Knowledge Discovery and Data Mining: Towards a Unifying Framework

Usama Fayyad
Microsoft Research
One Microsoft Way
Redmond, WA 98052, USA
fayyad@microsoft.com

Gregory Piatetsky-Shapiro
GTE Laboratories, MS 44
Waltham, MA 02154, USA
gps@gte.com

Padhraic Smyth
Information and Computer Science
University of California
Irvine, CA 92717-3425, USA
smyth@.ics.uci.edu

Abstract

This paper presents a first step towards a unifying framework for Knowledge Discovery in Databases. We describe links between data mining, knowledge discovery, and other related fields. We then define the KDD process and basic data mining algorithms, discuss application issues and conclude with an analysis of challenges facing practitioners in the field.

keywords: Knowledge Discovery in Databases (KDD), Data mining, overview article, large databases, automated analysis, issues and challenges in data mining.

To appear: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*, Portland, Oregon, August 2-4, 1996, AAAI Press. http://www-aig.jpl.nasa.gov/kdd96

Knowledge Discovery and Data Mining: Towards a Unifying Framework

Usama Fayyad
Microsoft Research
One Microsoft Way
Redmond, WA 98052, USA
fayyad@microsoft.com

Gregory Piatetsky-Shapiro
GTE Laboratories, MS 44
Waltham, MA 02154, USA
gps@gte.com

Padhraic Smyth
Information and Computer Science
University of California, Irvine
CA 92717-3425, USA
smyth@.ics.uci.edu

Abstract

This paper presents a first step towards a unifying framework for Knowledge Discovery in Databases. We describe links between data mining, knowledge discovery, and other related fields. We then define the KDD process and basic data mining algorithms, discuss application issues and conclude with an analysis of challenges facing practitioners in the field.

1 Introduction

Across a wide variety of fields, data are being collected and accumulated at a dramatic pace. There is an urgent need for a new generation of computational techniques and tools to assist humans in extracting useful information (knowledge) from the rapidly growing volumes of data, These techniques and tools are the subject of the emerging field of knowledge discovery in databases (KDD). This paper is an initial step towards a common framework that we hope will allow us to understand the variety of activities in this multidisciplinary field and how they fit together. We view the knowledge discovery process as a *set* of various activities for making sense of data. At the core of this process is the application of *data mining* methods for pattern[1] discovery. We examine how data mining is used and outline some of its methods. Finally, we look at practical application issues of KDD and enumerate challenges for future research and development.

2 KDD, Data Mining, and Relation to other Fields

Historically the notion of finding useful patterns in data has been given a variety of names including data mining, knowledge extraction, information discovery, information harvesting, data archaeology, and data pattern processing. The term *data mining* has been mostly used by statisticians, data analysts, and the management information systems (MIS) communities. It has also gained popularity in the database field. The term KDD was coined at the first KDD workshop in 1989 (Piatetsky-Shapiro 1991) to emphasize that "knowledge" is the end product of a data-driven discovery. It has been popularized in artificial intelligence and machine learning.

In our view KDD refers to the overall *process* of discovering useful knowledge from data while *data mining* refers to a particular *step* in this process. Data mining is the application of specific algorithms for extracting patterns from data. The distinction between the KDD process and the data mining step (within the process) is a central point of this paper. The additional steps in the KDD process, such as data preparation, data selection, data cleaning, incorporating appropriate prior knowledge, and proper interpretation of the results of mining, are essential to ensure that useful knowledge is derived from the data. Blind application of data mining methods (rightly criticised as "data dredging" in the statistical literature) can be a dangerous activity easily leading to discovery of meaningless patterns.

KDD has evolved, and continues to evolve, from the intersection of research fields such as machine learning, pattern recognition, databases, statistics, artificial intelligence, knowledge acquisition for expert systems, data visualization, and high performance computing. The unifying goal is extracting high-level knowledge from low-level data in the context of large data sets.

KDD overlaps with machine learning and pattern recognition in the study of particular data mining theories and algorithms: means for modeling data and extracting patterns. KDD focuses on aspects of finding *understandable* patterns that can be interpreted as *useful or interesting knowledge*, and puts a strong emphasis on working with large sets of real-world data. Thus, scaling properties of algorithms to large data sets are of fundamental interest.

