Engineering Applications of Artificial Intelligence 22 (2009) 109–116

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

A relational approach to probabilistic classification in a transductive setting

Donato Malerba *, Michelangelo Ceci, Annalisa Appice

Dipartimento di Informatica, Università degli Studi di Bari, Via Orabona 4, I-70126 Bari, Italy

ARTICLE INFO

Article history:
Received 21 April 2008
Accepted 28 April 2008
Available online 24 June 2008

Keywords:
Multi-relational data mining
Transductive classification
Relational probabilistic classification
Relational learning
Transduction

ABSTRACT

Transduction is an inference mechanism adopted from several classification algorithms capable of exploiting both labeled and unlabeled data and making the prediction for the given set of unlabeled data only. Several transductive learning methods have been proposed in the literature to learn transductive classifiers from examples represented as rows of a classical double-entry table (or relational table). In this work we consider the case of examples represented as a set of multiple tables of a relational database and we propose a new relational classification algorithm, named TRANSC, that works in a transductive setting and employs a probabilistic approach to classification. Knowledge on the data model, i.e., foreign keys, is used to guide the search process. The transductive learning strategy iterates on a *k*-NN based re-classification of labeled and unlabeled examples, in order to identify borderline examples, and uses the relational probabilistic classifier Mr-SBC to bootstrap the transductive algorithm. Experimental results confirm that TRANSC outperforms its inductive counterpart (Mr-SBC).

1. Introduction

During recent years, there has been a growing interest in learning algorithms capable of utilizing both labeled and unlabeled data for prediction tasks, such as classification. The reason for this attention is the cost of assigning labels which can be very high for large datasets. Two main settings have been proposed in the literature to exploit information contained in both labeled and unlabeled data: the *semi-supervised setting* and the *transductive setting* (Seeger, 2000). The former is a type of inductive learning, since the learned function is used to make predictions on any possible example. The latter asks for less—it is only interested in making predictions for the given set of unlabeled data. Since transduction needs no general hypothesis, it appears to be an easier problem than (semi-supervised) induction and it is likely to become much more popular in the future.

Several transductive learning methods have been proposed in the literature for support vector machines (Bennett, 1999; Gammerman et al., 1998; Joachims, 1999; Chen et al., 2003), for *k*-NN classifiers (Joachims, 2003) and even for general classifiers (Kukar and Kononenko, 2002). However, all of these transductive learning algorithms assume (un)labeled input examples are represented as rows of a classical double-entry table (or database relation), whose columns correspond to elementary (nominal, ordinal or numeric) single-valued attributes. This tabular representation of data, also known as *propositional* or *feature-vector* representation, turns out to be too restrictive for several applications, whose units of analysis have quite a complex structure, composed of several related objects with different properties. These units of analysis can be naturally modeled as a set of tables $T_1, \ldots, T_n$, such that each table T_i describes a specific type of objects involved in the units of analysis, while foreign key constraints explicitly model relationships between objects. Units analyzed by spatial data mining algorithms provide us with a clear example of such complex structures: objects of different types are organized in separate *layers*, i.e., database relations with their own distinct sets of attributes (including a geometry attribute), and locational properties of objects implicitly define several spatial relationships (e.g., topological).

To analyze these complex units of analysis, several (*multi-*) *relational data mining* (MRDM) methods have been reported in the literature (Džeroski and Lavrač, 2001). They can be applied directly to data distributed over several relations to find relational patterns which involve multiple relations. Relational patterns can be expressed not only in SQL, but also in first order logic (or predicate calculus), which explains why many MRDM algorithms originate from the field of inductive logic programming (ILP) (Muggleton, 1992; De Raedt, 1992; Lavrač and Džeroski, 1994).

Two MRDM methods have been reported in the literature for prediction in a transductive setting. Krogel and Scheffer (2004) investigate a transformation (known as *propositionalization*) of a relational description of gene interaction data into a classical

* Corresponding author. Tel./fax: +39 80 5443269.
E-mail addresses: malerba@di.uniba.it (D. Malerba), ceci@di.uniba.it (M. Ceci), appice@di.uniba.it (A. Appice).

0952-1976/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.engappai.2008.04.005

double-entry table, and then study transduction with the well-known transductive support vector machines. Therefore, transduction is not explicitly investigated on relational representations and it is based on propositionalization, which is fraught with many difficulties in practice (De Raedt, 1998; Getoor, 2001). Taskar et al. (2001) build, on the framework of Probabilistic Relational Models, a generative probabilistic model that captures interactions between examples, some of which have class labels, while others do not. However, given sufficient data, a discriminative model generally provides significant improvements in classification accuracy over generative models (Vapnik, 1995). Therefore, we intend this work to be a further step toward the investigation of methods which originate from the intersection of these two promising research areas, namely transduction and relational data mining. In particular, by developing transductive classification algorithms which work on relational data and are based on a discriminative model, we aim to extend the benefits of using unlabeled data to a wide range of applications characterized by complex units of analysis.

A further motivation for this study is related to the contiguity of two important concepts which characterize studies on transductive learning and relational learning, namely the *smoothness assumption* and the *relational correlation*. The former is typical of semi-supervised learning and refers to the fact that if two points x_1 and x_2 in a high-density region are close, then the corresponding outputs y_1 and y_2 should also be so (Chapelle et al., 2006a). Although still debated, this assumption seems to be fundamental also in the transductive setting (Chapelle et al., 2006b). In relational learning, *relational autocorrelation* refers to the fact that the values of a given attribute are highly uniform among objects that share a common neighbor. As reported in Jensen and Neville (2002), a high relational autocorrelation seems to characterize several relational learning problems reported in the literature. Our intuition is that when a high relational autocorrelation affects the dependent variable, the semi-supervised smoothness assumption is likely to hold and the transductive setting can return better results than the inductive setting. This intuition is clearer in spatial domains, where closeness of points corresponds to a spatial distance measure and relational autocorrelation is a manifestation of the (positive) spatial autocorrelation. To corroborate our insight we observe the interesting convergence of two opinions: transduction is most useful when the standard i.i.d. assumption is violated (Chapelle et al., 2006a), and statistical independence of examples is contradicted by many relational datasets (Jensen and Neville, 2002).

