Granular Computing and Imperfect Data in ILP

Chunnian Liu
School of Computer Science, Beijing Polytechnic University (BPU)
Beijing 100022, P.R. China, Email: cslcn@bjpu.edu.cn
Ning Zhong
Dept. of CSSE, Yamaguchi University
Tokiwa-Dai 2557, Ube 755, Japan, Email: zhong@ai.csse.yamaguchi-u.ac.jp

Abstract: *Inductive Logic Programming (ILP), as any other machine learning or KDD methods, has to deal with imperfect data when applied to real-world problems. Granular Computing (GrC) is a superset of various theories (such as rough sets, fuzzy sets and interval computation) used to handle incompleteness, uncertainty, vagueness, etc. in information systems. This paper investigates the feasibility of applying GrC (especially the rough set theory) to deal with imperfect data in ILP. We list various kinds of imperfect data in ILP (including noise data, too sparse data, missing data, indiscernible data, and too strict bias). For each kind of imperfect data, we try to point out the resulting problem and the potential solution using GrC (or a particular form of GrC such as the rough set theory). The presentation includes formalisms suggested by the authors or other researchers, as well as some general ideas which may give rise to more concrete results in future research.*

Keywords: ILP (Inductive Logic Programming), imperfect data, Granular Computing (GrC), Rough Set Theory.

1 Introduction

Inductive Logic Programming (ILP, see [2, 7, 8, 9]) comes from Logic Programming (LP) and can be regarded as a new method in machine learning with the advantages of more expressive power and ease of using background knowledge. If databases are involved, ILP is also relevant to Knowledge Discovery and Data Mining (KDD, see [1, 3]). In a simplified form, the normal problem setting of ILP is as follows:

Given:

- The target predicate p.
- The positive examples E^+ and the negative examples E^- (two sets of ground atoms of p).
- Background knowledge B (a finite set of definite clauses).

To find:

- Hypothesis H (the defining clauses of p) which is *correct* with respect to E^+ and E^-, i.e.
  1. $H \cup B$ is *complete* with respect to E^+ (that is: For all $e \in E^+$, $H \cup B$ implies e). We also say that $H \cup B$ *covers* all positive examples.
  2. $H \cup B$ is *consistent* with respect to E^- (that is: For no $e \in E^-$, $H \cup B$ implies e). We also say that $H \cup B$ *rejects* any negative examples.

To make the ILP problem meaningful, we assume the following prior conditions:

1. B is not complete with respect to E^+. (Otherwise there will be no learning task at all, because the background knowledge itself is the solution).
2. $B \cup E^+$ is consistent with respect to E^- (Otherwise there will be no solution to the learning task).

In the above normal problem setting for ILP, everything is assumed correct and perfect. But in large, real-world empirical learning, data are not always perfect. In contrary, uncertainty, incompleteness, vagueness, impreciseness, etc. are frequently observed in training examples, in background knowledge, as well as in the induced hypothesis. Thus ILP has to deal with imperfect data. In this aspect, the theory, measurement, techniques and experiences are much less mature for ILP than in the traditional attribute-value learning methods (compare [15], for example).

Another problem is that almost all practical ILP systems impose some restrictions on the hypothesis space and/or on the search strategies, in order to solve real-world problems with reasonable computing resource (time and/or space). These restrictions are called *declarative bias*. But if the bias is too strong, we may miss some useful solutions or have no solution at all.

All the problems concerning imperfect data or too strong bias in ILP have a common feature. In these situations, while it is impossible to differentiate distinct objects, we may consider granules – sets of objects drawn together by similarity, indistinguishability, or functionality. Even when precise solutions in terms of individual objects can be obtained, we may still prefer to granules in order to have an efficient and practical solution. The emerging theory of Granular Computing (GrC) (see [18, 19, 16, 17]) is a superset of various theories (such as rough sets, fuzzy sets and interval computation) used to handle incompleteness, uncertainty, vagueness, etc. in information systems. GrC grasps the essential concept – granules, and makes use of them in general problem solving. We believe that GrC (especially the rough set theory) can play an important role in dealing with imperfect data and/or too strong bias in ILP, and present in this paper some initial investigations. The main idea is, when we use granules instead of individual objects, we are actually relaxing the strict requirements in the standard normal problem setting for ILP, so that rough but useful hypotheses can be induced from imperfect data.

