Machine learning applications in anthropology: automated discovery over kinship structures

Sally Jo Cunningham
Department of Computer Science
University of Waikato
Hamilton, New Zealand
email: sallyjo@cs.waikato.ac.nz

Abstract: A common problem in anthropological field work is generalizing rules governing social interactions and relations (particularly kinship) from a series of examples. One class of machine learning algorithms is particularly well-suited to this task: inductive logic programming systems, as exemplified by FOIL. A knowledge base of relationships among individuals is established, in the form of a series of single-predicate facts. Given a set of positive and negative examples of a new relationship, the machine learning programs build a Horn clause description of the target relationship. The power of these algorithms to derive complex hypotheses is demonstrated for a set of kinship relationships drawn from the anthropological literature. FOIL extends the capabilities of earlier anthropology-specific learning programs by providing a more powerful representation for induced relationships, and is better able to learn in the face of noisy or incomplete data.

1. Introduction

Social anthropology is one of the least "mathematized" and "computerized" of the social sciences - a surprising fact, given the vast amount of quantitative data that field anthropologists record, classify, and sift. The relegation of mathematical anthropology

to a minority interest can be traced to a backlash against its earliest expression, in craniology and craniometry (the attempt to classify human races and cultures through measurement of skull capacities) (De Meur, 1986).

One of the main strongholds of mathematic research in anthropology has been the investigation of mathematical structures underlying kinship relationships (see, for example, De Meur, 1986; Ascher, 1991). These investigations have primarily been directed at providing techniques for visualizing and generalizing over complex relationships. The rules for determining descent and relationships have been a primary factor in organizing societies, by regulating permissible marriages, social interactions, inheritance of property, and a host of other social ties. Unfortunately for field workers, these rules are often not directly codified as a set of general rules or laws, as is common in literate societies; rather, the relationships may be specified as a set of positive and negative examples, with the generalized form of the relation implicit in these examples. For example, Findler (1992) cites a Papua New Guinea tribe in which "extremely forceful" injunctions against a male having sexual relationships with his son's wife or mother-in law could be expressed to the field worker only by naming the people who could and could not engage in sexual intercourse.

Further, when general rules are available they may not concisely or accurately describe the *actual* relationships existing between members of that society. An ideal marriage partner, for example, may be a paternal cousin; this ideal may be reported to the anthropologist studying that social group, but data on actual marriages in the society reveals other (perhaps as common!) prior relationships between husband and wife. A kinship rule may also be specified in an overly general, again idealized form: a rule among the Mundugumor, for example, that a man should marry his father's father's father's father's sister's son's son's daughter has been shown logically (and in practise) to reduce to marrying his father's father's sister's son's daughter (Gregory, 1986).

Among the few specifically anthropological computer programs, several deal with deriving these types of implicit relationships from field data. These programs were developed by N.V. Findler and his associates over several decades (Findler and McKinzie, 1969; Findler, 1973; Findler, 1992a-b). The earliest software support offered the ability to query an arbitrary set of kinship relations, given a directed graph of "birth information" (name, gender, and parents' names) and "marriage information" (husband/wife names). Later programs added additional operators to support the definition of more complex kinship ties (Findler, 1973). This software was still restricted to tracing known relationships, however; the user specified a logical definition of a relationship - for example, that a grandfather is a father's father or mother's father - and the database of relational descriptions could be queried for fully or partially instantiated patterns of that type (for example, "who is the grandfather of Joe?"). A final program provides an additional learning facility that can form hypotheses of relationship rules given a set of positive and negative examples (Findler, 1992). The program learns the smallest logical subset of the directed graph of kinship relationships that covers the example set, by eliminating arcs labelled with values irrelevant to the exemplars. The algorithm used cannot deal with noisy data, and cannot directly deal with recursive relationships.

This paper considers the ability of a general-purpose machine learning tool to process this type of anthropological data: relational learning (also called inductive logic programming) schemes, as exemplified by FOIL (Quinlan, 1990). Interestingly, FOIL was originally conceived as a technique for extending then-current machine learning algorithms to describe a set of simple, Western kinship relations; FOIL was shown to provide more accurate classifications on this task than earlier neural network programs (Quinlan, 1990; Hinton, 1986). These elementary kinship ties continue to be used frequently in illustrating inductive logic programming algorithms, but this learning task has been treated as a "toy" and has not been seriously studied in the machine learning literature.