Upgrading transductive classification algorithms devised for double-entry tabular data to multi-relational data is not a trivial task. First, we have to choose a strategy for the classification of unlabeled data and make it suitable for relational data. Second, we have to choose a classifier which can handle relational data and is based on a discriminative model. Third, we have to define a distance measure between examples described by several relations. Solutions to these issues are described in the following sections. They have been implemented in a new relational classification algorithm, named TRANSC (TRANsductive Structural Classifier), which exploits knowledge on the data model, namely foreign keys, to guide the search process. TRANSC works in a transductive setting and employs a probabilistic approach to classification. Information on the potential uncertainty of classification conveyed by probabilistic inference is useful when small changes in the attribute values of a test case may result in sudden changes of the classification. It is also useful when missing (or imprecise) information may prevent a new object from being classified at all.

This paper presents an extension of the preliminary work reported in Ceci et al. (2007). It is organized as follows. In the next section, the transductive learning strategy is described. It is based on an iterative k-NN based re-classification of training and working examples which aims at identifying ‘‘borderline'' examples, i.e., examples for which the classification is more uncertain. The relational probabilistic classifier used to bootstrap the transductive algorithm is presented in Section 3, while the relational dissimilarity measure used for k-NN is defined in Section 4. A theoretical analysis of the computational complexity of TRANSC is reported in Section 5. Experimental results are reported and discussed in Section 6. Finally, Section 7 concludes and presents ideas for further work.

2. The transductive learning strategy

Let D be a dataset labeled according to an unknown target function, whose range is a finite set $Y = \{C_1, C_2, \ldots, C_L\}$. Observations in D are described by a set of attributes X and by the attribute Y. The transductive classification problem is formalized as follows:

Input:

- a training set $\mathrm{TS} \subset D$, and
- the projection of the working set $\mathrm{WS} = D - \mathrm{TS}$ on X.

Output: a prediction of the class value of each example in the working set WS which is as accurate as possible.

The learner receives full information (including labels) on the examples in TS and partial information (without labels) on the examples in WS and is required to predict the class values only of the examples in WS. The original formulation of the problem of function estimation in a transductive (*distribution-free*) setting requires TS to be sampled from D without replacement. This means that, unlike the standard inductive setting, the examples in the training (and working) set are supposed to be mutually dependent. Vapnik also introduced a second (*distributional*) transduction setting, in which the learner receives training and working sets, which are assumed to be drawn i.i.d. from some unknown distribution. As shown in Vapnik (1998, Theorem 8.1), error bounds for learning algorithms in the distribution-free setting also apply to the more popular distributional transductive setting. Therefore, in this work we focus our attention on the first setting.

In the case of relational data, the problem of transductive classification can be more precisely formulated as follows:

Given:

- a database schema S which consists of a set of h relational tables $\{T_0, \ldots, T_{h-1}\}$, a set PK of primary keys on the tables in S, and a set FK of foreign key constraints on the tables in S;
- a target relation $T \in S$ and a target discrete attribute Y in T, different from the primary key of T, whose domain is the finite set $\{C_1, C_2, \ldots, C_L\}$;
- the projection T' of T on all attributes of T except Y;
- a training (working) set that is an instance TS (WS) of the database schema S with known (unknown) values for Y.

Find: the most accurate prediction of Y for examples in WS.

This problem is solved by TRANSC by accessing both the full representation of examples in the training set (T and its joined tables) and the partial representation of examples in the working set (T' and its joined tables). Indeed, an example in TS (WS) is represented as one tuple $t \in \mathrm{TS}.T$ ($t \in \mathrm{WS}.T'$) and all tuples related to t in TS (WS) according to FK.

In keeping with the main idea expressed in Joachims (1999), we iteratively refine the classification by changing the class of "borderline" training and working examples, i.e., of those examples whose classification is more uncertain. We propose an algorithm (see Algorithm 1) which starts with a given classification and, at each iteration, alternates a step during which examples are re-classified and a step during which the class of borderline examples is changed.

Algorithm 1. Top level transductive algorithm description

```
transductiveClassifier(initialClassification, TS, WS)
classification1 $\leftarrow$ initialClassification;
changedExamples $\leftarrow \phi$;
i $\leftarrow$ 0;
repeat
    prevClassification $\leftarrow$ classification1;
    prevChangedExamples $\leftarrow$ changedExamples;
    classification2 $\leftarrow$ reclassifyExamplesKNN(classification1, TS, WS);
    (classification1, changedExamples) $\leftarrow$ changeClass(classification2);
until ((++i $\geqslant$ MAX_ITERS) OR (computeOverlap(prevChangedExamples,changedExamples) $\geqslant$ MAXOVERLAP))
return prevClassification
```

The initial classification of $E \in \mathrm{WS} \cup \mathrm{TS}$ is obtained according to the classification function preclass defined as follows:

$$\mathrm{preclass}(E) = \begin{cases} \mathrm{class}(E) & \text{if } E \in \mathrm{TS}, \\ \mathrm{BayesianClassification}(E) & \text{if } E \in \mathrm{WS}, \end{cases}$$

where BayesianClassification(E) is the initial probabilistic classifier built from the training set TS (see next section).

