Data Mining: Concepts and Techniques

Jiawei Han and **Micheline Kamber**

Simon Fraser University

Preface

Our capabilities of both generating and collecting data have been increasing rapidly in the last several decades. Contributing factors include the widespread use of bar codes for most commercial products, the computerization of many business, scientific and government transactions and managements, and advances in data collection tools ranging from scanned texture and image platforms, to on-line instrumentation in manufacturing and shopping, and to satellite remote sensing systems. In addition, popular use of the World Wide Web as a global information system has flooded us with a tremendous amount of data and information. This explosive growth in stored data has generated an urgent need for new techniques and automated tools that can intelligently assist us in transforming the vast amounts of data into useful information and knowledge.

This book explores the concepts and techniques of *data mining*, a promising and flourishing frontier in database systems and new database applications. Data mining, also popularly referred to as *knowledge discovery in databases (KDD)*, is the automated or convenient extraction of patterns representing knowledge implicitly stored in large databases, data warehouses, and other massive information repositories.

Data mining is a multidisciplinary field, drawing work from areas including database technology, artificial intelligence, machine learning, neural networks, statistics, pattern recognition, knowledge based systems, knowledge acquisition, information retrieval, high performance computing, and data visualization. We present the material in this book from a *database perspective*. That is, we focus on issues relating to the feasibility, usefulness, efficiency, and scalability of techniques for the discovery of patterns hidden *in large databases*. As a result, this book is not intended as an introduction to database systems, machine learning, or statistics, etc., although we do provide the background necessary in these areas in order to facilitate the reader's comprehension of their respective roles in data mining. Rather, the book is a comprehensive introduction to data mining, presented with database issues in focus. It should be useful for computing science students, application developers, and business professionals, as well as researchers involved in any of the disciplines listed above.

Data mining emerged during the late 1980's, has made great strides during the 1990's, and is expected to continue to flourish into the new millennium. This book presents an overall picture of the field from a database researcher's point of view, introducing interesting data mining techniques and systems, and discussing applications and research directions. An important motivation for writing this book was the need to build an organized framework for the study of data mining — a challenging task owing to the extensive multidisciplinary nature of this fast developing field. We hope that this book will encourage people with different backgrounds and experiences to exchange their views regarding data mining so as to contribute towards the further promotion and shaping of this exciting and dynamic field.

To the teacher

This book is designed to give a broad, yet in depth overview of the field of data mining. You will find it useful for teaching a course on data mining at an advanced undergraduate level, or the first-year graduate level. In addition, individual chapters may be included as material for courses on selected topics in database systems or in artificial intelligence. We have tried to make the chapters as self-contained as possible. For a course taught at the undergraduate level, you might use chapters 1 to 8 as the core course material. Remaining class material may be selected from among the more advanced topics described in chapters 9 and 10. For a graduate level course, you may choose to cover the entire book in one semester.

Each chapter ends with a set of exercises, suitable as assigned homework. The exercises are either short questions

that test basic mastery of the material covered, or longer questions which require analytical thinking.

To the student

We hope that this textbook will spark your interest in the fresh, yet evolving field of data mining. We have attempted to present the material in a clear manner, with careful explanation of the topics covered. Each chapter ends with a summary describing the main points. We have included many figures and illustrations throughout the text in order to make the book more enjoyable and "reader-friendly". Although this book was designed as a textbook, we have tried to organize it so that it will also be useful to you as a reference book or handbook, should you later decide to pursue a career in data mining.

What do you need to know in order to read this book?

- You should have some knowledge of the concepts and terminology associated with database systems. However, we do try to provide enough background of the basics in database technology, so that if your memory is a bit rusty, you will not have trouble following the discussions in the book. You should have some knowledge of database querying, although knowledge of any specific query language is not required.
- You should have some programming experience. In particular, you should be able to read pseudo-code, and understand simple data structures such as multidimensional arrays.
- It will be helpful to have some preliminary background in statistics, machine learning, or pattern recognition. However, we will familiarize you with the basic concepts of these areas that are relevant to data mining from a database perspective.

To the professional

This book was designed to cover a broad range of topics in the field of data mining. As a result, it is a good handbook on the subject. Because each chapter is designed to be as stand-alone as possible, you can focus on the topics that most interest you. Much of the book is suited to applications programmers or information service managers like yourself who wish to learn about the key ideas of data mining on their own.

The techniques and algorithms presented are of practical utility. Rather than selecting algorithms that perform well on small "toy" databases, the algorithms described in the book are geared for the discovery of data patterns hidden in large, real databases. In Chapter 10, we briefly discuss data mining systems in commercial use, as well as promising research prototypes. Each algorithm presented in the book is illustrated in pseudo-code. The pseudo-code is similar to the C programming language, yet is designed so that it should be easy to follow by programmers unfamiliar with C or C++. If you wish to implement any of the algorithms, you should find the translation of our pseudo-code into the programming language of your choice to be a fairly straightforward task.

Organization of the book

The book is organized as follows.

Chapter 1 provides an introduction to the multidisciplinary field of data mining. It discusses the evolutionary path of database technology which led up to the need for data mining, and the importance of its application potential. The basic architecture of data mining systems is described, and a brief introduction to the concepts of database systems and data warehouses is given. A detailed classification of data mining tasks is presented, based on the different kinds of knowledge to be mined. A classification of data mining systems is presented, and major challenges in the field are discussed.

Chapter 2 is an introduction to data warehouses and OLAP (On-Line Analytical Processing). Topics include the concept of data warehouses and multidimensional databases, the construction of data cubes, the implementation of on-line analytical processing, and the relationship between data warehousing and data mining.

Chapter 3 describes techniques for preprocessing the data prior to mining. Methods of data cleaning, data integration and transformation, and data reduction are discussed, including the use of concept hierarchies for dynamic and static discretization. The automatic generation of concept hierarchies is also described.

Chapter 4 introduces the primitives of data mining which define the specification of a data mining task. It describes a data mining query language (DMQL), and provides examples of data mining queries. Other topics include the construction of graphical user interfaces, and the specification and manipulation of concept hierarchies.

Chapter 5 describes techniques for concept description, including characterization and discrimination. An attribute-oriented generalization technique is introduced, as well as its different implementations including a generalized relation technique and a multidimensional data cube technique. Several forms of knowledge presentation and visualization are illustrated. Relevance analysis is discussed. Methods for class comparison at multiple abstraction levels, and methods for the extraction of characteristic rules and discriminant rules with interestingness measurements are presented. In addition, statistical measures for descriptive mining are discussed.

Chapter 6 presents methods for mining association rules in transaction databases as well as relational databases and data warehouses. It includes a classification of association rules, a presentation of the basic Apriori algorithm and its variations, and techniques for mining multiple-level association rules, multidimensional association rules, quantitative association rules, and correlation rules. Strategies for finding interesting rules by constraint-based mining and the use of interestingness measures to focus the rule search are also described.

Chapter 7 describes methods for data classification and predictive modeling. Major methods of classification and prediction are explained, including decision tree induction, Bayesian classification, the neural network technique of backpropagation, k-nearest neighbor classifiers, case-based reasoning, genetic algorithms, rough set theory, and fuzzy set approaches. Association-based classification, which applies association rule mining to the problem of classification, is presented. Methods of regression are introduced, and issues regarding classifier accuracy are discussed.

Chapter 8 describes methods of clustering analysis. It first introduces the concept of data clustering and then presents several major data clustering approaches, including partition-based clustering, hierarchical clustering, and model-based clustering. Methods for clustering continuous data, discrete data, and data in multidimensional data cubes are presented. The scalability of clustering algorithms is discussed in detail.

Chapter 9 discusses methods for data mining in advanced database systems. It includes data mining in object-oriented databases, spatial databases, text databases, multimedia databases, active databases, temporal databases, heterogeneous and legacy databases, and resource and knowledge discovery in the Internet information base.

Finally, in Chapter 10, we summarize the concepts presented in this book and discuss applications of data mining and some challenging research issues.

Errors

It is likely that this book may contain typos, errors, or omissions. If you notice any errors, have suggestions regarding additional exercises or have other constructive criticism, we would be very happy to hear from you. We welcome and appreciate your suggestions. You can send your comments to:

```
Data Mining: Concept and Techniques
Intelligent Database Systems Research Laboratory
Simon Fraser University,
Burnaby, British Columbia
Canada V5A 1S6
Fax: (604) 291-3045
```

Alternatively, you can use electronic mails to submit bug reports, request a list of known errors, or make constructive suggestions. To receive instructions, send email to dk@cs.sfu.ca with "Subject: help" in the message header. We regret that we cannot personally respond to all e-mails. The errata of the book and other updated information related to the book can be found by referencing the Web address: http://db.cs.sfu.ca/Book.

Acknowledgements

We would like to express our sincere thanks to all the members of the data mining research group who have been working with us at Simon Fraser University on data mining related research, and to all the members of the DBMiner system development team, who have been working on an exciting data mining project, DBMiner, and have made it a real success. The data mining research team currently consists of the following active members: Julia Gitline,

Kan Hu, Jean Hou, Pei Jian, Micheline Kamber, Eddie Kim, Jin Li, Xuebin Lu, Behzad Mortazav-Asl, Helen Pinto, Yiwen Yin, Zhaoxia Wang, and Hua Zhu. The DBMiner development team currently consists of the following active members: Kan Hu, Behzad Mortazav-Asl, and Hua Zhu, and some partime workers from the data mining research team. We are also grateful to Helen Pinto, Hua Zhu, and Lara Winstone for their help with some of the figures in this book.

More acknowledgements will be given at the final stage of the writing.

Contents

Chapter 1

Introduction

This book is an introduction to what has come to be known as *data mining* and *knowledge discovery in databases*. The material in this book is presented from a database perspective, where emphasis is placed on basic data mining concepts and techniques for uncovering interesting data patterns hidden in *large data sets*. The implementation methods discussed are particularly oriented towards the development of *scalable* and *efficient* data mining tools.

In this chapter, you will learn how data mining is part of the natural evolution of database technology, why data mining is important, and how it is defined. You will learn about the general architecture of data mining systems, as well as gain insight into the kinds of data on which mining can be performed, the types of patterns that can be found, and how to tell which patterns represent useful knowledge. In addition to studying a classification of data mining systems, you will read about challenging research issues for building data mining tools of the future.

1.1 What motivated data mining? Why is it important?

Necessity is the mother of invention.

— English proverb.

The major reason that data mining has attracted a great deal of attention in information industry in recent years is due to the wide availability of huge amounts of data and the imminent need for turning such data into useful information and knowledge. The information and knowledge gained can be used for applications ranging from business management, production control, and market analysis, to engineering design and science exploration.

Data mining can be viewed as a result of the natural evolution of information technology. An evolutionary path has been witnessed in the database industry in the development of the following functionalities (Figure 1.1): *data collection and database creation, data management* (including data storage and retrieval, and database transaction processing), and *data analysis and understanding* (involving data warehousing and data mining). For instance, the early development of data collection and database creation mechanisms served as a prerequisite for later development of effective mechanisms for data storage and retrieval, and query and transaction processing. With numerous database systems offering query and transaction processing as common practice, data analysis and understanding has naturally become the next target.