Indeed, surprisingly few "real world" applications of FOIL and its descendants are reported. It appears that the additional inductive power offered by inductive logic programming algorithms is not required to process many existing data sets – a conjecture borne out by a recent, startling discovery that a classifier that bases its decisions on a single attribute often performs as well as more sophisticated algorithms! At least part of the reason for this result is that many of the standard test databases used in the machine learning community embody very simple underlying structures (Holte, 1993). Ironically, the "toy" kinship learning problems may be among the few existing types of data sets that are particularly well-suited to relational learning programs.

This paper is organized as follows: Section 2 describes FOIL in more detail; Section 3 presents a set of kinship description predicates derived from commonalities in kinship relationships presented in the anthropological literature; Section 4 discusses the applicability of FOIL to a variety of kinship relationships, and explores its suitability as a general tool for anthropologists; and Section 5 presents our conclusions.

2. Inductive Logic Programming and FOIL

Most learning programs arising out of the machine learning community have been limited to performing induction over a single table of values: objects are modelled as vectors of attribute/value pairs, with each object in the domain possessing the same number and type of attributes. Each object in the set of examples (the training set) is assigned to exactly one of a mutually exclusive and exhaustive set of classes, and this training set is generalized to classify new examples. The generalization is usually in the form of a decision tree or rule set for predicting a novel object on the basis of its attributes.

While this representation is sufficient for many domains, it is clearly inadequate for describing kinship patterns. Simply recording parent-child relationships is difficult: if each row in the table describes an individual and his/her children, then each child must be recorded as a separate attribute value. However, people have different numbers of children, so a fixed upper bound of attributes must be specified. For example, given an upper limit of five children we could represent a given parent-child relationship as:

% John has three children: Ann, Bill, and Susan

parent	child1	child2	child3	child4	child5
john	ann	bill	susan	n/a	n/a

But kinship relations typically cover more than one type of link between individuals; for example, an aunt is the sister of a mother or father. These additional link types would have to be added as attributes to each tuple. Further, consider the less specific relationships such as "ancestor". Representation of a general concept such as this would be exceedingly awkward, if not impossible.

FOIL provides a more powerful object representation than the flat table of attribute values. Objects are described by what is essentially a set of tables – multiple predicates in the form of Prolog-like "facts" (tuples satisfying the associated predicate). For example, the fact that Jack is the parent of three children, is male, and is married to Jill could be expressed by:

```
parent (jack, ann)
parent (jack, bill)
parent (jack, susan)
husband (jack, jill)
male (jack)
```

FOIL accepts as input a set of predicate tuples (which may include more than one relation), and a set of examples of the target relation (the relation to be generalized). The system then builds a function-free Horn clause description of the target (essentially, a program written in a subset of Prolog that describes the relation). For example, suppose we wish to induce a description of "father" from a set of data including

information above such as the parentage, marriage, and sex of each person. In addition to these kinship tuples we also provide a set of examples of "father" tuples (eg, "father (jack, bill)"). FOIL will then induce the following Horn clause description:

father(A,B) :- parent(A,B), not(female(A)).

Literally, this rule states that A is defined to be a father of B if A is a parent of B and A is also not female. Notice that the input may include irrelevant predicates, such as "husband", which are discarded by FOIL. Induced rules may include negated predicates, as above, but cannot include predicates containing constants (such as parent (A, Tom)). FOIL can perform comparisons between numeric values, but not other numeric operations (a common weakness in machine learning algorithms).

FOIL is also able to handle limited amounts of noise in the data. Suppose, for example, that a list of seven “father” tuples contains the tuple (jane, lloyd), where jane has been described as female and the parent of lloyd. If the example set contains “enough” correct data to counterbalance the erroneous tuples, then FOIL will induce a rule set and give notice of example tuples not explained by the rules. For this case, the output appears as:

```
***Warning:  the following definition
***does not cover 1 tuple in the relation

father(A,B) :- parent(A,B), male(A).
```

Note that the definition has changed slightly, from stating that a father is not female to stating that a father must be male, but that the final result is still correct in spite of the noise. Unfortunately, noise may have a more serious effect on the rule sets, introducing incorrect clauses and predicates to cover the noisy data. This problem is most serious when the noise level is relatively high, or when the pattern of noise causes the data to appear to represent a different concept from the one intended.

