8-1-1999

DATA MINING AND PROBABILISTIC NETWORK-BASED MECHANISM FOR DATA WAREHOUSING

Mei-Ling Shyu
Purdue University School of Electrical and Computer Engineering

R. L. Kashyap
Purdue University School of Electrical and Computer Engineering

Shyu, Mei-Ling and Kashyap, R. L., "DATA MINING AND PROBABILISTIC NETWORK-BASED MECHANISM FOR DATA WAREHOUSING" (1999). *ECE Technical Reports.* Paper 44.
http://docs.lib.purdue.edu/ecetr/44

DATA MINING AND PROBABILISTIC NETWORK-BASED MECHANISM FOR DATA WAREHOUSING

MEI-LING SHYU
R. L. KASHYAP

TR-ECE 99-11
AUGUST 1999

SCHOOL OF ELECTRICAL
AND COMPUTER ENGINEERING
PURDUE UNIVERSITY
WEST LAFAYETTE, INDIANA 47907-1285

DATA MINING AND PROBABILISTIC NETWORK-BASED MECHANISM FOR DATA WAREHOUSING [1]

Mei-Ling Shyu and R. L. Kashyap

School of Electrical and Computer Engineering
1285 EE Building
Purdue University
West Lafayette, IN 47907-1285

'This work has been partially supported by National Science Foundation under contract IRI 9619812.

TABLE OF CONTENTS

ABSTRACT

A good *multimedia database management system (MDBMS)* should be able to store, retrieve, and manage rich semantic data in multimedia database systems. Due to the complexity of real-world applications, the number of databases and the volumes of data in databases have increased tremendously. With the explosive growth in the amount and complexity of data, how to effectively manage the network of databases and utilize the large amount of data becomes important.

For this purpose, a probabilistic network-based mechanism for constructing a federation of data warehouses and speeding up information retrieval to facilitate the functionality of an MDBMS is proposed. Our solution procedure consists of three steps. First, we build the probabilistic network by reasoning the probability distributions and mining the generalized affinity-based associations from a set of historical data collected from the network of operational databases. By doing so, the summarized and useful knowledge can be discovered. Second, we derive a similarity measure method to construct a federation of data warehouses so as to reduce the number of inter-warehouse accesses required for queries. Those databases with high similarity values are placed in the same data warehouse. The similarity value is measured via a stochastic process from the mined probability distributions. Third, a second stochastic process generates a list of possible paths with respect to a given query and specifies the particular media objects over the constructed data warehouses so as to speed up multimedia query processing and information retrieval.

1. INTRODUCTION

1.1 Motivation and Problem Definition

The exponential growth of computer networks and data-collection technology has generated an incredibly large offer of products and services for the users of the computer networks. Recently, multimedia information systems have emerged as a fruitful area for research due to the recent progress in high-speed communication networks, large capacity storage devices, digitized media, and data compression technologies over the last few years. Multimedia information has been used in several applications including manufacturing, medicine, education, business, entertainment, etc. A typical example is to browse through or query a list of merchandise by reading prices (text data), listening to descriptions (audio data), and viewing demonstrations (video data). With the increasing use of multimedia database systems, there is the need for a *multimedia database management system (MDBMS)* which has the capabilities to provide a suitable environment for storing, managing, and retrieving data in multimedia systems. Here, MDBMSs are considered since an MDBMS stores and manages not only images, audio, graphics, animation, and full-motion video, but also text as in traditional text-based databases. Also, data access and manipulation for multimedia databases are more complicated than those of the conventional databases since it is necessary to incorporate diverse media with diverse characteristics.

As the application complexity increases, the number of databases and the volumes of data in databases have increased tremendously in such an information-providing environment. With the explosive growth in the amount and complexity of data, the need to extend the database technology to effectively manage the databases and utilize the large amount of data has posed a great challenge to the database research community. The advanced data storage technology and database management systems have increased our capabilities to collect and store data of all kinds. However, our ability to interpret and analyze the data is still limited, creating an urgent need to accelerate discovery of information in databases. In response to such a demand, data warehousing and data mining techniques [5, 7, 47] are emerging to extract the previously unknown and potentially useful knowledge, to provide better decision support, and to increase business opportunities. The overall knowledge discovery in databases *(KDD)* process includes the preprocessing and postprocessing of data as well as interpretation of the discovered patterns as knowledge, while the data mining methods and algorithms aim solely at extracting patterns from raw data [16]. As pointed out by [29], there is a need and an opportunity for an at least partially-.automated form of *KDD*, or data mining to handle the huge size of real-world database systems.

Data warehousing employs database technologies for storing and maintaining data. A data warehouse is subject-oriented and contains integrated data [18]. A data warehouse does not create value by itself; value comes from the data in the warehouse [46]. Different data warehousing views can be generated for different applications. In data warehouse views, the data values are aggregated from a collection of data from the underlying operational databases. In data warehousing, the summarized and consolidated information from data is more important than the detailed individual data, and may not be available directly from the data. For this purpose, data mining techniques are used. Data mining is a process for extracting nontrivial, implicit, previously unknown and potentially useful information from data. Many other terms such as knowledge discovery, *knowledge* mining, knowledge extraction, data archaeology, data dredging, and data analysis carry a similar or slightly different meaning in the existing articles and documents [7].

