Extraction of Spatial Proximity Patterns by Concept Generalization

Edwin M. Knorr and **Raymond T. Ng**
Department of Computer Science
University of British Columbia
Vancouver, B.C., V6T 1Z4, Canada
{knorr,rng}@cs.ubc.ca

Abstract

We study the spatial data mining problem of how to extract a special type of proximity relationship—namely that of distinguishing two *clusters* of points based on the types of their neighbouring *features*. The points in the clusters may represent houses on a map, and the features may represent spatial entities such as schools, parks, golf courses, etc. Classes of features are organized into concept hierarchies. We develop algorithm GenDis which uses concept generalization to identify the distinguishing features or concepts which serve as discriminators. Furthermore, we study the issue of which discriminators are "better" than others by introducing the notion of maximal discriminators, and by using a ranking system to quantitatively weigh maximal discriminators from different concept hierarchies.

Introduction

In recent years, there has been considerable research in detecting patterns hidden in data (Agrawal *et al.* 1992; Agrawal, Imielinski, & Swami 1993; Borgida & Brachman 1993). A reasonable and rather popular approach to spatial data mining is the use of clustering techniques to analyze the spatial distribution of data (Ng & Han 1994; Ester, Kriegel, & Xu 1995; Zhang, Ramakrishnan, & Livny 1996). While such techniques are effective and efficient in identifying spatial clusters, they do not support further analysis and discovery of the properties of the clusters. To this end, we have developed an approximate, but efficient, algorithm (Knorr 1995) to discover knowledge about the clusters by analyzing the features that are in close proximity to the clusters. More specifically, given a spatial cluster Cl, the algorithm finds the top-k features that are closest to Cl in an aggregate sense. An aggregate notion of proximity is needed because the distribution of points in a cluster may not be uniform. For example, a particular golf course may appear in a cluster's top-10 list if the golf course is relatively close to many of the houses in the cluster. On the other hand, a particular shopping centre which is actually closer to the cluster (in terms of feature boundary to cluster boundary distance) may not appear in the top-10 list if few houses are relatively close to the shopping centre.

It is also important to identify common *classes* of features which are in close proximity to most (or all) of the input clusters (Knorr & Ng 1996). This notion of *commonality extraction* is important because, for example, it is often unlikely that one particular golf course is close to every cluster, even though each cluster may have *some* golf course close to it—though not necessarily the same one. If such is the case, then a generalized statement can be made concerning the fact that the clusters tend to be near golf courses. Such statements can be useful in terms of knowledge discovery because they describe generic types of features common to multiple clusters.

While aggregate proximity relationships and commonality extraction can be quite valuable, we are also interested in determining the distinguishing features (or classes of features) between two clusters. For example, if an expensive housing cluster and a poor housing cluster are given as input, we may find that concepts such as "golf courses", "private schools", and "social services centres" are discriminators, in which the expensive housing cluster is close to a golf course or a private school, but the poor housing cluster is not; and, the poor housing cluster is close to a social services centre, whereas the expensive housing cluster is not.

In this paper, we describe an algorithm called GenDis, which finds "discriminating" classes of features that serve to distinguish one cluster from another. We use concept generalization to extract the discriminators. Attribute-oriented concept generalization (Han, Cai, & Cercone 1992; Lu, Han, & Ooi 1993) has been shown to be quite useful in guiding the discovery of general patterns. The work in this paper differs from the attribute-oriented approach in a number of

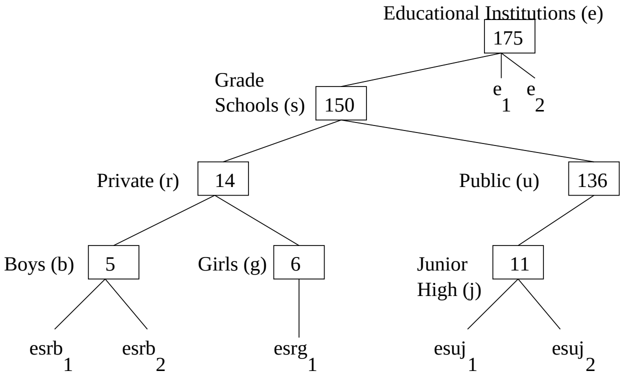

Figure 1: Educational Institutions Concept Hierarchy

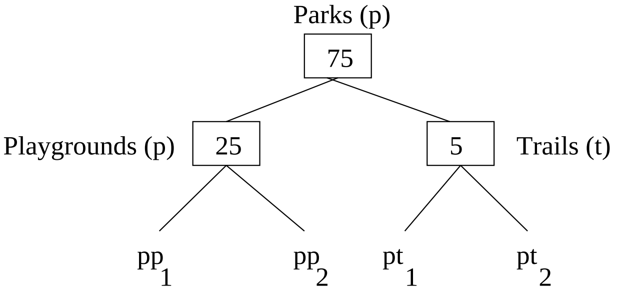

Figure 2: Parks Concept Hierarchy

ways. First, to identify discriminating patterns, we do not use set differences and thresholds. Second, our notion of maximal discriminators is unique. Finally, we introduce a way of ranking the discriminating concepts.

