SkILL - a Stochastic Inductive Logic Learner

Joana Côrte-Real†, Theofrastos Mantadelis†, Inês Dutra†, Ricardo Rocha†, and Elizabeth Burnside‡

†CRACS & INESC TEC, University of Porto
Rua do Campo Alegre, 1021/1055, 4169-007 Porto, Portugal
{jcr, theo.mantadelis, ines, ricroc}@dcc.fc.up.pt

‡Department of Radiology
University of Wisconsin, MA, USA
eburnside@uwhealth.org

***Abstract*—Probabilistic Inductive Logic Programming (PILP) is a relatively unexplored area of Statistical Relational Learning which extends classic Inductive Logic Programming (ILP). Within this scope, we introduce SkILL, a Stochastic Inductive Logic Learner, which takes probabilistic annotated data and produces First Order Logic (FOL) theories. Data in several domains such as medicine and bioinformatics have an inherent degree of uncertainty, and because SkILL can handle this type of data, the models produced for these areas are closer to reality. SkILL can then use probabilistic data to extract non-trivial knowledge from databases, and also address efficiency issues by introducing an efficient search strategy for finding hypotheses in PILP environments. SkILL's capabilities are demonstrated using a real world medical dataset in the breast cancer domain.**

I. Introduction

Statistical Relational Learning (SRL) [17] is a well-known collection of techniques whose main objective is to produce interpretable probabilistic classifiers, often in the form of understandable logical sentences. Researchers have often focused their efforts on defining logic languages to handle probabilistic data ([20], [11], [19], [6]), but few works have been dedicated to *learning* understandable rules from that probabilistic knowledge. In this work, we introduce SkILL – a Stochastic Inductive Logic Learner – which can combine the rule learning capability of classic Inductive Logic Programming (ILP) ([14], [9]) with uncertain knowledge by generating FOL rules based on a richer and more expressive representation of the data than traditional ILP supports.

ILP stands out from traditional machine learning due to its suitability to handle relational data but suffers from exponential growth of its hypotheses search space with the size of the dataset. Introducing probabilistic information to describe data may implicitly reduce the search space since useful information can still be extracted, for instance, from marginal distributions. Also, compressing data in such a way could also be used in order to protect private sensitive data. Furthermore, in cases where the full conditional probability table is not known, information can still be used efficiently in the computation of a hypothesis, for instance, by adding values from the literature to the background knowledge.

The contributions of this work are: (i) the SkILL system, which is a general purpose PILP tool that supports logical constructs, arbitrary variable logical terms and annotated disjunctions [21]; (ii) an efficient search strategy to traverse the PILP search space, and (iii) an analysis of a real-world medical dataset pertaining breast cancer.

II. SkILL

SkILL is a Probabilistic Inductive Logic Programming (PILP) tool which can extract non-trivial knowledge (FOL theories) from probabilistic data. SkILL's inputs include Probabilistic Background Knowledge (PBK), representing the basic information known about the problem, which can be composed of both rules and facts, either probabilistic or not. Additionally, the observations that the system is attempting to explain, called Probabilistic Examples (PE), are annotated with their *expected values*, which can represent either statistical information or the degree of belief in an example (using type I or type II probability structures [4], respectively). PBK and PE can be seen as the probabilistic versions of ILP's Background Knowledge (BK) and Examples input.

The aim of the SkILL tool is to find a hypothesis in the valid search space which minimizes a *loss function* w.r.t. the given PE. Hypotheses are formed by a single clause or by a set of disjunct clauses, and their length is equal to the number of clauses they contain. Combining two hypotheses of length one results in a hypothesis of length two, and so on. The result of this combination is the logical disjunction of the clauses in all hypotheses.

SkILL's hypotheses search space is composed of all combinations of hypotheses of length one, up to a maximum user-defined length. It is easy to see that this search space is exponential and so traversing it exhaustively would not be scalable. As such, SkILL's algorithm selects two different populations of fixed size at every step (Primary and Secondary), so as to maintain the complexity of the algorithm constant for user-defined sizes of populations and maximum length. Hypotheses populations are ranked according to user-defined metrics, so as to allow for either combining only good hypotheses, or stochastically selecting candidates.