Since the 1960's, database and information technology has been evolving systematically from primitive file processing systems to sophisticated and powerful databases systems. The research and development in database systems since the 1970's has led to the development of relational database systems (where data are stored in relational table structures; see Section 1.3.1), data modeling tools, and indexing and data organization techniques. In addition, users gained convenient and flexible data access through query languages, query processing, and user interfaces. Efficient methods for **on-line transaction processing** (OLTP), where a query is viewed as a read-only transaction, have contributed substantially to the evolution and wide acceptance of relational technology as a major tool for efficient storage, retrieval, and management of large amounts of data.

Database technology since the mid-1980s has been characterized by the popular adoption of relational technology and an upsurge of research and development activities on new and powerful database systems. These employ ad-

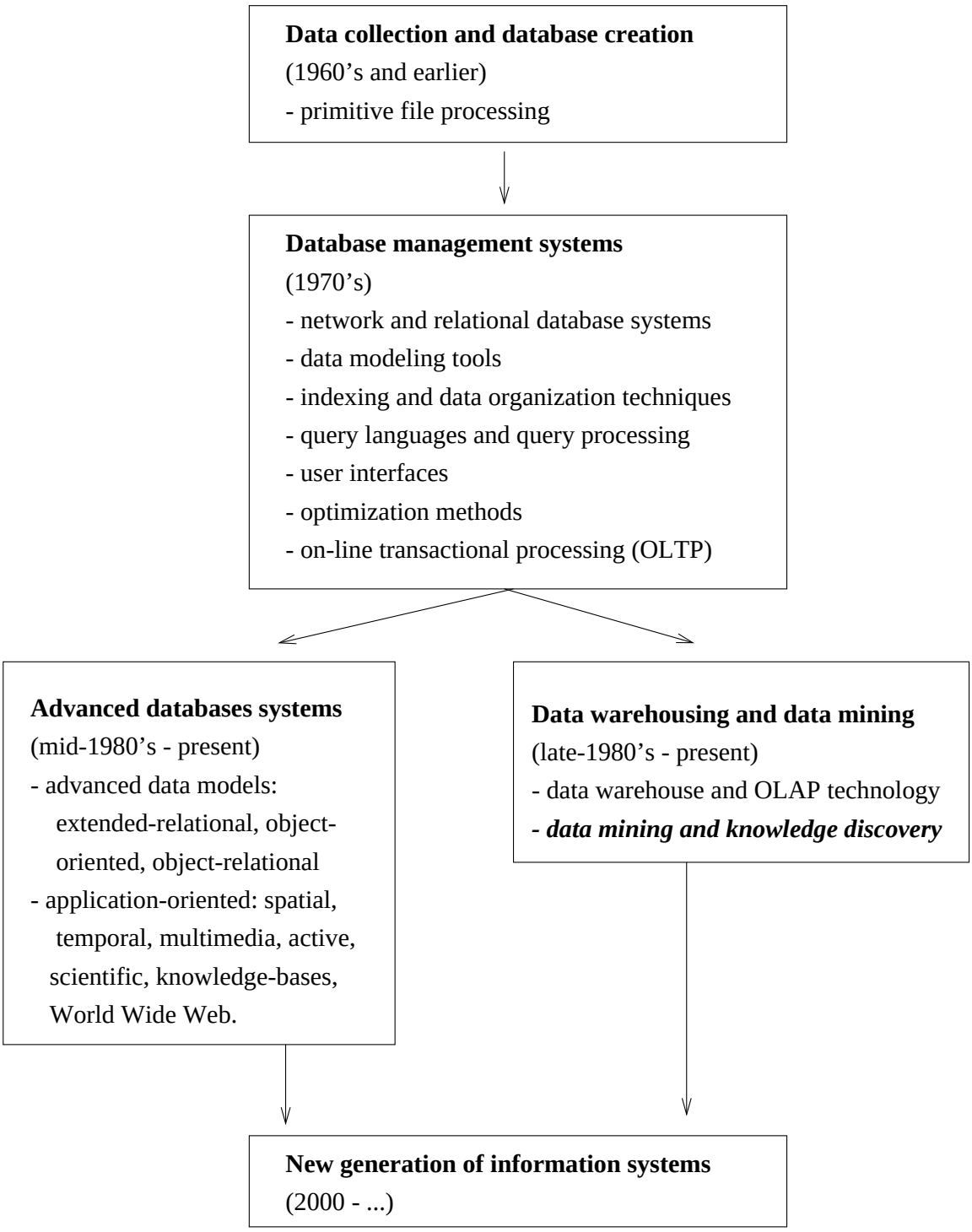

Figure 1.1: The evolution of database technology.

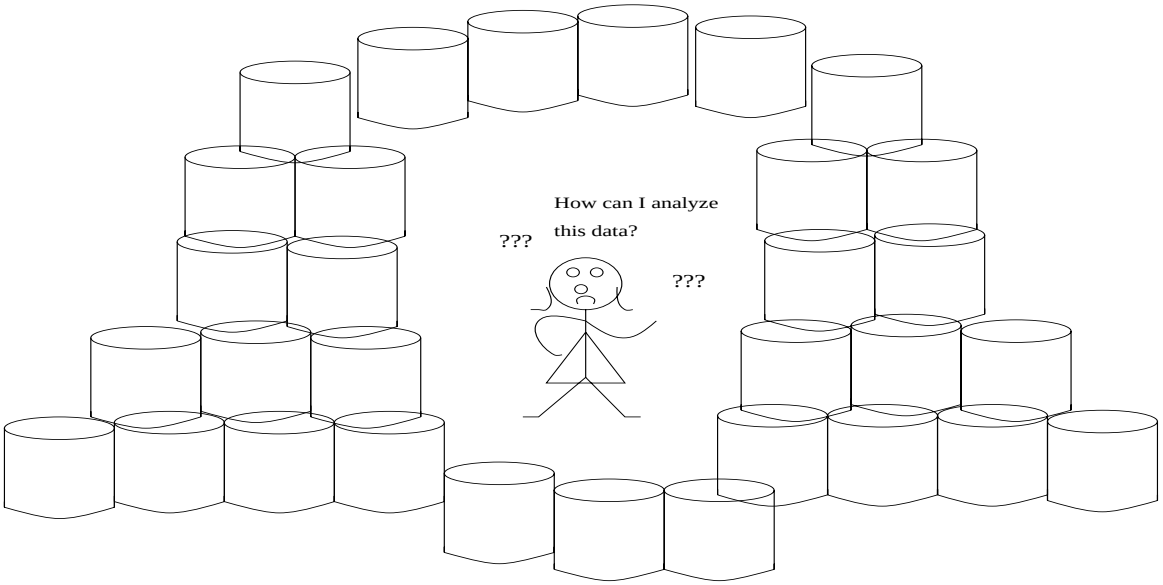

Figure 1.2: We are data rich, but information poor.

vanced data models such as extended-relational, object-oriented, object-relational, and deductive models Application-oriented database systems, including spatial, temporal, multimedia, active, and scientific databases, knowledge bases, and office information bases, have flourished. Issues related to the distribution, diversification, and sharing of data have been studied extensively. Heterogeneous database systems and Internet-based global information systems such as the World-Wide Web (WWW) also emerged and play a vital role in the information industry.

The steady and amazing progress of computer hardware technology in the past three decades has led to powerful, affordable, and large supplies of computers, data collection equipment, and storage media. This technology provides a great boost to the database and information industry, and makes a huge number of databases and information repositories available for transaction management, information retrieval, *and data analysis*.

Data can now be stored in many different types of databases. One database architecture that has recently emerged is the **data warehouse** (Section 1.3.2), a repository of multiple heterogeneous data sources, organized under a unified schema at a single site in order to facilitate management decision making. Data warehouse technology includes data cleansing, data integration, and **On-Line Analytical Processing (OLAP)**, that is, analysis techniques with functionalities such as summarization, consolidation and aggregation, as well as the ability to view information at different angles. Although OLAP tools support multidimensional analysis and decision making, additional data analysis tools are required for in-depth analysis, such as data classification, clustering, and the characterization of data changes over time.

The abundance of data, coupled with the need for powerful data analysis tools, has been described as a "*data rich but information poor*" situation. The fast-growing, tremendous amount of data, collected and stored in large and numerous databases, has far exceeded our human ability for comprehension *without powerful tools* (Figure 1.2). As a result, data collected in large databases become "data tombs" — data archives that are seldom revisited. Consequently, important decisions are often made based not on the information-rich data stored in databases but rather on a decision maker's intuition, simply because the decision maker does not have the tools to extract the valuable knowledge embedded in the vast amounts of data. In addition, consider current expert system technologies, which typically rely on users or domain experts to *manually* input knowledge into knowledge bases. Unfortunately, this procedure is prone to biases and errors, and is extremely time-consuming and costly. Data mining tools which perform data analysis may uncover important data patterns, contributing greatly to business strategies, knowledge bases, and scientific and medical research. The widening gap between data and information calls for a systematic development of *data mining tools* which will turn data tombs into "golden nuggets" of knowledge.

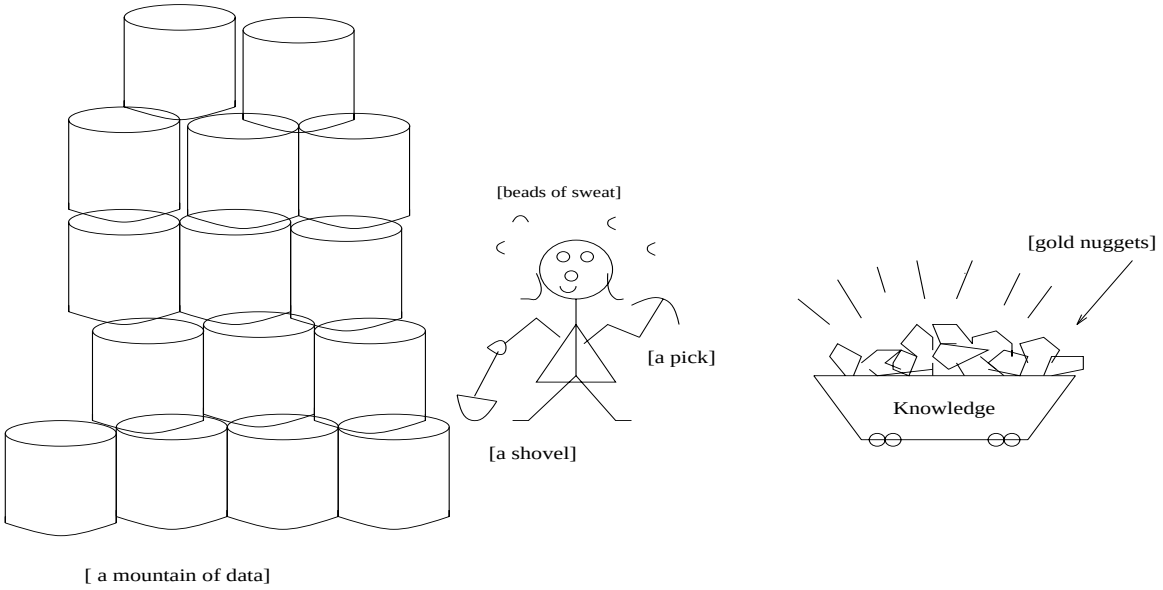

Figure 1.3: Data mining - searching for knowledge (interesting patterns) in your data.

