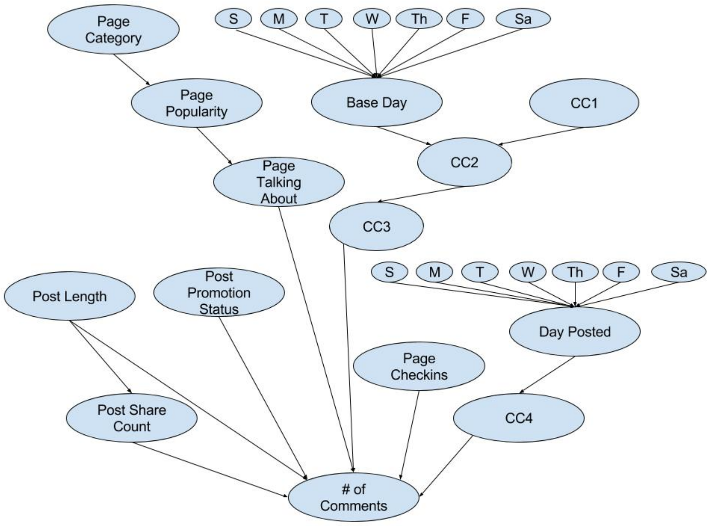

Figure 1: Bayesian Network for Facebook comment volume

As the number of variables in the dataset is large, so dimensionality reduction has been performed over the baseDay and postDay and other statistics related variables are dropped as they are required to be calculated anyway. The dimensions have been reduced to 28.

Structure Learning is a NP-Hard problem therefore for the sake of simplicity the Bayesian Network has been constructed based on manual human intuition and tweaked on the basis of inference in order to get an efficient structure for the problem. Since we have proposed the Bayesian network based on intuition, there is no underlying assumption in learning the structure of network graph.

Some of the Local Independencies which are identified based on our proposed Bayesian network are:

Table 1: Local Independencies

(Comments _|_ baseDay, pageCategory, pagePopularity, cc2, cc1, postDay | postLength, postPromotion, cc4, pageCheckins, postShareCt, pageTalkingAbout, cc3)

(postLength _|_ postPromotion, pageCategory, cc1, cc3, cc2, cc4, baseDay, pagePopularity, postDay, pageTalkingAbout, pageCheckins)

(pageTalkingAbout _|_ postLength, postPromotion, cc4, cc1, cc3, cc2, pageCategory, baseDay, postShareCt, postDay, pageCheckins | pagePopularity)

(pageCategory _|_ postLength, postPromotion, cc4, cc1, cc3, cc2, pageCheckins, baseDay, postShareCt, postDay)

Independencies which can be inferred from the graph using d-separation for variable Comments are:

Table 2: Local Independencies using d-separation

(Comments _|_ postPromotion, pageCheckins, cc1, cc3, cc2, cc4, baseDay, postShareCt, pagePopularity, postDay, pageTalkingAbout, pageCategory | postLength)

(Comments _|_ postLength, postPromotion, pageCheckins, cc1, cc3, cc2, pageCategory, baseDay, postShareCt, pagePopularity, pageTalkingAbout | cc4)

(Comments _|_ postLength, pageCheckins, cc1, cc3, cc2, cc4, baseDay, postShareCt, pagePopularity, postDay, pageTalkingAbout, pageCategory | postPromotion)

(Comments _|_ postLength, postPromotion, pageCheckins, cc1, cc3, cc2, cc4, baseDay, postShareCt, pagePopularity, postDay, pageTalkingAbout | pageCategory)

(Comments _|_ postLength, postPromotion, pageCheckins, cc3, cc2, cc4, baseDay, postShareCt, pagePopularity, postDay, pageTalkingAbout, pageCategory | cc1)

(Comments _|_ postLength, postPromotion, pageCheckins, cc3, postDay, cc4, postShareCt, pagePopularity, pageTalkingAbout, pageCategory | cc2)

(Comments _|_ postLength, postPromotion, pageCheckins, postDay, cc4, postShareCt, pagePopularity, pageTalkingAbout, pageCategory | cc3)

(Comments _|_ postLength, postPromotion, cc4, cc1, cc3, cc2, pageCategory, baseDay, postShareCt, pagePopularity, postDay, pageTalkingAbout | pageCheckins)