SkILL supports three metrics to rank and evaluate hypotheses: RMSE (root mean squared error), PAcc (probabilistic accuracy) and Random (stochastic selection). The RMSE metric penalizes predictions farther from the expected values, while PAcc is the generalization of the discrete accuracy to the probabilistic setting as introduced by De Raedt and Thon [18] and also used by Muggleton [13]. SkILL can handle classical ILP modes such as output/input and constants, and also encode expert information as probabilistic facts, annotated disjunctions and probabilistic rules in the PBK.

III. SkILL's Search Algorithm

Algorithm 1 presents SkILL's main algorithm to traverse the hypotheses search space. It takes as input the PBK and PE, plus parameters corresponding to: the maximum length of the hypotheses to be generated (MaxHypLength); the size of the two auxiliary sets of hypotheses used for combination and generation of new longer hypotheses (Psize and Ssize);

two metrics to rank the selection of hypotheses used for combination (PRankMetric and SRankMetric); and finally, a metric that is used to decide what is the best hypothesis found (EvalMetric).

Algorithm 1: SkILL Algorithm

```
1  Input = PBK, PE, MaxHypLength, Psize, Ssize, PRankMetric,
   SRankMetric, EvalMetric
2  Output = Best hypothesis according to EvalMetric
3  Hyps1 = HypsN = AllHyps = generate_hyps_length_one(PBK, PE)
4  for Length = 2; Length ≤ MaxHypLength; Length++ do
5  |   Primary = select_members(HypsN, Psize, PRankMetric)
6  |   Secondary = select_members(Hyps1, Ssize, SRankMetric)
7  |   HypsN = generate_combinations(Primary, Secondary)
8  |   AllHyps = AllHyps ∪ HypsN
9  end
10 return best_hypothesis(AllHyps, EvalMetric)
```

Initially, the algorithm uses the TopLog engine [15] from the GILPS ILP system to generate all possible hypotheses of length one (line 3 in Alg. 1). A hypothesis of length one is constructed from literals that are contained in the PBK and are selected by the user. Hypotheses of length one are generated in such a way as to always entail at least one of the probabilistic examples. SkILL improves on this approach by removing hypotheses which are permutations of each other (i.e., syntactically distinct but semantically equal). This approach, which is the state-of-the-art in ILP, results in hypotheses of length one that mirror patterns contained in the observations w.r.t. the PBK.

Once hypotheses of length one are generated, the algorithm proceeds by generating hypotheses with length greater than one (lines 4–9 in Alg. 1) until reaching MaxHypLength. Combining hypotheses in order to generate new hypotheses with larger length is not a trivial task; possible combinations are $\binom{N}{K}$ with N being the total number of length one hypotheses and K the maximum hypothesis length. SkILL's search strategy selects candidate hypotheses for two different sets, named *Primary* and *Secondary* (lines 5–6 in Alg. 1), and new hypotheses are then generated by combining members from each set (line 7 in Alg. 1).

In each iteration of the algorithm, the primary set is filled with the Psize best hypotheses, according to a given *ranking metric* (argument PRankMetric), from the set of hypotheses generated in the previous iteration (1 clause hypotheses when searching for 2 clauses hypotheses; 2 clauses hypotheses when searching for 3 clauses hypotheses; etc). The secondary set is populated with Ssize hypotheses from the set of hypotheses of length one, according to SRankMetric. Depending on the ranking metrics chosen, the system can generate hypotheses in a fully stochastic way, use best hypotheses or create a heterogeneous mix. The stochastic component of the selection is distinct for each iteration.

Finally, all hypotheses are evaluated according to the given evaluation metric (argument EvalMetric), and the best generated hypothesis for all different lengths is returned (line 10 in Alg. 1).