1.2 So, what is data mining?

Simply stated, **data mining** refers to *extracting or "mining" knowledge from large amounts of data*. The term is actually a misnomer. Remember that the mining of gold from rocks or sand is referred to as *gold* mining rather than rock or sand mining. Thus, "data mining" should have been more appropriately named "knowledge mining from data", which is unfortunately somewhat long. "Knowledge mining", a shorter term, may not reflect the emphasis on mining from large amounts of data. Nevertheless, mining is a vivid term characterizing the process that finds a small set of precious nuggets from a great deal of raw material (Figure 1.3). Thus, such a misnomer which carries both "data" and "mining" became a popular choice. There are many other terms carrying a similar or slightly different meaning to data mining, such as **knowledge mining from databases, knowledge extraction, data/pattern analysis, data archaeology**, and **data dredging**.

Many people treat data mining as a synonym for another popularly used term, "**Knowledge Discovery in Databases**", or **KDD**. Alternatively, others view data mining as simply an essential step in the process of knowledge discovery in databases. Knowledge discovery as a process is depicted in Figure 1.4, and consists of an iterative sequence of the following steps:

- **data cleaning** (to remove noise or irrelevant data),
- **data integration** (where multiple data sources may be combined)[1],
- **data selection** (where data relevant to the analysis task are retrieved from the database),
- **data transformation** (where data are transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations, for instance)[2],
- **data mining** (an essential process where intelligent methods are applied in order to extract data patterns),
- **pattern evaluation** (to identify the truly interesting patterns representing knowledge based on some **interestingness measures**; Section 1.5), and
- **knowledge presentation** (where visualization and knowledge representation techniques are used to present the mined knowledge to the user).

[1] A popular trend in the information industry is to perform data cleaning and data integration as a preprocessing step where the resulting data are stored in a data warehouse.

[2] Sometimes data transformation and consolidation are performed before the data selection process, particularly in the case of data warehousing.

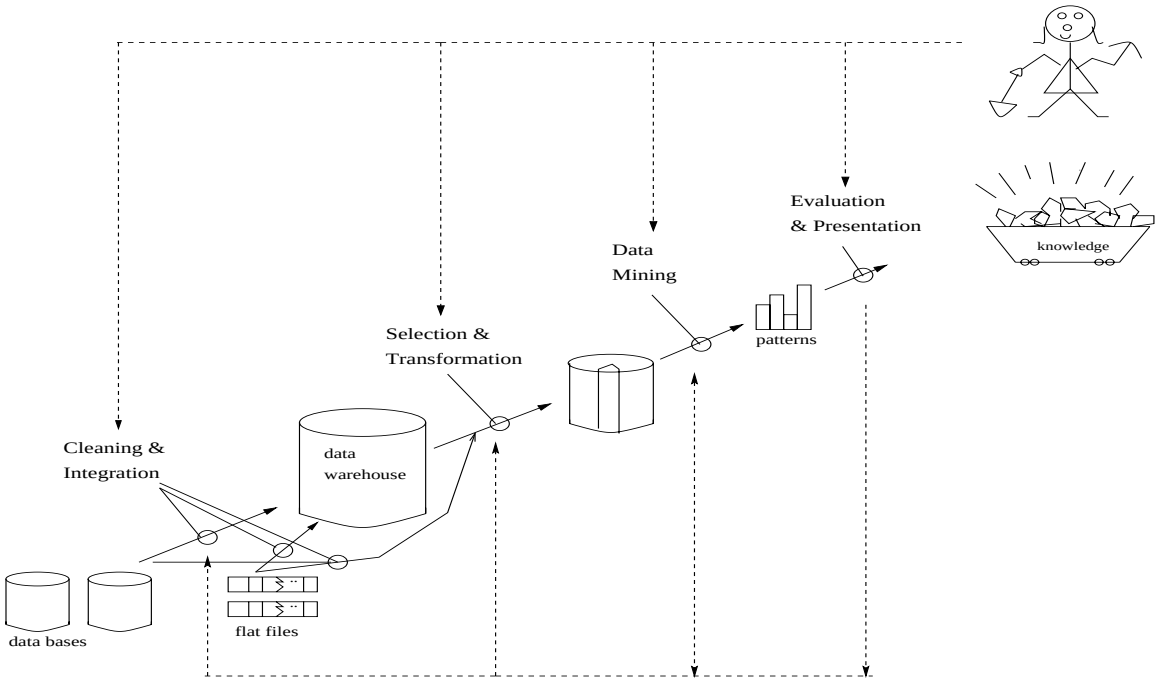

Figure 1.4: Data mining as a process of knowledge discovery.

The data mining step may interact with the user or a knowledge base. The interesting patterns are presented to the user, and may be stored as new knowledge in the knowledge base. Note that according to this view, data mining is only one step in the entire process, albeit an essential one since it uncovers hidden patterns for evaluation.

We agree that data mining is a knowledge discovery process. However, in industry, in media, and in the database research milieu, the term "data mining" is becoming more popular than the longer term of "knowledge discovery in databases". Therefore, in this book, we choose to use the term "data mining". We adopt a broad view of data mining functionality: **data mining** is the process of discovering interesting knowledge from *large* amounts of data stored either in databases, data warehouses, or other information repositories.

Based on this view, the architecture of a typical data mining system may have the following major components (Figure 1.5):

1. **Database, data warehouse, or other information repository.** This is one or a set of databases, data warehouses, spread sheets, or other kinds of information repositories. Data cleaning and data integration techniques may be performed on the data.

2. **Database or data warehouse server.** The database or data warehouse server is responsible for fetching the relevant data, based on the user's data mining request.

3. **Knowledge base.** This is the domain knowledge that is used to guide the search, or evaluate the interestingness of resulting patterns. Such knowledge can include **concept hierarchies**, used to organize attributes or attribute values into different levels of abstraction. Knowledge such as user beliefs, which can be used to assess a pattern's interestingness based on its unexpectedness, may also be included. Other examples of domain knowledge are additional interestingness constraints or thresholds, and metadata (e.g., describing data from multiple heterogeneous sources).

4. **Data mining engine.** This is essential to the data mining system and ideally consists of a set of functional modules for tasks such as characterization, association analysis, classification, evolution and deviation analysis.

5. **Pattern evaluation module.** This component typically employs interestingness measures (Section 1.5) and interacts with the data mining modules so as to *focus* the search towards interesting patterns. It may access interestingness thresholds stored in the knowledge base. Alternatively, the pattern evaluation module may be

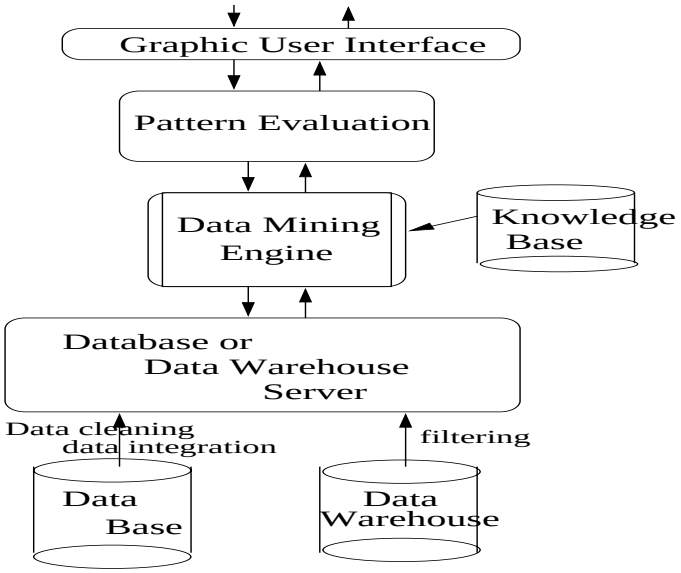

Figure 1.5: Architecture of a typical data mining system.

integrated with the mining module, depending on the implementation of the data mining method used. For efficient data mining, it is highly recommended to push the evaluation of pattern interestingness as deep as possible into the mining process so as to confine the search to only the interesting patterns.

6. **Graphical user interface.** This module communicates between users and the data mining system, allowing the user to interact with the system by specifying a data mining query or task, providing information to help focus the search, and performing exploratory data mining based on the intermediate data mining results. In addition, this component allows the user to browse database and data warehouse schemas or data structures, evaluate mined patterns, and visualize the patterns in different forms.

From a data warehouse perspective, data mining can be viewed as an advanced stage of on-line analytical processing (OLAP). However, data mining goes far beyond the narrow scope of summarization-style analytical processing of data warehouse systems by incorporating more advanced techniques for data understanding.

While there may be many "data mining systems" on the market, not all of them can perform true data mining. A data analysis system that does not handle large amounts of data can at most be categorized as a machine learning system, a statistical data analysis tool, or an experimental system prototype. A system that can only perform data or information retrieval, including finding aggregate values, or that performs deductive query answering in large databases should be more appropriately categorized as either a database system, an information retrieval system, or a deductive database system.

Data mining involves an integration of techniques from multiple disciplines such as database technology, statistics, machine learning, high performance computing, pattern recognition, neural networks, data visualization, information retrieval, image and signal processing, and spatial data analysis. We adopt a database perspective in our presentation of data mining in this book. That is, emphasis is placed on *efficient* and *scalable* data mining techniques for *large* databases. By performing data mining, interesting knowledge, regularities, or high-level information can be extracted from databases and viewed or browsed from different angles. The discovered knowledge can be applied to decision making, process control, information management, query processing, and so on. Therefore, data mining is considered as one of the most important frontiers in database systems and one of the most promising, new database applications in the information industry.

1.3 Data mining — on what kind of data?

In this section, we examine a number of different data stores on which mining can be performed. In principle, data mining should be applicable to any kind of information repository. This includes relational databases, data

warehouses, transactional databases, advanced database systems, flat files, and the World-Wide Web. Advanced database systems include object-oriented and object-relational databases, and specific application-oriented databases, such as spatial databases, time-series databases, text databases, and multimedia databases. The challenges and techniques of mining may differ for each of the repository systems.

Although this book assumes that readers have primitive knowledge of information systems, we provide a brief introduction to each of the major data repository systems listed above. In this section, we also introduce the fictitious *AllElectronics* store which will be used to illustrate concepts throughout the text.

1.3.1 Relational databases

A database system, also called a **database management system (DBMS)**, consists of a collection of interrelated data, known as a **database**, and a set of software programs to manage and access the data. The software programs involve mechanisms for the definition of database structures, for data storage, for concurrent, shared or distributed data access, and for ensuring the consistency and security of the information stored, despite system crashes or attempts at unauthorized access.

A **relational database** is a collection of **tables**, each of which is assigned a unique name. Each table consists of a set of **attributes** (*columns* or *fields*) and usually stores a large number of **tuples** (*records* or *rows*). Each tuple in a relational table represents an object identified by a unique *key* and described by a set of attribute values.

Consider the following example.

Example 1.1 The *AllElectronics* company is described by the following relation tables: *customer, item, employee*, and *branch*. Fragments of the tables described here are shown in Figure 1.6. The attribute which represents key or composite key component of each relation is underlined.

