2009 IEEE International Conference on Data Mining Workshops

LNBC : A Link-Based Naive Bayes Classifier

Bahareh Bina
Computer Science Dept.
Simon Fraser University
Burnaby,Canada
bba18@cs.sfu.ca

Oliver Schulte
Computer Science Dept.
Simon Fraser University
Burnaby,Canada
oschulte@cs.sfu.ca

Hassan Khosravi
Computer Science Dept.
Simon Fraser University
Burnaby,Canada
hkhosrav@cs.sfu.ca

***Abstract*—Many databases store data in relational format, with different types of entities and information about links between the entities. Link-based classification is the problem of predicting the class label of a target entity given information about features of the entity and about features of the related entities. A natural approach to link-based classification is to upgrade standard classification methods from the propositional, single-table testing. In this paper we propose a new classification rule for upgrading naive Bayes classifiers (NBC). Previous work on relational NBC has achieved the best results with link independency assumption which says that the probability of each link to an object is independent from the other links to the object. We formalize our method by breaking it into two parts: (1) the independent influence assumption: that the influence of one path from the target object to a related entity is independent of another. We consider object-path independency and (2) the independent feature assumption of NBC: that features of the target entity and a related entity are probabilistically independent given a target class label. We derive a new relational NBC rule that places more weight on the target entity features than formulations of the link independency assumption. The new NBC rule yields higher accuracies on three benchmark datasets—Mutagenesis, MovieLens, and Cora—with average improvements ranging from 2% to 10%.**

I. Introduction

Most real-world structured data are stored in tables representing multiple relations. Relational data mining aims at discovering interesting knowledge directly from multiple tables. An important task in relational data mining is *link-based classification* which is the problem of predicting a *target attribute*, or *class label*, of a target entity given information about features (attributes) of the entity and about features (attributes) of the related entities [7]. A natural approach to link-based classification is to upgrade propositional learners developed for single-table classification [10] to deal with relational data. We propose a relational version of the naive Bayes classifier (NBC) [5]. NBC uses only the features of the target entity; it assumes that such features are probabilistically independent given a class label for the target entity. The required probabilities for classification are usually estimated using the observed frequency counts. NBC has been widely investigated in the propositional setting. It produces competitive classification accuracy [5], [8], [4]; in addition it is easy to learn and easy to understand.

To upgrade NBC in order to deal with relational data we introduce a new *independent influence assumption* (IIA): treating the influence of the object and each of its linked entities on the target class label as independent of the influence of the object and its other linked entities. In cases with many related links to an object, the information from related links is overweighted, reducing the impact of the class object's attributes. We introduce the link-based Naive Bayes classifier (LNBC), which places higher weights on attributes of the class object and the prior probability of the class.

For empirical evaluation, we compare LNBC with the state of the art relational NBC, the Graph-NB method of [11], which uses the same classification rule as [14], [6]. LNBC yields higher accuracies on three benchmark datasets—Mutagenesis, MovieLens, and Cora—with average improvements ranging from 2% to 10%.

The rest of the paper is organized as follows. Related work is discussed in section II. Section III presents preliminaries. Our algorithm (LNBC) is introduced in Section IV, and experimental results are presented in Section V. Finally, we conclude this study in Section VI.

II. Related Work

While several relational versions of the NBC classifier, dealing with relational data directly, have been examined, two versions that are most widely used and perform best in evaluations are RNBC [14], and Graph-NB [11].

RNBC is based on the *independent value assumption* (IVA). To explain the intuition behind the independent value assumption, it is helpful to consider link-based classification in terms of multisets: Whereas in single-table classification, a given predictive feature has only one value, in link-based classification, a feature may have several different values, corresponding to the number of links between the target object and predictive feature's entity. For example, if we seek to predict the intelligence of a student given her GPA and the difficulty of the courses she has taken, the feature $difficulty$ has as many values as the number of courses the student has taken. IVA assumes that different values of a feature are independent of each other.