3. Kinship structures: the predicate descriptions underlying induction over kinship ties

The predicate descriptions modelling kinship ties will dictate the types of kinship structures that can be induced. An examination of the anthropological literature suggests that the set of examples should contain the following information about each person in the kinship set, as a minimum, in order to be able to infer the richest variety of relationships:

- a unique identifying symbol for each individual (usually the person's name)
- the individual's sex
- the individual's age
- primary biological kinship ties (father, mother, son, daughter, brother, sister)
- direct marriage ties (husband/wife)
- prior relations, if they remain in force (earlier, generally pre-marriage or prior marriage, ties by blood or matrimony)
- class membership (participation in tribe, moiety, clan, etc.)
- status (whether the individual is still living, or is deceased)

These predicates and their instantiations specify a *kinship structure* – a network in which nodes represent individuals, and connecting arcs are labelled with the relationship between the two individuals. In addition, each node is labelled as well

(with name, sex, age, and class membership information). Note that this formulation is a directed graph, as the predicates are not commutative (ie, the unidirectional predicates *husband(A,B)* and *wife(B,A)* are used, rather than the less specific, bidirectional *spouse(A,B)*). In specifying the primary kinship ties, some redundancy will improve induction efficiency: rules for brother/sister relationships can, for example, be induced from parent/child information, but if sibling ties appear often in other kinship definitions then it may be more effective to limit inductive searches by explicitly including brother/sister predicates. As will be discussed below, this type of redundancy may also encourage induction of rules that are more accurate (in the sense of being semantically closer to culturally defined rules, rather than in the sense of correctly classifying existing examples).

This information is not exhaustive; some relations may exist that cannot be characterized by the above set of predicates. This set does, however, cover the underlying relationships and individual information necessary to define most kinship ties (Findler, 1992a).

4. Representational issues for kinship data: FOIL's suitability for anthropological machine learning

As discussed above, FOIL was originally designed to learn Western kinship relations over a toy set of 104 relationships among 24 individuals (Quinlan, 1990). The appropriateness of this type of representation for non-Western relations has not been explored, nor have inductive logic programming algorithms been applied to real genealogical databases. In this section, we consider a set of complex relationships drawn from the anthropological literature, and discuss the ability of FOIL to generalize these kinship patterns.

example 1: age-determinate relationships

While Western relationships do not incorporate age comparisons between individuals, other cultures often do – for example, the Confucian distinction between an older brother and a younger brother. Given a set of kinship structure predicates as described in Section 3 and a set of positive examples of older brother/younger brother pairs, FOIL induces the definition that A is the older brother of B if the age of A is greater than the age of B, and if A and B are brothers:

olderbrother(A,B) :- age(A,C), age(B,D), C>D, brother(A,B), brother(B,A).

In the above example, a "brother" predicate exists in the kinship structure. Logically this predicate is redundant, and the system should be able to construct a correct definition in its absence. Running the same example set with the "brother" predicate removed from the kinship network, FOIL derives the following rule:

olderbrother(A,B) :- age(A,C), age(B,D), father(E,A), C>D, father(E,B).

Note that the concept of "brother" is indirectly defined as "two individuals with the same father". The original example set did not include negative examples sister/brother pairs, and so the rule above does not require "brothers" to be male!

Further, it is difficult to tailor the example set to force FOIL to induce the "proper" rule definition in this case. Additional runs were conducted over different example sets, again consisting of correct positive examples; these sets, however, also contained male/female predicates for all individuals and sister/brother pairs. The runs produced the following definitions

(i) olderbrother(A,B) :- age(A,C), age(B,D), father(E,A), mother(F,A), D<=10, male(A).

(ii) olderbrother(A,B) :- age(A,C), father(D,A), father(D,B), C>29.