The examples are then re-classified by means of a version of the k-NN algorithm (Mitchell, 1997), tailored for transductive inference in MRDM. The idea is to classify each example $E \in \{\mathrm{TS} \cup \mathrm{WS}\}$ on the basis of a k-sized neighborhood $N_k(E) = \{E_1, \ldots, E_k\}$, consisting of the k examples of $\mathrm{TS} \cup \mathrm{WS}$ closest to E with respect to a dissimilarity measure d. This is obtained by estimating the L-dimensional class probability vector associated to the example E, i.e., $y' = (y_1(E), \ldots, y_L(E))$, where $y_i(E) = P(\mathrm{class}(E) = C_i)$, $P(\mathrm{class}(E) = C_i) \geqslant 0$ for each $i = 1, \ldots, L$, and $\sum_{i=1,\ldots,L} P(\mathrm{class}(E) = C_i) = 1$. More precisely, each $P(\mathrm{class}(E) = C_i)$ is estimated as follows:

$$P(\mathrm{class}(E) = C_i) = \frac{|\{E_j \in N_k(E) | C_{E_j} = C_i\}|}{k}, \tag{1}$$

where C_{E_j} is the class value associated to E_j at the previous step (at the first step, C_{E_j} is the class label returned by preclass(E_j)). It should be noted that $P(\mathrm{class}(E) = C_i)$ is estimated according to the transductive inference principle, as both training and working examples are taken into account in the process.

The changeClass procedure is in charge of changing the classification of the borderline examples. Unlike what was proposed in Joachims (1999), where examples on the border are identified by means of support vectors, we consider the examples for which the entropy of the decision made by the classifier is maximum. The entropy for each example E is computed from the probabilities associated with each class C_i:

$$\mathrm{Entropy}(E) = -\sum_{i=1,\ldots,L} P(\mathrm{class}(E) = C_i) \times \log(P(\mathrm{class}(E) = C_i)). \tag{2}$$

The examples are ordered according to the entropy function and the class label of at most the first k examples, having $\mathrm{Entropy}(E) > \mathrm{MINENTROPY}$, is changed. In particular, each selected example E is assigned the most likely class C_i for E among those remaining after the old class of E has been excluded. The threshold k is the same used for k-NN and is necessary in order to avoid changing the class of several examples that would lead to erroneously changing the class of entire "clusters".

Two distinct stopping criteria are used. The first criterion stops the execution of the algorithm when the maximum number of iterations (MAX_ITERS) is reached. This guarantees the termination of the algorithm. In any case, our experiments showed that this criterion is rarely attained when the parameter MAX_ITERS is as small as 10. The second criterion stops execution when a cycle processes the same examples as the previous one. For this purpose, the overlap between two sets of examples is determined. The computeOverlap function returns the ratio between the cardinality of the intersection between the sets of examples and that of their union.

3. The relational probabilistic classifier

The initial classification of examples in the working set is based on a probabilistic classifier, named Mr-SBC (Ceci et al., 2003), which upgrades the classical naïve Bayes classifier (Domingos and Pazzani, 1997) to multi-relational data. Given an example E, a classical naïve Bayes classifier assigns E to the class C_i that maximizes the *posterior probability* $P(C_i|E)$. By applying the Bayes theorem, $P(C_i|E)$ is expressed as follows:

$$P(C_i|E) = \frac{P(C_i)P(E|C_i)}{P(E)}. \tag{3}$$

Since $P(E)$ is independent of the class C_i, it does not affect the classification, therefore $P(C_i|E) \propto P(C_i)P(E|C_i)$.

The basic idea in Mr-SBC is that of constructing a set of relational patterns (first order definite clauses) $\mathfrak{R}$ to describe the example E, and then using $\mathfrak{R}$ to define a suitable decomposition of the likelihood $P(E|C_i)$ *à la* naïve Bayesian classifier to simplify the probability estimation problem.

The construction of a first order definite clause to be included in $\mathfrak{R}$ is based on the notion of a foreign key path, which is an ordered sequence of tables $\vartheta = \{T_{i_1}, T_{i_2}, \ldots T_{i_s}\}$, such that $T_{i_j} \in S$ ($j = 1 \ldots s$). In its original version, Mr-SBC considers only foreign key paths ϑ, where each table T_{i_j} has a foreign key to table $T_{i_{j-1}}$ ($j = 2, \ldots, s$). In this work, we generalize the definition of foreign key path, in order to take into account the case that a foreign key can be from table T_{i_j} to table $T_{i_{j-1}}$ or vice versa.

To formally define the set of first order definite clauses $\mathfrak{R}$, some definitions have to be introduced.

Definition 1 (*Structural predicate*). A binary predicate p is a structural predicate associated to the pair of tables $\langle T_i, T_j \rangle \in S$ if a foreign key in T_i exists which references a table $T_j \in S$, or vice versa. The first argument of p represents the primary key of T_j and the second argument represents the primary key of T_i, or vice versa.

Definition 2 (*Property predicate*). A binary predicate p is a property predicate associated to a table $T_i \in S$ and an attribute Att of T_i if the first argument of p represents the primary key of T_i and the second argument represents a value observed for Att in T_i. The attribute Att is neither the primary key of T_i nor a foreign key in T_i.

Hence, we can formally define a first order definite clause to be associated to a foreign key path as follows:

Definition 3 (*First order definite clause based on a foreign key path*). A first order definite clause associated to the foreign key path ϑ is a clause in the form:

$$p_0(A_1, y) \leftarrow p_1(A_1, A_2), p_2(A_2, A_3), \ldots, p_{s-1}(A_{s-1}, A_s), p_s(A_s, c)$$

or

$$p_0(A_1, y) \leftarrow p_1(A_1, A_2), p_2(A_2, A_3), \ldots, p_{s-1}(A_{s-1}, A_s),$$

where

(1) p_0 is a property predicate associated to both the target table T and the target attribute Y.
(2) $\vartheta = \{T_{i_1}, \ldots, T_{i_{s-1}}\}$ is a foreign key path such that for each $k = 1, \ldots, s-1$: p_k is a structural predicate associated to the pair of tables $\langle T_{i_{k-1}}, T_{i_k} \rangle$ of ϑ. p_1 is associated to the target table T and table T_{i_1} of ϑ.
(3) p_s is an optional property predicate associated to both table $T_{i_{s-1}}$ and an attribute Att of $T_{i_{s-1}}$.

