

# Relational Learning Problems and Simple Models

Foster Provost  
NYU Stern School of Business  
44 West 4<sup>th</sup> Street  
New York, NY 10012, U.S.A.  
fprovost@stern.nyu.edu

Claudia Perlich  
NYU Stern School of Business  
44 West 4<sup>th</sup> Street  
New York, NY 10012, U.S.A.  
cperlich@stern.nyu.edu

Sofus A. Macskassy  
NYU Stern School of Business  
44 West 4<sup>th</sup> Street  
New York, NY 10012, U.S.A.  
smacskas@stern.nyu.edu

In recent years, we have seen remarkable advances in algorithms for relational learning, especially statistically based algorithms. These algorithms have been developed in a wide variety of different research fields and problem settings. It is important scientifically to understand the strengths, weaknesses, and applicability of the various methods. However, we are stymied by a lack of a common framework for characterizing relational learning.

What are the dimensions along which relational learning problems and potential solutions should be characterized? Jensen (1998) outlined dimensions that are applicable to relational learning, including various measures of size, interconnectivity and variety; items to be characterized include the data, the (true) model, the background knowledge, and so on. Additionally, individual research papers will characterize aspects of relational learning that they are considering and are ignoring. However, there are few studies or even position papers that examine various methods, contrasting them along common dimensions (one notable exception being the paper by Jensen and Neville (2002b)).

It also is not clear whether straightforward measures of size, interconnectivity, or variety will be the best dimensions. In this paper we argue that other sorts of dimensions are at least as important. In particular, the aforementioned dimensions characterize the learning problem (i.e., the training data and the true model). Equally important are characteristics of the context for using the learned model—which have important implications for learning. For illustration, let us discuss three context characteristics, and their implications for studying relational learning algorithms.

**i) Training data versus background knowledge.** Relational learning incorporates background knowledge in a more principled manner than is possible with traditional, single-table learning. Here we consider one particular type of background knowledge: descriptions of known objects, of the same type as those to be classified. Dealing with such objects is not an issue for typical (propositional) learning approaches, because data items are assumed to be independent. In contrast, relational models may be able to take advantage of relations between the data to be classified and background-knowledge entities. But what distinguishes background knowledge from training data? Comparison of methods is difficult if they incorporate different assumptions.

Consider the simple question:

Q1: *Should it be possible for the learned model to take advantage of links to specific training entities?*<sup>1</sup>

There is not a simple answer; it depends on the application to which the learning is being applied. This is not a new observation, but to our knowledge it is not dealt with uniformly by researchers and their methods (which makes comparisons difficult).

Let us clarify by defining two, possibly overlapping, sets of data: training data (T) and background knowledge (B). Only B will be available when the model is in use. Q1 then becomes: Is  $T \subseteq B$ ? And if we define  $T' = T - (T \cap B)$ , then models should *not* be allowed to consider links to entities in  $T'$ . This is a simple example, but it is not trivial—which we will discuss more below.

---

<sup>1</sup> To be clear, we are focusing on the links to specific entities themselves, rather than to properties of the entities.

Once such a dimension is agreed upon, we can define mutually acceptable comparative studies, for example that vary the size of T and of B, their overlap, and so on. We also can discuss whether all of B is present at training time, or if new background knowledge may become available when the model is used.

**ii) Base-level learning vs. learning determinations.**

Assume that models may refer only to background-knowledge entities. The question still remains, should they? For example,<sup>2</sup> is it useful to learn that Hitchcock directs horror movies if Hitchcock is dead and is not going to direct any more movies? Wouldn't it be better to learn that `director` strongly determines `genre`?

Q2 (Q1'): *Should it be possible for the model to take advantage of links to specific background entities?*

Again, this depends on the application, and in particular, on the level of generalization desired. Is it appropriate to learn base-level models or higher-level models (or both)? Such higher-level learning has been called learning *determinations* (Russell, 1986; Schlimmer, 1993). For example, in a traditional, single-table setting one may learn that Mexicans speak Spanish and Brazilians speak Portuguese, or at a higher level learn that `Country-of-origin` determines `Language`. A determination is a higher-order regularity that, once known, can be used in a completely new context for learning from very few data, for analogical reasoning, etc.

If the application of the model is going to be in a completely different context (new entities are not linked to previous background knowledge), it may be appropriate to learn higher-level regularities.<sup>3</sup> If the model is going to be used in the same or similar contexts (e.g., it might encounter more Mexicans), base-level learning may be quite appropriate. If the context is uncertain, it may be appropriate to learn both.