IV. Experimental Settings

SKILL runs on top of the Yap Prolog system [1], uses GILPS [15] as the basis hypotheses generator and MetaProbLog [10] (an extension of ProbLog [6]) as the probabilistic representation language. Knowledge is thus annotated according to ProbLog syntax and the MetaProbLog engine is used to evaluate the probabilities of the generated theories. This section presents two sets of experiments: (i) a comparison against ProbFOIL+ which focuses on performance and scalability, and (ii) a study of a real-world medical dataset of non-definite biopsies.

A. Comparison with ProbFOIL+

SkILL's execution time and probabilistic accuracy were compared against the probabilistic rule learner ProbFOIL+ [2] using a probabilistic dataset of 44 probabilistic facts about family relations. The dataset is composed of literals mother, father and parent (Prolog rule), as well as of 10 examples of the target predicate grandmother, all with probability 1.00. ProbFOIL+ differs from SkILL because in addition to learning a probabilistic hypothesis, it also calculates values for each clause of the hypothesis which minimize the error over all examples. However, to the best of the authors' knowledge, ProbFOIL+ does not support constant arguments or annotated disjunctions, making it impossible to compare to the dataset presented in the remainder of this section. Both SkILL and ProbFOIL+ were tested using three settings: (i) attempt to use only the mother literal in the hypothesis, (ii) attempt to use mother and father literals, and (iii) attempt to use the three available literals. Table I presents reported execution time and probabilistic accuracy for SkILL and ProbFOIL+ in these three settings.

TABLE I. Execution time and probabilistic accuracy of SkILL and ProbFOIL+ in family dataset

Setting	Execution time (s)		Probabilistic accuracy (%)	
	SkILL	ProbFOIL+	SkILL	ProbFOIL+
mother	0.46	0.95	70.0	98.4
mother + father	0.73	2.91	91.0	99.7
mother + father + parent	0.96	45.31	97.0	99.7

In every setting in Table I the hypotheses induced by both systems were correct and equivalent to each other w.r.t their literals. ProbFOIL+'s independent error minimization technique allows for higher accuracy in every experiment, but as the dataset grows up to 44 facts, SkILL's accuracy is only marginally lower to that of ProbFOIL+. Also, SkILL's runtime is always shorter than ProbFOIL+'s, in every setting. SkILL is clearly more scalable than ProbFOIL+, since its runtime only doubles for 3 literals when compared to 1, whereas ProbFOIL+ gets 50 times slower in the same case.

Whilst SkILL appears to under-perform accuracy-wise, both systems learn the same logical rules for every setting. If the best hypothesis found by the ProbFOIL+ system is not very accurate, then the added tuning mechanism will make a significant difference in accuracy results, as can be seen in Table I. However, in the case where a hypothesis already has high accuracy, both systems produce comparable results. As datasets grow larger, and/or have more literals available for hypotheses construction, the best hypotheses available tend to have higher accuracy, which leads to similar results for SkILL and ProbFOIL+. This phenomenon is present in the three-literal setting of this experiment and in the following experiment.

B. Knowledge Extraction

Breast cancer diagnosis guidelines suggest that patients presenting suspicious breast lesions should be sent to perform

a diagnostic mammogram and possibly an ultrasound, and a core needle biopsy to further define this abnormality. The biopsy is very important in determining malignancy of a lesion and usually yields definitive results; however, in 5% to 15% of cases, the results are non-definitive [16]. Routine practice usually sends all patients with non-definitive biopsies to excision, even though only a small fraction of them (10-20%) have in fact a malignant finding confirmed after the procedure - the remainder of them did not need to be subjected to surgery.

Although non-definitive biopsies are relatively rare, sending every woman that has a non-definitive biopsy to excision is not a good practice. Machine learning methods have been used to mitigate this and other problems by allowing to produce models of the data that can distinguish between benign and malignant cases [8], [3]. However, in the medical domain it is crucial to represent data in a way that experts can understand and reason about, and as such ILP can successfully be used to produce such models. Furthermore, probabilistic ILP allows for incorporating in the PBK the confidence of physicians in observations and known values from the literature.