- The relation *customer* consists of a set of attributes, including a unique customer identity number (*cust_ID*), customer name, address, age, occupation, annual income, credit information, category, etc.

- Similarly, each of the relations *employee*, *branch*, and *items*, consists of a set of attributes, describing their properties.

- Tables can also be used to represent the relationships between or among multiple relation tables. For our example, these include *purchases* (customer purchases items, creating a sales transaction that is handled by an employee), *items_sold* (lists the items sold in a given transaction), and *works_at* (employee works at a branch of *AllElectronics*). □

Relational data can be accessed by **database queries** written in a relational query language, such as SQL, or with the assistance of graphical user interfaces. In the latter, the user may employ a menu, for example, to specify attributes to be included in the query, and the constraints on these attributes. A given query is transformed into a set of relational operations, such as join, selection, and projection, and is then optimized for efficient processing. A query allows retrieval of specified subsets of the data. Suppose that your job is to analyze the *AllElectronics* data. Through the use of relational queries, you can ask things like "Show me a list of all items that were sold in the last quarter". Relational languages also include aggregate functions such as `sum`, `avg` (average), `count`, `max` (maximum), and `min` (minimum). These allow you to find out things like "Show me the total sales of the last month, grouped by branch", or "How many sales transactions occurred in the month of December?", or "Which sales person had the highest amount of sales?".

When data mining is applied to relational databases, one can go further by *searching for trends or data patterns*. For example, data mining systems may analyze customer data to predict the credit risk of new customers based on their income, age, and previous credit information. Data mining systems may also detect deviations, such as items whose sales are far from those expected in comparison with the previous year. Such deviations can then be further investigated, e.g., has there been a change in packaging of such items, or a significant increase in price?

Relational databases are one of the most popularly available and rich information repositories for data mining, and thus they are a major data form in our study of data mining.

customer

cust_ID	name	address	age	income	credit_info	...
C1	Smith, Sandy	5463 E. Hastings, Burnaby, BC, V5A 4S9, Canada	21	$27000	1	...
...	...	...	...	...	...	...

item

item_ID	name	brand	category	type	price	place_made	supplier	cost
I3	hi-res-TV	Toshiba	high resolution	TV	$988.00	Japan	NikoX	$600.00
I8	multidisc-CDplay	Sanyo	multidisc	CD player	$369.00	Japan	MusicFront	$120.00
...	...	...	...	...	...	...	...	...

employee

empl_ID	name	category	group	salary	commission
E55	Jones, Jane	home entertainment	manager	$18,000	2%
...	...	...	...	...	...

branch

branch_ID	name	address
B1	City Square	369 Cambie St., Vancouver, BC V5L 3A2, Canada
...	...	...

purchases

trans_ID	cust_ID	empl_ID	date	time	method_paid	amount
T100	C1	E55	09/21/98	15:45	Visa	$1357.00
...	...	...	...	...	...	...

items_sold

trans_ID	item_ID	qty
T100	I3	1
T100	I8	2
...	...	...

works_at

empl_ID	branch_ID
E55	B1
...	...

Figure 1.6: Fragments of relations from a relational database for *AllElectronics*.

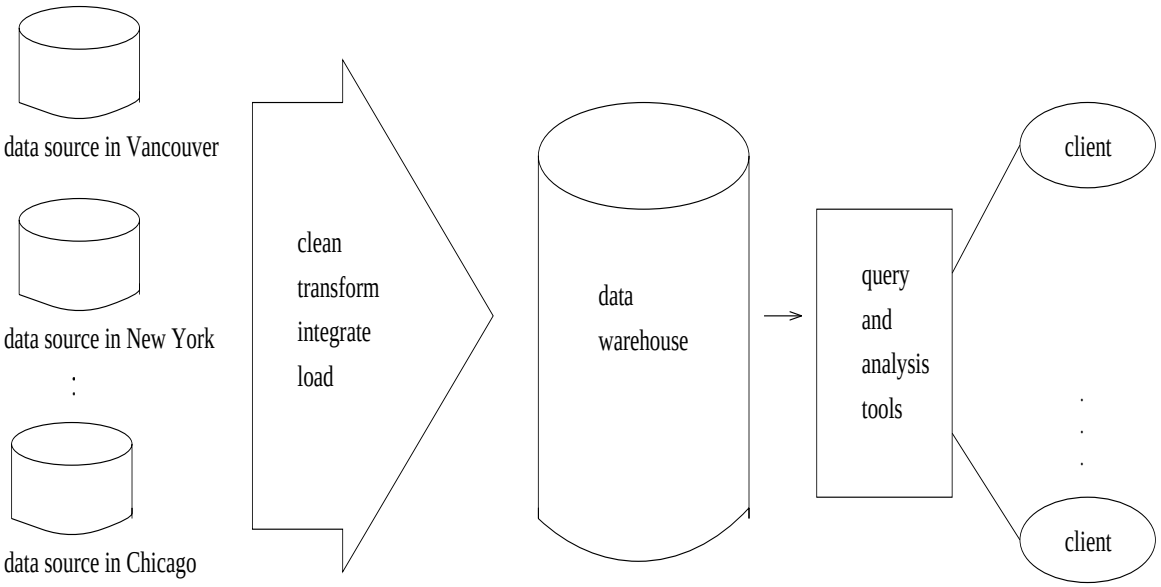

Figure 1.7: Architecture of a typical data warehouse.

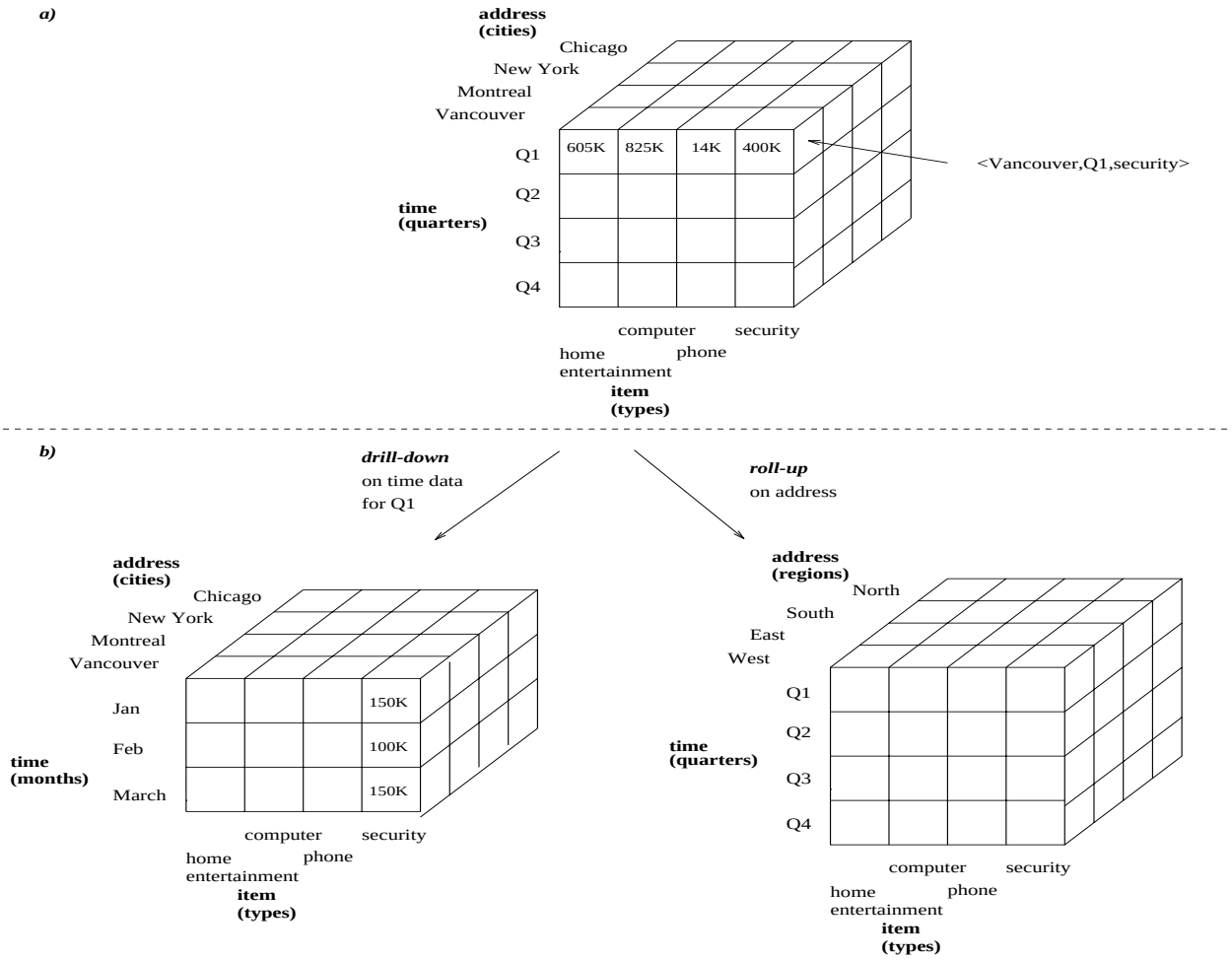

Figure 1.8: A multidimensional data cube, commonly used for data warehousing, *a)* showing summarized data for *AllElectronics* and *b)* showing summarized data resulting from drill-down and roll-up operations on the cube in *a)*.

1.3.2 Data warehouses

Suppose that *AllElectronics* is a successful international company, with branches around the world. Each branch has its own set of databases. The president of *AllElectronics* has asked you to provide an analysis of the company's sales per item type per branch for the third quarter. This is a difficult task, particularly since the relevant data are spread out over several databases, physically located at numerous sites.

If *AllElectronics* had a data warehouse, this task would be easy. A **data warehouse** is a repository of information collected from multiple sources, stored under a unified schema, and which usually resides at a single site. Data warehouses are constructed via a process of data cleansing, data transformation, data integration, data loading, and periodic data refreshing. This process is studied in detail in Chapter 2. Figure 1.7 shows the basic architecture of a data warehouse for *AllElectronics*.

In order to facilitate decision making, the data in a data warehouse are *organized around major subjects*, such as customer, item, supplier, and activity. The data are stored to provide information from a *historical perspective* (such as from the past 5-10 years), and are typically *summarized*. For example, rather than storing the details of each sales transaction, the data warehouse may store a summary of the transactions per item type for each store, or, summarized to a higher level, for each sales region.

A data warehouse is usually modeled by a multidimensional database structure, where each **dimension** corresponds to an attribute or a set of attributes in the schema, and each **cell** stores the value of some aggregate measure, such as *count* or *sales_amount*. The actual physical structure of a data warehouse may be a relational data store or a **multidimensional data cube**. It provides a multidimensional view of data and allows the precomputation and

sales

trans_ID	list of item_ID's
T100	I1, I3, I8, I16
...	...

Figure 1.9: Fragment of a transactional database for sales at *AllElectronics*.

fast accessing of summarized data.