**iii) Linking to the target values.** In traditional flat-table learning, supervised induction *algorithms* reference the target values in the training set. However, it does not make sense for a learned propositional *model* to reference the target values of other examples (because they are assumed to be i.i.d.). However,

---

<sup>2</sup> Thanks to David Jensen for the example.

<sup>3</sup> There actually are three levels of regularities that should be considered. Between determinations and patterns referring to specific entities are models that refer to attributes of entities (and not to the entities themselves). For example, a model could take advantage of differences between European directors and American directors.

relational learning does not assume independent entities, and in fact tries to take advantage of linkages between entities. The target values of the linked entities can be treated in different ways, and again, if methods have different treatments comparison is difficult.

Consider another simple question:

Q3: *Should it be possible for the model to consider the target values of linked entities?*

Again, there is not a simple answer: it depends on the application. This question is related to the question of training data versus background knowledge. By our previous discussion, the model should not be able to access the target values of entities in T'. (The induction algorithm, of course, will access these.)

Issues i through iii illustrate dimensions that are qualitatively different from the size, connectivity, homogeneity, etc., of the data or models. These are characteristics of the application that influence what sort of modeling should be done.

**Example 1.** We want build a model to predict the box-office receipts of movies, using data such as those represented in the Internet Movie Database (IMDB) (Jensen & Neville, 2002a). How do we answer our three questions? We assert that the answer to all three should be "yes." We should not forget about prior movies. (And who knows, a long-lost Hitchcock movie may resurface.) We should not ignore the box-office receipts of prior movies. !

**Example 2.** We want to predict the subtopic of published academic papers within the area of machine learning, based on their relationships to each other through citations or common authors (McCallum et al., 2000; Taskar et al., 2001). How do we answer our three questions? Again, it depends on the application context. If the models are to be applied to the same area (machine learning) from which the training data were selected, the answer to all three should be "yes." We should not forget about other papers we know about. We should not ignore these papers' subtopics. !

We believe that these three questions will be answered in the affirmative for many applications of relational learning.<sup>4</sup> This has implications for the design and evaluation of relational learning algorithms. Here we discuss only one implication: the answers to these

---

<sup>4</sup> The answer to one or more questions of course also may be negative. For example, consider classifying web pages from a site different from that used for training (Slattery & Mitchell, 2000).

questions may bear on the baseline classification procedure to which other methods are compared. If it is possible for a model to reference the class values of training/background entities, very simple models may perform well.

Consider the following two closely related relational classifiers. Both perform simple combinations of evidence from an entity’s relational neighbors. More specifically, these classifiers take advantage of relational “homophily”—the tendency of entities to be related to other similar entities, a form of relational autocorrelation (Jensen and Neville 2002c). The difference between the two classifiers is whether they take advantage of the class labels of linked entities or of explicit class-membership probabilities of linked entities.

**Definition.** The *degree- $k$  relational-neighbor classifier ( $k$ -RN)* estimates the probability of an entity  $e$  being a member of class  $i$  as the (weighted) proportion of the background entities linked to  $e$  by paths of length  $k$ , that belong to class  $i$ . !

**Definition.** The *degree- $k$  probabilistic relational-neighbor classifier ( $k$ -pRN)* estimates the class-membership probability of an entity  $e$  as the normalized sum of the class-membership probabilities of the background ( $B = T$ ) entities to which  $e$  is linked by paths of length  $k$ . !

For a domain of company affiliation classification (Bernstein et al., 2003), for high-autocorrelation affiliations a 1-RN model performs remarkably well—as well as a complicated (and much slower) multi-document text-classification procedure devised specifically for this application, and better than methods based on correlations in stock performance. It clearly would be a mistake to omit this simple model from a comparison of learning techniques for this domain. A  $k$ -RN model also works quite well (compared to other methods) for classifying initial public offerings (Perlich & Provost, 2003).

We suggest using a simple, homophily-based classifier (such as one of these relational neighbor classifiers) as a baseline because homophily is ubiquitous in relational data. Jensen & Neville (2002c) found high relational autocorrelation for almost all attributes they examined in linked movie data. Homophily-based classification is one of the primary techniques used in fraud detection [Fawcett & Provost, 1997; Cortes et al., 2001]. Chakrabarti et al. (1998) take advantage of homophily to classify hypertext documents. Furthermore, homophily with respect to a wide variety of descriptive variables has been observed in the interpersonal

relationships that define social groups, and is one of the basic premises of theories of social structure (Blau, 1977).