KDD also has much in common with statistics, particularly exploratory data analysis methods. The sta-

[1] Throughout this paper we use the term "pattern" to designate *pattern* or *model* extracted from the data.

tistical approach offers precise methods for quantifying the inherent uncertainty which results when one tries to infer general patterns from a particular sample of an overall population. KDD software systems often embed particular statistical procedures for sampling and modeling data, evaluating hypotheses, and handling noise within an overall knowledge discovery framework. In contrast to traditional approaches in statistics, KDD approaches typically employ more search in model extraction and operate in the context of larger data sets with richer data structures.

In addition to its strong relation to the database field (the 2nd 'D' in KDD), another related area is *data warehousing*, which refers to the popular business trend for collecting and cleaning transactional data to make them available for on-line analysis and decision support. A popular approach for analysis of data warehouses has been called OLAP (*on-line analytical processing*), after a set of principles proposed by Codd (1993). OLAP tools focus on providing multidimensional data analysis, which is superior to SQL in computing summaries and breakdowns along many dimensions. OLAP tools are targeted towards simplifying and supporting interactive data analysis, while the KDD tool's goal is to automate as much of the process as possible.

3 Basic Definitions

We define KDD (Fayyad, Piatetsky-Shapiro, & Smyth 1996) as

Knowledge Discovery in Databases is the *non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data.*

Here *data* is a set of facts (e.g., cases in a database) and *pattern* is an expression in some language describing a subset of the data or a model applicable to that subset. Hence, in our usage here, extracting a *pattern* also designates fitting a model to data, finding structure from data, or in general any high-level description of a set of data. The term *process* implies that KDD is comprised of many steps, which involve data preparation, search for patterns, knowledge evaluation, and refinement, all repeated in multiple iterations. By *non-trivial* we mean that some search or inference is involved, i.e. it is not a straightforward computation of predefined quantities like computing the average value of a set of numbers. The discovered patterns should be *valid* on new data with some degree of certainty. We also want patterns to be *novel* (at least to the system, and preferably to the user) and *potentially useful*, i.e., lead to some benefit to the user/task. Finally, the patterns should be *understandable*, if not immediately then after some post-processing.

The above implies that we can define quantitative measures for evaluating extracted patterns. In many cases, it is possible to define measures of certainty (e.g., estimated prediction accuracy on new data) or utility (e.g. gain, perhaps in dollars saved due to better predictions or speed-up in response time of a system). Notions such as novelty and understandability are much more subjective. In certain contexts understandability can be estimated by simplicity (e.g., the number of bits to describe a pattern). An important notion, called **interestingness** (e.g. see Piatetsky-Shapiro & Matheus 1994, Silberschatz & Tuzhilin 1995), is usually taken as an overall measure of pattern value, combining validity, novelty, usefulness, and simplicity. Interestingness functions can be explicitly defined or can be manifested implicitly via an ordering placed by the KDD system on the discovered patterns or models.

Data Mining is a step in the KDD process consisting of applying data analysis and discovery algorithms that, under acceptable computational efficiency limitations, produce a particular enumeration of patterns over the data (see Section 5 for more details).

Note that the space of patterns is often infinite, and the enumeration of patterns involves some form of search in this space. Practical computational constraints place severe limits on the subspace that can be explored by a data mining algorithm.

KDD Process is the *process* of using the database along with any required selection, preprocessing, subsampling, and transformations of it; to apply data mining methods (algorithms) to enumerate patterns from it; and to evaluate the products of data mnining to identify the subset of the enumerated patterns deemed "knowledge".

The data mining component of the KDD process is concerned with the algorithmic means by which patterns are extracted and enumerated from data. The overall KDD process (Figure 1) includes the *evaluation* and possible *interpretation* of the "mined" patterns to determine which patterns may be considered new "knowledge." The KDD process also includes all of the additional steps described in Section 4. The notion of an overall user-driven process is not unique to KDD: analogous proposals have been put forward in statistics (Hand 1994) and in machine learning (Brodley and Smyth 1996).

