Knowledge Discovery in Relational Databases

Midterm project report:
Sogol Balali, Arezoo Rajabi, Mandana Hamidi
{balalis, rajabia, hamidim}@onid.oregonstate.edu

March 12, 2015

Abstract

The main objective of this project is to learn new concept from structured dataset such as relational database. We studied the behavior of three relational machine learning algorithm including First Order Inductive Logic (FOIL) , Top-Down Inductive Decision Tree (TILDE) and Mixture Model Membership. The results show that the TILDE algorithm performs better than FOIL algorithm, ...

1 Introdoction

Relational data is common method to storage data and many systems use them as input for knowledge discovery. Recently researchers have been interested in data mining in relational data to find uncovered relations which are useful for the owner. Consider a dataset that includes the data of customers' purchase history [Getoor and Mihalkova, 2011]. One of the beneficial knowledge in this data set is the categories of items that are purchased together. This dataset does not contain the categories explicitly but many methods are proposed to find the hidden relation in relational data. A formal definition for relations learning methods are brought in the below:
Problem definition: In relational database each object or example is shown in as a tuple in tables. Here, we know some facts and not to find a set of rules which define the relations between objects to extract knowledge from data base. In other words, for a given set of classes C and set of classified objects E and a background theory of B, the problem is an hypothesis H such that $\forall e \in E, H \wedge e \wedge B \rightarrow c$, where $c \in C$.

In this projects, we plan to compare different relation learning methods' performance. In the rest of this paper, we explain the related works (Section 2), experimental settings (Section 3) and a description of what you have done so far and the problems and issues you have faced in implementing / developing your solutions (Section 4).

2 Related Work

There are some approaches to relational data mining such as Statistical Relational Learning (SRL) [Blockeel and De Raedt, 1998, Getoor and Mihalkova, 2011], multi-view learning [Xu et al., 2013], propositionalization [Kramer et al., 2000], and etc.
In this section we explain three main relational learning algorithms that we selected for solving the problem. The first algorithm is FOIL, which we plan to select it as a baseline, the second algorithm is TILDE, which

is a first order logic extension of the C4.5 decision tree algorithm, and the last one is Mixed Membership Model, which solves the problem form the probabilistic graphical model perspective.

2.1 First Order Inductive Learner (FOIL)

FOIL is first-order supervised learning algorithm that uses a divide and conquer strategy to define literals which specifies the class of objects. FOIL's input includes information about the target relation which we want to learn rules about [Quinlan and Cameron-Jones, 1993]. The objects that do not belong to the target class can be given as a input to this method to learn rules better. The tuples that in the target class are shown by $\oplus$ and the tuples are not in this class are shown by $\oslash$. This algorithm starts with all $\oplus$ and $\oslash$ objects and after that it constructs a function-free Horn clause to explain some of $\oplus$ tuples. Then it removes the covered these tuples and continues with rest of the tuples. FOIL starts with left-hand side of clause and specifies it by adding littorals to the right hand side.

- Start by defining right hand clause:
  $R(V_1, V_2, V_3, ..., V_k) \leftarrow$
  Set a Training set T that contains all $\oplus$ and $\oslash$ tuples.
- While T contains $\oslash$ tuples and it does get complex
  - Find a literal L to ass to right hand side of the clause
  - Create new Training set by removing the $\oplus$ tuples that covers by the clause and the $\oslash$ tuples that are rejected by the new clause.
- Prune the clause by unnecessary literals.

The main step in this algorithm is to determine appropriate literals to append to the clause. Two main features that each literal must provides are:

- The fist one evaluate each literals based on the number of $\oslash$ tuples removed and $\oplus$ tuples covered by that.
- The literal has to introduce new variables that are useful in future literals.

In this regard this algorithm uses the Gain metric to measure the amount of information is covered by a literals. Let T_+ denote the number of $\oplus$ tuples in training set T and T'_+ denotes the number of $\oplus$ tuples that remain in training set after adding literal L. Therefore, the amount of information is in training set T is :

$$I(T) = -log(\frac{T_+}{T}) \tag{1}$$

And the gain information by applying the literal L is:

$$gain(L) = s \times (I(T) - I(T')) \tag{2}$$

2.2 Top-down Induction of first-order Logical Decision trees (TILDE)

TILDE algorithm is one of the methods for learning relations in relational database. TILDE can be considered as a first order logic extension of the C4.5 decision tree algorithm, which is a state-of-the-art decision tree learner for attribute-value problems. However, instead of testing attribute values at the nodes of the

tree, TILDE tests logical predicates. This provides the advantages of both propositional decision trees (i.e. efficiency and pruning techniques) and the use of first-order logic (i.e. increased expressiveness). First-order logic enables us to use a background knowledge (which is not possible with non relational data mining algorithms).