978-0-7695-3902-7/09 $26.00 © 2009 IEEE
DOI 10.1109/ICDMW.2009.116

Graph-NB is based on the link independency assumption. It assumes that related links to an object are independent of each other. Although these two methods have two different approaches to classification, they have the same rule for classification which is obtained by the product taken over attributes of the target entity conditioned on the class label, times another product whose factors are the probabilities of the features of linked entities, conditional on a class label for the target entity, times the probability of the class label to find the most probable class label. The main difference to our rule is the different weights assigned to the probabilities of the class label and the target entity features.

An intuitively different approach to relational NBC from the multiset view is based on rules, similar to Inductive Logic Programming [9], [11], [3]. These approaches also assign less weight to the target entity features and class label than our NBC rule. The rule-based approaches focus on generating informative rules or pruning uninformative ones. As our focus in this paper is not rule search but rather how to evaluate rules to produce a prediction, we simply include all possible foreign key paths, that is, all possible rules, and use the data to weight them.

A practically important problem in relational classification is collective classification in which all the class labels of linked entities are not determined [12], [17], [1]. For example, the class labels of papers cited by the paper considered for classification may be unknown. Because of our focus on evaluating the IIA assumption, we do not consider such problem in this paper, and only cases with determined classes for all linked objects are covered.

III. Preliminaries

A standard **relational database** contains a set of tables, each with key fields, descriptive attributes, and possibly foreign key pointers. A **database instance** specifies the tuples contained in the tables of a given database schema. We assume that tables in the relational schema are divided into *entity tables* and *Relationship tables.* This is the case whenever a relational schema is derived from an entity-relationship model (ER model) [15, Ch.2.2]. Tables presenting objects are referred to as Entities and the tables showing the relationships between them are referred to as relationship tables. The symbol X refers to the class target table, symbol S refers to other entity tables, and the symbol R refers to relationship tables or links.

Table I shows a relational schema for a database related to a university. In our university example, there are three objects or entity tables: $Student$, $Course$, and $Professor$. There are two relationship tables; $Registered$ with foreign key pointers to the $Student$ and $Course$ tables whose tuples indicate which students have registered in which courses and $Taught-by$ with foreign key pointers to the $Course$ and $Professor$ tables whose tuples indicate courses taught by professors.

IV. Link-based Naive Bayes Classification

We upgrade NBC to deal with relational data as follows. In single table classification, object $x = (x_1, x_2, \ldots, x_n)$ is classified into the class c according to the following formula.

$$P(c|x_1, x_2, \ldots, x_n) \propto \prod_{j=1}^{n} P(x_j|c)P(c) \qquad (1)$$

This formula is derived from the naive Bayes assumption that features of an entity are independent conditional on its class label. Whereas this independence assumption is not completely true in most real data, research has shown that naive Bayesian classifiers perform well [4].

We define a relational naive Bayes classifier in two steps: first, computing the influence of a single linked object on the class label prediction, and second, treating the influences as independent by multiplying them. Our notation follows [11].

A. Evaluating the influence of a single linked object

First consider the case in which object s is linked to target object x by a single relation R, that is, $R(x, s)$ holds. Let $r_1, \ldots, r_m$ be the attributes of the relation (link) R, and let $s_1, \ldots, s_h$ be the attributes of the related object. Figure 1 illustrates a link between a target entity and a related object, and the attributes associated with the link.

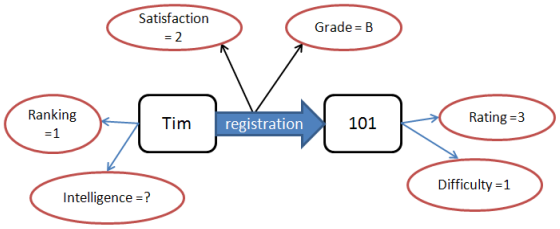

Figure 1. Course 101 is related to the target entity Tim by the Registration relation.

The class label prediction is proportional to

$$P(c|x_1, \ldots, x_n, r_1, \ldots, r_m, s_1, \ldots, s_h) \propto P(c) \prod_{i=1}^{n} P(x_i|c) \prod_{j=1}^{m} P(r_j|c) \prod_{k=1}^{h} P(s_k|c) \qquad (2)$$

The independence assumption in Equation (2) is illustrated in the Bayes network of Figure 2.