In the following sections, we will discuss various kinds of imperfect data in ILP (including noise data, too sparse data, missing data, indiscernible data, as well as too strong bias). For each kind of imperfect data, we try to point out the resulting problem and the potential solution using GrC (or a particular form of GrC such as the rough set theory). The presentation includes formalisms suggested by the author or other researchers, as well as some general ideas which may give rise to more concrete results in future research.

2 Insufficient Background Knowledge

Background knowledge B is essential in ILP learning, and the ease of using background knowledge is one of the major advantages of ILP over traditional attribute-value learning methods. However, if B lacks essential predicates (or essential clauses of some predicates), it is possible that no non-trivial hypothesis H can be induced. (Note that E^+ itself can be always regarded as a hypothesis, but it is trivial). For example, if two persons have the same descriptions other than their names, no meaningful hypothesis will be induced to distinguish them unless the background knowledge B contains some predicates/clauses that relate persons by their names in a sensitive way. In some cases, even a large amount of positive examples are given, some examples are not generalized by hypotheses if some background knowledge is missing. This has been a big topic in the research area of ILP. Recently, [6] uses *event calculus* to solve this problem.

We try to use a special form of GrC (rough sets) to provide a "rough" solution. Here the problem is that even a learning task can be expressed in the normal problem setting for ILP, it is possible that no meaningful hypothesis can be induced due to the lack of essential predicates/clauses in the background knowledge. In order to learn something useful in these cases, the requirement in the normal problem setting of ILP that H should be "correct with respect to E^+ and E^-" has to be relaxed. In [4], we propose the following rough problem setting for incomplete background knowledge, using Pawlak's rough sets [10, 11, 12]:

Given:

- The target predicate p (the set of all ground atoms of p is U)
- An equivalence relation R on U (we have the approximation space $A = (U, R)$)
- $E^+ \subseteq U$ and $E^- \subseteq U$ satisfying the prior condition: $B \cup E^+$ is consistent with respect to E^-
- Background knowledge B (may lack essential predicates/clauses)

Considering the following rough sets:

1. $E^{++} = \overline{apr}_A(E^+)$, containing all positive examples, and those negative examples $E^{-+} = \{e' \in E^- | \exists_{e \in E^+} eRe'\}$
2. $E^{--} = E^- - E^{-+}$, containing the "pure" (remaining) negative examples
3. $E_{++} = \underline{apr}_A(E^+)$, containing "pure" positive examples. That is: $E_{++} = E^+ - E^{+-}$, where $E^{+-} = \{e \in E^+ | \exists_{e' \in E^-} eRe'\}$
4. $E_{--} = E^- + E^{+-}$ containing all negative examples and "non-pure" positive examples

To find:

1. Hypothesis H^+ (the defining clauses of p) which is correct with respect to E^{++} and E^{--};
2. Hypothesis H^- (the defining clauses of p) which is correct with respect to E_{++} and E_{--}.

With this rough problem setting, we may induce H^+ which covers all positive examples and a few negative examples, rejecting other negative examples. We may also induce H^- which covers most positive examples, rejecting all negative examples. These hypotheses are rough (because the problem itself is rough), but still useful. On the other hand, if we insist in the normal problem setting for ILP, these hypotheses are not considered as "solutions".

In practical sense, this rough setting can provide a framework for the user to test various heuristics. Suppose that a learning task (in the normal setting) fails, and the user suspects that the background knowledge B is not sufficient. Then the user may "invent" some equivalence relation R among examples, and try the rough setting to see if some rough solutions can be induced. The test on the rough setting can also help the user to identify the missing part of B.

3 Indiscernible Examples

In the KDD community, people have to deal with situations such as: two patients showing the same symptoms have got different diagnostic results; two person records in a database having the same values for condition attributes have different decision attribute values; etc. In ILP, the similar situation is that example e belongs to E^+ as well as to E^-. Then the problem cannot be expressed in the normal problem setting at all, because the prior condition 2 ($B \cup E^+$ is consistent with respect to E^-) is violated. In order to learn something useful in these cases, the requirement in the normal problem setting of ILP that H should be "correct with respect to E^+ and E^-" has also to be relaxed. In [4] we also propose a rough problem setting for indiscernible examples, which turns out as a special case of the above rough setting:

Given:

- The target predicate p (the set of all ground atoms of p is U)
- $E^+ \subseteq U$ and $E^- \subseteq U$ where $E^+ \cap E^- \neq \emptyset$
- Background knowledge B

The rough sets to consider and the hypotheses to find:

- Taking the identity relation I as a special equivalence relation R, the remaining description is the same as in the above rough setting.