Concept Hierarchies

In a GIS context, we define a feature as a natural or man-made place of interest. Natural features may include mountains, lakes, islands, etc., and man-made features may be schools, parks, golf courses, shopping centres, etc. We define a *concept* to be a class of features. Each concept is part of some hierarchy. The trees shown in Figures 1 and 2 are two concept hierarchies, to which we will refer throughout this paper. In the educational institutions hierarchy, one subclass of educational institutions (shorthand "e") is grade schools ("s"). Grade schools in turn are classified into private ("r") and public ("u") schools, which can be further sub-classified into less general concepts. Specific instances of schools appear at the leaf level. The leaves are considered to be trivial concepts. We use shorthand notation to identify any node in a tree. For example, in Figure 1, the feature $esrb_1$ is a boys' private grade school, and the feature e_1 is an educational institution that is not a grade school (e.g., art school, university, technical college). The cardinalities of the concepts are also listed. In our example, there are 175 educational institutions, 150 of which are grade schools. Of those 150 grade schools, 14 are private grade schools—and of those 14, 5 are exclusively for boys, 6 are exclusively for girls, and 3 are co-ed (not shown). For simplicity, only those concepts that are relevant to our discussion are shown—namely those concepts relating to specific features which appear in at least one of the original top-k lists. Recall that a top-k list for a given cluster contains the k features "nearest" the cluster—nearest in an aggregate sense.

Algorithm GenDis: Extraction of Maximal Discriminators

Motivation and Definition

Due to limited space, we limit our discussion to the extraction of patterns for "dissimilar" clusters. More specifically, given two clusters as input, we aim to find discriminating features, or simply *discriminators*, that distinguish one cluster from the other.

A natural way of detecting discriminators is to use set differences on the two lists. (This is the underlying principle used in the attribute-oriented approach.) Consider the concept hierarchies in Figures 1 and 2, and:

- suppose the top-k list associated with an expensive housing cluster Cl_e contains $esrb_1$, $esrg_1$, and pp_1 (plus a number of features that are found in the top-k list of a poor housing cluster Cl_p)
- suppose the top-k list associated with Cl_p contains pt_1 and $esuj_2$ (plus a number of features common to Cl_e)

Set differences on these two lists yield all 5 mentioned features as discriminators. The presence of $esrb_1$, $esrg_1$, or pp_1 distinguishes Cl_e from Cl_p—and the presence of pt_1 or $esuj_2$ distinguishes Cl_p from Cl_e. While this approach of using set differences is easy to compute, drawing distinctions based solely on individual features can be somewhat limiting. For example, closer scrutiny of these 5 features and the clusters to which they belong reveals that private schools are close to the expensive cluster, but not to the poor cluster; and that a public school is close to the poor cluster, but not to the expensive cluster. Thus, while $esrb_1$ and $esuj_2$ are both educational institutions, and pp_1 and pt_1 are both parks, a key question is: *how different* is $esrb_1$ from $esuj_2$, and pp_1 from pt_1? This question motivates our study of maximal discriminators, defined below.

Concept generalization can help answer the "how different" question by highlighting the differences between features. For example, the difference between

$esuj_2$ and $esrb_1$ is that the former is a junior high public school, whereas the latter is a boys' private school. This observation can be obtained by generalizing $esuj_2$ to $esuj$, and by generalizing $esrb_1$ to $esrb$. This leads to two questions. First, in ascending the concept hierarchy, how many levels of generalization are most appropriate? We defer the answer to the next paragraph. Second, is this kind of highlighting by generalization always possible? To answer this question, suppose in our example that $esuj_2$ were $esrb_2$ instead. Although $esrb_1$ and $esrb_2$ are distinct entities, generalizing both features yields the same class of boys' private schools. In effect, rather than highlighting the differences, generalization in this case underscores the similarities or the lack of differences between the features. Thus, in evaluating the differences between features, concept generalization is useful in both highlighting the differences and identifying the lack of differences, whatever the case may be.