These rules correctly categorize this particular set of kinship data, but do not correspond to the commonplace definition of "older brother". Several lessons can be drawn at this point. The example set has to be carefully chosen to illustrate relationships that both must and must not occur in the defined kinship tie; positive examples alone (here, of brother pairs) may not be sufficient to permit FOIL to induce the desired concepts. The primary relationships used to represent the kinship network directly affect the semantic plausibility of induced rules, so that including a richer set of base relations (such as “male” and “female”) can allow FOIL to come closer to expressing the underlying associations between pairs in an example. Finally, it may not be possible to infer the “commonplace definition” from a limited set of examples; FOIL may be sidetracked into modelling incidental relationships that appear significant only because of the small size of a data set. For example, rules (i) and (ii) include specific age information that appears because, for the instances in the data sets, the younger brother always happens to be no older than 10, while the older brother always happens to be older than 29. One heuristic for detecting when FOIL has located a good description of the relationships implicit in the data might be to continue to add cases to the example set until the answer stops changing.

example 2: disambiguating relationships with multiple definitions

Some complex relationships may have more than one definition. For example, Surinamese Negro dialects do not distinguish between "mother's brother's daughter" and "wife", or between "mother's brother's son" and "brother-in-law" (Findler, 1992a-b). Given only positive examples, FOIL constructs the following rule sets for these two relationships:

mother's brother's son == brother-in-law

relation1(A,B) :- wife(C,A), brother(C,B).
relation1(A,B) :- father(C,B), mother(D,A), male(B), brother(C,D).

mother's brother's daughter == wife

relation2(A,B) :- wife(A,B).
relation2(A,B) :- father(C,B), mother(D,A), brother(C,D).

Here, FOIL correctly characterizes the two kinship ties underlying both of these Surinamese relationships. Note that the learning algorithm is powerful enough to create multiple rules categorizing different cases of the same relationship.

example 3: defining exogamy and incest

All cultures enforce rules on permissible and impermissible marriage partners. FOIL is able to induce exogamy and incest characterizations that are positively stated, but is not as effective in determining characterizations dependent entirely on the absence of a tie in the kinship structure. For example: in most Western societies a relationship is incestuous if it involves any of the following kinship ties: brother, sister, aunt, uncle, mother, father, grandparent, and cousin (matrilineal and patrilineal). Given a sufficiently rich example set illustrating each of these types of incestuous relationships, FOIL will induce a set of rules covering each category of incest.

However, suppose the learning task is to induce the Western definition of exogamy: a marriage is permissible between individuals related as second cousins *or to a lesser extent, or who are unrelated*. FOIL can build a rule set covering each of the relations covered in the example set: second cousins, third cousins, etc. FOIL cannot, however, use these examples to induce general rules stating "less related than second cousins", or "not related at all".

example 4: how many wives can the tribal chief have?

In some cultures the number of wives a man is permitted depends on his social, political, or religious status – for example, only the tribal chief may be allowed to have more than one wife (Findler, 1992a-b). Suppose we attempt to induce this definition of a chief, given a set of positive examples of chiefs who have zero, one or more wives and a set of negative examples of non-chiefs having one or no wife. The above kinship structure is not sufficient for this task; each husband/wife pair is represented as a separate predicate tuple, and FOIL cannot count or sum tuples. We must add another predicate, *number_of_wives(A,B)* (stating the number of wives (B) that person A currently is married to). Given the above example set, FOIL then produces the following rule:

chief(A) :- number_of_wives(A,B), B>=0.

Characteristics of a non-chief can be similarly induced:

non_chief(A) :- number_of_wives(A,B), B<=1.

For FOIL to succeed in this example, the user must know that the number of wives is important in characterizing a chief and add that information to the kinship structure. As noted above, the general kinship structure is not sufficient to support induction of all possible relationships, and may need to be augmented with special information for a given induction task. Machine learning, like human learning, is most successful when the learner very nearly knows what it is that isn't known.