4 The KDD Process

The KDD process is interactive and iterative, involving numerous steps with many decisions being made by

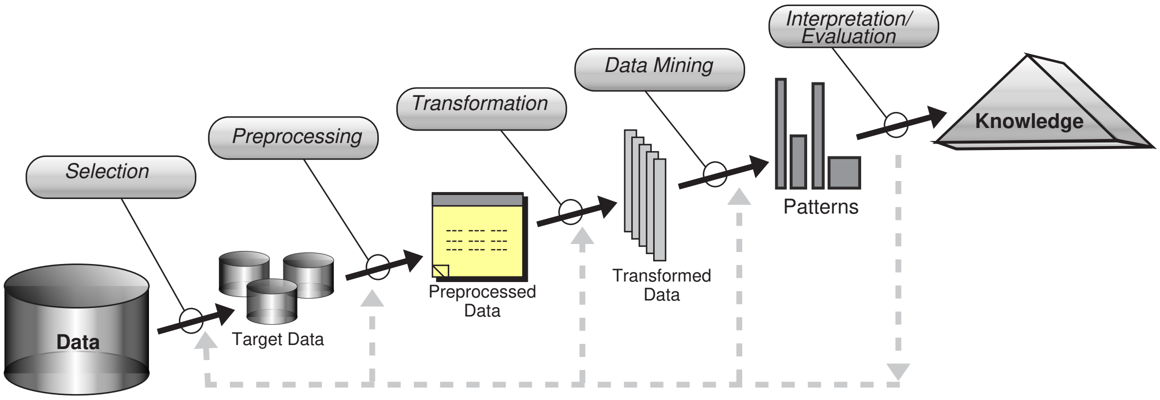

Figure 1: An overview of the steps comprising the KDD process.

the user. Brachman & Anand (1996) give a practical view of the KDD process emphasizing the interactive nature of the process. Here we broadly outline some of its basic steps:

1. Developing an understanding of the application domain and the relevant prior knowledge, and identifying the *goal* of the KDD process from the customer's viewpoint.
2. Creating a target data set: selecting a data set, or focusing on a subset of variables or data samples, on which discovery is to be performed.
3. Data cleaning and preprocessing: basic operations such as the removal of noise if appropriate, collecting the necessary information to model or account for noise, deciding on strategies for handling missing data fields, accounting for time sequence information and known changes.
4. Data reduction and projection: finding useful features to represent the data depending on the goal of the task. Using dimensionality reduction or transformation methods to reduce the effective number of variables under consideration or to find invariant representations for the data.
5. Matching the goals of the KDD process (step 1) to a particular data mining *method*: e.g., summarization, classification, regression, clustering, etc. Methods are described in Section 5.1, and in more detail in (Fayyad, Piatetsky-Shapiro, & Smyth 1996).
6. Choosing the data mining algorithm(s): selecting method(s) to be used for searching for patterns in the data. This includes deciding which models and parameters may be appropriate (e.g. models for categorical data are different than models on vectors over the reals) and matching a particular data mining method with the overall criteria of the KDD process (e.g., the end-user may be more interested in understanding the model than its predictive capabilities — see Section 5.2).
7. Data mining: searching for patterns of interest in a particular representational form or a set of such representations: classification rules or trees, regression, clustering, and so forth. The user can significantly aid the data mining method by correctly performing the preceding steps.
8. Interpreting mined patterns, possibly return to any of steps 1–7 for further iteration. This step can also involve visualization of the extracted patterns/models, or visualization of the data given the extracted models.
9. Consolidating discovered knowledge: incorporating this knowledge into another system for further action, or simply documenting it and reporting it to interested parties. This also includes checking for and resolving potential conflicts with previously believed (or extracted) knowledge.

The KDD process can involve significant iteration and may contain loops between any two steps. The basic flow of steps (although not the potential multitude of iterations and loops) is illustrated in Figure 1. Most previous work on KDD has focused on step 7 – the data mining. However, the other steps are as important for the successful application of KDD in practice.

Having defined the basic notions and introduced the KDD process, we now focus on the data mining component, which has by far received the most attention in the literature.

5 The Data Mining Step of the KDD Process

The data mining component of the KDD process often involves repeated iterative application of particular data mining methods. The objective of this section is to present a very brief overview of the primary goals of data mining, a description of the methods used to address these goals, and a very brief overview of data mining algorithms which incorporate these methods.

