J. Yaghob (Ed.): ITAT 2015 pp. 172–178
Charles University in Prague, Prague, 2015

Benchmarking Classifier Performance with Sparse Measurements

Jan Motl[1]

Czech Technical University, Prague, Czech Republic,
jan.motl@fit.cvut.cz,
WWW home page: http://relational.cvut.cz

Abstract: The presented paper describes a methodology, how to perform benchmarking, when classifier performance measurements are sparse. The described methodology is based on missing value imputation and was demonstrated to work, even when 80% of measurements are missing, for example because of unavailable algorithm implementations or unavailable datasets. The methodology was then applied on 29 relational classifiers & propositional tools and 15 datasets, making it the biggest meta-analysis in relational classification up to date.

1 Introduction

You can't improve what you can't measure. However, in some fields the comparison of different approaches is demanding. For example, in the field of relational classifiers, essentially each classifier uses different syntax and requires data in different format, making the comparison of relational classifiers difficult. Despite these obstacles, each author of a new relational classifier attempts to prove that his algorithm is better than some previous algorithm and takes the burden of comparing his algorithm to a small set of algorithms on a limited set of datasets. But how can we compare the algorithms, if they are not evaluated on the same set of datasets?

1.1 Literature Review

The biggest meta-analysis of relational classifiers (to our best knowledge) is "Is mutagenesis still challenging?" [27], where 19 algorithms are examined. While this analysis is highly interesting, it limits itself on comparison of the classifiers on a single dataset.

The biggest analysis in the regard of used datasets is from Dhafer [21], where a single algorithm is tested on 10 datasets and 20 tasks (some datasets have multiple targets).

If we are interested into comparison of multiple algorithms on multiple datasets, the counts are comparably smaller. For example, in the article from Bina [2] 6 algorithms on 5 datasets are examined.

This meta-analysis presents comparison of 29 algorithms on 15 datasets.

1.2 Baseline

Traditionally, a set of algorithms is evaluated on a set of datasets. And then the algorithms are ordered with one (or multiple) of the following methods:

- Average measure
- Average ranking
- Count of wins

	Algo. 1	**Algo. 2**	**Algo. 3**
Dataset A	0.55	0.5	0.45
Dataset B	0.65	0.6	0.55
Dataset C	0.95	1	0.9
Average accuracy	0.72	0.7	0.63
Average ranking	1.33	1.67	3
# Wins	2	1	0

Table 1: Hypothetical evaluation of classifiers based on accuracy (bigger is better) with three ordering methods. In this scenario, all the methods are in agreement (Algorithm 1 is always the best).

	Algo. 1	**Algo. 2**	**Algo. 3**
Dataset A	0.55	0.5	-
Dataset B	0.65	0.6	-
Dataset C	-	1	0.9
Average accuracy	0.6	0.7	0.9
Average ranking	1	1.67	2
# Wins	2	1	0

Table 2: With sparse measurements, average measure predicts that Algorithm 3 is the best while the rest of the methods predict that Algorithm 1 is the best.

The different ordering methods [6] are illustrated on an example in Table 1. In this hypothetical scenario 3 algorithms are evaluated on 3 datasets with accuracy. Based on each ordering method, the first algorithm is the best and the third algorithm is the worst.

But what if not all the measures are available? With the same data, but some missing, we can get different results (Table 2). Based on average accuracy the third algorithm is the best. But we are getting this result only because the third algorithm was evaluated on the datasets with high average accuracy (i.e. easy datasets), while the rest of the algorithms were evaluated on datasets with lower average accuracy (i.e. hard datasets).

Average ranking and count of wins are more robust to missing values. However, neither of them is infallible. Imagine that someone publishes an algorithm and its

weaker version and measures the accuracy not only on the common datasets but also on thousands of randomly generated datasets that are never ever going to be classified by any other algorithm. Then the stronger version of the classifier is going to score at least a thousand wins and place on the first position on the leaderboard regardless of the score on the few common datasets.