(Comments _|_ postLength, postPromotion, pageCheckins, cc1, cc3, cc2, cc4, postShareCt, pagePopularity, postDay, pageTalkingAbout, pageCategory | baseDay)

(Comments _|_ postLength, postPromotion, pageCheckins, cc1, cc3, cc2, cc4, baseDay, pagePopularity, postDay, pageTalkingAbout, pageCategory | postShareCt)

(Comments _|_ postLength, postPromotion, pageCheckins, cc1, cc3, cc2, cc4, baseDay, postShareCt, postDay, pageTalkingAbout | pagePopularity)

(Comments _|_ postLength, postPromotion, pageCheckins, cc1, cc3, cc2, cc4, baseDay, postShareCt, pagePopularity, pageTalkingAbout, pageCategory | postDay)

(Comments _|_ postLength, postPromotion, pageCheckins, cc1, cc3, cc2, cc4, baseDay, postShareCt, postDay | pageTalkingAbout)

5 Algorithms Used

A graphical model specifies a complete joint probability distribution (JPD) over all the variables. Given the JPD, we can answer all possible inference queries by marginalization (summing out over irrelevant variables). In this project we have used Approximate inference algorithms based on Monte Carlo methods.

5.1 Approximation algorithms

Many models of interest, such as those with repetitive structure, as in multivariate time-series or image analysis, have large induced width, which makes exact inference very slow. We must therefore resort to approximation techniques. Unfortunately, approximate inference is **#P-hard**, but we can nonetheless come up with approximations which often work well in practice. Below is a list of the major techniques.[5]

- Variational methods. The simplest example is the mean-field approximation, which exploits the law of large numbers to approximate large sums of random variables by their means. In particular, we essentially decouple all the nodes, and introduce a new parameter, called a variational parameter, for each node, and iteratively update these parameters so as to minimize the cross-entropy (KL distance) between the approximate and true probability distributions. Updating the variational parameters becomes a proxy for inference. The mean-field approximation produces a lower bound on the likelihood. More sophisticated methods are possible, which give tighter lower (and upper) bounds.

- Sampling (Monte Carlo) methods. The simplest kind is importance sampling, where we draw random samples x from P(X), the (unconditional) distribution on the hidden variables, and then weight the samples by their likelihood, P(y|x), where y is the evidence. A more efficient approach in high dimensions is called Monte Carlo Markov Chain (MCMC), and includes as special cases Gibbs sampling and the Metropolis-Hasting algorithm.

- "Loopy belief propogation". This entails applying Pearl's algorithm to the original graph, even if it has loops (undirected cycles). In theory, this runs the risk of double counting, but Yair Weiss and others have proved that in certain cases (e.g., a single loop), events are double counted "equally", and hence "cancel" to give the right answer. Belief propagation is equivalent to exact inference on a modified graph, called the universal cover or unwrapped/ computation tree, which has the same local topology as the original graph. This is the same as the Bethe and cavity/TAP approaches in statistical physics. Hence there is a deep connection between belief propagation and variational methods that people are currently investigating.

- Bounded cutset conditioning. By instantiating subsets of the variables, we can break loops in the graph. Unfortunately, when the cutset is large, this is very slow. By instantiating only a subset of values of the cutset, we can compute lower bounds on the probabilities of interest. Alternatively, we can sample the cutsets jointly, a technique known as block Gibbs sampling.

- Parametric approximation methods. These express the intermediate summands in a simpler form, e.g., by approximating them as a product of smaller factors. "Minibuckets" and the Boyen-Koller algorithm fall into this category.

Approximate inference is a huge topic in itself

As the network model is hybrid, approximate inference algorithm has been used to generate samples from the model based on data and infer the relationship of one or more variables over the others.[3] Approximate inference algorithms are used for determining the mean and entropy of each distribution and relative entropy between distributions.

For a child node with continuous parents, Conditional Probability Distribution of child node can be represented as a linear Gaussian of continuous parents. If a continuous child has both discrete and continuous parents then its CPD is defined by different set of parameters for every value of the discrete parent. These parameters can be learned by Linear Regression where the target is child node and training set is parent data.