The knowledge discovery *goals* are defined by the intended use of the system. We can distinguish two types of goals: **Verification**, where the system is limited to verifying the user's hypothesis, and **Discovery**, where the system autonomously finds new patterns. We further subdivide the Discovery goal into **Prediction**, where the system finds patterns for the purpose of predicting the future behaviour of some entities; and **Description**, where the system finds patterns for the purpose of presenting them to a user in a human-understandable form. In this paper we are primarily concerned with discovery-oriented data mining.

Most data mining methods are based on tried and tested techniques from machine learning, pattern recognition, and statistics: classification, clustering, regression, and so forth. The array of different algorithms under each of these headings can often be quite bewildering to both the novice and experienced data analyst. It should be emphasized that of the very many data mining methods advertised in the literature, there are really only a few fundamental techniques. The actual underlying model representation being used by a particular method (i.e., the functional form of f in the mapping $x \rightarrow f(x)$) usually comes from a composition of a small number of well-known options: polynomials, splines, kernel and basis functions, threshold/Boolean functions, etc. Thus, algorithms tend to differ primarily in goodness-of-fit criterion used to evaluate model fit, or in the search method used to find a good fit.

5.1 Data Mining Methods

Although the boundaries between prediction and description are not sharp (some of the predictive models can be descriptive, to the degree that they are understandable, and vice versa), the distinction is useful for understanding the overall discovery goal. The relative importance of prediction and description for particular data mining applications can vary considerably. However, in the context of KDD, description tends to be more important than prediction. This is in contrast to many machine learning and pattern recognition applications where prediction is often the primary goal. The goals of prediction and description are achieved via the following primary data mining methods.

Classification: learning a function that maps (classifies) a data item into one of several predefined classes.

Regression: learning a function which maps a data item to a real-valued prediction variable and the discovery of functional relationships between variables.

Clustering: identifying a finite set of categories or clusters to describe the data. Closely related to clustering is the method of *probability density estimation* which consists of techniques for estimating from data the joint multi-variate probability density function of all of the variables/fields in the database.

Summarization: finding a compact description for a subset of data, e.g., the derivation of summary or association rules and the use of multivariate visualization techniques.

Dependency Modeling: finding a model which describes significant *dependencies* between variables (e.g., learning of belief networks).

Change and Deviation Detection: discovering the most significant changes in the data from previously measured or normative values

5.2 The Components of Data Mining Algorithms

Having outlined the *general methods* of data mining, the next step is to construct *specific algorithms* to implement these methods. One can identify three primary components in any data mining algorithm: *model representation*, *model evaluation*, and *search*. This reductionist view is not necessarily complete or fully encompassing: rather, it is a convenient way to express the key concepts of data mining algorithms in a relatively unified and compact manner—(Cheeseman 1990) outlines a similar structure.

Model Representation: the language used to describe discoverable patterns. If the representation is too limited, then no amount of training time or examples will produce an accurate model for the data. It is important that a data analyst fully comprehend the representational assumptions which may be inherent in a particular method. It is equally important that an algorithm designer clearly state which representational assumptions are being made by a particular algorithm. Note that more powerful representational power for models increases the danger of overfitting the training data resulting in reduced prediction accuracy on unseen data.

Model Evaluation Criteria: quantitative statements (or "fit functions") of how well a particular pattern (a model and its parameters) meet the goals of the KDD process. For example, predictive models are often judged by the empirical prediction accuracy on some test set. Descriptive models can be evaluated along the dimensions of predictive accuracy, novelty, utility, and understandability of the fitted model.

Search Method: consists of two components: *parameter search* and *model search*. Once the model representation (or family of representations) and the model evaluation criteria are fixed, then the data mining problem has been reduced to purely an optimization task: find the parameters/models from the selected family which optimize the evaluation criteria. In *parameter search* the algorithm must search for the parameters which optimize the model evaluation criteria given observed data and a fixed model representation. *Model search* occurs as a loop over the parameter search method: the model representation is changed so that a family of models are considered.

5.3 Data Mining Algorithms

There exist a wide variety of data mining algorithms. For a brief review of the most popular of these see (Fayyad, Piatetsky-Shapiro, & Smyth 1996) and the references therein. Popular model representations include decision trees and rules, nonlinear regression and classification, example-based methods (including nearest-neighbour and case-based reasoning methods), probabilistic graphical dependency models (including Bayesian networks), and relational learning models (including inductive logic programming).