If all the algorithms were evaluated on at least one common dataset, we could also order the algorithms just based on the common datasets. But if there isn't any dataset on which all the algorithms are evaluated, we have to come out with another solution.

The solution is to perform missing value imputation and convert the problem to the problem we can already solve.

2 Imputation

The proposed missing value imputation iteratively approximates:

$$acc \approx \overrightarrow{alg} * \overrightarrow{dat} \tag{1}$$

with following pseudocode:

```
acc = pivot(input, @mean)
alg = rowmean(acc)
dat = ones(1, ncol(acc))
for i = 1:nit
    dat = dat + colmean(acc - alg*dat)
    alg = alg + rowmean(acc - alg*dat)
end
```

Where:

input: Matrix with three columns: {algorithm name, dataset name, measured accuracy}.

acc: Matrix with accuracies, where algorithms are in rows and datasets in columns.

alg: Column vector with average accuracy of the algorithms over all the datasets. Initialized to average algorithm accuracy.

dat: Row vector with relative difficulty of the datasets. Initialized to a vector of ones.

nit: Parameter describing the count of iterations. 10 iterations are commonly sufficient.

2.1 Evaluation on a Dense Dataset

To assess the ability of the proposed imputation to properly order relational classifiers, a test on a related task was performed. Arguably the closest task to relational classification, which is well benchmarkable, is propositional classification - the most common type of classification, where a single table is classified.

Conveniently, accuracies of 179 propositional classifiers on 121 datasets were published in a recent study by Fernandez-Delgado [8]. Since not all the examined algorithms always finished successfully, e.g. due to colinearity of data, a dense submatrix of 179 algorithms on 14 datasets

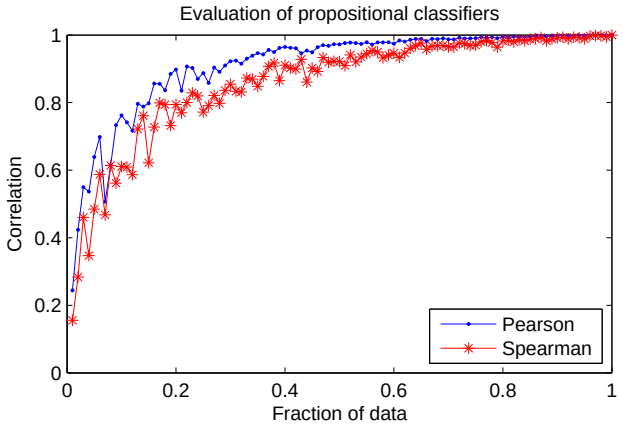

Figure 1: Correlation of the predicted algorithm order with the ground truth based on the proportion of missing data.

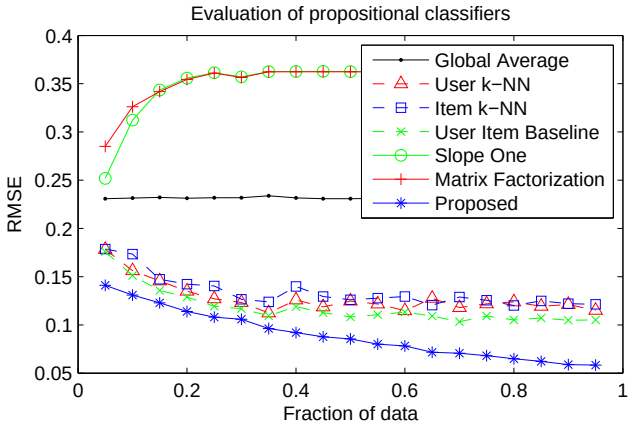

Figure 2: RMSE of the sampled and imputed submatrix with the dense submatrix.

was extracted. The dense submatrix was then randomly sampled with a variable count of measurements. The missing values were then imputed. The resulting learning curve, depicted in figure 1, suggests, that once 20% of all combinations algorithm $\times$ dataset are used, a fairly good estimate of the actual ordering can be estimated.