Another way in which attributes of a related entity can carry information about the class label is via a sequence of links between the related entity and the target entity. We call this sequence a foreign key path containing the chain of relationship tables $R_1, R_2, \ldots, R_n$ where each R_i has at least one common foreign key with R_{i+1}. An instance of a foreign key path u is a sequence of links $u = x, l_1, l_2, \ldots, l_{m_u}$. The last object linked to by u is

Student(*student_id*: integer, *intelligence*: string, *ranking*: string)
Course(*course_id*: integer, *difficulty*: string, *rating*: string)
Professor (*professor_id*: integer, *teaching_ability*: string, *popularity*: string)
Registered (*student_id*: integer, *Course_id*: integer, *grade*: string, *satisfaction*: string)
Taught_by (*course_id*: integer, *professor_id*: integer, *year*: string, *assessrate*: string)

Table I
A RELATIONAL SCHEMA FOR A UNIVERSITY MODEL.

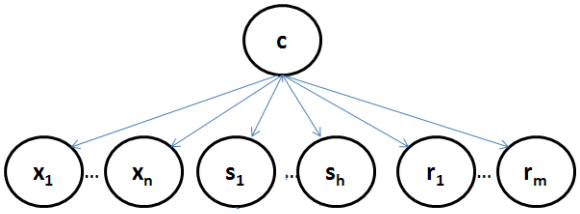

Figure 2. The Bayesian network attributes dependency model. Given the class label, the attributes of the target entity x_i, attributes of the linked entity s_k, and the attributes of the link r_j are independent of each other.

denoted as s_u. The attributes of the last link l_{m_u} in the foreign key path are denoted by $r_{u1}, \ldots, r_{um_u}$. Figure 3 illustrates an instance of a foreign key path that has length 2. For brevity, we sometimes refer simply to foreign key paths rather than instances of foreign key paths.

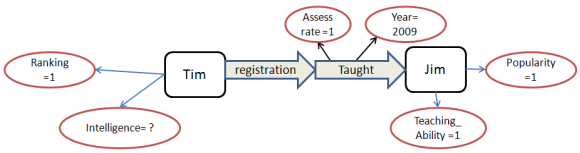

Figure 3. An (instance of) a foreign key path of length 2.

The generalization of equation (2) for foreign key path u is

$$P(c|u) = P(c|x_1, \ldots, x_n, r_{u1}, \ldots, r_{um_u}, s_{u1}, \ldots, s_{uh_u}) \propto P(c) \prod_{i=1}^{n} P(x_i|c) \prod_{j=1}^{m_u} P(r_{uj}|c) \prod_{k=1}^{h_u} P(s_{uk}|c). \quad (3)$$

B. Combining Influences of Related Objects

To combine information from different foreign key paths, we assume *object-path* independence, meaning that the influence of one path from the target object to a linked entity is independent of another. This assumption can be conceptualized as repeating the object we want to classify with each foreign key path and considering each "object-path" combination independently of each other. Figure 4 depicts this assumption for object Tim.

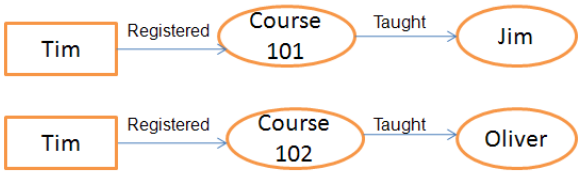

Figure 4. The object-path independence assumption.

Applying this assumption to Equation (3) we obtain our overall relational naive Bayese formula, where l is the number of instances of foreign key paths for the target entity.

$$P(c) \propto \prod_{u=1}^{l} P(c|u) = \prod_{u=1}^{l} \left(P(c) \prod_{i=1}^{n} P(x_i|c) \prod_{j=1}^{m_u} P(r_{uj}|c) \prod_{k=1}^{h_u} P(s_{uk}|c) \right) = \left(P(c) \prod_{i=1}^{n} P(x_i|c) \right)^{l} \prod_{u=1}^{l} \left(\prod_{j=1}^{m_u} P(r_{uj}|c) \prod_{k=1}^{h_u} P(s_{uk}|c) \right) \quad (4)$$

We refer to the classification method that selects a class label by maximizing the expression in Equation (4) as the **LNBC**. The LNBC method models the dependence between class label and link structure with the Naive Bayes assumption that different foreign key paths are independent given the class label.