Mr-SBC constructs $\Re$ by searching the first order definite clauses R_i based on a foreign key path ϑ, such that the antecedent of R_i covers at least one training example E. This means that given the first order definite clause R_i defined as follows:

$$R_i : p_0(A_1, y) \leftarrow p_{i_1}(A_1, A_2), p_{i_2}(A_2, A_3), \ldots, p_{i_{s-1}}(A_{s-1}, X_s), p_{i_s}(A_s, c),$$

R_i is introduced in $\Re$ if a training example E and a substitution θ exist, such that

$$\{p_{i_1}(A_1, A_2), p_{i_2}(A_2, A_3), \ldots, p_{i_{s-1}}(A_{s-1}, X_s), p_{i_s}(A_s, c)\}\theta \subseteq E.$$

The length of the foreign key path ϑ is less than or equal to a user-defined maximum length (MAX_LENGTH_PATH). The property predicate p_{i_s} is associated to either a discrete attribute or a continuous attribute Att of the table $T_{i_{s-1}}$. In the former case, p_{i_s} checks a condition in the form "Att $= v$", where v is a value in the range of Att, while in the latter case, p_{i_s} checks a condition in the form "Att $\in [v_1, v_2]$", where $[v_1, v_2]$ is a bin obtained by means of a discretization of Att based on an equal-width strategy. Indeed, a continuous attribute is discretized into Nb bins, where Nb is a user-defined parameter.

If $\Re(E) \subseteq \Re$ is the set of first order definite clauses, whose antecedent covers the example E, then the probability $P(E|C_i)$ is defined as

$$P(E|C_i) = P\left(\bigwedge_{R_j \in \Re(E)} \text{antecedent}(R_j) \middle| C_i \right). \quad (4)$$

The straightforward application of the naïve Bayes independence assumption to all literals in $\bigwedge_{R_j \in \Re(E)}$ antecedent(R_j) is not correct, since it may lead to underestimating $P(E|C_i)$ when several similar clauses in $\Re(E)$ are considered for the class C_i. Therefore, in this study, we employ a less biased procedure for the computation of the probabilities in Eq. (4), namely that adopted in the multi-relational naïve Bayesian classifier Mr-SBC (Ceci et al., 2003).

4. The relational dissimilarity measure

The re-classification of training and working examples is based on a dissimilarity measure d. The classical k-NN method assumes that examples correspond to points in the m-dimensional space $\mathbb{R}^m$ and the nearest neighbors of the example to classify are defined in terms of the standard Euclidean distance. However, in our multi-relational transductive formulation, examples cannot be associated to points of $\mathbb{R}^m$. This motivates the need for a different notion of a distance (dissimilarity) measure that applies to relational data.

TRANSC computes the dissimilarity between each pair of examples E_1 and E_2 by first converting the first order definite clauses, discovered by Mr-SBC, into a set of Boolean features and then using these features as input of some propositional dissimilarity measure.

The set of first order definite clauses extracted by Mr-SBC should be transformed before converting into Boolean features. This transformation involves only the first order definite clauses $R_i \in \Re$, whose property predicate p_{i_s} is defined on a continuous attribute Att. In this case, p_{i_s} models the condition $T_{i_{s-1}}$.Att $\in [v_1, v_2]$. However, this representation may cause information loss on the order relation of continuous values. To overcome this problem, we follow the idea formulated in Esposito et al. (2000) and transform the rule R_i into R_i', such that:

(1) $p_{i_j}' = p_{i_j}$, for each $j = 0, \ldots, s-1$, while
(2) the property predicate p_{i_s}' expresses the condition $T_{i_{s-1}}$.Att$\leqslant v_2$.

The advantage of R_i' with respect to R_i is that R_i' models as closer two examples E_1 and E_2, whose Att values belong to two consecutive bins, rather than two examples, whose Att values belong to distant bins.

Once the new set $\Re' = \{R_i'\}$ is constructed, Boolean features are derived from $\Re'$, in order to represent examples by means of a single relational table V. This is a form of propositionalization (Krogel et al., 2003) which allows us to use dissimilarity measures defined for classical propositional representations. The schema of V includes $|\Re'|$ attributes, that is, $V_1, \ldots, V_{|\Re'|}$: one attribute V_i for each first order definite clause R_i'. Each row of V corresponds to an example $E \in \{\text{TS} \cup \text{WS}\}$. If the antecedent of the first order clause R_i' *covers* E, then the i-th value of the row in V corresponding to E is set to true, false otherwise.

Example 1. Let us consider the example E:

$$\text{mutagenecity}(m, \text{yes}), \text{logp}(m, \text{true}), \text{atom}(m, a1), \text{atom}(m, a2), \text{atom}(m, a3), \text{charge}(a1, 0.2), \text{charge}(a2, 0.7), \ldots,$$

which describes a molecule m in terms of the "logP" property and the "mutagenicity" degree. The molecule is composed of one or more atoms and each atom is described by the "charge". This relational description of m can be converted into the Boolean vector (true, false), if $\Re = \{R_1, R_2\}$, where

$$R_1 : \text{molecule\_mutagenecity}(M, \text{true}) \leftarrow \text{atom}(M, A), \text{charge}(A, [0.5, 0.8]),$$

$$R_2 : \text{molecule\_mutagenecity}(M, \text{false}) \leftarrow \text{logp}(M, \text{false}).$$

Indeed, there is a substitution $\theta = \{M \leftarrow m, A \leftarrow a2\}$, such that antecedent$(R_1)\theta \subseteq E$, while there is no substitution θ, such that antecedent$(R_2)\theta \subseteq E$.