To capture the essence of the above discussion, and to determine the appropriate number of levels of generalization, we use the notion of the *smallest common ancestor* of a set of nodes in a tree. More formally, if $F_1, \ldots, F_u$ are all features in the same concept hierarchy, the smallest common ancestor of $F_1, \ldots, F_u$, denoted by $sca(\{F_1, \ldots, F_u\})$, is the root of the smallest subtree containing $F_1, \ldots, F_u$. Now, suppose F and G are two sets of features from the same concept hierarchy. We define the *maximal discriminator* of F and G, denoted by $md(F, G)$, as follows:

- If the subtree rooted at $sca(F)$ contains $sca(G)$, or if the subtree rooted at $sca(G)$ contains $sca(F)$, then $md(F, G)$ is NULL.

- Otherwise, let F' be the child of $sca(F \cup G)$ such that the subtree rooted at that child contains $sca(F)$, and let G' be the child of $sca(F \cup G)$ such that the subtree rooted at that child contains $sca(G)$. Then, $md(F, G)$ is the pair $\langle F', G' \rangle$.

For example, consider Figure 1 and the sets $F = \{esrb_1, esrg_1\}$ and $G = \{esuj_1, esuj_2\}$. By definition, $sca(F)$ is esr, $sca(G)$ is $esuj$, and $sca(F \cup G)$ is es. Furthermore, F' is esr and G' is esu. Thus, the maximal discriminator is $\langle esr, esu \rangle$, which corresponds to private schools and public schools—the observation we want, as discussed above.

Consider $md(\{esrb_1\}, \{esuj_1, esrb_2\})$ as another example. This time $sca(\{esuj_1, esrb_2\})$ is es, whose subtree contains $esrb_1 = sca(\{esrb_1\})$. Thus, the maximal discriminator in this case is NULL, indicating that the sets $\{esrb_1\}$ and $\{esuj_1, esrb_2\}$ are not considered to be sufficiently different.

```
1        Initialize answer set S to empty set
2        For each concept hierarchy
2.1       Let F be the set of features from
              this hierarchy for one cluster
2.2       Let G be the set of features from
              this hierarchy for the other cluster
2.3       If both F and G are empty
2.3.1         goto 2.6
2.4       If either F or G is empty
2.4.1           add <C,nil> to S, where C is
                 the root of the concept hierarchy
2.5       else
2.5.1           compute md(F,G) as defined
2.5.2           if md(F,G) is not null
2.5.2.1            add md(F,G) to S
2.6      End-for
3        Compute and report the final rankings of
          the discriminators in S
```

Figure 3: Algorithm GenDis for Extracting Maximal Discriminators

One may wonder what the word "maximal" in maximal discriminator means. It is used to describe the situation in which the sets F and G are generalized to the fullest possible extent. Any further generalization will render the sets identical (corresponding to $sca(F \cup G)$). Thus, the maximal discriminator reports the broadest difference between two sets. In our first example, the broadest difference is simply the distinction between private and public schools—as indicated by $md(F, G) = \langle esr, esu \rangle$.

A useful by-product of the notion of smallest common ancestors and maximal discriminators is the ability to report more specific information. In general, by following the path from F' to $sca(F)$, and by following the path from G' to $sca(G)$, we get more specific levels of distinction. The most specific level of distinction occurs with the pair $\langle sca(F), sca(G) \rangle$; however, in practice, if $sca(F)$ (or $sca(G)$) is a leaf, then it may make more sense to report the parent of the leaf. For example, suppose $F = \{esrb_1\}$ and $G = \{esuj_2\}$, and suppose "St. George's School" is the name of feature $esrb_1$ and "Pierre Elliot Trudeau School" is the name of feature $esuj_2$. A user who is unfamiliar with these feature names may prefer to see a more general level of distinction—namely, the distinction of a boys' private school versus a junior high public school. We leave the desired level of distinction as an application issue, but mention it for completeness.

Algorithm GenDis

Figure 3 presents the outline of Algorithm GenDis for extracting maximal discriminators for two clusters, using multiple concept hierarchies. Let us apply the algorithm on clusters $Cl_e = \{esrb_1, esrg_1, pp_1\}$ and $Cl_p = \{pt_1, esuj_2\}$. Suppose the first iteration of the for-loop considers the educational institutions hierarchy, in which $F = \{esrb_1, esrg_1\}$ and $G = \{esuj_2\}$. In Step 2.5.2.1, $md(F,G)$ is the pair $\langle esr, esu\rangle$, which is added to the answer set S. The pair corresponds to private schools and public schools, which highlight the distinction (in terms of kinds of features) between the two clusters. In the next iteration, the parks hierarchy is used, in which $F = \{pp_1\}$ and $G = \{pt_1\}$. From Figure 2, $md(F,G)$ is the pair $\langle pp, pt\rangle$, which corresponds to playgrounds and trails. Thus, $\langle pp, pt\rangle$ is added to S, yielding a second discriminating class of features.