Comparison with Graph-NB: [11] applied a different formulation of the Naive Bayes assumption for the Graph-NB method. In our notation the Graph-NB formula is as follows.

$$\operatorname*{argmax}_c P(c) \prod_{i=1}^{n} P(x_i|c) \prod_{u=1}^{l} \left(\prod_{j=1}^{m_u} P(r_{uj}|c) \prod_{k=1}^{h_u} P(s_{uk}|c) \right) \quad (5)$$

There is a main difference with our LNBC equation (4). The Graph-NB formula assumes independence of links but not of object-link pairs. This is illustrated by the difference between Figure 4 and Figure 5.

Formula 4 indicates that the prior probability $P(c)$ and probabilities of attributes of the object $P(x_i|c)$ are repeated

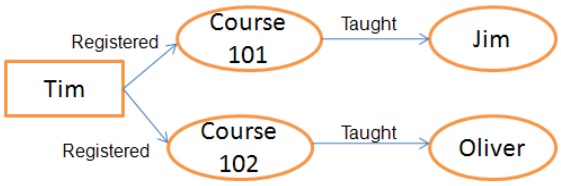

Figure 5. The link independency assumption of the Graph-NB formula.

by the number of foreign key path instances. This means that their influence is much larger compared to Graph-NB. In Graph-NB, these probabilities play minor roles when the object has many links, and attributes of links and related entities dominate the final decision, although more distant attributes usually carry less information compared to target class attributes.

V. Evaluation

We conduct experiments on three relational datasets to show the performance of the proposed method:

To evaluate the method the following metrics are used:

- Precision: shows how precise is a classifier in locating positive instances
- Recall: shows how thorough is the coverage of positive instances
- Accuracy: shows how much is the overall correctness of the model
- F-measure: shows harmonic mean of precision and recall

We performed the experiments on the following three real world databases.

A. Dataset

Mutagenesis Database. This dataset is widely used in the ILP area. We use the *regression-friendly* dataset. It contains 4 tables totals to 15218 tuples. Mutagenesis has two entity tables: $Mole$ with 188 instances, and $Atom$ with 4893 kind of atoms, and two relationships: $Moleatm$ which shows the atoms composing a mole with 4893 tuples and $Bond$ which shows the bonds between atoms with 5244 tuples. There is a cycle between table $Atom$ and $Bond$. To eliminate the cycle, table $Atom$ is duplicated. The schema of the dataset with duplicate table $Atom$ is shown in the figure 6. The target table Mole has 125 Mutagenes moles with positives Labels(the class attribute), and 63 non Mutagenes moles with negatives labels.

We conduct two experiments on this dataset based on the two levels of background knowledge : BK_0 and BK_2. In BK_0, the descriptive attributes of the target table(Mole) are not considered in the experiment, so for each target object, we just consider atoms, bonds, bond types, atom types, and partial charges on atoms. In BK_2, we classify objects based on all the attributes. In order to compare the performance of the proposed algorithm with other methods for which experimental results are available in the literature, a 10-fold cross-validation was performed, and continuous properties lumo, logp, and charge were discretisized into 10 equal size intervals(i.e., each interval contains the same number of individuals).

MovieLens Database. This dataset is drawn from the UC Irvine machine learning repository. It contains two entity tables: $User$ with 941 tuples and $Item$ with 1,682 tuples, and one relationship table $Rated$ with 80,000 ratings. The $User$ table has 3 descriptive attributes $age, gender, occupation$. We discretized the attribute age into three bins with equal frequency. The table $Item$ represents information about the movies. It has 17 Boolean attributes that indicate the genres of a given movie; a movie may belong to several genres at the same time. For example, a movie may have the value T for both the war and the $action$ attributes. For classification task, we choose Gender of $User$ as a class label $(P(Gender = F) = 0.3)$. Evaluation is done by five-fold cross-validation.