A comparison of the proposed imputation method to other imputation methods in regard to Root Mean Square Error (RMSE) is in figure 2. The reference methods are from RapidMiner and their parameters were optimized with grid search.

2.2 Theoretical Evaluation

According to the No-Free-Lunch theorem [41], the best classifier will not be the same for all the data sets. Hence we shouldn't even attempt to measure and average classifier's performance over a wide set of datasets. But merely describe strengths and weaknesses of different classifiers. In this respect, the selected imputation method fails because it is not able to model interactions between datasets and algorithms. Nevertheless, in the practice, some classifiers appear to be systematically better then other [8]. If all

we want is to order classifiers based on their expected accuracy on a set of datasets, the absence of ability to model interactions is irrelevant.

Another property of the used methodology is that it doesn't permit mixture of measures. That is unfortunate since some articles [18] report only accuracy, while other articles [10] report only precision, recall and F-measure.

A special attention is necessary when we are comparing results from several authors, because not only the evaluation methodology can differ (e.g. 10-fold cross-validation vs. single hold-out sample), but also datasets can differ despite the common name. For example, the canonical version of East-West dataset (further abbreviated as *Trains*) contains 10 instances [30]. However, some authors prefer an extended version of the dataset with 20 instances [35]. Nevertheless, data quality is the pitfall common to all analyses. To alleviate the problem with different datasets in the future, a new (and the first) relational repository was based at `relational.fit.cvut.cz`. Further discussion about the collected data is provided in the next section.

The final limitation of the method is that it doesn't provide trustworthy confidence intervals. The first reason is that measures for the same algorithm and dataset are averaged and treated as a single measure. The second reason is that the algorithm performs missing value imputation, violating the assumption of sample independence.

A list summarizing the advantages and constrains of the proposed method follows:

Advantages:

- Permits benchmarking with sparse measures.
- Respects that some datasets are tougher than others.
- Allows conflicting measurements (for example, by different authors).

Disadvantages:

- Neglects interactions between datasets and algorithms.
- Requires one common measure (e.g. we cannot mix accuracy and F-measure).
- Requires comparably prepared datasets (e.g. using the same instance count).
- Doesn't provide confidence intervals.

3 Classification of Relational Data

The proposed methodology how to benchmark with sparse measurements is applied on relational classifiers, including propositional tools. In the following paragraphs description of the collected measures, benchmarked algorithms and datasets follow. The collected data can be downloaded from `motl.us\benchmarking`.

Algorithm	Algorithm type	Reference
Aleph	ILP	[35]
CILP++*	Neural Network	[9]
CrossMine	ILP	[42]
E-NB*	Probabilistic	[37]
FOIL	ILP	[23]
FORF-NA	Decision Tree	[40]
Graph-NB	Probabilistic	[26]
HNBC*	Probabilistic	[37]
kFOIL	Kernel	[23]
Lynx-RSM*	Propositionalization	[29]
MLN	Probabilistic	[36]
MRDTL-2	Decision Tree	[25]
MRNBC*	Probabilistic	[37]
MVC*	Multi-View	[11]
MVC-IM*	Multi-View	[12]
MulSVM*	Propositionalization	[43]
nFOIL*	Probabilistic	[22]
PIC*	Probabilistic	[37]
RELAGGS	Propositionalization	[17]
RPT*	Probabilistic	[28]
RSD	Propositionalization	[18]
RollUp	Propositionalization	[16]
SINUS	Propositionalization	[18]
SimFlat*	Propositionalization	[12]
SDF*	Decision Tree	[2]
TILDE	Decision Tree	[37]
TreeLiker-Poly	ILP	[20]
TreeLiker-RelF	ILP	[20]
Wordification*	Propositionalization	[35]

Table 3: List of 29 relational classifiers and propositional algorithms used in the meta-analysis. A star by the algorithm name marks algorithms, for which measurements by someone else than by the algorithm authors was not found.