Example 1.2 A data cube for summarized sales data of *AllElectronics* is presented in Figure 1.8a). The cube has three **dimensions**: address (with *city* values *Chicago, New York, Montreal, Vancouver*), time (with *quarter* values *Q1, Q2, Q3, Q4*), and item (with item *type* values *home entertainment, computer, phone, security*). The aggregate value stored in each cell of the cube is *sales_amount*. For example, the total sales for $Q1$ of items relating to security systems in Vancouver is \$400K, as stored in cell ⟨Vancouver, Q1, security⟩. Additional cubes may be used to store aggregate sums over each dimension, corresponding to the aggregate values obtained using different SQL group-bys, e.g., the total sales amount per city and quarter, or per city and item, or per quarter and item, or per each individual dimension. □

In research literature on data warehouses, the data cube structure that stores the primitive or lowest level of information is called a **base cuboid**. Its corresponding higher level multidimensional (cube) structures are called (non-base) **cuboids**. A base cuboid together with all of its corresponding higher level cuboids form a **data cube**.

By providing multidimensional data views and the precomputation of summarized data, data warehouse systems are well suited for **On-Line Analytical Processing**, or **OLAP**. OLAP operations make use of background knowledge regarding the domain of the data being studied in order to allow the presentation of data at *different levels of abstraction*. Such operations accommodate different user viewpoints. Examples of OLAP operations include **drill-down** and **roll-up**, which allow the user to view the data at differing degrees of summarization, as illustrated in Figure 1.8b). For instance, one may drill down on sales data summarized by *quarter* to see the data summarized by *month*. Similarly, one may roll up on sales data summarized by *city* to view the data summarized by *region*.

Although data warehouse tools help support data analysis, additional tools for data mining are required to allow more in depth and automated analysis. Data warehouse technology is discussed in detail in Chapter 2.

1.3.3 Transactional databases

In general, a **transactional database** consists of a file where each record represents a transaction. A transaction typically includes a unique transaction identity number (*trans_ID*), and a list of the **items** making up the transaction (such as items purchased in a store). The transactional database may have additional tables associated with it, which contain other information regarding the sale, such as the date of the transaction, the customer ID number, the ID number of the sales person, and of the branch at which the sale occurred, and so on.

Example 1.3 Transactions can be stored in a table, with one record per transaction. A fragment of a transactional database for *AllElectronics* is shown in Figure 1.9. From the relational database point of view, the *sales* table in Figure 1.9 is a nested relation because the attribute "list of item_ID's" contains a set of *items*. Since most relational database systems do not support nested relational structures, the transactional database is usually either stored in a flat file in a format similar to that of the table in Figure 1.9, or unfolded into a standard relation in a format similar to that of the *items_sold* table in Figure 1.6. □

As an analyst of the *AllElectronics* database, you may like to ask "Show me all the items purchased by Sandy Smith" or "How many transactions include item number I3?". Answering such queries may require a scan of the entire transactional database.

Suppose you would like to dig deeper into the data by asking "Which items sold well together?". This kind of *market basket data analysis* would enable you to bundle groups of items together as a strategy for maximizing sales. For example, given the knowledge that printers are commonly purchased together with computers, you could offer

an expensive model of printers at a discount to customers buying selected computers, in the hopes of selling more of the expensive printers. A regular data retrieval system is not able to answer queries like the one above. However, data mining systems for transactional data can do so by identifying sets of items which are frequently sold together.

1.3.4 Advanced database systems and advanced database applications

Relational database systems have been widely used in business applications. With the advances of database technology, various kinds of advanced database systems have emerged and are undergoing development to address the requirements of new database applications.

The new database applications include handling spatial data (such as maps), engineering design data (such as the design of buildings, system components, or integrated circuits), hypertext and multimedia data (including text, image, video, and audio data), time-related data (such as historical records or stock exchange data), and the World-Wide Web (a huge, widely distributed information repository made available by Internet). These applications require efficient data structures and scalable methods for handling complex object structures, variable length records, semi-structured or unstructured data, text and multimedia data, and database schemas with complex structures and dynamic changes.

In response to these needs, advanced database systems and specific application-oriented database systems have been developed. These include object-oriented and object-relational database systems, spatial database systems, temporal and time-series database systems, text and multimedia database systems, heterogeneous and legacy database systems, and the Web-based global information systems.

While such databases or information repositories require sophisticated facilities to efficiently store, retrieve, and update large amounts of complex data, they also provide fertile grounds and raise many challenging research and implementation issues for data mining.

1.4 Data mining functionalities — what kinds of patterns can be mined?

We have observed various types of data stores and database systems on which data mining can be performed. Let us now examine the kinds of data patterns that can be mined.

Data mining functionalities are used to specify the kind of patterns to be found in data mining tasks. In general, data mining tasks can be classified into two categories: **descriptive** and **predictive**. Descriptive mining tasks characterize the general properties of the data in the database. Predictive mining tasks perform inference on the current data in order to make predictions.

In some cases, users may have no idea of which kinds of patterns in their data may be interesting, and hence may like to search for several different kinds of patterns in parallel. Thus it is important to have a data mining system that can mine multiple kinds of patterns to accommodate different user expectations or applications. Furthermore, data mining systems should be able to discover patterns at various granularities (i.e., different levels of abstraction). To encourage interactive and exploratory mining, users should be able to easily "play" with the output patterns, such as by mouse clicking. Operations that can be specified by simple mouse clicks include adding or dropping a dimension (or an attribute), swapping rows and columns (**pivoting**, or axis rotation), changing dimension representations (e.g., from a 3-D cube to a sequence of 2-D cross tabulations, or **crosstabs**), or using OLAP roll-up or drill-down operations along dimensions. Such operations allow data patterns to be expressed from different angles of view and at multiple levels of abstraction.

Data mining systems should also allow users to specify hints to guide or focus the search for interesting patterns. Since some patterns may not hold for all of the data in the database, a measure of certainty or "trustworthiness" is usually associated with each discovered pattern.

Data mining functionalities, and the kinds of patterns they can discover, are described below.

1.4.1 Concept/class description: characterization and discrimination

Data can be associated with classes or concepts. For example, in the *AllElectronics* store, classes of items for sale include *computers* and *printers*, and concepts of customers include *bigSpenders* and *budgetSpenders*. It can be useful to describe individual classes and concepts in summarized, concise, and yet precise terms. Such descriptions of a class or a concept are called **class/concept descriptions**. These descriptions can be derived via (1) *data*

characterization, by summarizing the data of the class under study (often called the **target class**) in general terms, or (2) *data discrimination*, by comparison of the target class with one or a set of comparative classes (often called the **contrasting classes**), or (3) both data characterization and discrimination.

Data **characterization** is a summarization of the general characteristics or features of a target class of data. The data corresponding to the user-specified class are typically collected by a database query. For example, to study the characteristics of software products whose sales increased by 10% in the last year, one can collect the data related to such products by executing an SQL query.

There are several methods for effective data summarization and characterization. For instance, the data cube-based OLAP roll-up operation (Section 1.3.2) can be used to perform user-controlled data summarization along a specified dimension. This process is further detailed in Chapter 2 which discusses data warehousing. An *attribute-oriented induction* technique can be used to perform data generalization and characterization without step-by-step user interaction. This technique is described in Chapter 5.

The output of data characterization can be presented in various forms. Examples include **pie charts**, **bar charts**, **curves**, **multidimensional data cubes**, and **multidimensional tables**, including crosstabs. The resulting descriptions can also be presented as **generalized relations**, or in rule form (called **characteristic rules**). These different output forms and their transformations are discussed in Chapter 5.

Example 1.4 A data mining system should be able to produce a description summarizing the characteristics of customers who spend more than $1000 a year at *AllElectronics*. The result could be a general profile of the customers such as they are 40-50 years old, employed, and have excellent credit ratings. The system should allow users to drill-down on any dimension, such as on "employment" in order to view these customers according to their occupation. □

Data **discrimination** is a comparison of the general features of target class data objects with the general features of objets from one or a set of contrasting classes. The target and contrasting classes can be specified by the user, and the corresponding data objects retrieved through data base queries. For example, one may like to compare the general features of software products whose sales increased by 10% in the last year with those whose sales decreased by at least 30% during the same period.

The methods used for data discrimination are similar to those used for data characterization. The forms of output presentation are also similar, although discrimination descriptions should include comparative measures which help distinguish between the target and contrasting classes. Discrimination descriptions expressed in rule form are referred to as **discriminant rules**. The user should be able to manipulate the output for characteristic and discriminant descriptions.

Example 1.5 A data mining system should be able to compare two groups of *AllElectronics* customers, such as those who shop for computer products regularly (more than 4 times a month) vs. those who rarely shop for such products (i.e., less than three times a year). The resulting description could be a general, comparative profile of the customers such as 80% of the customers who frequently purchase computer products are between 20-40 years old and have a university education, whereas 60% of the customers who infrequently buy such products are either old or young, and have no university degree. Drilling-down on a dimension, such as *occupation*, or adding new dimensions, such as *income_level*, may help in finding even more discriminative features between the two classes. □

Concept description, including characterization and discrimination, is the topic of Chapter 5.

1.4.2 Association analysis

Association analysis is the discovery of *association rules* showing attribute-value conditions that occur frequently together in a given set of data. Association analysis is widely used for market basket or transaction data analysis.

More formally, **association rules** are of the form $X \Rightarrow Y$, i.e., "$A_1 \wedge \cdots \wedge A_m \rightarrow B_1 \wedge \cdots \wedge B_n$", where A_i (for $i \in \{1, \ldots, m\}$) and B_j (for $j \in \{1, \ldots, n\}$) are attribute-value pairs. The association rule $X \Rightarrow Y$ is interpreted as "database tuples that satisfy the conditions in X are also likely to satisfy the conditions in Y".

Example 1.6 Given the *AllElectronics* relational database, a data mining system may find association rules like

$$age(X, \text{“}20-29\text{”}) \wedge income(X, \text{“}20-30K\text{”}) \Rightarrow buys(X, \text{“}CD\ player\text{”}) \qquad [support = 2\%, confidence = 60\%]$$

meaning that of the *AllElectronics* customers under study, 2% (**support**) are 20-29 years of age with an income of 20-30K and have purchased a CD player at *AllElectronics*. There is a 60% probability (**confidence**, or certainty) that a customer in this age and income group will purchase a CD player.

Note that this is an association between more than one attribute, or predicate (i.e., *age, income*, and *buys*). Adopting the terminology used in multidimensional databases, where each attribute is referred to as a dimension, the above rule can be referred to as a **multidimensional association rule**.

Suppose, as a marketing manager of *AllElectronics*, you would like to determine which items are frequently purchased together within the same transactions. An example of such a rule is

$$contains(T, \text{“}computer\text{”}) \Rightarrow contains(T, \text{“}software\text{”}) \qquad [support = 1\%, confidence = 50\%]$$

meaning that if a transaction T contains “computer”, there is a 50% chance that it contains “software” as well, and 1% of all of the transactions contain both. This association rule involves a single attribute or predicate (i.e., *contains*) which repeats. Association rules that contain a single predicate are referred to as **single-dimensional association rules**. Dropping the predicate notation, the above rule can be written simply as “$computer \Rightarrow software$ [1%, 50%]”. □

In recent years, many algorithms have been proposed for the efficient mining of association rules. Association rule mining is discussed in detail in Chapter 6.