Cora Database. This data set is drawn from Cora, a database of computer science research papers extracted automatically from the web using machine learning techniques [13]. It contains one entity table: $Paper$ with 2200 tuples, and one relationship table $citation$ with 5400 citations. The $Paper$ table has a descriptive attribute $topic$ and a bag of words indicating words that have occurred in the paper. Table $citation$ only contains the id of paper which is citing and the id of paper which is cited. To remove the cycle we duplicate table $Paper$. We select papers in machine-learning area and for classification task, we identify whether paper topic is Neural Networks $(P(+) = 0.32)$.

B. Results

Figure 7 shows the accuracy of the proposed algorithm LNBC, and Graph-NB for the three datasets. On the Cora dataset, the accuracy by our algorithm is improved by 2%, and on the MovieLens, LNBC also shows accuracy improvement by 10%. The accuracy on Mutagenesis is increased by about 4%.

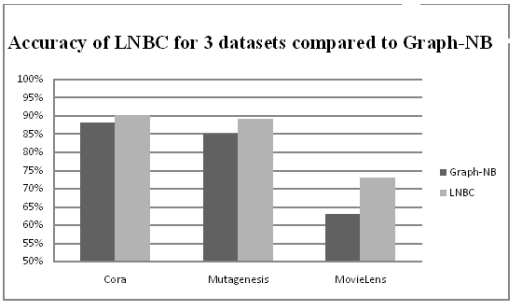

Figure 7. The accuracy results

To observe the effect of weighting the target object

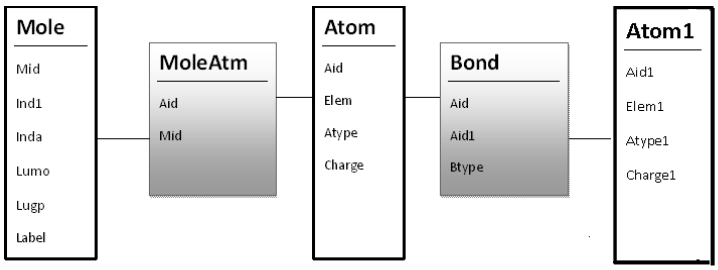

Figure 6. The relational schema of Mutagenesis dataset.

attributes, we conducted experiments on mutagensis with 2 levels of background knowledge. In BK_0 the target object does not have any descriptive attributes, so the difference of the LNBC and Graph-NB is just on the weight of the prior class probability. We compare the performance of LNBC on Mutagenesis dataset with other methods for which experimental results are available in the literature. In Figures 8, 9 results for Graph-NB are taken from [11]. Results for Cross-Mine are taken from [16]. Results for FOIL, TILDE are taken from [2]. Results for 1BC and 1BC2 are taken from [6]. The results on BK_0 provided in Figure 9 show

Algorithm	Accuracy
LNBC	89.2%
Graph-NB	85%
CrossMine	88.8%
Foil	61%
TILDE	85%
1BC	87.2%
1BC2	72.9%

Figure 8. Comparison of accuracy results on Mutagenesis dataset on BK-2

that the accuracy of LNBC is inefficient compared to other algorithms. The reason behind that is just weighting the prior makes the results biased towards the most probable class label. So LNBC can work well on datasets in which the target object has descriptive attributes. Figure 10 shows the Precision, Recall, and F-measure of LNBC and RNBC for the three datasets.

VI. Conclusion

A goal of relational classification is to make predictions that utilize information not only about the target table but also about related objects. Naive Bayes Classifiers are a well-established predictive method for propositional single table data. We proposed a new way of utilizing the independence

Algorithm	Accuracy
LNBC	70%
Graph-NB	77%
CrossMine	68.8%
Foil	61%
TILDE	75%
1BC	71%
1BC2	73%

Figure 9. Comparison of accuracy results on Mutagenesis dataset on BK-0

	Precision	Recall	F-measure
Graph-NB	0.98	0.84	0.90
LNBC	0.96	0.89	0.92

(a) Mutagenesis results

	Precision	Recall	F-measure
Graph-NB	0.78	0.73	0.754
LNBC	0.73	0.93	0.82

(b) MovieLens results

	Precision	Recall	F-measure
Graph-NB	0.759	0.789	0.769
LNBC	0.835	0.736	0.776

(c) Cora results

Figure 10. The Precision, Recall,F-measure of LNBC and Graph-NB for three datasets.