In this study, we use 130 biopsies dating from January 2006 to December 2011, which were prospectively given a non-definitive diagnosis at radiologic-histologic correlation conferences. 21 cases were determined to be malignant after surgery, and the remaining 109 proved to be benign. For all of these cases, several sources of variables were systematically collected including variables related to demographic and historical patient information (age, personal history, family history etc), mammographic BI-RADS descriptors (mass shape, mass margins, calcifications etc), pathological information after biopsy (type of disease, if it is incidental or not, number of foci etc), biopsy procedure information (needle gauge, type of procedure etc), and other relevant facts about the patient. Probabilistic data was also gathered: namely the confidence in malignancy for each case (before excision), assigned by a group of physicians analysing that case. Furthermore, and since physicians base their conclusions in literature values from the universe of all biopsies, values were added in the PBK as the probability of malignancy given a feature value (is_malignant features). For example, it is well known among radiologists expert in mammography that if a mass has a spiculated margin, the probability that the associated finding is malignant is around 90%.

This experiment uses as examples the probabilities assigned by group of physicians representing their estimation of the malignancy of each case. **Learning from non-discrete classes is a unique characteristic of SkILL that combines interpretable rules with a non-boolean classification model**. The resulting theory is presented in Figure 1 (hypothesis found using: PAcc metric both for ranking and evaluation; primary/secondary population of 20/200; and generating hypotheses until length 3).

The is_malignant hypothesis found is composed of two clauses and has a probabilistic accuracy of 90% and prediction accuracy of 94%, when using a threshold of 0.5. This learnt predicate does use the following probabilistic facts annotated from the medical literature:

1) 0.70:: is_malignant(mass_margin(microlob))
2) 0.50:: is_malignant(mass_shape(irregular))
3) 0.20:: is_malignant(mass_margin(indistinct))

These predicates describe the margins and shapes of a mass and state the probability of malignancy, in medical literature, given that the mass margin is microlobulated, the mass shape is irregular or the mass margin is indistinct. Furthermore, SkILL's hypotheses can be used for probabilistic prediction of malignancy of a tumor. In this experiment we then used the generated hypothesis to predict the values of the examples compared this approach against a Naive Bayes classifier using the same data (as described by Kuusisto *et al* in [7]). We found that the probabilities produced by SkILL are much closer to the expected values than the probability values produced by the Naive Bayes classifier, making SkILL's predictions much closer to the actual values that the physicians use to asses their patients. In other words, SkILL can produce predictors that are better calibrated than other traditional probabilistic models.

```
is_malignant(Finding) ←
  is_mass(Finding, Mass) ∧
  mass_shape(Mass, irregular) ∧
  mass_density(Mass, high) ∧
  mass_margin(Mass, microlob) ∧
  0.70::is_malignant(mass_margin(microlob))
is_malignant(Finding) ←
  is_mass(Finding, Mass) ∧
  mass_shape(Mass, irregular) ∧
  0.50::is_malignant(mass_shape(irregular)) ∧
  mass_margin(Mass, indistinct) ∧
  0.20::is_malignant(mass_margin(indistinct))
```

Fig. 1. Probabilistic hypothesis for malignancy of non-definitive biopsies

We also concluded that the hypothesis of length 2 presented in Figure 1 is only marginally better than the two hypotheses of length 1 composing it (probabilistic accuracies of 89.6% and 86.8% against the combined 90%). However, it is obvious that there exist problems that would greatly benefit from classifiers with multiple rules which can give more insight into the system being analysed. In this aspect, SkILL takes advantage of its clever search and combination of hypotheses, being able to explore a more qualitative portion of the full space, whilst being able to perform both classification and prediction, efficiently extending the classical ILP approach.