3.1 Measure Selection

Since almost all relational classifiers in the literature are evaluated on classification accuracy (with exceptions like CLAMF [10], ACORA [34] or SAYU [4], which are evaluated in the literature only with measures based on precision & recall) but only a few were evaluated with a different measure (like precision & recall, F-measure, AUC or AUC-PR), the meta-analysis limits itself to classification accuracy. The methods how to measure accuracy may differ, but only testing accuracies (not training) were collected.

Other interesting measures, like runtime or memory consumption, were not evaluated, as they are rarely published. And even if they were published, they would be hardly comparable as the measurements are platform dependent.

Dataset	Target	#Instances	Reference
Alzheimer	acetyl	1326	[15]
Alzheimer	amine	686	[15]
Alzheimer	memory	642	[15]
Alzheimer	toxicity	886	[15]
Carcinogenesis	carcinogenic	329	[38]
ECML	insurance	7329	[19]
Financial	loan status	682	[1]
Hepatitis	biopsy	690	[35]
IMDb	ratings	281449	[2]
KRK	depth-of-win	1000	[18]
Mondial	religion	185	[39]
MovieLens	age	941	[2]
Musk-small	musk	92	[7]
Musk-large	musk	102	[7]
Mutagenesis	mutagenic	188	[5]
Thrombosis	degree	770	[3]
Trains	direction	10	[30]
UW-CSE	advisedBy	339	[36]

Table 4: List of 15 datasets with their targets (Alzheimer dataset has multiple targets) used in the meta-analysis.

3.2 Algorithm Selection

The selection of relational classifiers and propositionalization tools was restricted to algorithms, which:

- Were published in conference or journal paper.
- Were benchmarked on at least four datasets.
- Were evaluated on classification accuracy.

The list of compared algorithms is in table 3.

3.3 Dataset Selection

Datasets were selected based on the following criteria:

- The dataset has a defined classification target.
- The dataset consists of at least two tables.
- The dataset is used by at least four algorithms.

The used relational datasets are listed in table 4.

4 Results

Box plot in Figure 3 depicts estimated average classification accuracies of 29 algorithms on 15 datasets (18 tasks). The input data consists of 26% of all combinations algorithm × dataset, making the estimates solid (recall figure 1). The accuracies were estimated with 1000 bootstrap samples. Whiskers depict 1.5 IQR.

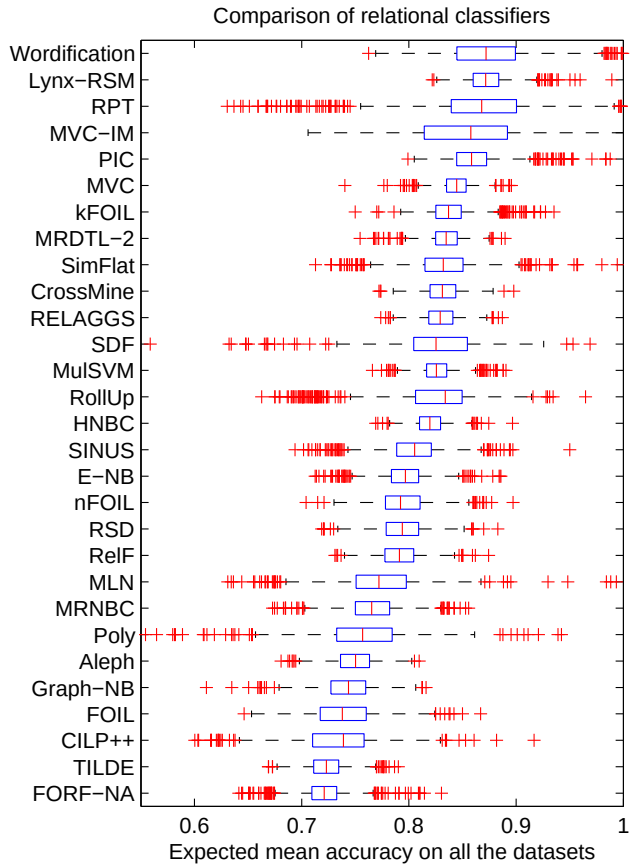

Figure 3: Box plot with expected accuracies.