Since we borrowed our two examples from the existing literature, we can ask how the simple relational classifiers perform in comparison to more-complex methods.

**Example 1, revisited.** Neville et al. (2002) learn models for predicting (inter alia) box-office receipts of movies, using Relational Probability Trees (RPTs) and Relational Bayesian Classifiers (RBCs). The models estimate the probability that a movie “will be” a blockbuster (the box-office receipts exceed \$2million). Neville et al. found areas under the ROC curve (AUCs) of 0.82 and 0.85 for RPTs and RBCs, respectively, using a set of eight attributes on related entities, such as the most prevalent genre of the movie’s studio.

How well does a homophily-based classifier perform on this problem? Consider 2-RN based on a particular link type (call it  $2\text{-RN}_{\langle \text{link type} \rangle}$ ). Links between movies are through various other entities (actors, studios, production companies, etc.), and we consider the links to be typed by the entity through which they pass (e.g.,  $2\text{RN}_{\text{producer}}$  means: how often does the producer produce blockbusters). The relational-neighbor classifiers achieved AUCs of 0.78 and 0.79 for  $\text{NumLinks}_{\text{production-company}}$  for  $2\text{-RN}_{\text{producer}}$  (respectively). Simply averaging the homophily scores for the various links achieves  $\text{AUC} = 0.82$ .<sup>5</sup>

Let us pretend that the experimental designs are completely comparable, and ask: Do the more complex models produced by the relational learners perform substantially better than simple classifiers? Before answering this we first must agree on issues i—iii above.

**Example 2, revisited.** Taskar et al. (2001) learn Probabilistic Relational Models (PRMs) for classifying academic papers within machine learning into one of seven possible subtopics. Figure 1 shows the accuracy of the PRM as larger proportions of the data are used as training data and labeled background knowledge (here,  $T=B$ ). They varied the proportion of known classes in 10% increments, performing 5-fold cross-validation.

---

<sup>5</sup> Learning a linear combination of  $2\text{-RN}_{\langle \rangle}$  for several link types (actor, director, producer, production company) and the number of links (as suggested by David Jensen) of several link types achieves  $\text{AUC}=0.85$ . Although this does involve some learning, and perhaps a quirk of data entry, it nevertheless only uses a single attribute (the class value) from related entities.

How well does a simple homophily-based classifier perform on this problem? With a moderate amount of labeled background papers, a 1-pRN model performs remarkably well—as well as the PRM. Figure 1 compares the classification accuracy of 1-pRN with the reported results from the PRM. Specifically, using the same data as the prior study, we varied the proportion of papers for which the class initially is known from 10% to 60%, in 5% increments. We performed a 10-fold cross-validation at each setting. For classification, unknown classes of related papers were taken to be the class prior (as defined by the known classes). As shown in Figure 1, although 1-pRN has accuracy of only 50% initially, the accuracy is comparable to that of the PRM once half of the papers are labeled.

The poor performance when few of the classes are known is not surprising, as there is little evidence on which the relational neighbor classifier can base its prediction. The PRM uses a form of belief propagation from labeled to unlabeled entities, which may account for the high performance even with only 10% of the entities labeled. Also shown in Figure 1, iterative 1-pRN adds simple belief propagation to the relational-neighbor classifier. Specifically, it estimates the classes of the unknown papers by repeatedly updating the class-probabilities of each initially unknown node (using 1-pRN) until some stopping criterion is met (in this case, we simply let it run for 100 iterations). While there is little difference from 1-pRN when a large percentage of class labels are known initially, iterative 1-pRN shows a marked improvement when fewer class labels are known. In fact, it is quite competitive with the PRM.

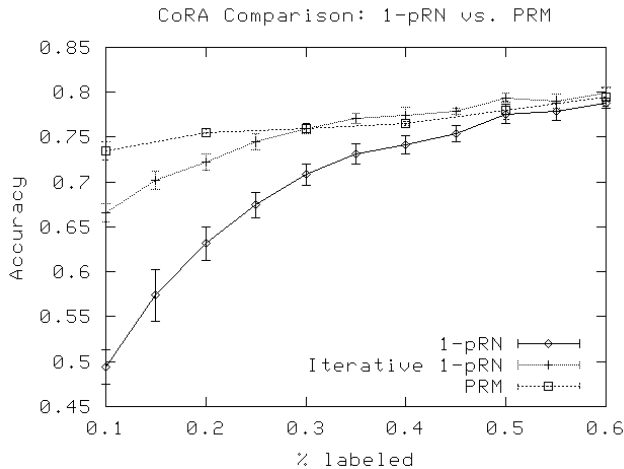


Figure 1: Probabilistic 1-pRN vs. PRM

Again, if we assume that the experimental designs are comparable, these results provide additional evidence

that for this application context (i.e., “yes” to all three questions) such simple models should receive attention.