An important point is that each technique typically suits some problems better than others. For example, decision tree classifiers can be very useful for finding structure in high-dimensional spaces and are also useful in problems with mixed continuous and categorical data (since tree methods do not require distance metrics). However, classification trees may not be suitable for problems where the true decision boundaries between classes are described by a 2nd-order polynomial (for example). Thus, there is no 'universal' data mining method and choosing a particular algorithm for a particular application is something of an art. In practice, a large portion of the applications effort can go into properly formulating the problem (asking the right question) rather than in optimizing the algorithmic details of a particular data mining method (Hand 1994; Langley and Simon 1995).

6 Application Issues

For a survey of applications of KDD as well as detailed examples, see (Piatetsky-Shapiro et al 1996) for industrial applications and (Fayyad, Haussler, & Stolorz 1996) for applications in science data analysis. Here, we examine criteria for selecting potential applications, which can be divided into practical and technical categories. The practical criteria for KDD projects are similar to those for other applications of advanced technology, and include the potential impact of an application, absence of simpler alternative solutions, and strong organizational support for using technology. For applications dealing with personal data one should also consider the privacy/legal issues (Piatetsky-Shapiro 1995).

The technical criteria include considerations such as the *availability of sufficient data (cases)*. In general, the more fields there are and the more complex the patterns being sought, the more data are needed. However, strong prior knowledge (see below) can reduce the number of needed cases significantly. Another consideration is the *relevance of attributes*. It is important to have data attributes relevant to the discovery task: no amount of data will allow prediction based on attributes that do not capture the required information. Furthermore, *low noise levels (few data errors)* is another consideration. High amounts of noise make it hard to identify patterns unless a large number of cases can mitigate random noise and help clarify the aggregate patterns. *Changing and time-oriented data*, while making the application development more difficult, makes it potentially much more useful, since it is easier to retrain a system than to retrain a human. Finally, and perhaps one of the most important considerations is *prior knowledge*. It is very useful to know something about the domain — what are the important fields, what are the likely relationships, what is the user utility function, what patterns are already known, and so forth.

6.1 Research and Application Challenges

We outline some of the current primary research and application challenges for KDD. This list is by no means exhaustive and is intended to give the reader a feel for the types of problems that KDD practitioners wrestle with.

Larger databases: Databases with hundreds of fields and tables, millions of records, and multi-gigabyte size are quite commonplace, and terabyte (10^{12} bytes) databases are beginning to appear. Methods for dealing with large data volumes include more efficient algorithms (Agrawal et al. 1996), sampling, approximation methods, and massively parallel processing (Holsheimer et al. 1996).

High dimensionality: Not only is there often a very large number of records in the database, but there can also be a very large number of fields (attributes, variables) so that the dimensionality of the problem is high. A high dimensional data set creates problems in terms of increasing the size of the search space for model induction in a combinatorially explosive manner. In addition, it increases the chances that a data mining algorithm will find spurious patterns that are not valid in general. Approaches to this problem include methods to reduce the effective dimensionality of the problem and the use of prior knowledge to identify irrelevant variables.

Overfitting: When the algorithm searches for the best parameters for one particular model using a limited set of data, it may model not only the general patterns in the data but also any noise specific to that data set, resulting in poor performance of the model on test data. Possible solutions include cross-validation, regularization, and other sophisticated statistical strategies.

Assessing statistical significance: A problem (related to overfitting) occurs when the system is searching over many possible models. For example, if a system tests N models at the 0.001 significance level, then on average, with purely random data, $N/1000$ of these models will be accepted as significant. This point is frequently missed by many initial attempts at KDD. One way to deal with this problem is to use methods which adjust the test statistic as a function of the search, e.g., Bonferroni adjustments for independent tests, or randomization testing.

Changing data and knowledge: Rapidly changing (non-stationary) data may make previously discovered patterns invalid. In addition, the variables measured in a given application database may be modified, deleted, or augmented with new measurements over time. Possible solutions include incremental methods for updating the patterns and treating change as an opportunity for discovery by using it to cue the search for patterns of change only (Matheus, Piatetsky-Shapiro, and McNeill 1996). See also (Mannila, Toivonen, & Verkamo 1995; Agrawal & Psaila 1995).