4.1 Validation

The ordering of algorithms from the meta-analysis should roughly correspond to the ordering of the algorithms in the individual articles. Differences in the orderings are evaluated with Spearman correlation in table 5.

As we can see, the orderings in the literature can contradict – once RELLAGS is better than CrossMine, once CrossMine is better than RELLAGS.

5 Discussion

Summary of figure 3 based on algorithm type is in figure 4. Interestingly, kernel and multi-view approaches are averagely the most accurate algorithms. But some propositionalization algorithms, namely Wordification, Lynx-RSM and RPT (Relational Probabilistic Tree) beat them. Nevertheless, note that propositionalization algorithms are overrepresented in the meta-analysis, making it more likely that some of them place at extreme positions.

Also note, that performance of algorithms is influenced by their setting. While algorithms in figure 3 are ordered based on the average accuracy per combination of algorithm × dataset, algorithms in figure 5 are ordered based

Ordering	Spearman Correlation	Reference
TILDE < FORF-NA < Graph-NB < RelF < Poly < SDF	0.89	[2]
MRNBC < TILDE < E-NB < HNBC < PIC	0.9	[37]
TILDE < RELLAGS < CrossMine < MVC-IM	1.0	[12]
FOIL < TILDE < CrossMine < RELLAGS < MVC	0.9	[31]

Table 5: Comparison of algorithm ordering in the literature with ordering from the imputation.

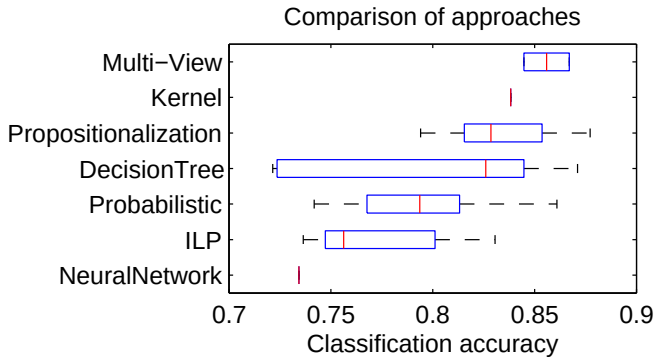

Figure 4: Estimated accuracies by algorithm type.

on the maximal known accuracy per combination of algorithm $\times$ dataset. Notably, with the right setting, RSD improves its ranking by 5 positions.

Finally, it is trusted that each individual author measured accuracy correctly. For example, in the article from 2014 [24] authors of Wordification applied cross-validation only on the propositional classifiers, leaving discretization and propositionalization out of the cross-validation. This design can lead to overly optimistic estimates of the accuracy. In the follow up article from 2015 [35] the authors of Wordification are already applying cross-validating on the whole process flow. Wordification accuracies used in this article are exclusively from [35].

5.1 Fairness of Comparison

The comparison solely based on the classification accuracy can be unfair. For example, CLAMF* [10] and CoMoVi* [32] are designed to work with temporal datasets. Hence in Financial dataset they estimate probability of a loan default only from the data before the time of a loan application, while classifiers designed for the statical datasets also use data at and after the time of the loan application. All classifiers used in the meta-analysis treat all the datasets as if they were statical.

Validation of temporal datasets can be furthermore complicated by repeated target events. For example, a customer may apply for a loan many times. And now there are two perfectly plausible goals - we may want to calculate probability of default of a current customer with history of loans or probability of default of a new customer. Generally, the second task is tougher because one of the best predictors of customer's behavior is the customer's past behavior. Nevertheless, all datasets in the meta-analysis have exactly one target value per classified object (including Financial dataset). Note that difference between within/across classification in IMDb and MovieLens datasets is another issue [33].