V. Related Work

The PILP setting was first introduced by Raedt and Kerstig in 2004 [17] and, in 2011, Raedt and Thon presented the first PILP system ProbFOIL [18]. ProbFOIL is capable of performing induction over probabilistic examples and on background knowledge encoded as ProbLog probabilistic facts. A number of relevant metrics such as precision, accuracy and m-estimate are adapted from the discrete ILP domain for use in the new setting, and ProbFOIL's search for a hypothesis is guided based on probabilistic accuracy of the theories. Recently, an extension of ProbFOIL was presented (ProbFOIL+) [2], which can also tune the probabilistic value of clauses to minimize the error. However, these systems do not take advantage of the probabilistic nature of the data to guide their search strategy, using instead an exhaustive approach. Additionally, and unlike SkILL, ProbFOIL and ProbFOIL+ don't support literals with constants as arguments, and can not handle mutually exclusive blocks of facts.

Probabilistic Explanation Based Learning (PEBL) [5] can find the most likely FOL clause which explains a set of positive

examples in terms of a database of probabilistic facts. The explanation clause is the combination of predicates which yields the highest probability based on the examples, and is found by constructing variabilized refutation proofs for the given examples using SLD resolution. However, since PEBL is a deductive system, information about the expected structure of the explanation should be provided as predicates (which are often recursive).

Orthogonally, Markov Logic Networks (MLNs) [19] also combine structure learning using a FOL framework with a probabilistic Markov Random Fields approach [19]. An MLN is a set of pairs of logic formulae and weights, where the latter are calculated based on the number of true groundings of the respective formula. Structure learning for MLNs softens the hypotheses by using probabilities and as such produces better classifiers, as shown in [19]. However, MLNs still consider crisp background knowledge, not taking into account the possibility of probabilistic logic facts. Additionally, and whilst MLNs are capable of structure learning, the final classifier is an MLN itself, which does not have the advantage of readability, especially when problem sizes are larger.

Finally, Meta-Interpretive Learning [13] – which is a technique aimed at performing predicate invention in ILP using abduction – can also be used to perform probabilistic structure learning by calculating prior and posterior distributions on the hypotheses space according to the examples explained by a given hypothesis [12]. This approach is similar to structure learning for MLNs in the sense that a relation exists between simultaneously grounded entities in the data and that hypotheses are ranked according to how many of these possible configurations they explain. However, probabilistic background knowledge is also not supported by meta-interpretive learning, since it does not support probabilistic facts.

VI. Conclusions

This work presents the PILP learner SkILL, which extends classic ILP learners by incorporating probabilistic facts and rules in its BK, as well as by using probabilistic examples. This system generates FOL rules (hypotheses) that can be used for classification and prediction and which produce probabilistic values. SkILL addresses efficiency issues in hypotheses generation by limiting the number of candidate hypotheses in its search space. This is done by selecting two populations according to different metrics, and only the combination of the members of those populations will be performed. Therefore, the number of SkILL's evaluation candidates does not increase exponentially, making it scalable for hypotheses containing several clauses. SkILL was compared against the PILP learner ProbFOIL+ using the family dataset and both systems were found to generate the same final hypotheses. SkILL's algorithm performed up to 2 orders of magnitude faster and presented the same accuracy as ProbFOIL+ for larger datasets. Finally, SkILL was used to extract non-trivial knowledge from a dataset of non-definitive biopsies which was annotated with probabilistic literature values and rules. Results showed that these annotations to the BK were in fact used in the physician's mental models and therefore useful for prediction of malignancy.

Acknowledgements

We would like to thank Vítor Santos Costa and Hendrik Blockeel for their suggestions. Joana Côrte-Real is funded by the FCT grant SFRH/BD/52235/2013. This work is financed by the FCT Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within project UID/EEA/50014/2013, and by NIH NLM grant R01-LM010921.

References

[1] V. Santos Costa, R. Rocha, and L. Damas. The YAP Prolog System. *Journal of Theory and Practice of Logic Programming*, 12(1 & 2):5–34, 2012.

[2] L. De Raedt, A. Dries, I. Thon, G. Van den Broeck, and M. Verbeke. Inducing Probabilistic Relational Rules from Probabilistic Examples. In *Proceedings of 24th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1835–1843. AAAI Press, 2015.