1.4.3 Classification and prediction

Classification is the processing of finding a set of **models** (or functions) which describe and distinguish data classes or concepts, for the purposes of being able to use the model to predict the class of objects whose class label is unknown. The derived model is based on the analysis of a set of **training data** (i.e., data objects whose class label is known).

The derived model may be represented in various forms, such as *classification (IF-THEN) rules*, *decision trees*, *mathematical formulae*, or *neural networks*. A **decision tree** is a flow-chart-like tree structure, where each node denotes a test on an attribute value, each branch represents an outcome of the test, and tree leaves represent classes or class distributions. Decision trees can be easily converted to classification rules. A **neural network** is a collection of linear threshold units that can be trained to distinguish objects of different classes.

Classification can be used for predicting the class label of data objects. However, in many applications, one may like to predict some missing or unavailable *data values* rather than class labels. This is usually the case when the predicted values are numerical data, and is often specifically referred to as **prediction**. Although prediction may refer to both data value prediction and class label prediction, it is usually confined to data value prediction and thus is distinct from classification. Prediction also encompasses the identification of distribution *trends* based on the available data.

Classification and prediction may need to be preceded by **relevance analysis** which attempts to identify attributes that do not contribute to the classification or prediction process. These attributes can then be excluded.

Example 1.7 Suppose, as sales manager of *AllElectronics*, you would like to classify a large set of items in the store, based on three kinds of responses to a sales campaign: *good response*, *mild response*, and *no response*. You would like to derive a model for each of these three classes based on the descriptive features of the items, such as *price, brand, place_made, type*, and *category*. The resulting classification should maximally distinguish each class from the others, presenting an organized picture of the data set. Suppose that the resulting classification is expressed in the form of a decision tree. The decision tree, for instance, may identify *price* as being the single factor which best distinguishes the three classes. The tree may reveal that, after *price*, other features which help further distinguish objects of each class from another include *brand* and *place_made*. Such a decision tree may help you understand the impact of the given sales campaign, and design a more effective campaign for the future. □

Chapter 7 discusses classification and prediction in further detail.

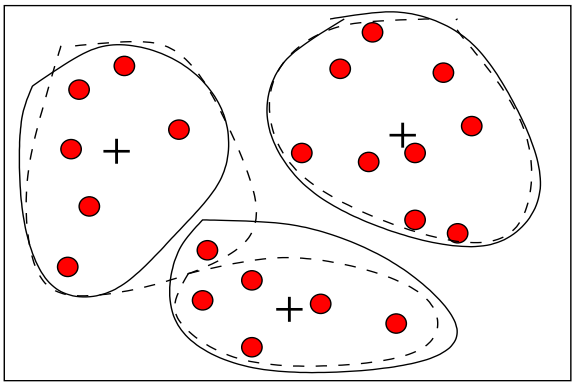

Figure 1.10: A 2-D plot of customer data with respect to customer locations in a city, showing three data clusters. Each cluster 'center' is marked with a '+'.

1.4.4 Clustering analysis

Unlike classification and predication, which analyze class-labeled data objects, **clustering** analyzes data objects without consulting a known class label. In general, the class labels are not present in the training data simply because they are not known to begin with. Clustering can be used to generate such labels. The objects are clustered or grouped based on the principle of *maximizing the intraclass similarity and minimizing the interclass similarity.* That is, clusters of objects are formed so that objects within a cluster have high similarity in comparison to one another, but are very dissimilar to objects in other clusters. Each cluster that is formed can be viewed as a class of objects, from which rules can be derived. Clustering can also facilitate **taxonomy formation**, that is, the organization of observations into a hierarchy of classes that group similar events together.

Example 1.8 Clustering analysis can be performed on *AllElectronics* customer data in order to identify homogeneous subpopulations of customers. These clusters may represent individual target groups for marketing. Figure 1.10 shows a 2-D plot of customers with respect to customer locations in a city. Three clusters of data points are evident. □

Clustering analysis forms the topic of Chapter 8.

1.4.5 Evolution and deviation analysis

Data **evolution analysis** describes and models regularities or trends for objects whose behavior changes over time. Although this may include characterization, discrimination, association, classification, or clustering of *time-related* data, distinct features of such an analysis include time-series data analysis, sequence or periodicity pattern matching, and similarity-based data analysis.

Example 1.9 Suppose that you have the major stock market (time-series) data of the last several years available from the New York Stock Exchange and you would like to invest in shares of high-tech industrial companies. A data mining study of stock exchange data may identify stock evolution regularities for overall stocks and for the stocks of particular companies. Such regularities may help predict future trends in stock market prices, contributing to your decision making regarding stock investments. □

In the analysis of time-related data, it is often desirable not only to model the general evolutionary trend of the data, but also to identify data deviations which occur over time. **Deviations** are differences between measured values and corresponding references such as previous values or normative values. A data mining system performing deviation analysis, upon the detection of a set of deviations, may do the following: describe the characteristics of the deviations, try to explain the reason behind them, and suggest actions to bring the deviated values back to their expected values.

Example 1.10 A decrease in *total sales* at *AllElectronics* for the last month, in comparison to that of the same month of the last year, is a deviation pattern. Having detected a significant deviation, a data mining system may go further and attempt to explain the detected pattern (e.g., did the company have more sales personnel last year in comparison to the same period this year?). □

Data evolution and deviation analysis are discussed in Chapter 9.

1.5 Are all of the patterns interesting?

A data mining system has the potential to generate thousands or even millions of patterns, or rules. *Are all of the patterns interesting?* Typically not — only a small fraction of the patterns potentially generated would actually be of interest to any given user.

This raises some serious questions for data mining: *What makes a pattern interesting? Can a data mining system generate all of the interesting patterns? Can a data mining system generate only the interesting patterns?*

To answer the first question, a pattern is **interesting** if (1) it is *easily understood* by humans, (2) *valid* on new or test data with some degree of *certainty*, (3) potentially *useful*, and (4) *novel*. A pattern is also interesting if it validates a hypothesis that the user *sought to confirm*. An interesting pattern represents **knowledge**.

Several **objective measures of pattern interestingness** exist. These are based on the structure of discovered patterns and the statistics underlying them. An objective measure for association rules of the form $X \Rightarrow Y$ is rule **support**, representing the percentage of data samples that the given rule satisfies. Another objective measure for association rules is **confidence**, which assesses the degree of certainty of the detected association. It is defined as the conditional probability that a pattern Y is true given that X is true. More formally, support and confidence are defined as

$$support(X \Rightarrow Y) = Prob\{X \cup Y\}.$$

$$confidence(X \Rightarrow Y) = Prob\{Y|X\}.$$

In general, each interestingness measure is associated with a threshold, which may be controlled by the user. For example, rules that do not satisfy a confidence threshold of say, 50%, can be considered uninteresting. Rules below the threshold likely reflect noise, exceptions, or minority cases, and are probably of less value.

Although objective measures help identify interesting patterns, they are insufficient unless combined with subjective measures that reflect the needs and interests of a particular user. For example, patterns describing the characteristics of customers who shop frequently at *AllElectronics* should interest the marketing manager, but may be of little interest to analysts studying the same database for patterns on employee performance. Furthermore, many patterns that are interesting by objective standards may represent common knowledge, and therefore, are actually uninteresting. **Subjective interestingness measures** are based on user beliefs in the data. These measures find patterns interesting if they are **unexpected** (contradicting a user belief) or offer strategic information on which the user can act. In the latter case, such patterns are referred to as **actionable**. Patterns that are expected can be interesting if they confirm a hypothesis that the user wished to validate, or resemble a user's hunch.

The second question, "*Can a data mining system generate* all *of the interesting patterns?*", refers to the **completeness** of a data mining algorithm. It is unrealistic and inefficient for data mining systems to generate all of the possible patterns. Instead, a focused search which makes use of interestingness measures should be used to control pattern generation. This is often sufficient to ensure the completeness of the algorithm. Association rule mining is an example where the use of interestingness measures can ensure the completeness of mining. The methods involved are examined in detail in Chapter 6.

Finally, the third question, *"Can a data mining system generate* only *the interesting patterns?"*, is an optimization problem in data mining. It is highly desirable for data mining systems to generate only the interesting patterns. This would be much more efficient for users and data mining systems, since neither would have to search through the patterns generated in order to identify the truely interesting ones. Such optimization remains a challenging issue in data mining.

Measures of pattern interestingness are essential for the efficient discovery of patterns of value to the given user. Such measures can be used after the data mining step in order to rank the discovered patterns according to their interestingness, filtering out the uninteresting ones. More importantly, such measures can be used to guide and

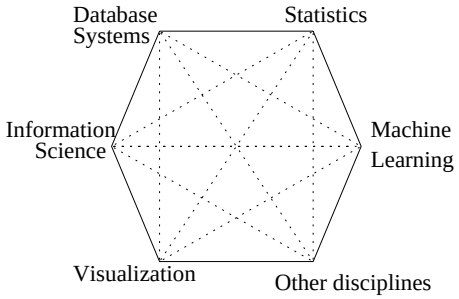

Figure 1.11: Data mining as a confluence of multiple disciplines.

constrain the discovery process, improving the search efficiency by pruning away subsets of the pattern space that do not satisfy pre-specified interestingness constraints.

Methods to assess pattern interestingness, and their use to improve data mining efficiency are discussed throughout the book, with respect to each kind of pattern that can be mined.

1.6 A classification of data mining systems

Data mining is an interdisciplinary field, the confluence of a set of disciplines (as shown in Figure 1.11), including database systems, statistics, machine learning, visualization, and information science. Moreover, depending on the data mining approach used, techniques from other disciplines may be applied, such as neural networks, fuzzy and/or rough set theory, knowledge representation, inductive logic programming, or high performance computing. Depending on the kinds of data to be mined or on the given data mining application, the data mining system may also integrate techniques from spatial data analysis, information retrieval, pattern recognition, image analysis, signal processing, computer graphics, Web technology, economics, or psychology.

Because of the diversity of disciplines contributing to data mining, data mining research is expected to generate a large variety of data mining systems. Therefore, it is necessary to provide a clear classification of data mining systems. Such a classification may help potential users distinguish data mining systems and identify those that best match their needs. Data mining systems can be categorized according to various criteria, as follows.

- Classification according to the *kinds of databases* mined.

  A data mining system can be classified according to the kinds of databases mined. Database systems themselves can be classified according to different criteria (such as data models, or the types of data or applications involved), each of which may require its own data mining technique. Data mining systems can therefore be classified accordingly.

  For instance, if classifying according to data models, we may have a relational, transactional, object-oriented, object-relational, or data warehouse mining system. If classifying according to the special types of data handled, we may have a spatial, time-series, text, or multimedia data mining system, or a World-Wide Web mining system. Other system types include heterogeneous data mining systems, and legacy data mining systems.

- Classification according to the *kinds of knowledge* mined.

  Data mining systems can be categorized according to the kinds of knowledge they mine, i.e., based on data mining functionalities, such as characterization, discrimination, association, classification, clustering, trend and evolution analysis, deviation analysis, similarity analysis, etc. A comprehensive data mining system usually provides multiple and/or integrated data mining functionalities.