Finally, the similarity between the pair of examples E_1 and E_2 can be computed by means of the Kendall, Sokal–Michener similarity measure (Esposito et al., 2000), computed on their feature vector representation $V(E_1)$ and $V(E_2)$ stored in V, that is:

$$s(E_1, E_2) = \frac{\text{cardinality}(V(E_1)\ XNOR\ V(E_2))}{|\mathfrak{R}'|}, \tag{5}$$

where cardinality(•) returns the number of true values occurring in the Boolean vector describing the example in V. The similarity coefficient computed in Eq. (5) takes values in the unit interval: $s(E_1, E_2) = 1$, if the two vectors match perfectly, while $s(E_1, E_2) = 0$, if the two vectors are orthogonal. The dissimilarity between the pair of examples $\langle E_1, E_2\rangle$ is then computed as

$$d(E_1, E_2) = 1 - s(E_1, E_2). \tag{6}$$

5. Learning complexity

The time complexity of the TRANSC strongly depends on the number of rules ($|\mathfrak{R}|$) learned by Mr-SBC and the maximum number of literals in a single rule (MAX_LENGTH_PATH). In particular, the construction of the V table is preliminar to the execution of Algorithm 1 and, in the worst case, its time complexity is

$$O(\underbrace{a_{\mathrm{TW}}^{p}}_{\text{joins cost}} \cdot |\mathfrak{R}|), \tag{7}$$

where

- $p = \text{MAX\_LENGTH\_PATH}$;
- a_{TW} is the number of tuples in a single table of $\mathrm{TS} \cup \mathrm{WS}$ (for simplicity, we suppose that all tables have the same number of tuples).[1]

The time complexity of Algorithm 1 is

$$O\left(\left(\underbrace{a_{\mathrm{TW}} \cdot a_{\mathrm{TW}} \cdot |\mathfrak{R}|}_{\text{reclassifyExamplesKNN}} + \underbrace{a_{\mathrm{TW}} \cdot L}_{\text{changeClass}}\right) \cdot \text{MAX\_ITERS}\right) = O(a_{\mathrm{TW}}^2 \cdot |\mathfrak{R}|), \tag{8}$$

where L is the number of classes.

By combining (8) and (7), the complexity of TRANSC is

$$O(\text{Mr-SBC\_complexity} + a_{\mathrm{TW}}^{p} \cdot |\mathfrak{R}|), \tag{9}$$

under the reasonable condition that $\text{MAX\_LENGTH\_PATH} \geqslant 2$ ($p \geqslant 2$).

It can be proved that, in the worst case, the time complexity of Mr-SBC is

$$O(a_{\mathrm{TS}}^{p} \cdot L \cdot b^{(p-1)} \cdot s \cdot v), \tag{10}$$

where

- a_{TS} is the number of tuples in a single relational table of TS (for simplicity, we suppose that a_{TS} is constant),
- b is the number of tables directly related to a table by means of foreign keys (for simplicity, we suppose that b is constant),
- s is the number of attributes per table (for simplicity, we suppose that s is constant),
- v is the number of distinct values per attribute (for simplicity, we suppose that v is constant).

Since $|\mathfrak{R}|$ in the worst case is $|\mathfrak{R}| = L \cdot b^{(p-1)} \cdot s \cdot v$, we have that the complexity of TRANSC increases the complexity of Mr-SBC of a factor $(a_{\mathrm{TW}}/a_{\mathrm{TS}})^p$.

6. Experiments

An empirical evaluation of our algorithm was carried out on both the Mutagenesis dataset, which has been used to test several MRDM algorithms, and on two real-world spatial data collections concerning North West England (NWE) Census Data and Munich Census Data, respectively.

We compared the performance of TRANSC to that of Mr-SBC in order to identify the advantages of employing a transductive reformulation of the problem of relational probabilistic classification in real-world applications, where few labeled examples are available and manual annotation is fairly expensive.

The two algorithms are compared on the basis of the average misclassification error on the same M-fold cross-validation (CV) of each dataset. For each dataset, the target table is first divided into M blocks of nearly equal size and then a subset of tuples related to the tuples of the target table block is extracted by means of foreign key constraints. In this way, M database examples are created. For each trial, both TRANSC and Mr-SBC are trained on a single database instance and tested on the hold-out $M-1$ database instances, forming the working set. It should be noted that the error rates reported in this work are significantly higher than those reported in other literature (Ceci et al., 2003; Ceci and Appice, 2006) because of this peculiar experimental design. Indeed, unlike the standard CV approach, here one fold at a time is set aside to be used as the *training set* (and not as the *test set*). Small training set sizes allow us to validate the transductive approach, but result in high error rates as well.

Since the performance of the transductive classifier TRANSC may vary significantly, depending on the size (k) of the neighborhood used to predict the class value of each working example, experiments for different k values are performed in order to set the optimal value. In theory, we should experiment with each value of k ranging in the interval $[1, N]$, where $N = |\mathrm{TS}.T| + |\mathrm{WS}.T'|$. However, as observed in Wettschereck (1994), it is not necessary to consider all possible values of k during CV to obtain the best performance. This can be well approximated by means of CV on no more than about 10 values of k. A similar consideration has also been reported in Gora and Wojna (2002), where it is shown that the search for the optimal k can be substantially reduced from $[1, N]$ to $[1, \sqrt{N}]$, without too much inaccuracy in the approximation. Hence, we have decided to consider in our experiments only $k \in \{\eta i | i = 1, \ldots, q\}$, where $\eta = \sqrt{N}/q$ is the step value and q is the number of steps.

Classifiers mined in all experiments in this study are obtained by setting the following parameters:

- MAX_LENGTH_PATH = 3, coherently with the length of sensible foreign key paths in the selected databases;
- MAX_ITERS = 10, since experiments showed that the number of iterations is almost always less than 10;
- MINENTROPY = 0.65, since all the experiments are performed on two-class problems for which entropy varies in the unit interval;
- MAXOVERLAP = 0.5, since it is the middle value of the range of possible values $[0, 1]$;
- $q = 5$, since no more than approximately 10 values of k are necessary.