4 Too Strong Bias

Declarative bias (restrictions on the hypothesis space and/or on the search strategies) is necessary in any inductive learning. We can cite the famous statement: "No Bias, no prediction" [5]. But if the bias is too strong, we may miss some useful solutions or have no solution at all.

In his gRS-ILP model [13, 14], Siromoney characterizes this problem by (inherently) indistinguishable examples. He considers $S = (E, B, L)$ where L is the declarative bias. The equivalence relation R on S is defined as: eRe' iff for any induced logic program $P = H \cup B$, either both or none of e and e' are implied by P. If $e \in E^+$ and $e' \in E^-$, we have a rough setting. Note that here the equivalence relation R is conceptual (in most cases it cannot be computed), determined by the nature of the declarative bias L.

In the practical sense, the gRS-ILP model can be used to definitely describe a part of the input data (the examples E), when no hypothesis can describe the whole input data. For example, if one only insists in the consistency of P with E^-, ignoring its completeness with E^+, one may get a logic program P_{cons} which can definitely describe a part of the positive examples.

5 GrC for Other Imperfect Data in ILP

In this section, we consider other kinds of imperfect data in ILP, and point out how the underlying ideas of Granular Computing (GrC) are relevant to these situations, hoping that more concrete results will come in future research.

- Noise data.

  This includes erroneous argument values in examples, and/or erroneous classification of examples as belonging to E^+ or E^-. The ILP community has developed several *statistical and information theoretical measures* incorporated in search heuristics and/or stopping criteria to deal with noise during the search for a "correct" theory H (see [2] for details). As in the previous sections, here the underlying idea is also to weaken the requirement of 100% correctness with respect to the examples (allowing a few negative examples to be implied by $H \cup B$ and a few positive examples not to be implied) in order to avoid hypotheses overfitting the examples but having low prediction accuracy. It would be interesting to re-formulate these measures uniformly within the GrC framework, as [15] did for traditional attribute-value learning methods.

  Another approach is to detect and eliminate the noisy examples in a pre-processing phase, followed by a normal ILP session. The noise detection is based on Rissanen's *minimum description length* (MDL, see [2]). Because the actual MDL is usually not computable, approximation is needed. We believe that GrC can play a role in this aspect.

- Too sparse data.

  This means that the training examples are too sparse to induce reliable hypothesis H. The noise-handling mechanisms mentioned above usually also take care of too sparse data, because

the problems in both cases are similar: to avoid hypotheses overfitting the examples but having low prediction accuracy.

[20] propose a probabilistic mechanism considering unseen instances, as a way to deal with too sparse data. It would be interesting to reformulate the mechanism within the GrC framework and extend it to ILP.

- Missing values.

  This means that some arguments of some examples have unknown values. A simple way to deal with this problem is to induce a missing value from other examples, e.g. the value occurring in the same argument place of the majority of other examples. If we group examples having missing values together with corresponding normal examples by appropriate relations, the rough set theory can be applied to this problem.

- Discretization of continuous values.

  Logic Programming (hence ILP) is not good in handling continuous values. We may incorporate Constraint Logic Programming (CLP) into ILP (see [3]). Another approach is to discretize continuous values, and in this case granules appear. We can sort the examples according to the continuous argument A, and identify intervals of A so that within each of them the examples have the same classification (as positive or negative). Then we can replace each interval of A with a distinct value of a new discrete domain.

6 Conclusion and Future Work

Granular Computing (GrC) is a rich and flexible paradigm that may have a great impact on the study of intelligent information systems. This paper addressed the problem of imperfect data handling in Inductive Logic Programming (ILP) using some ideas, concepts and methods of GrC (or a particular form of GrC). Although some part of this work is still in the initial shape, we believe that the general ideas presented here may give rise to more concrete results in future research.

Future work in this direction includes:

- Trying to find more concrete formalisms and methods to deal with noise data, too sparse data, missing values, etc. in ILP within the framework of GrC.