[3] P. Ferreira, N. Fonseca, I. Dutra, R. Woods, and E. Burnside. Predicting Malignancy from Mammography Findings and Image-Guided Core Biopsies. *International Journal of Data Mining and Biomedicine*, 11(3):257–276, 2015.

[4] J. Halpern. An Analysis of First-Order Logics of Probability. *Artificial intelligence*, 46(3):311–350, 1990.

[5] A. Kimmig, L. De Raedt, and H. Toivonen. Probabilistic Explanation Based Learning. In *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, pages 176–187. Springer, 2007.

[6] A. Kimmig, B. Demoen, L. De Raedt, V. Santos Costa, and R. Rocha. On the Implementation of the Probabilistic Logic Programming Language ProbLog. *Theory and Practice of Logic Programming*, 11(2 & 3):235–262, 2011.

[7] F. Kuusisto, I. Dutra, M. Elezaby, E. A. Mendonça, J. Shavlik, and E. Burnside. Leveraging expert knowledge to improve machine-learned decision support systems. In *Summit on Clinical Research Informatics*, pages 87–91. AMIA, 2015.

[8] F. Kuusisto, I. Dutra, H. Nassif, Y. Wu, M. Klein, H. Neuman, J. Shavlik, and E. Burnside. Using Machine Learning to Identify Benign Cases with Non-Definitive Biopsy. In *International Conference on e-Health Networking, Application & Services*, page 283–285. IEEE, 2013.

[9] N. Lavrac and S. Dzeroski. *Relational Data Mining*. Springer, September 2001.

[10] T. Mantadelis and G. Janssens. Nesting Probabilistic Inference. *Computing Research Repository*, abs/1112.3785, 2011.

[11] S. Muggleton. Learning Stochastic Logic Programs. *Electronic Transactions on Artificial Intelligence*, 4(B):141–153, 2000.

[12] S. Muggleton, D. Lin, J. Chen, and A. Tamaddoni-Nezhad. MetaBayes: Bayesian Meta-Interpretive Learning using Higher-Order Stochastic Refinement. In *Inductive Logic Programming*, pages 1–17. Springer, 2014.

[13] S. Muggleton, D. Lin, N. Pahlavi, and A. Tamaddoni-Nezhad. Meta-interpretive learning: application to grammatical inference. *Machine learning*, 94(1):25–49, 2014.

[14] S. Muggleton and L. De Raedt. Inductive Logic Programming: Theory and Methods. *Journal of Logic Programming*, 19/20:629–679, 1994.

[15] S. Muggleton, J. Santos, C. Almeida, and A. Tamaddoni-Nezhad. TopLog: ILP Using a Logic Program Declarative Bias. In *International Conference on Logic Programming*, pages 687–692. Springer, 2008.

[16] B. Poole, J. Wecsler, P. Sheth, S. Sener, L. Wang, L. Larsen, D. Tripathy, and J. Lang. Malignancy rates after surgical excision of discordant breast biopsies. *Journal of Surgical Research*, 195(1):152–157, 2014.

[17] L. De Raedt and K. Kersting. Probabilistic inductive logic programming. In *International Conference on Algorithmic Learning Theory*, pages 19–36. Springer, 2004.

[18] L. De Raedt and I. Thon. Probabilistic Rule Learning. In *Inductive Logic Programming*, pages 47–58. Springer, 2011.

[19] M. Richardson and P. Domingos. Markov Logic Networks. *Machine learning*, 62(1-2):107–136, 2006.

[20] T. Sato and Y. Kameya. PRISM: A language for symbolic-statistical modeling. In *International Joint Conference on Artificial Intelligence*, volume 97, pages 1330–1339. Morgan Kaufmann, 1997.

[21] J. Vennekens, S. Verbaeten, and M. Bruynooghe. Logic Programs with Annotated Disjunctions. In *Logic Programming*, pages 431–445. Springer, 2004.