Also the goals of modeling can differ. For example Markov Logic Network* in [14] is evaluated as generative model, so accuracies reported are over all predicates, not just the target one. And accuracies can vary substantially with respect to the chosen target predicate. All the algorithms in the meta-analysis are evaluated in a discriminative setting.

Additionally, not all classifiers are designed to perform well on a wide spectrum of datasets. Indeed, there are algorithms like MOLFEA [13] that are designed to work only on a narrow subset of datasets. A possible specialization of the algorithms in the meta-analysis is not taken in the consideration.

At last different authors may have different ethos. Algorithms that were evaluated only by the algorithm authors (to our best knowledge) were marked with a star in table 3. And many algorithms that place at the top of the ranking are stared algorithms. However, this trend can also be explained with following hypotheses:

- Recent algorithms tend to be better than the old algorithms. And recent algorithms (like Wordification [35]) did not have enough time to accumulate references.
- New algorithms tend to look better in comparison to mediocre algorithms than in comparison to the best algorithm in the field. Hence authors prefer to compare their algorithms against mediocre algorithms.
- Third-party evaluators do not have the knowledge and resources to find the best algorithm setting. Hence popular algorithms have, on average, low accuracy. This problem is partially mitigated by considering only the best reported accuracies in figure 5.

Overall, comparison of measurements from several sources is not a simple task at all.

*The algorithm is not included in the meta-analysis because it's accuracy wasn't measured on enough datasets.

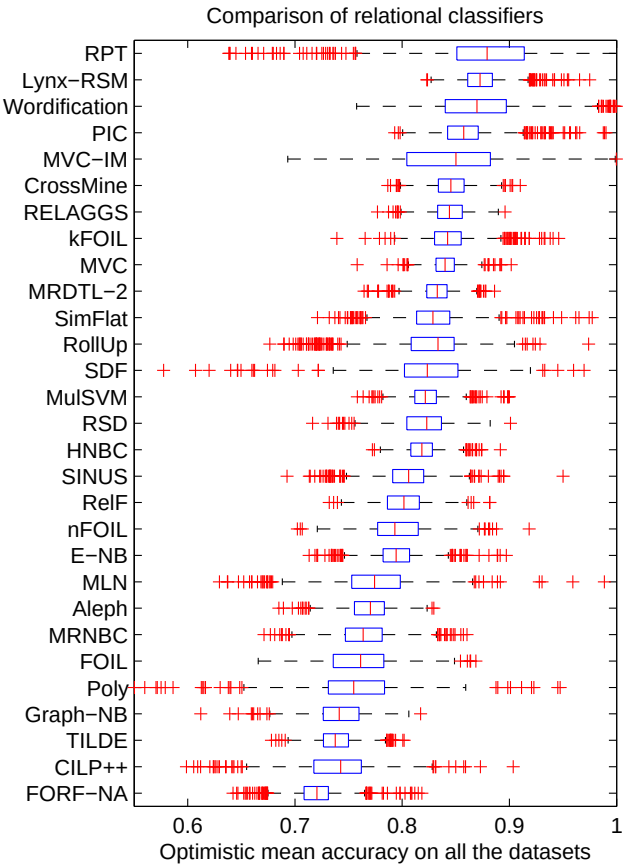

Figure 5: Box plot with optimistic accuracies.

6 Conclusion

Based on the performed analysis, Wordification, Lynx−RSM and Relational Probabilistic Tree on average outperform other 26 algorithms for relational classifications. Other promising categories of relational classifiers are multi-view and kernel based approaches.

Acknowledgement

The research reported in this paper has been supported by the Czech Science Foundation (GAČR) grant 13-17187S.