In Conditional Linear Gaussian network, every discrete variable has only discrete parents and every continuous variable has a CLG CPD. It is important to note that in such networks a continuous variable cannot have discrete child node and distribution is a mixture of weighted average of Gaussians. In case of discrete child with a Continuous parent, threshold technique can be used to determine the probability but there can be a problem of abrupt change in probability with parent's value. It can be taken care by using the logistic model. Learning of parameters is done using linear regression model over the parent node data for a particular node.

The process of conditioning (also called probability propagation or inference or belief updating) is performed via a “flow of information” through the network. Note that this information flow is not limited to the directions of the arcs. In our probabilistic system, this becomes the task of computing the posterior probability distribution for a set of query nodes, given values for some evidence (or observation) nodes.[1]

6 Performance

Following performance metric evaluation has been used for results.

Mean of a Distribution is evaluated as:

$$E[p(x)] = \sum_x xp(x)$$

For the Bayesian Network conditional dependencies are accounted for metric evaluation such as:

$$p(x) = \Pi_{i=1}^{D} p(X_i|pa(X_i))$$

Entropy of a distribution is:

$$H[p(x)] = -\sum_x p(x)lnp(x)$$

For N samples, it can be estimated as:

$$\hat{H}[p(x)] = -\frac{1}{N}\sum_{k=1}^{N} lnp(x_{\text{k}})$$

Relative Entropy (K-L divergence) between two probability distributions over a random variable x is a measure of distance between them and is evaluated as:

$$KL(p||q) = -\sum_x p(x)[lnq(x) - lnp(x)]$$

For N samples, it can be estimated as:

$$K\hat{L}(p||q) = -\frac{1}{N}\sum_i p(x)[lnq(x_k) - lnp(x_k)]$$

6.1 Properties of Divergence

Few properties associated with Divergence are as follows. Detail derivation is referenced[2]

- Divergence is not symmetric. That is, KL(p||q) = KL(q||p) is not necessarily true.
- Divergence is always non-negative.
- Divergence is a convex function on the domain of probability distributions.

The results of the Hybrid Bayesian network model is determined as above mentioned performance evaluation metric and the results are:

Mean : 8.29779874544
Entropy : 4.60252656202
KL Divergence : 0.00284625541208

Some of the domain specific queries which were inferred and answered by proposed Bayesian Network model are:

Domain Specific queries		
Node queried	Evidence	Probability
Comments = 100	postLength = 25	0.000229574653351
postDay	NA	0.158
pageCategory = 4	postShareCt = 116	0.037
pageCategory = 24	NA	0.102

Some of the probabilities observed are small because of the large variance in the input dataset over the variables which causes the Gaussian Distribution to be spread across a larger range of values.

Since approximate inference algorithm is NP-Hard problem[8], conditioning probabilities associated with minor restrictions makes it possible to calculate in polynomial time of n (number of nodes in the graph). In the proposed model, the complexity has been restricted to polynomial time of n.

7 Application

Above proposed approach is just an example of what kind of problems could be answered using Bayesian Network model. An important application of this model can be used by advertising commercials to predict the spread of their products across social network. In addition to this, there can be multiple variations such as predicting the likes volume for an advertised product post, sentiment analysis over the comment section to treat them as a feedback from the consumers directly and in a most efficient way.

References

[1] URL: `http://www.csse.monash.edu.au/bai/book/BAI_Chapter2.pdf`.

[2] Mark Braverman. *Information Theory in Computer Science*. URL: `https://www.cs.princeton.edu/courses/archive/fall11/cos597D/L03.pdf`.

[3] Daphne Koller & Nir Friedman. *Probabilistic Graphical Models Principles and Techniques*. MIT Press,

[4] “libpgm Documentation”. In: *http://pythonhosted.org/libpgm/* ().

[5] Kevin Murphy. *A Brief Introduction to Graphical Models and Bayesian Networks*. URL: `http://www.cs.ubc.ca/~murphyk/Bayes/bnintro.html`.

[6] Kamaljot Singh. *Facebook Comment Volume Dataset Data Set*. URL: `https://archive.ics.uci.edu/ml/datasets/Facebook+Comment+Volume+Dataset`.

[7] Sargur Srihari. *Advanced Machine Learning Course*. URL: `http://www.cedar.buffalo.edu/~srihari/CSE674/index.html`.

[8] Liessman Sturlaugson and John W. Sheppard. *Inference Complexity in Continuous Time Bayesian Networks*. URL: `http://auai.org/uai2014/proceedings/individuals/64.pdf`.