Missing and noisy data: This problem is especially acute in business databases. U.S. census data reportedly has error rates of up to 20%. Important attributes may be missing if the database was not designed with discovery in mind. Possible solutions include more sophisticated statistical strategies to identify hidden variables and dependencies (Heckerman 1996; Smyth et al. 1996).

Complex relationships between fields: Hierarchically structured attributes or values, relations between attributes, and more sophisticated means for representing knowledge about the contents of a database will require algorithms that can effectively utilize such information. Historically, data mining algorithms have been developed for simple attribute-value records, although new techniques for deriving relations between variables are being developed (Djoko, Cook, & Holder 1995; Dzeroski 1996).

Understandability of patterns: In many applications it is important to make the discoveries more understandable by humans. Possible solutions include graphical representations (Buntine 1996; Heckerman 1996), rule structuring, natural language generation, and techniques for visualization of data and knowledge. Rule refinement strategies (e.g. Major & Mangano 1995) can be used to address a related problem: the discovered knowledge may be implicitly or explicitly redundant.

User interaction and prior knowledge: Many current KDD methods and tools are not truly *interactive* and cannot easily incorporate prior knowledge about a problem except in simple ways. The use of domain knowledge is important in all of the steps of the KDD process as outlined in Section 4. Bayesian approaches (e.g. Cheeseman 1990) use prior probabilities over data and distributions as one form of encoding prior knowledge. Others employ deductive database capabilities to discover knowledge that is then used to guide the data mining search (e.g. Simoudis et al. 1995).

Integration with other systems: A stand-alone discovery system may not be very useful. Typical integration issues include integration with a DBMS (e.g. via a query interface), integration with spreadsheets and visualization tools, and accommodating real-time sensor readings. Examples of integrated KDD systems are described by Simoudis, Livezey, and Kerber (1995), and Stolorz et al (1995).

7 Concluding Remarks

We have presented some definitions of basic notions in the KDD field. A primary aim is to clarify the relation between knowledge discovery and data mining. We provided an overview of the KDD process and basic data mining methods. Given the broad spectrum of data mining methods and algorithms, our brief overview is inevitably limited in scope: there are many data mining techniques, particularly specialized methods for particular types of data and domains. Although various algorithms and applications may appear quite different on the surface, it is not uncommon to find that they share many common components. Understanding data mining and model induction at this component level clarifies the task of any data mining algorithm and makes it easier for the user to understand its overall contribution and applicability to the KDD process.

This paper represents a step towards a common framework that we hope will ultimately provide a unifying vision of the common overall goals and methods used in KDD. We hope this will eventually lead to a better understanding of the variety of approaches in this multi-disciplinary field and how they fit together.

Acknowledgments

We thank Sam Uthurusamy, Ron Brachman, and KDD-96 referees for their valuable suggestions and ideas.

Bibliography

Agrawal, R. and Psaila, G. 1995. Active Data Mining, In *Proceedings of KDD-95: First International Conference on Knowledge Discovery and Data Mining*, pp. 3-8, Menlo Park, CA: AAAI Press.

Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., and Verkamo, I. 1996. Fast Discovery of Association Rules, in *AKDDM*, AAAI/MIT Press, 307–328.

Brachman, R. and Anand, T. 1996. The Process of Knowledge Discovery in Databases: A Human Centered Approach, in *AKDDM*, AAAI/MIT Press, 37–58.

Brodley, C. E., and Smyth, P. 1996 Applying classification algorithms in practice. *Statistics and Computing*, to appear.

Buntine, W. 1996. Graphical Models for Discovering Knowledge, in *AKDDM*, AAAI/MIT Press, 59–82.

Cheeseman, P. 1990. On Finding the Most Probable Model. In *Computational Models of Scientific Discovery and Theory Formation*, Shrager, J. and Langley P. (eds). Los Gatos, CA: Morgan Kaufmann, 73–95.

Codd, E.F. 1993. Providing OLAP (On-line Analytical Processing) to User-Analysts: An IT Mandate. E.F. Codd and Associates.