  Moreover, data mining systems can also be distinguished based on the granularity or levels of abstraction of the knowledge mined, including generalized knowledge (at a high level of abstraction), primitive-level knowledge (at a raw data level), or knowledge at multiple levels (considering several levels of abstraction). An advanced data mining system should facilitate the discovery of knowledge at multiple levels of abstraction.

- Classification according to the *kinds of techniques* utilized.

Data mining systems can also be categorized according to the underlying data mining techniques employed. These techniques can be described according to the degree of user interaction involved (e.g., autonomous systems, interactive exploratory systems, query-driven systems), or the methods of data analysis employed (e.g., database-oriented or data warehouse-oriented techniques, machine learning, statistics, visualization, pattern recognition, neural networks, and so on). A sophisticated data mining system will often adopt multiple data mining techniques or work out an effective, integrated technique which combines the merits of a few individual approaches.

Chapters 5 to 8 of this book are organized according to the various kinds of knowledge mined. In Chapter 9, we discuss the mining of different kinds of data on a variety of advanced and application-oriented database systems.

1.7 Major issues in data mining

The scope of this book addresses major issues in data mining regarding mining methodology, user interaction, performance, and diverse data types. These issues are introduced below:

1. **Mining methodology and user-interaction issues.** These reflect the kinds of knowledge mined, the ability to mine knowledge at multiple granularities, the use of domain knowledge, ad-hoc mining, and knowledge visualization.

   - *Mining different kinds of knowledge in databases.*
     Since different users can be interested in different kinds of knowledge, data mining should cover a wide spectrum of data analysis and knowledge discovery tasks, including data characterization, discrimination, association, classification, clustering, trend and deviation analysis, and similarity analysis. These tasks may use the same database in different ways and require the development of numerous data mining techniques.
   - *Interactive mining of knowledge at multiple levels of abstraction.*
     Since it is difficult to know exactly what can be discovered within a database, the data mining process should be *interactive*. For databases containing a huge amount of data, appropriate sampling technique can first be applied to facilitate interactive data exploration. Interactive mining allows users to focus the search for patterns, providing and refining data mining requests based on returned results. Specifically, knowledge should be mined by drilling-down, rolling-up, and pivoting through the data space and knowledge space interactively, similar to what OLAP can do on data cubes. In this way, the user can interact with the data mining system to view data and discovered patterns at multiple granularities and from different angles.
   - *Incorporation of background knowledge.*
     Background knowledge, or information regarding the domain under study, may be used to guide the discovery process and allow discovered patterns to be expressed in concise terms and at different levels of abstraction. Domain knowledge related to databases, such as integrity constraints and deduction rules, can help focus and speed up a data mining process, or judge the interestingness of discovered patterns.
   - *Data mining query languages and ad-hoc data mining.*
     Relational query languages (such as SQL) allow users to pose ad-hoc queries for data retrieval. In a similar vein, high-level **data mining query languages** need to be developed to allow users to describe ad-hoc data mining tasks by facilitating the specification of the relevant sets of data for analysis, the domain knowledge, the kinds of knowledge to be mined, and the conditions and interestingness constraints to be enforced on the discovered patterns. Such a language should be integrated with a database or data warehouse query language, and optimized for efficient and flexible data mining.
   - *Presentation and visualization of data mining results.*
     Discovered knowledge should be expressed in high-level languages, visual representations, or other expressive forms so that the knowledge can be easily understood and directly usable by humans. This is especially crucial if the data mining system is to be interactive. This requires the system to adopt expressive knowledge representation techniques, such as trees, tables, rules, graphs, charts, crosstabs, matrices, or curves.

- *Handling outlier or incomplete data.*

  The data stored in a database may reflect outliers — noise, exceptional cases, or incomplete data objects. These objects may confuse the analysis process, causing overfitting of the data to the knowledge model constructed. As a result, the accuracy of the discovered patterns can be poor. Data cleaning methods and data analysis methods which can handle outliers are required. While most methods discard outlier data, such data may be of interest in itself such as in fraud detection for finding unusual usage of telecommunication services or credit cards. This form of data analysis is known as **outlier mining**.

- *Pattern evaluation: the interestingness problem.*

  A data mining system can uncover thousands of patterns. Many of the patterns discovered may be uninteresting to the given user, representing common knowledge or lacking novelty. Several challenges remain regarding the development of techniques to assess the interestingness of discovered patterns, particularly with regard to subjective measures which estimate the value of patterns with respect to a given user class, based on user beliefs or expectations. The use of interestingness measures to guide the discovery process and reduce the search space is another active area of research.

2. **Performance issues.** These include efficiency, scalability, and parallelization of data mining algorithms.

   - *Efficiency and scalability of data mining algorithms.*

     To effectively extract information from a huge amount of data in databases, data mining algorithms must be **efficient** and **scalable**. That is, the running time of a data mining algorithm must be predictable and acceptable in large databases. Algorithms with exponential or even medium-order polynomial complexity will not be of practical use. From a database perspective on knowledge discovery, efficiency and scalability are key issues in the implementation of data mining systems. Many of the issues discussed above under *mining methodology and user-interaction* must also consider efficiency and scalability.

   - *Parallel, distributed, and incremental updating algorithms.*

     The huge size of many databases, the wide distribution of data, and the computational complexity of some data mining methods are factors motivating the development of **parallel and distributed data mining algorithms**. Such algorithms divide the data into partitions, which are processed in parallel. The results from the partitions are then merged. Moreover, the high cost of some data mining processes promotes the need for **incremental** data mining algorithms which incorporate database updates without having to mine the entire data again "from scratch". Such algorithms perform knowledge modification incrementally to amend and strengthen what was previously discovered.

3. **Issues relating to the diversity of database types.**

   - *Handling of relational and complex types of data.*

     There are many kinds of data stored in databases and data warehouses. Can we expect that a single data mining system can perform effective mining on all kinds of data? Since relational databases and data warehouses are widely used, the development of efficient and effective data mining systems for such data is important. However, other databases may contain complex data objects, hypertext and multimedia data, spatial data, temporal data, or transaction data. It is unrealistic to expect one system to mine all kinds of data due to the diversity of data types and different goals of data mining. Specific data mining systems should be constructed for mining specific kinds of data. Therefore, one may expect to have different data mining systems for different kinds of data.

   - *Mining information from heterogeneous databases and global information systems.*

     Local and wide-area computer networks (such as the Internet) connect many sources of data, forming huge, distributed, and heterogeneous databases. The discovery of knowledge from different sources of structured, semi-structured, or unstructured data with diverse data semantics poses great challenges to data mining. Data mining may help disclose high-level data regularities in multiple heterogeneous databases that are unlikely to be discovered by simple query systems and may improve information exchange and interoperability in heterogeneous databases.

The above issues are considered major requirements and challenges for the further evolution of data mining technology. Some of the challenges have been addressed in recent data mining research and development, *to a*

certain extent, and are now considered *requirements*, while others are still at the research stage. The issues, however, continue to stimulate further investigation and improvement. Additional issues relating to applications, privacy, and the social impact of data mining are discussed in Chapter 10, the final chapter of this book.

1.8 Summary

- **Database technology** has evolved from primitive file processing to the development of database management systems with query and transaction processing. Further progress has led to the increasing demand for efficient and effective data analysis and data understanding tools. This need is a result of the explosive growth in data collected from applications including business and management, government administration, scientific and engineering, and environmental control.
- **Data mining** is the task of discovering interesting patterns from large amounts of data where the data can be stored in databases, data warehouses, or other information repositories. It is a young interdisciplinary field, drawing from areas such as database systems, data warehousing, statistics, machine learning, data visualization, information retrieval, and high performance computing. Other contributing areas include neural networks, pattern recognition, spatial data analysis, image databases, signal processing, and inductive logic programming.
- A **knowledge discovery process** includes data cleaning, data integration, data selection, data transformation, data mining, pattern evaluation, and knowledge presentation.
- Data patterns can be mined from many different kinds of **databases**, such as relational databases, data warehouses, and transactional, object-relational, and object-oriented databases. Interesting data patterns can also be extracted from other kinds of **information repositories**, including spatial, time-related, text, multimedia, and legacy databases, and the World-Wide Web.
- A **data warehouse** is a repository for long term storage of data from multiple sources, organized so as to facilitate management decision making. The data are stored under a unified schema, and are typically summarized. Data warehouse systems provide some data analysis capabilities, collectively referred to as **OLAP (On-Line Analytical Processing)**. OLAP operations include drill-down, roll-up, and pivot.
- **Data mining functionalities** include the discovery of concept/class descriptions (i.e., characterization and discrimination), association, classification, prediction, clustering, trend analysis, deviation analysis, and similarity analysis. Characterization and discrimination are forms of data summarization.
- A pattern represents **knowledge** if it is easily understood by humans, valid on test data with some degree of certainty, potentially useful, novel, or validates a hunch about which the user was curious. Measures of **pattern interestingness**, either *objective* or *subjective*, can be used to guide the discovery process.
- **Data mining systems** can be **classified** according to the kinds of databases mined, the kinds of knowledge mined, or the techniques used.
- Efficient and effective data mining in large databases poses numerous **requirements** and great **challenges** to researchers and developers. The issues involved include data mining methodology, user-interaction, performance and scalability, and the processing of a large variety of data types. Other issues include the exploration of data mining applications, and their social impacts.

Exercises

1. What is data mining? In your answer, address the following:
   (a) Is it another hype?
   (b) Is it a simple transformation of technology developed from databases, statistics, and machine learning?
   (c) Explain how the evolution of database technology led to data mining.
   (d) Describe the steps involved in data mining when viewed as a process of knowledge discovery.

2. Present an example where data mining is crucial to the success of a business. What data mining functions does this business need? Can they be performed alternatively by data query processing or simple statistical analysis?

3. How is a data warehouse different from a database? How are they similar to each other?

4. Define each of the following data mining functionalities: characterization, discrimination, association, classification, prediction, clustering, and evolution and deviation analysis. Give examples of each data mining functionality, using a real-life database that you are familiar with.

5. Suppose your task as a software engineer at *Big-University* is to design a data mining system to examine their university course database, which contains the following information: the name, address, and status (e.g., undergraduate or graduate) of each student, and their cumulative grade point average (GPA). Describe the architecture you would choose. What is the purpose of each component of this architecture?

6. Based on your observation, describe another possible kind of knowledge that needs to be discovered by data mining methods but has not been listed in this chapter. Does it require a mining methodology that is quite different from those outlined in this chapter?

7. What is the difference between discrimination and classification? Between characterization and clustering? Between classification and prediction? For each of these pairs of tasks, how are they similar?

8. Describe three challenges to data mining regarding data mining methodology and user-interaction issues.

9. Describe two challenges to data mining regarding performance issues.

Bibliographic Notes

The book *Knowledge Discovery in Databases*, edited by Piatetsky-Shapiro and Frawley [26], is an early collection of research papers on knowledge discovery in databases. The book *Advances in Knowledge Discovery and Data Mining*, edited by Fayyad et al. [10], is a good collection of recent research results on knowledge discovery and data mining. Other books on data mining include *Predictive Data Mining* by Weiss and Indurkhya [37], and *Data Mining* by Adriaans and Zantinge [1]. There are also books containing collections of papers on particular aspects of knowledge discovery, such as *Machine Learning & Data Mining: Methods and Applications*, edited by Michalski, Bratko, and Kubat [20], *Rough Sets, Fuzzy Sets and Knowledge Discovery*, edited by Ziarko [39], as well as many tutorial notes on data mining, such as *Tutorial Notes of 1999 International Conference on Knowledge Disocvery and Data Mining* (*KDD99*) published by ACM Press.