### Final words

In summary, we have argued (1) that we should come to an agreement on dimensions such as these for characterizing relational learning tasks, and (2) that we should be aware of the power of simple models when certain problem formulations are chosen. In particular, we advocate use of the (homophily-based) Relational Neighbor classifiers as baselines for evaluating generalization performance.

### Acknowledgements

We thank David Jensen for many discussions that have clarified our thinking on these issues, Jennifer Neville for providing us with the details of their study (and, several years ago, the initial impetus for thinking about accessing the target values), and Ben Taskar and Andrew McCallum for providing us with versions of the Cora data set. Avi Bernstein, Scott Clearwater, and Shawndra Hill worked closely with us on the development and study of kRN and closely related models. This work is sponsored in part by the Defense Advanced Research Projects Agency (DARPA) and Air Force Research Laboratory, Air Force Materiel Command, USAF, under agreement F30602-01-2-585.

### References

- [Bernstein et al., 2003] A. Bernstein, S. Clearwater, and F. Provost. The relational vector-space model and industry classification. *IJCAI-2003 Workshop on Learning Statistical Models from Relational Data*, 2003. (Working paper CDeR #IS-03-02, Stern School of Business, New York University.)
- [Blau, 1977] P. Blau. *Inequality and Heterogeneity: A Primitive Theory of Social Structure*. New York: The Free Press, 1977.
- [Cortes et al., 2001] C. Cortes, D. Pregibon, and C. Volinsky. Communities of interest. *Proceedings of the 4th International Conference on Advances in Intelligent Data Analysis (IDA-2001)*, pp. 105—114.
- [Fawcett & Provost, 1997] T. Fawcett and F. Provost. Adaptive fraud detection. *Data Mining and Knowledge Discovery* 1 (3):291—316, 1997.
- [Jensen, 1998] D. Jensen. *Quantitative criteria to characterize KD-ML research for C-XNT*, 1998.
- [Jensen and Neville, 2002a] D. Jensen J. and Neville. Data mining in social networks. Invited presentation to the National Academy of Sciences Workshop on

- Dynamic Social Network Modeling and Analysis. Washington, DC. November 7-9, 2002.
- [Jensen and Neville, 2002b] D. Jensen and J. Neville. Schemas and models. In S. Dzeroski, L. De Raedt, and S. Wrobel, editors, *Proceedings of the Workshop on Multi-Relational Data Mining (MRDM-2002)*, pages 56—70.
- [Jensen and Neville, 2002c] D. Jensen and J. Neville, "Linkage and autocorrelation cause feature selection bias in relational learning," In *Proceedings of the Nineteenth International Conference on Machine Learning*, 2002.
- [McCallum et al., 2000] A. McCallum, K. Nigam, J. Rennie, and K. Seymore. Automating the construction of Internet portals with machine learning. *Information Retrieval*, 3(2): 127—163, 2000.
- [Neville et al., 2002] J. Neville, D. Jensen, L. Friedland, and M. Hay. Learning relational probability trees. University of Massachusetts, (Computer Science Department) *Technical Report 02-55*. Revised Feb. 2003.
- [Perlich and Provost, 2003] C. Perlich and F. Provost. Aggregation-based feature invention and relational concept classes. Working paper *CDeR #IS-03-03*, Stern School of Business, New York University, 2003. *To appear in KDD-2003*.
- [Russel, 1986] S. J. Russell. Preliminary steps toward the automation of induction. In *Proceedings of the National Conference on Artificial Intelligence*, pp. 477—484, 1986.
- [Schlimmer, 1993] J. C. Schlimmer. Efficiently inducing determinations: A complete and systematic search algorithm that uses optimal pruning. In *Proceedings of the Tenth International Conference on Machine Learning*, pp. 284—290, 1993.
- [Slattery and Mitchell, 2000] S. Slattery and T. Mitchell. Discovering test set regularities in relational domains. In *Proceedings of the Seventeenth International Conference on Machine Learning (ICML-2000)*, pp. 895—902, 2000.
- [Taskar et al. 2001] B. Taskar, E. Segal, and D. Koller. Probabilistic classification and clustering in relational data. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence*, pp. 870—878.