Conclusions

How does FOIL compare to induction programs developed specifically for anthropologists? Findler's set of kinship learning programs include task-specific functions such as the ability to list all information available for a given individual, or to display connecting links between two people (Findler and McKinzie, 1969; Findler, 1973; Findler, 1992) – a capability beyond the scope of a general purpose induction program like FOIL. FOIL and Findler's programs use similar operators in defining kinship structures, and permit the user to present both positive and negative examples of the relation to be defined. However, the output formats of Findler's programs are limited to sample graphs illustrating the induced relation(s), while FOIL presents its inductions in the more powerful Horn clause format, which can include recursive relations and negated predicates. Finally, Findler's programs cannot induce over noisy data, while FOIL can handle limited amounts of noise (both in the form of inaccurate and incomplete data). Incomplete data can cause FOIL to derive rules that, while they accurately describe the example set, may not correspond to a common sense understanding of the domain. If the level of errors in the example set is low, then FOIL *may* be able to produce correct results in spite of the noise; the truism still applies, however, that better quality data is likely to yield better quality information.

A major weakness of FOIL, shared by the learning component of Findler's programs, is an inability to incorporate existing domain theory in any other form than predicate descriptions of objects. As discussed above, this limitation may require the user to add redundant information to the kinship structure (eg, explicitly stating brother/sister relationships rather than forcing the system to induce this relationship, perhaps incorrectly, each time it re-appears in another kinship definition). FOCL, a descendant of FOIL, addresses this problem by allowing the learning algorithm to use intensionally defined predicates (predicates defined by a rule, rather than a set of examples). This

additional capacity permits FOCL to re-use definitions induced in earlier runs of the program as well as information directly supplied by the user.

How well-suited is FOIL to inducing information in the anthropological domain? FOIL's strengths lie in symbolic induction, rather than numeric manipulation, and in its ability to induce over multiple relations (predicates). These capabilities are a good match to many problems in the domain of social anthropology: social relations are characterized by a rich set of ties that must be represented in the example set or background data, a wide variety of inducible relationships that require a powerful representation such as the Horn clause to express them, and a generally symbolic rather than numeric structure.

Success in learning over kinship structures, like any machine learning or statistical analysis task, requires an adequate domain theory to support the types of learning required. Further, the set of positive and negative examples of the target relation must be carefully chosen as well, so that the pertinent facets of the underlying concept are represented. Neither modelling task is trivial – a point glossed over in earlier discussions of applications of machine learning to this type of data.

Examples in this paper and others illustrate FOIL's ability to accurately induce relationships based on primarily genealogical data (see, for example, Quinlan, 1990). This capability can be useful for anthropologists both to induce descriptions of new, previously uncharacterized relationships, and as a machine learning tool to confirm or refute known rules through their instantiations in the data. This latter use may ultimately prove to be more practical; as discussed in the introduction, the socially defined rules may not concisely or accurately describe the *actual* relationships between members of that society. FOIL provides a powerful tool to induce these existing relationships.

Acknowledgments: I would like to thank the anonymous referees for their excellent comments, which were very helpful in improving this paper.

References

Ascher, M. *Ethnomathematics: a multicultural view of mathematical ideas*. Pacific Grove, CA, USA: Brooks/Cole Publishing Company, 1991.

De Meur, Gisele, ed. *New Trends in Mathematical Anthropology*. London: Routledge & Kegan Paul, 1986.

Findler, N.V. and McKinzie, W.R. "On a computer program that generates and queries kinship structures." *Behavioral Science,* 14 (1969), pp. 334-343.

Findler, N.V. (1973): "Kinship structures revisited." *Behavioral Science,* 18 (1969), 68-71.

Findler, N.V. "Automatic rule discovery for field work in anthropology." *Computers and the Humanities,* 26(4) (1992a), 285-292.

Findler, N.V. "An excursion into social and cultural anthropology by artificial intelligence – an automated discovery system to identify rules of inheritance, succession, marriage, injunction against incest, and exogamy." *Computers in Human Behavior,* 8 (1992b), 367-377.

Gregory, C.A. "A matrix approach to the calculus of kinship relations." In De Meur (1986), pp. 139-166.

Hinton, G.E. "Learning distributed representations of concepts." *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*, Lawrence Erlbaum Press, 1986.

Holte, R.C. “Very simple classification rules perform well on most commonly used datasets.” *Machine Learning*, 11 (1993), 63-91.

Quinlan, J.R. "Learning logical definitions from relations." *Machine learning*, 5 (1990), 239-266.