Djoko, S., Cook, D., and Holder, L. 1995. Analyzing the Benefits of Domain Knowledge in Substructure Discovery, in *Proceedings of KDD-95: First International Conference on Knowledge Discovery and Data Mining,* Menlo Park, CA: The AAAI Press.

Dzeroski, S. 1996. Inductive Logic Programming for Knowledge Discovery in Databases, in *AKDDM*, AAAI/MIT Press.

Fayyad, U. M., G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, 1996. *Advances in Knowledge Discovery and Data Mining,* (AKDDM), AAAI/MIT Press.

Fayyad, U.M., Haussler, D. and Stolorz, Z. 1996. KDD for Science Data Analysis; Issues and Examples. *Proc. 2nd Int. Conf. on Knowledge Discovery and Data Mining (KDD-96)*, Menlo Park, CA: AAAI Press.

Fayyad, U.M., Piatetsky-Shapiro, G., and Smyth, P. 1996. From Data Mining to Knowledge Discovery: An Overview, in *AKDDM*, AAAI/MIT Press, pp. 1–30.

Hand, D. J. 1994. Deconstructing statistical questions. *J. Royal. Stat. Soc. A*, 317–356.

Heckerman, D. 1996. Bayesian Networks for Knowledge Discovery, in *AKDDM*, AAAI/MIT Press, 273–306.

Holsheimer, M., Kersten, M.L., Mannila, H., and Toivonen, H. 1996. Data Surveyor: Searching the Nuggets in Parallel, in *AKDDM*, AAAI/MIT Press.

Langley, P. and Simon, H. A. 1995. Applications of machine learning and rule induction. *Communications of the ACM*, 38, 55–64.

Major, J. and Mangano, J. 1995. Selecting among Rules Induced from a Hurricance Database. *Journal of Intelligent Information Systems* 4(1):39–52.

Mannila, H., Toivonen, H. and Verkamo, A.I. 1995. Discovering Frequent Episodes in Sequences, In *Proceedings of KDD-95: First International Conference on Knowledge Discovery and Data Mining*, pp. 210-215, Menlo Park, CA: AAAI Press.

Matheus, C., Piatetsky-Shapiro, G., and McNeill, D. 1996. Selecting and Reporting What is Interesting: The KEFIR Application to Healthcare Data, in *AKDDM*, Cambridge, MA: AAAI/MIT Press, 495-516.

Piatetsky-Shapiro, G. 1991. Knowledge Discovery in Real Databases, *AI Magazine*, Winter 1991.

Piatetsky-Shapiro, G., Matheus, C. 1994. The Interestingness of Deviations. In *Proceedings of KDD-94*. Fayyad, U. M. and Uthurusamy, R., (eds.), AAAI Press report WS-03, Menlo Park, CA: AAAI Press.

Piatetsky-Shapiro, G. 1995. Knowledge Discovery in Personal Data vs. Privacy – a Mini-symposium. *IEEE Expert*, April.

Piatetsky-Shapiro, G., Brachman, R., Khabaza, T., Kloesgen, W., and Simoudis, E., 1996. An Overview of Issues in Developing Industrial Data Mining and Knowledge Discovery Applications, In Proceedings of KDD-96, Menlo Park, CA: AAAI Press.

Silberschatz, A. and Tuzhilin, A. 1995. On Subjective Measures of Interestingness in Knowledge Discovery. In *Proceedings of KDD-95: First International Conference on Knowledge Discovery and Data Mining*, pp. 275-281, Menlo Park, CA: AAAI Press.

Simoudis, E., Livezey, B., and Kerber, R. 1995. Using Recon for Data Cleaning, In *Proceedings of KDD-95: First International Conference on Knowledge Discovery and Data Mining*, pp. 275-281, Menlo Park, CA: AAAI Press.

Smyth, P., Burl, M., Fayyad, U. and Perona, P. 1996. Modeling Subjective Uncertainty in Image Annotation, in *AKDDM*, AAAI/MIT Press, 517–540.

Stolorz, P. et al. 1995. Fast Spatio-Temporal Data Mining of Large Geophysical Datasets, In *Proceedings of KDD-95: First International Conference on Knowledge Discovery and Data Mining*, pp. 300–305, AAAI Press.