6.1. Benchmark relational data application

The Mutagenesis dataset concerns the problem of identifying some mutagenic compounds. We have considered, similarly to most experiments on data mining algorithms reported in literature, the "regression friendly" dataset consisting of 188

[1] For the sake of notational simplicity, we assume that $\mathrm{TW} = \mathrm{TS} \cup \mathrm{WS}$ is an instance of the database schema S, built by taking the union of the corresponding tables in TS and WS.

Table 1
Background knowledge for Mutagenesis data

Background	Description
BK_0	Data obtained with the molecular modeling package QUANTA. For each compound it obtains the atoms, bonds, bond types, atom types, and partial charges on atoms
BK_1	Definitions in BK_0 plus indicators *ind1* and *inda* in molecule table
BK_2	Variables (attributes) logp and lumo are added to definitions in BK_1

molecules. A study on this dataset (Srinivasan et al., 1999) has identified five levels of background knowledge. Each subset is constructed by augmenting a previous subset and provides richer descriptions of the examples. Table 1 shows the first three sets of background knowledge, the ones we have used in our experiments, where $BK_i \subset BK_{i+1}$ for $i = 0, 1$. The larger the background knowledge set, the more complex the learning problem. Experiments are run according to the 10-fold CV framework ($M = 10$).

The predictive accuracy of TRANSC was measured by considering the values $k \in \{2, 5, 8, 10, 13\}$. For each setting BK_i ($i = 0, 1, 2$), the average misclassification error of both TRANSC and Mr-SBC is reported in Table 2.

Results clearly show that TRANSC performs better than Mr-SBC in all settings. This is more evident for the most complex setting (BK_2). Another observation concerns the sensitivity of results to the k value. In particular, it can be noted that accuracy increases with high values of k, but at the same time accuracy decreases when k approximates $\sqrt{N}$. In addition, the accuracy is also affected by the number of bins in the Mr-SBC discretization. From Table 3 we can observe that better performances of TRANSC and Mr-SBC are obtained for Nb = 10 and 15 when discretization permits a good compromise between specificity and generality. However, it is interesting to notice that the percentage of error reduction of TRANSC increases with Nb = 20. This means that TRANSC is less sensitive to discretization than Mr-SBC. This is probably due to a flattening of probabilities in Mr-SBC when intervals are small and attribute values do not permit discrimination between classes.

Table 2
TRANSC vs. Mr-SBC on Mutagenesis data: average misclassification error on the working sets

Experiment	TRANSC					Mr-SBC (%)
	$k = 2$ (%)	$k = 5$ (%)	$k = 8$ (%)	$k = 10$ (%)	$k = 13$ (%)	
Avg. BK_0 error	21.16	20.27	22.27	22.77	23.75	24.40
%error reduction	13.27	16.90	8.71	6.66	2.66	
Avg. BK_1 error	23.75	22.27	22.75	22.75	24.27	24.42
%error reduction	2.77	8.80	6.86	6.86	0.61	
Avg. BK_2 error	16.67	16.58	16.58	17.67	20.31	23.33
%error reduction	28.57	28.95	28.95	24.29	12.97	

The number of bins (Nb) for the discretization is set to 15.

Table 3
TRANSC vs. Mr-SBC on Mutagenesis (BK_2) data: working set results varying number of bins (Nb) in Mr-SBC discretization

	TRANSC					Mr-SBC (%)
	$k = 2$ (%)	$k = 5$ (%)	$k = 8$ (%)	$k = 10$ (%)	$k = 13$ (%)	
Error						
Nb = 5	20.75	19.16	20.77	20.75	24.25	23.42
Nb = 10	17.61	17.63	17.63	19.63	21.75	22.92
Nb = 15	16.67	16.58	16.58	17.67	20.31	23.33
Nb = 20	16.58	17.11	17.69	18.28	18.81	24.92
%error reduction						
Nb = 5	11.43	18.21	11.31	11.43	−3.52	
Nb = 10	23.20	23.08	23.08	14.36	5.13	
Nb = 15	28.57	28.95	28.95	24.29	12.97	
Nb = 20	33.48	31.37	29.01	26.65	24.53	

6.2. *Spatial data applications*

We have also tested TRANSC on two different spatial data collections, that is, the NWE Census Data and the Munich Census Data.

The NWE Census Data are obtained from both census and digital map data provided by the European project SPIN![2] These data concern Greater Manchester, one of the five counties of NWE. Greater Manchester is divided into 10 metropolitan districts, each of which is in turn divided into censual sections (wards), for a total of 214 wards. Census data are available at ward level and provide socio-economic statistics (e.g., mortality rate—the percentage rate of deaths with respect to the number of inhabitants) as well as some measures of the deprivation of each ward according to information provided by the Census, combined into single index scores. We have employed the Jarman Underprivileged Area Score (which is designed to estimate the need for primary care), the indices developed by Townsend and Carstairs (used to perform health-related analyses), and the Department of the Environment (DoE) index (which is used in targeting urban regeneration funds). The higher the index value, the more deprived the ward.

The goal of the classification task is to predict the value of the Jarman index (low or high) deprivation factor by exploiting both the other deprivation factors, mortality rate and geographical factors, represented in some linked topographic maps. Spatial analysis is possible thanks to the availability of vectorized boundaries of the 1998 census wards as well as of other Ordnance Survey digital maps of NWE, where several interesting layers, such as urban area (115 spatial objects), green area (9), road net (1687), rail net (805) and water net (716) can be found. The objects on each layer have been stored as tuples of relational tables, also including information on the object type (TYPE). For instance, an urban area may be either a "large urban area" or a "small urban area". Topological non-disjoint relationships between wards and objects in all these layers are materialized as relational tables (WARDS_URBAN_AREAS, WARDS_GREEN_AREAS, WARDS_ROADS, WARDS_RAILS and WARDS_WATERS). The number of tuples in these tables is 5313 (381 wards-urban areas, 13 wards-green areas, 2798 wards-roads, 1054 wards-rails and 1067 wards-waters).