KDD Nuggets is a regular, free electronic newsletter containing information relevant to knowledge discovery and data mining. Contributions can be e-mailed with a descriptive subject line (and a URL) to "gps@kdnuggets.com". Information regarding subscription can be found at "http://www.kdnuggets.com/subscribe.html". *KDD Nuggets* has been moderated by Piatetsky-Shapiro since 1991. The Internet site, *Knowledge Discovery Mine*, located at "http://www.kdnuggets.com/", contains a good collection of KDD-related information.

The research community of data mining set up a new academic organization called ACM-SIGKDD, a Special Interested Group on Knowledge Discovery in Databases under ACM in 1998. The community started its first international conference on knowledge discovery and data mining in 1995 [12]. The conference evolved from the four *international workshops on knowledge discovery in databases*, held from 1989 to 1994 [7, 8, 13, 11]. ACM-SIGKDD is organizing its first, but the fifth international conferences on knowledge discovery and data mining (KDD'99). A new journal, *Data Mining and Knowledge Discovery*, published by Kluwers Publishers, has been available since 1997.

Research in data mining has also been published in major textbooks, conferences and journals on databases, statistics, machine learning, and data visualization. References to such sources are listed below.

Popular textbooks on database systems include *Database System Concepts, 3rd ed.*, by Silberschatz, Korth, and Sudarshan [30], *Fundamentals of Database Systems, 2nd ed.*, by Elmasri and Navathe [9], and *Principles of Database and Knowledge-Base Systems, Vol. 1*, by Ullman [36]. For an edited collection of seminal articles on database systems, see *Readings in Database Systems* by Stonebraker [32]. Overviews and discussions on the achievements and research challenges in database systems can be found in Stonebraker et al. [33], and Silberschatz, Stonebraker, and Ullman [31].

Many books on data warehouse technology, systems and applications have been published in the last several years, such as *The Data Warehouse Toolkit* by Kimball [17], and *Building the Data Warehouse* by Inmon [14]. Chaudhuri and Dayal [3] present a comprehensive overview of data warehouse technology.

Research results relating to data mining and data warehousing have been published in the proceedings of many international database conferences, including *ACM-SIGMOD International Conference on Management of Data (SIGMOD), International Conference on Very Large Data Bases (VLDB), ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS), International Conference on Data Engineering (ICDE), International Conference on Extending Database Technology (EDBT), International Conference on Database Theory (ICDT), International Conference on Information and Knowledge Management (CIKM)*, and *International Symposium on Database Systems for Advanced Applications (DASFAA)*. Research in data mining is also published in major database journals, such as *IEEE Transactions on Knowledge and Data Engineering (TKDE), ACM Transactions on Database Systems (TODS), Journal of ACM (JACM), Information Systems, The VLDB Journal, Data and Knowledge Engineering*, and *International Journal of Intelligent Information Systems (JIIS)*.

There are many textbooks covering different topics in statistical analysis, such as *Probability and Statistics for Engineering and the Sciences, 4th ed.* by Devore [4], *Applied Linear Statistical Models, 4th ed.* by Neter et al. [25], *An Introduction to Generalized Linear Models* by Dobson [5], *Applied Statistical Time Series Analysis* by Shumway [29], and *Applied Multivariate Statistical Analysis, 3rd ed.* by Johnson and Wichern [15].

Research in statistics is published in the proceedings of several major statistical conferences, including *Joint Statistical Meetings*, *International Conference of the Royal Statistical Society*, and *Symposium on the Interface: Computing Science and Statistics*. Other source of publication include the *Journal of the Royal Statistical Society*, *The Annals of Statistics*, *Journal of American Statistical Association*, *Technometrics*, and *Biometrika*.

Textbooks and reference books on machine learning include *Machine Learning* by Mitchell [24], *Machine Learning, An Artificial Intelligence Approach*, Vols. 1-4, edited by Michalski et al. [21, 22, 18, 23], *C4.5: Programs for Machine Learning* by Quinlan [27], and *Elements of Machine Learning* by Langley [19]. The book *Computer Systems that Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems*, by Weiss and Kulikowski [38], compares classification and prediction methods from several different fields, including statistics, machine learning, neural networks, and expert systems. For an edited collection of seminal articles on machine learning, see *Readings in Machine Learning* by Shavlik and Dietterich [28].

Machine learning research is published in the proceedings of several large machine learning and artificial intelligence conferences, including the *International Conference on Machine Learning (ML), ACM Conference on Computational Learning Theory (COLT), International Joint Conference on Artificial Intelligence (IJCAI)*, and *American Association of Artificial Intelligence Conference (AAAI)*. Other sources of publication include major machine learning, artificial intelligence, and knowledge system journals, some of which have been mentioned above. Others include *Machine Learning (ML), Artificial Intelligence Journal (AI)* and *Cognitive Science*. An overview of classification from a statistical pattern recognition perspective can be found in Duda and Hart [6].

Pioneering work on data visualization techniques is described in *The Visual Display of Quantitative Information* [34] and *Envisioning Information* [35], both by Tufte, and *Graphics and Graphic Information Processing* by Bertin [2]. *Visual Techniques for Exploring Databases* by Keim [16] presents a broad tutorial on visualization for data mining. Major conferences and symposiums on visualization include *ACM Human Factors in Computing Systems (CHI)*, *Visualization*, and *International Symposium on Information Visualization*. Research on visualization is also published in *Transactions on Visualization and Computer Graphics*, *Journal of Computational and Graphical Statistics*, and *IEEE Computer Graphics and Applications*.

Bibliography

[1] P. Adriaans and D. Zantinge. *Data Mining.* Addison-Wesley: Harlow, England, 1996.

[2] J. Bertin. *Graphics and Graphic Information Processing.* Berlin, 1981.

[3] S. Chaudhuri and U. Dayal. An overview of data warehousing and OLAP technology. *ACM SIGMOD Record*, 26:65–74, 1997.

[4] J. L. Devore. *Probability and Statistics for Engineering and the Science, 4th ed.* Duxbury Press, 1995.

[5] A. J. Dobson. *An Introduction to Generalized Linear Models.* Chapman and Hall, 1990.

[6] R. Duda and P. Hart. *Pattern Classification and Scene Analysis.* Wiley: New York, 1973.

[7] G. Piatetsky-Shapiro (ed.). *Notes of IJCAI'89 Workshop Knowledge Discovery in Databases (KDD'89).* Detroit, Michigan, July 1989.

[8] G. Piatetsky-Shapiro (ed.). *Notes of AAAI'91 Workshop Knowledge Discovery in Databases (KDD'91).* Anaheim, CA, July 1991.

[9] R. Elmasri and S. B. Navathe. *Fundamentals of Database Systems, 2nd ed.* Bemjamin/Cummings, 1994.

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

[11] U.M. Fayyad and R. Uthurusamy (eds.). *Notes of AAAI'94 Workshop Knowledge Discovery in Databases (KDD'94).* Seattle, WA, July 1994.

[12] U.M. Fayyad and R. Uthurusamy (eds.). *Proc. 1st Int. Conf. Knowledge Discovery and Data Mining (KDD'95).* AAAI Press, Aug. 1995.

[13] U.M. Fayyad, R. Uthurusamy, and G. Piatetsky-Shapiro (eds.). *Notes of AAAI'93 Workshop Knowledge Discovery in Databases (KDD'93).* Washington, DC, July 1993.

[14] W. H. Inmon. *Building the Data Warehouse.* John Wiley, 1996.

[15] R. A. Johnson and D. W. Wickern. *Applied Multivariate Statistical Analysis, 3rd ed.* Prentice Hall, 1992.

[16] D. A. Keim. Visual techniques for exploring databases. In *Tutorial Notes, 3rd Int. Conf. on Knowledge Discovery and Data Mining (KDD'97)*, Newport Beach, CA, Aug. 1997.

[17] R. Kimball. *The Data Warehouse Toolkit.* John Wiley & Sons, New York, 1996.

[18] Y. Kodratoff and R. S. Michalski. *Machine Learning, An Artificial Intelligence Approach, Vol. 3.* Morgan Kaufmann, 1990.

[19] P. Langley. *Elements of Machine Learning.* Morgan Kaufmann, 1996.

[20] R. S. Michalski, I. Bratko, and M. Kubat. *Machine Learning and Data Mining: Methods and Applications.* John Wiley & Sons, 1998.

[21] R. S. Michalski, J. G. Carbonell, and T. M. Mitchell. *Machine Learning, An Artificial Intelligence Approach, Vol. 1.* Morgan Kaufmann, 1983.

[22] R. S. Michalski, J. G. Carbonell, and T. M. Mitchell. *Machine Learning, An Artificial Intelligence Approach, Vol. 2.* Morgan Kaufmann, 1986.

[23] R. S. Michalski and G. Tecuci. *Machine Learning, A Multistrategy Approach, Vol. 4.* Morgan Kaufmann, 1994.

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

[25] J. Neter, M. H. Kutner, C. J. Nachtsheim, and L. Wasserman. *Applied Linear Statistical Models, 4th ed.* Irwin: Chicago, 1996.

[26] G. Piatetsky-Shapiro and W. J. Frawley. *Knowledge Discovery in Databases.* AAAI/MIT Press, 1991.

[27] J. R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann, 1993.

[28] J.W. Shavlik and T.G. Dietterich. *Readings in Machine Learning.* Morgan Kaufmann, 1990.

[29] R. H. Shumway. *Applied Statistical Time Series Analysis.* Prentice Hall, 1988.

[30] A. Silberschatz, H. F. Korth, and S. Sudarshan. *Database System Concepts, 3ed.* McGraw-Hill, 1997.

[31] A. Silberschatz, M. Stonebraker, and J. D. Ullman. Database research: Achievements and opportunities into the 21st century. *ACM SIGMOD Record*, 25:52–63, March 1996.

[32] M. Stonebraker. *Readings in Database Systems, 2ed.* Morgan Kaufmann, 1993.

[33] M. Stonebraker, R. Agrawal, U. Dayal, E. Neuhold, and A. Reuter. DBMS research at a crossroads: The vienna update. In *Proc. 19th Int. Conf. Very Large Data Bases*, pages 688–692, Dublin, Ireland, Aug. 1993.

[34] E. R. Tufte. *The Visual Display of Quantitative Information.* Graphics Press, Cheshire, CT, 1983.

[35] E. R. Tufte. *Envisioning Information.* Graphics Press, Cheshire, CT, 1990.

[36] J. D. Ullman. *Principles of Database and Knowledge-Base Systems, Vol. 1.* Computer Science Press, 1988.

[37] S. M. Weiss and N. Indurkhya. *Predictive Data Mining.* Morgan Kaufmann, 1998.

[38] S. M. Weiss and C. A. Kulikowski. *Computer Systems that Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems.* Morgan Kaufman, 1991.

[39] W. Ziarko. *Rough Sets, Fuzzy Sets and Knowledge Discovery.* Springer-Verlag, 1994.