In TILDE a set of rules describe relations. These rules are represented as a regression tree in which the leaves describe probabilities of each rules. TILDE uses a form of information gain heuristic over relational features and learns the probabilities for different right hands sides of production rules. TILDE is one the other well-known machine learning to construct a top down decision tree. However a few learning systems have made use of decision tree techniques. The main reason of that is corresponding to the discrepancies between representation of inductive logic programming and structure underlying a decision tree. The TILDE is a first order logical decision tree based on ID3 and introduces a logical representation for relational decision trees. Moreover, this algorithm applies a refinement operator that improves the computation of the set of tests considered at a node.

2.3 Mixed Membership Model

One the popular methods to model relational data and analyze them is membership block models. Finding the relations in protein-protein interaction network, discovering groups in social network are the most famous problems that are defined in this area.

The key idea in these methods is that the relational data is mapped to a graph $G(N, Y)$, where $Y(p, q)$ maps pairs of nodes to values(edge weights) and then maximize the log-likelihood of $E[P(Y|\alpha, B)]$ in which B is group interaction probability matrix. Assume there are K groups in given data. $\pi_i(j)$ indicates the probability of object i belongs to group j. The interaction between groups is defined by a matrix of Bernoulli rates $B_{(K\times K)}$ where$B(e, f)$ indicates the probability of having an edge between an object from group e and an object from group f. The mixed membership model is drawn as follow [Airoldi et al., 2009]:

- for each $p \in N$ draw a K dimensional mixed membership vector $\vec{\pi_p} \sim Dirichlet(\vec{\alpha})$.
- for each pair of $(p, q) \in N \times N$:
  - Draw a membership indicator for initiator $\vec{z}_{p\to q} \sim multinomial(\vec{\pi}_p)$
  - Draw a membership indicator for initiator $\vec{z}_{p\leftarrow q} \sim multinomial(\vec{\pi}_q)$
  - sample the value of the interaction , $Y(p, q) \sim Bernoulli(\vec{z}^{\top}_{p\to q} B \vec{z}_{p\leftarrow q})$

The joint probability of Y and the latent variables $\{\vec{\pi}_{1:N}, Z_{\to}, Z_{\leftarrow}\}$ can be written in the following factors form:

$$P(Y, \{\vec{\pi}_{1:N}, Z_{\to}, Z_{\leftarrow}|\vec{\alpha}, B) = \prod_{p,q} P(Y(p, q)|\vec{z}_{p\to q}, B, \vec{z}_{p\leftarrow q}) P(\vec{z}_{p\to q}|\vec{\pi}_p) P(\vec{z}_{p\leftarrow q}|\vec{\pi}_q) \prod_{p} P(\vec{\pi}_p|\vec{\alpha}) \quad (3)$$

EM method usually is used to optimize the objective function $E[P_{Y|B,\alpha}]$ such that during the E- step, the posterior distribution over the unknown variable quantities $\{\vec{\pi}_{1:N}, Z_{\to}, Z_{\leftarrow}\}$ is updated. In this regard, mean- field variational methods has been used.
During M-step, the empirical Bayes estimates of the hyper-parameters are computed. The M-step is equivalent to finding the MLE using expected sufficient statistics under the variational distribution.
The block models methods are introduced for relational methods and the main problem is to match this method to our problem is to define $Y(x, y)$.

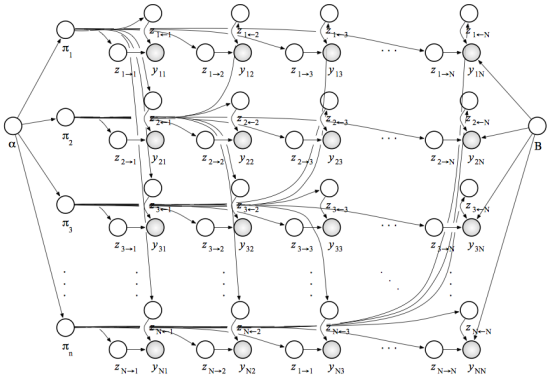

Figure 1: Mixed membership block model

3 Experimental settings

In order to compare the performance of relational learning algorithms we plan to carry out the following different evaluation methods on the UW-CSE datase.