The Munich Census Data concern the level of monthly rent per square meter for flats in Munich, expressed in German Marks.[3] The data were collected in 1998 by Infratest Sozialforschung to develop the 1999 Munich rental guide. This dataset contains 2180 geo-referenced flats situated in the 446 subquarters of Munich, obtained by first dividing the Munich metropolitan area into three

[2] http://www.ais.fraunhofer.de/KD/SPIN/

[3] http://www.di.uniba.it/~ceci/micFiles/munich_db.tar.gz

areal zones and then by dividing each of these zones into 64 districts. The vectorized boundaries of subquarters, districts and zones, as well as the map of public transport stops, consisting of public train stops (56 subway (U-Bahn) stops, 15 rapid train (S-Bahn) stops and 1 railway station), within Munich are available for this study. The objects included in these layers are stored in different relational tables (SUBQUARTERS, TRANSPORT_STOPS and FLATS). Information on the "area" of subquarters is stored in the corresponding table. Transport stops are described by means of their type (U-Bahn, S-Bahn or Railway station), while flats are described by means of their "monthly rent per square meter", "floor space in square meters" and "year of construction".

The target attribute was represented by the "monthly rent per square meter", whose values have been discretized into the two values low $= [2.0, 14.0]$ or high $=]14.0, 35.0]$. The spatial arrangement of data is defined by both the "close_to" relation between Munich metropolitan subquarter areas and the "inside" relation between public train stops and metropolitan subquarters. Both of these topological relations are materialized as relational tables (CLOSE_TO and INSIDE).

The average misclassification errors of TRANSC and Mr-SBC are reported in Tables 4 and 5. The results are obtained according to both a 10-fold CV of the data and a 20-fold CV of the same data. In the case of the NWE Census Data, we set $k \in \{4, 7, 9, 11, 14\}$, while in the case of the Munich Census Data we set $k \in \{9, 18, 27, 36, 45\}$. In both datasets, results confirm an improved accuracy for the transductive setting with respect to the inductive one. The gain depends on the k value and this result is more evident in the case of 10-fold CV. In 20-fold CV, there is an error propagation through algorithm iterations, due to the presence of few training examples. A deeper analysis of the results of 10-fold CV confirms that accuracy increases with high values of k ($k = 11$ for NWE and $k = 36$ for Munich), but at the some time accuracy decreases when k approximates $\sqrt{N}$. This poses the problem of determining some criterion to automatically approximate the best k value.

Table 4
TRANSC vs. Mr-SBC on NWE Census Data: average misclassification error on the working sets

Experiment	TRANSC					Mr-SBC (%)
	$k = 4$ (%)	$k = 7$ (%)	$k = 9$ (%)	$k = 11$ (%)	$k = 14$ (%)	
Avg. 10-CV error	23.38	21.10	19.79	18.04	19.08	22.71
%error reduction	−2.97	7.06	12.84	20.56	15.99	
Avg. 20-CV error	33.87	34.41	33.82	33.20	33.28	34.31
%error reduction	0.00	−1.60	0.15	1.96	1.75	

Number of bins (Nb) in Mr-SBC discretization is set to 10.

Table 5
TRANSC vs. Mr-SBC on Munich Census Data: average misclassification error on the working sets (Nb = 40)

Experiment	TRANSC					Mr-SBC (%)
	$k = 9$ (%)	$k = 18$ (%)	$k = 27$ (%)	$k = 36$ (%)	$k = 45$ (%)	
Avg. 10-CV error	28.99	28.61	28.36	28.30	28.15	31.23
%error reduction	7.17	8.41	9.19	9.40	9.86	
Avg. 20-CV error	37.25	36.30	36.73	36.67	36.78	37.79
%error reduction	1.44	3.94	2.81	2.98	2.68	

7. Conclusions

In this work we have investigated the combination of transductive inference with principled probabilistic MRDM classification, in order to face the challenges posed by real-world applications, characterized by both complex and heterogeneous data, which are naturally modeled as several tables of a relational database and the availability of a small (large) set of labeled (unlabeled) data. Our proposal builds on a multi-relational naïve Bayesian classifier (Mr-SBC), which is learned from the training (i.e., labeled) examples and is then used to perform a preliminary labeling of the working (i.e., unlabeled) data. The initial classification of the examples, comprising the working set, is then refined iteratively over a finite number of steps, each of which consists of a k-NN classification of all examples and a subsequent reclassification of some borderline examples. Neighbors are determined by computing a dissimilarity measure, defined for relational representations of examples.

The proposed transductive multi-relational classifier (TRANSC) has been compared to its inductive counterpart (Mr-SBC) in an empirical study, involving both a benchmark relational dataset and two spatial datasets. Results are in favor of TRANSC and the percentage of accuracy improvement of the transductive setting with respect to the inductive one appears to be better than the small improvement observed in Joachims (2003), when SVMs are compared in both the inductive and the transductive setting. As future work, we intend to extend the empirical investigation, in order to corroborate our intuition that transductive inference has benefits over inductive inference when applied to relational datasets, which are characterized by a strong relational autocorrelation.

References

Bennett, K.P., 1999. Combining support vector and mathematical programming methods for classification. In: Scholkopf, B., Burges, C., Smola, A., (Eds.), Advances in Kernel Methods: Support Vector Learning, MIT Press, Cambridge, MA, pp. 307–326.

Ceci, M., Appice, A., 2006. Spatial associative classification: propositional vs structural approach. Journal of Intelligent Information Systems 27 (3), 191–213.