assumptions of the Naive Bayes Classifier in link-based classification. The key distinguishing features of our method are: (1) The influence of features of an object that is related to the target object by a link chain is computed using the Naive Bayes independence assumptions. (2) The features of the link chain may also be taken into consideration. (3) If an object is related to the target object by more than one link chain, the influence of the object is increased; each link chain-object pair is counted separately. The main effect of this scheme is to place higher weight on the probabilities of the class labels and the features of the target entities. Empirical evaluation on three real-world datasets showed improved predictive performance when the target entity had descriptive attributes or features. A task for future work is to add additional refinements of relational Bayes learning such as (1) pruning of uninformative features, (2) data-sensitive discretization of continuous data, and (3) considering explicitly the probability of a foreign key path as in [3].

REFERENCES

[1] Jacob Abernethy, Olivier Chapelle, and Carlos Castillo. Web spam identification through content and hyperlinks. In *AIRWeb '08: Proceedings of the 4th international workshop on Adversarial information retrieval on the web*, pages 41–44, New York, NY, USA, 2008. ACM.

[2] Hendrik Blockeel and Luc De Raedt. Top-down induction of first-order logical decision trees. *Artificial Intelligence*, 101(1-2):285–297, 1998.

[3] Michelangelo Ceci, Annalisa Appice, and Donato Malerba. Mr-sbc: A multi-relational naïve bayes classifier. In *PKDD*, pages 95–106, 2003.

[4] Pedro Domingos and Michael Pazzani. Beyond independence: Conditions for the optimality of the simple bayesian classifier. In *Machine Learning*, pages 105–112. Morgan Kaufmann, 1996.

[5] Richard O. Duda and Peter E. Hart. *Pattern Classification and Scene Analysis*. John Wiley & Sons Inc, 1973.

[6] Peter A. Flach and Nicolas Lachiche. Naive bayesian classification of structured data. *Mach. Learn.*, 57(3):233–269, 2004.

[7] Getoor and Diehl. Link mining:a survey. In *SIGKDD*, 2005.

[8] George Harrison John. *Enhancements to the data mining process*. PhD thesis, Stanford, CA, USA, 1997.

[9] Stefan Kramer, Nada Lavravc, and Peter Flach. Propositionalization approaches to relational data mining. pages 262–286, 2000.

[10] Wim Van Laer and Luc de Raedt. How to upgrade propositional learners to first-order logic: A case study. In *Relational Data Mining*. Springer Verlag, 2001.

[11] Hongyan Liu, Xiaoxin Yin, and Jiawei Han. An efficient multi-relational naïve bayesian classifier based on semantic relationship graph. In *MRDM '05: Proceedings of the 4th international workshop on Multi-relational mining*, pages 39–48, New York, NY, USA, 2005. ACM.

[12] Qing Lu and Lise Getoor. Link-based classification. In Tom Fawcett, Nina Mishra, Tom Fawcett, and Nina Mishra, editors, *ICML*, pages 496–503. AAAI Press, 2003.

[13] Andrew Mccallum, Kamal Nigam, Jason Rennie, and Kristie Seymore. A machine learning approach to building domain-specific search engines. In *In Proceedings of the 16th International Joint Conference on Artificial Intelligence*, pages 662–667, 1999.

[14] Jennifer Neville, David Jensen, and Brian Gallagher. Simple estimators for relational bayesian classifiers. *Data Mining, IEEE International Conference on*, 0:609, 2003.

[15] J. D. Ullman. *Principles of database systems*. 2. Computer Science Press, 1982.

[16] X. Yin, J. Han, J. Yang, and P.S. Yu. Crossmine: Efficient classification across multiple database relations. In *Proceedings of ICDE*, 2004.

[17] Xiaojin Zhu Zhuxj, Zoubin Ghahramani, and John Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *In ICML*, pages 912–919, 2003.