Like the situation for identifying and quantifying commonalities (Knorr & Ng 1996), maximal discriminators from different concept hierarchies should be ranked (i) to give an idea of how strong the discriminators are, and (ii) to take into account the varying cardinalities of different concepts and hierarchies. This can be done as follows. Given the pair $\langle F', G'\rangle$ as a maximal discriminator, the score is defined by the maximum of the cardinalities of F' and G', normalized by the total cardinality of the concept hierarchy. These scores are then ranked. For example, from Figures 1 and 2, we see that the score for $\langle esr, esu\rangle$ is 136/175 (approximately 0.78), and the score for $\langle pp, pt\rangle$ is 25/75 (approximately 0.33). Although the scores depend on the cardinalities and granularities of the various concept hierarchies, larger scores are generally favoured since they often reflect the fact that different types of discriminating features having relatively low probabilities of occurrence appear in both clusters.

The score for $\langle C, nil\rangle$, where C is the root of a concept hierarchy (e.g., $\langle e, nil\rangle$) is 1 because, as shown in Step 2.4.1 of Algorithm GenDis, this corresponds to a situation where concepts from the hierarchy rooted at C appear in one of the two top-k lists, but not both. Of course, those cases where $md(F,G)$ is NULL are not included in the list of discriminators—and hence, the rankings—since F and G are not considered to be sufficiently different.

From a complexity standpoint, smallest common ancestors can be computed in $O(1)$ time, with $O(n)$ pre-processing time, where n is the total number of nodes (Harel & Tarjan 1984). It is easy to see that the complexity of computing maximal discriminators is equally low.

Future Work

In future work, we will investigate how to generalize the extraction of maximal discriminators from two clusters to n clusters, in an efficient manner. In other words, given n clusters and their n top-k lists, we aim to distinguish cluster Cl_i from the remaining $n-1$ clusters, for all $i \in \{1, \ldots, n\}$.

Acknowledgments

This research has been partially sponsored by NSERC Grants OGP0138055 and STR0134419, IRIS-2 Grants HMI-5 and IC-5, and a CITR Grant on "Distributed Continuous-Media File Systems."

References

Agrawal, R.; Ghosh, S.; Imielinski, T.; Iyer, B.; and Swami, A. 1992. An interval classifier for database mining applications. In *Proc. 18th VLDB*, 560–573.

Agrawal, R.; Imielinski, T.; and Swami, A. 1993. Mining association rules between sets of items in large databases. In *Proc. ACM SIGMOD*, 207–216.

Borgida, A., and Brachman, R. 1993. Loading data into description reasoners. In *Proc. ACM SIGMOD*, 217–226.

Ester, M.; Kriegel, H.; and Xu, X. 1995. Knowledge discovery in large spatial databases: Focusing techniques for efficient class identification. In *Proc. 4th Int'l Symposium on Large Spatial Databases*, 67–82.

Han, J.; Cai, Y.; and Cercone, N. 1992. Knowledge discovery in databases: an attribute-oriented approach. In *Proc. 18th VLDB*, 547–559.

Harel, D., and Tarjan, R. 1984. Fast algorithms for finding nearest common ancestors. *SIAM Journal on Computing* 13:338–355.

Knorr, E. M., and Ng, R. T. 1996. Finding aggregate proximity relationships and commonalities in spatial data mining. *IEEE Transactions on Knowledge and Data Engineering (Special Issue on Database Mining).* Forthcoming.

Knorr, E. M. 1995. Efficiently determining aggregate proximity relationships in spatial data mining. Master's thesis, Dept. of Computer Science, Univ. British Columbia.

Lu, W.; Han, J.; and Ooi, B. 1993. Discovery of general knowledge in large spatial databases. In *Proc. Far East Workshop on Geographic Information Systems*, 275–289.

Ng, R., and Han, J. 1994. Efficient and effective clustering methods for spatial data mining. In *Proc. 20th VLDB*, 144–155.

Zhang, T.; Ramakrishnan, R.; and Livny, M. 1996. Birch: An efficient data clustering method for very large databases. In *Proc. ACM SIGMOD*, Forthcoming.