Ceci, M., Appice, A., Malerba, D., 2003. Mr-SBC: a multi-relational naive bayes classifier. In: Lavrac, N., Gamberger, D., Blockeel, H., Todorovski, L. (Eds.), Proceedings of the Seventh European Conference on Principles and Practice of Knowledge Discovery in Databases, PKDD 2003, Lecture Notes in Artificial Intelligence, vol. 2838. Springer, Berlin, pp. 95–106.

Ceci, M., Appice, A., Barile, N., Malerba, D., 2007. Transductive learning from relational data. In: Perner, P. (Ed.), Machine Learning and Data Mining in Pattern Recognition, MLDM, Lecture Notes in Computer Science, vol. 4571. Springer, Berlin, pp. 324–338.

Chapelle, O., Scholkopf, B., Zien, A., 2006a. Semi-Supervised Learning. MIT Press, Cambridge, MA.

Chapelle, O., Scholkopf, B., Zien, A., 2006b. A discussion of semi-supervised learning and transduction. In: Chapelle, O., Scholkopf, B., Zien, A. (Eds.), Semi-Supervised Learning. MIT Press, Cambridge, MA, pp. 457–462.

Chen, Y., Wang, G., Dong, S., 2003. Learning with progressive transductive support vector machines. Pattern Recognition Letters 24, 1845–1855.

De Raedt, L., 1992. Interactive Theory Revision. Academic Press, London.

De Raedt, L., 1998. Attribute-value learning versus inductive logic programming: the missing links. In: Page, D. (Ed.), Inductive Logic Programming, 8th International Workshop, ILP 1998, Lecture Notes in Artificial Intelligence, vol. 1446. Springer, Berlin, pp. 1–8.

Domingos, P., Pazzani, M., 1997. On the optimality of the simple Bayesian classifier under zeo-ones loss. Machine Learning 28 (2–3), 103–130.

Džeroski, S., Lavrač, N., 2001. Relational Data Mining. Springer, New York.

Esposito, F., Malerba, D., Tamma, V., Bock, H.H., 2000. Similarity and dissimilarity. In: Bock, H.H., Diday, E. (Eds.), Analysis of Symbolic Data: Exploratory Methods for Extracting Statistical Information from Complex Data, Springer, New York, NY, pp. 139–152.

Gammerman, A., Azoury, K., Vapnik, V., 1998. Learning by transduction. In: Proceedings of the 14th Annual Conference on Uncertainty in Artificial Intelligence, UAI 1998. Morgan Kaufmann, Los Altos, CA, pp. 148–155.

Getoor, L., 2001. Multi-relational data mining using probabilistic relational models: research summary. In: Knobbe, A., Van der Wallen, D.M.G. (Eds.), Proceedings of the 1st Workshop in Multi-relational Data Mining, Freiburg, Germany.

Gora, G., Wojna, A., 2002. RIONA: a classifier combining rule induction and k-nn method with automated selection of optimal neighbourhood. In: Elomaa, T., Mannila, H., Toivonen, H. (Eds.), Proceedings of the 13th European Conference on Machine Learning, ECML 2002, Lecture Notes in Artificial Intelligence, vol. 2430. Springer, Berlin, pp. 111–123.

Jensen, D., Neville, J., 2002. Linkage and autocorrelation cause feature selection bias in relational learning. In: Proceedings of the Nineteenth International Conference on Machine Learning.

Joachims, T., 1999. Transductive inference for text classification using support vector machines. In: Proceedings of the 16th International Conference on Machine Learning, ICML 1999. Morgan Kaufmann, Los Altos, CA, pp. 200–209.

Joachims, T., 2003. Transductive learning via spectral graph partitioning. In: Proceedings of the 20th International Conference on Machine Learning, ICML 2003. Morgan Kaufmann, Los Altos, CA.

Krogel, M., Rawles, S., Zelezny, F., Flach, P., Lavrac, N., Wrobel, S., 2003. Comparative evaluation of approaches to propositionalization. In: Horvath, T., Yamamoto, A. (Eds.), Proceedings of the International Conference on Inductive Logic Programming, Lecture Notes in Artificial Intelligence, vol. 2835. Springer, Berlin, pp. 197–214.

Krogel, M.-A., Scheffer, T., 2004. Multi-relational learning, text mining, and semi-supervised learning for functional genomics. Machine Learning 57 (1–2), 61–81.

Kukar, M., Kononenko, I., 2002. Reliable classifications with machine learning. In: Elomaa, T., Mannila, H., Toivonen, H. (Eds.), Proceedings of the 13th European Conference on Machine Learning, ECML 2002. Springer, Berlin, pp. 219–231.

Lavrač, N., Džeroski, S., 1994. Inductive Logic Programming: Techniques and Applications. Ellis Horwood, Chichester, UK.

Mitchell, T., 1997. Machine Learning. McGraw-Hill, New York, USA.

Muggleton, S., 1992. Inductive Logic Programming. Academic Press, London.

Seeger, M., 2000. Learning with labeled and unlabeled data.

Srinivasan, A., King, R.D., Muggleton, S., 1999. The role of background knowledge: using a problem from chemistry to examine the performance of an ILP program. Technical Report PRG-TR-08-99, Oxford University Computing Laboratory.

Taskar, B., Segal, E., Koller, D., 2001. Probabilistic classification and clustering in relational data. In: Nebel, B. (Ed.), IJCAI. Morgan Kaufmann, Los Altos, CA, pp. 870–878 ISBN 1-55860-777-3.

Vapnik, V., 1995. The Nature of Statistical Learning Theory. Springer, New York, NY, USA.

Vapnik, V., 1998. Statistical Learning Theory. Wiley, New York.

Wettschereck, D., 1994. A study of distance-based machine learning algorithms. Ph.D. Thesis, Oregon State University.