- Giving quantitative measures associated with hypotheses induced of ILP, either in its normal problem setting or in the new rough settings.

- Developing ILP algorithms and systems based on rough problem settings.

Acknowledgements

C. Liu's work is supported by the Natural Science Foundation of China (NSFC), Beijing Municipal Natural Science Foundation (BMNSF) and Chinese 863 High-Tech Program.

References

[1] S. Dzeroski, "Inductive Logic Programming and Knowledge Discovery in Databases", In *Advances in KDD*, MIT Press, 117-151, 1996.

[2] N. Lavrac, S. Dzeroski, and I. Bratko "Handling Imperfect Data in Inductive Logic Programming", in L.de Raedt (Eds), *Advances in Inductive Logic Programming*, IOS Press, 48-64, 1996.

[3] C. Liu, N. Zhong, and S. Ohsuga, "Constraint ILP and its Application to KDD", In *Proc. of IJCAI-97 Workshop on Frontiers of ILP*, Nagoya, Japan, 103-104, 1997.

[4] C. Liu and N. Zhong, "Rough Problem Settings for Inductive Logic Programming", in *Proc. RSFDGrC'99*, Springer LNAI 1711, 168-177, 1999.

[5] T.M. Mitchell, *Machine Learning*, McGraw-Hill, 1997.

[6] S. Moyle and S. Muggleton, "Learning Programs in the Event Calculus", in *Proc. 7th International Workshop on ILP*, 205-212, 1997.

[7] S. Muggleton, "Inductive Logic Programming", *New Generation Computing*, 8(4):295-317, 1991.

[8] S. Muggleton (Eds), *Inductive Logic Programming*, Academic Press, 1992

[9] S-H. Nienhuys-Cheng and R. de Wolf, *Foundations of Inductive Logic Programming*, Springer LNAI 1228, 404 pages, 1997.

[10] Z. Pawlak, "Rough Sets", *International Journal of Computer and Information Science*, Vol.11, 341-356, 1982.

[11] Z. Pawlak, *Rough Sets: Theoretical Aspects of Reasoning about Data*, Kluwer Academic Publishers, Boston, 1991.

[12] Z. Pawlak, "Granularity of knowledge, indiscernibility and rough sets", In *Proc. 1998 IEEE International Conference on Fuzzy Systems*, 106-110, 1998.

[13] A. Siromoney, "A Rough Set Perspective of Inductive Logic Programming". In L. de Raedt and S. Muggleton (Eds), *Proc. the IJCAI'97 Workshop on Frontiers of Inductive Logic Programming*, 111-113, 1997.

[14] A. Siromoney and K. Inoue, "The Generic Rough Set Inductive Logic Programming Model and Motifs in Strings", In Zhong, N., Skowron, A., and Ohsuga, S. (eds.) "New Directions in Rough Sets, Data Mining, and Granular-Soft Computing", Springer LNAI 1711, 158-167, 1999.

[15] Y.Y. Yao and N. Zhong, "An Analysis of Quantitative Measures Associated with Rules", In N. Zhong and L. Zhou (Eds), *Methodologies for Knowledge Discovery and Data Mining*, Springer LNAI 1574, 479-488, 1999.

[16] Y.Y. Yao, "Granular Computing using Neighborhood Systems ", Roy, R., Furuhashi, T., and Chawdhry, P.K. (eds.) *Advances in Soft Computing: Engineering Design and Manufacturing*, Springer 539-553, 1999

[17] Y.Y. Yao and N. Zhong, "Granular Computing using Information Tables", *Granular Computing and Data Mining*, Physica-Verlag, 2000 (in press).

[18] L.A. Zadeh, "Fuzzy Sets and Information Granularity", in Gupta, N., Ragade, R. and Yager, R. (Eds.) *Advances in Fuzzy Set Theory and Applications*, North-Holland, Amsterdam, 3-18, 1979.

[19] L.A. Zadeh, "Toward a Theory of Fuzzy Information Granulation and Its Centrality in Human Reasoning and Fuzzy Logic", *Fuzzy Sets and Systems*, Vol.19, 111-127, 1997.

[20] N. Zhong, J. Dong, and S. Ohsuga, "Data Mining: A Probabilistic Rough Set Approach", In L. Polkowski and A. Skowron (Eds) *Rough Sets in Knowledge Discovery*, Physica-Verlag, 127-146, 1998.