References

[1] Berka, P.: Workshop notes on Discovery Challenge PKDD'99, 1999

[2] Bina, B., Schulte, O., Crawford, B., Qian, Z., Xiong, Y.: Simple decision forests for multi-relational classification. Decision Support Systems **54(3)** 2013, 1269–1279

[3] Coursac, I., Duteil, N.: PKDD 2001 Discovery Challenge – Medical Domain, 2001

[4] Davis, J., Burnside, E., Page, D.: View learning extended: inventing new tables for statistical relational learning. ICML Workshop on Open Problems in Statistical Relational Learning, 2006

[5] Debnath, A. K., Lopez de Compadre, R. L., Debnath, G., Shusterman, A. J., Hansch, C.: Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds. Correlation with molecular orbital energies and hydrophobicity. Journal of Medicinal Chemistry **34(2)** (1991), 786–797

[6] Demšar, J.: Statistical comparisons of classifiers over multiple data sets. The Journal of Machine Learning Research **7** (2006), 1–30

[7] Dietterich, T.: Solving the multiple instance problem with axis-parallel rectangles. Artificial Intelligence **89(1–2)** (1997), 31–71

[8] Fernández-Delgado, M., Cernadas, E., Barro, S., Amorim, D.: Do we need hundreds of classifiers to solve real world classification problems? Journal of Machine Learning Research **15** (2014), 3133–3181

[9] França, M. V. M., Zaverucha, G., D'Avila Garcez, A.: Fast relational learning using bottom clause propositionalization with artificial neural networks. Machine Learning **94(1)** (2014), 81–104

[10] Frank, R., Moser, F., Ester, M.: A method for multi-relational classification using single and multi-feature aggregation functions. Lecture Notes in Computer Science **4702** (2007), 430–437

[11] Guo, H., Viktor, H. L.: Mining relational databases with multi-view learning. Proceedings of the 4th International Workshop on Multi-Relational Mining – MRDM'05, 2005, 15–24

[12] Guo, H., Viktor, H. L.: Learning from skewed class multi-relational databases. Fundamenta Informaticae **89(1)** (2008), 69–94

[13] Helma, C., Kramer, S., De Raedt, L.: The molecular feature miner MOLFEA. In: Proceedings of the Beilstein Workshop 2002: Molecular Informatics: Confronting Complexity; Beilstein Institut, 2002, 1–16

[14] Khosravi, H., Schulte, O., Hu, J., Gao, T.: Learning compact Markov logic networks with decision trees. Machine Learning **89(3)** (2012), 257–277

[15] King, R. D., Sternberg, M., Srinivasan, A.: Relating chemical activity to structure: an examination of ILP successes. New Generation Computing **13 (3–4)** (1995), 411–433

[16] Knobbe, A., J., De Haas, M., Siebes, A.: Propositionalisation and aggregates. Lecture Notes in Computer Science **2168** (2001), 277–288

[17] Krogel, M. -A.: On propositionalization for knowledge discovery in relational databases. PhD Thesis, Otto-von-Guericke-Universität Magdeburg, 2005

[18] Krogel, M. -A., Rawles, S., Železný, F., Flach, P. A., Lavrač, N., Wrobel, S.: Comparative evaluation of approaches to propositionalization. In: Proceedings of the 13th International Conference on Inductive Logic Programming, volume 2835, 194–217, 2003

[19] Krogel, M. -A., Wrobel, S.: Facets of aggregation approaches to propositionalization. In: Inductive Logic Programming: 13th International Conference, 30–39, Springer, Berlin, 2003

[20] Kuželka, O.: Fast construction of relational features for machine learning. PhD Thesis, Czech Technical University, 2013

[21] Lahbib, D., Boullé, M., Laurent, D.: Itemset-based variable construction in multi-relational supervised learning. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 7842 LNAI:130–150, 2013

[22] Landwehr, N.: Integrating naive bayes and FOIL. Journal of Machine Learning Research **8** (2007), 481–507

[23] Landwehr, N., Passerini, A., De Raedt, L., Frasconi, P.: kFOIL: Learning simple relational kernels. Aaai **6** (2006), 389–394