3.1 Datasets

UW-CSE: We carried experiments on the UW-CSE data set by Richardson and Domingos [Richardson and Domingos, 2006], which consists of 12 relations, 2673 tuples, and 113 positive examples. Following [Frana et al., 2014] and [Picado et al., 2014], we generated negative examples using the closed-world assumption, and then sampled these to obtain five as many negative examples as positive examples.

3.2 Evaluation criteria

In order to evaluate the performance of the learning methods we applied 4 following well-known evaluation criteria:

- **Precision**: Precision measure the fraction of true found relations and is defined as below:

$$Precision = \frac{|R \cap Tr|}{|R|} \tag{4}$$

- **Recall**: Recall emphasizes only on the the fraction of the expected relations are returned by an algorithms. This criterion is defined as below:

$$Recall = \frac{|R \cap Tr|}{|Tr|} \tag{5}$$

- Accuracy:

- **F-measure:** Precision and Recall are the well-known criteria to evaluate the results. where, R is the round relations by the algorithm and Tr is the all true relations could be found in data. Consider an algorithms all possible relations, then the Recall would be one while the Precision would be so low. In the other hand, if the algorithm just return few true relation, then the Precision would be so high, while the recall would be so low.
  Both criteria do not work alone. Therefore, we use another criteria that is defined by combination of these two criteria. F-measure is harmonic mean of recall and precision:

$$F - measure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{6}$$

3.3 Learning systems

We plan to apply the following systems, which contains some of relational learning algorithm:
Aleph [1]**:** Aleph is a well known ILP system that can emulate several other ILP systems such as FOIL and Progol.
ACE [2]**:** A tool for relational learning that includes TILDE and several other relational learning algorithms and is based on an advanced special-purpose logical inference engine.

4 Our progress and Future Phases of Project

So far we have done the following steps:

1. We downloaded the UW-CSE dataset. Since the dataset contains only the positive examples, we generated a set of negative examples that are 5 times larger than the positive examples set
2. We setup the codes of the Aleph and also the ACE systems.
3. We generated the background knowledge file that was needed for TILDE algorithm.

The next steps of the project:

1. Run the TILDE and FOIL algorithms on the UW-CSE dataset and compute the precision, recall, and F-measure.
2. Proposing a mixed membership block model for the UW-CSE dataset and compare it with the other methods.

References

E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing. Mixed membership stochastic blockmodels. In *Advances in Neural Information Processing Systems*, pages 33–40, 2009.

H. Blockeel and L. De Raedt. Top-down induction of first-order logical decision trees. *Artif. Intell.*, 101 (1-2):285–297, May 1998. ISSN 0004-3702.

[1] http://www.cs.ox.ac.uk/activities/machlearn/Aleph/aleph.html.
[2] http://dtai.cs.kuleuven.be/ACE/

M. V. M. Frana, G. Zaverucha, and A. S. d'Avila Garcez. Fast relational learning using bottom clause propositionalization with artificial neural networks. *Machine Learning*, 94(1):81–104, 2014.

L. Getoor and L. Mihalkova. Learning statistical models from relational data. In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, pages 1195–1198. ACM, 2011.

S. Kramer, N. Lavrač, and P. Flach. Relational data mining. chapter Propositionalization Approaches to Relational Data Mining, pages 262–286. Springer-Verlag New York, Inc., New York, NY, USA, 2000. ISBN 3-540-42289-7.

J. Picado, A. Termehchy, and A. Fern. Schema independence of learning algorithms. *First international workshop on Big Uncertain Data (BUDA 2014)*, 2014. URL http://www.sigmod2014.org/buda.

J. R. Quinlan and R. M. Cameron-Jones. Foil: A midterm report. In *Machine Learning: ECML-93*, pages 1–20. Springer, 1993.

M. Richardson and P. Domingos. Markov logic networks. *Mach. Learn.*, 62(1-2): 107–136, Feb. 2006. ISSN 0885-6125. doi: 10.1007/s10994-006-5833-1. URL http://dx.doi.org/10.1007/s10994-006-5833-1.

C. Xu, D. Tao, and C. Xu. A survey on multi-view learning. *CoRR*, pages –1–1, 2013.