[24] Lavrač, N., Perovšek, M., Vavpetič, A.: Propositionalization online. In: ECML PKDD 2014, 456–459, Springer-Verlag, 2014

[25] Leiva, H. A., Atramentov, A., Honavar, V.: A multi-relational decision tree learning algorithm. Proceedings of the 13th International Conference on Inductive Logic Programming, 2002, 38–56

[26] Liu, H., Yin, X., Han, J.: An efficient multi-relational Naïve Bayesian classifier based on semantic relationship graph. Proceedings of the 4th International Workshop on Multi-Relational Mining – MRDM'05, Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2005), 2005, 39–48

[27] Lodhi, H., Muggleton, S.: Is mutagenesis still challenging? Proceedings of the 15th International Conference on Inductive Logic Programming, ILP 2005, Late-Breaking Papers., 2005, 35–40

[28] Macskassy, S. A., Provost, F.: A simple relational classifier. Technical Report, Stern New York University, 2003

[29] Di Mauro, N., Esposito, F.: Ensemble relational learning based on selective propositionalization. CoRR (2013), 1–10

[30] Michie, D., Muggleton, S., Page, D., Srinivasan, A.: To the international computing community: a new east-west challenge. Technical Report, Oxford University Computing Laboratory, Oxford, 1994

[31] Modi, S.: Relational classification using multiple view approach with voting. International Journal of Computer Applications **70(16)** (2013), 31–36

[32] Neto, R. O., Adeodato, P. J. L., Salgado, A. C., Filho, D. R., Machado, G. R.: CoMoVi: a framework for data transformation in credit behavioral scoring applications using model driven architecture. SEKE 2014 (2014), 286–291

[33] Neville, J., Gallagher, B., Eliassi-Rad, T., Wang, T.: Correcting evaluation bias of relational classifiers with network cross validation. Knowledge and Information Systems **30 (1)** (2012), 31–55

[34] Perlich, C., Provost, F.: Distribution-based aggregation for relational learning with identifier attributes. Machine Learning **62(1–2)** SPEC. ISS. (2006), 65–105

[35] Perovšek, M., Vavpetič, A., Kranjc, J., Cestnik, B., Lavrač, N.: Wordification: propositionalization by unfolding relational data into bags of words. Expert Systems with Applications **42(17–18)** (2015), 6442–6456

[36] Richardson, M., Domingos, P.: Markov logic networks. Machine Learning **62 (1–2)** SPEC. ISS. (February 2006), 107–136

[37] Schulte, O., Bina, B., Crawford, B., Bingham, D., Xiong, Y.: A hierarchy of independence assumptions for multi-relational Bayes net classifiers. Proceedings of the 2013 IEEE Symposium on Computational Intelligence and Data Mining, CIDM 2013 – 2013 IEEE Symposium Series on Computational Intelligence, SSCI 2013, 2013, 150–159

[38] Srinivasan, A., King, R. D., Muggleton, S. H., Sternberg, M. J. E.: Carcinogenesis predictions using ILP. Inductive Logic Programming **1297** (1997) 273–287

[39] Taskar, B., Abbeel, P., Koller, D.: Discriminative probabilistic models for relational data. UAI'02 Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence, 2002, 485–492

[40] Vens, C., Van Assche, A., Blockeel, H., Džeroski, S.: First order random forests with complex aggregates. Lecture Notes in Computer Science **3194** (2004), 323–340

[41] Wolpert, D.: The existence of a priori distinctions between learning algorithms. Neural Computation **8(7)** (1996), 1391–1420

[42] Yin, X., Han, J., Yang, J., Yu, P. S.: CrossMine: efficient classification across multiple database relations. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) **3848** LNAI(6) (2006), 172–195

[43] Zou, M., Wang, T., Li, H., Yang, D.: A general multi-relational classification approach using feature generation and selection. In: 6th International Conference, ADMA, 21–33, Springer-Verlag, 2010