Three of the most common methods in data mining are association rules [42, 43], data classification [6, 25] and data clustering [15, 50]. Association rules discover the co-occurrence associations among data. Data classification is the process that classifies a set of data into different classes according to some common properties and classification models. Finally, data clustering groups physical or abstract objects into disjoint sets that are similar in some respect. In traditional databases, data clustering places related or similar valued records or objects in the same page on disks for performance reasons. A good clustering method ensures that the intra-cluster similarity is high and the inter-cluster similarity is low. However, since the data in each multimedia database is structural in nature and the workloads are query intensive with mostly complex queries that tend to access information across multiple multimedia databases, we consider conceptual database clustering rather than data clustering in this study. Similar to data clustering, database clustering is to group related multimedia databases in the same cluster (data warehouse) such that the intra-cluster similarity is high and the inter-cluster similarity is low. Here, two multimedia databases are related in the sense that they either are accessed together frequently or have similar objects.

Though data warehousing and data mining are two separate topics and practices, they are closely related in applications [3]. For example, the growth of data warehousing in organizations is one important connection between data :mining and data warehousing. The raw material (raw data) in the data warehouses is the main source for data mining, i.e., to prepare clean and well-documented databases for the data warehouses. Data warehousing and data mining technologies have been beneficial to many industries such as the manufacturing, retail, transportation, healthcare, and telecommunications since they offer the capability to sort, analyze, manipulate, and query data. For example, by utilizing the data warehousing and data mining techniques in the *market-basket* analysis in a retail store, the retail store learns what items its customers frequently purchase together and places these items in market baskets. The goal is to understand the behavior of typical customers as they navigate the aisles of the store. Suppose that many customers will walk from one to the other of those items which are purchased together frequently. Then the owner of the store can put some high-profit items tempting such customers between those items to induce more impulse buying to increase profits. Finding patterns in the *biological data* from the fields of DNA and protein analysis improves the healthcare industry. Furthermore, several emerging applications in information providing services, such as on-line services and the World Wide Web, also call for various data warehousing/data mining techniques to better understand user behavior, to ameliorate the service provided, and to increase the business opportunities. To make use of the huge amount of data which is unexplorable under current techniques, data warehousing and data mining are the techniques needed.

The cost of query processing is pretty high when accessing these (databases. However, if the related databases are conceptually grouped together, the cost of query processing can be expected to be reduced since these databases usually belong to a certain application domain and are required consecutively on some query access path. Hence, the need to perform database clustering by discovering the summarized knowledge in the databases to accelerate query processing has become inevitable. This motivates us to cluster the network of databases into beneficial data warehouses based on the access behavior of the application queries. For those users who wish to access only parts of the databases, they can access the data from the appropriate clusters without going through the whole network of databases. In other words, the set of data warehouses provides a flexible means of sharing information among all the databases since it allows users to incrementally and dynamically access the information they want without being overwhelmed with all of the unstructured information.

1.2 Synthetic Outline of My Work

Unlike traditional text-based database systems, information in multimedia systems is highly volatile and semantically rich. Data access and rnanipulation for multimedia database systems are more complicated than those of the conventional database systems since it is necessary to incorporate diverse media with diverse characteristics. Therefore, a good MDBMS *(multimedia database management system)* is a necessity for a network of databases. Recent papers related to multimedia database systems can be categorized in the following application domains: speech recognition, word recognition, signal processing, handwriting recognition, and document/passage retrieval [2, 22, 26, 28, 32]. However, the focus of the above research is on the low-level feature recognition of multimedia data; while our approach addresses the need for a mechanism at the database management point of view. In addition, query processing in multimedia database systems, in general, involves not only the closely interrelated *communication cost* and *processing cost* but also the capability to handle the rich semantic data.

Toward this end, we have proposed and developed a probabilistic network based mechanism to support the dynamic discovery of knowledge units within a network of databases. The mechanism is based upon a core set of database constructs that characterize object database systems, along with a set of queries with the probabilistic descriptions of database access patterns. The mechanism provides a uniform framework for organizing, mining, managing, and utilizing database information. units within an environment of multiple, autonomous, interconnected multimedia 'databases. The network of databases is modeled as a probabilistic network with the affinity relations of the multimedia databases embedded in some probabilistic models. The proposed *Markov model mediator (MMM)* mechanism serves as the probabilistic model for each node in the probabilistic network. MMMs adopt both the *Markov Model* framework and the *mediator* concept. A Markov model is a well-researched mathematical construct which consists of a number of states connected by transitions:. [48, 49] define a mediator to be a program that collects information from one or more sources, processes and combines it, and exports the resulting information. In other words, a mediator is a device regarded as a framework for performing integration over multiple databases and expresses how the integration is to be achieved.

With the help of probabilistic networks, methods can be developed to discover useful information and knowledge for the multimedia database systems via probabilistic reasoning. The proposed MMM mechanism allows us to query different media types and manage the rich semantic multimedia data. In an MMM, each node represents a *media object* since the primitive constructed or manipulated entities in most multimedia systems are media objects. A media object could be a video clip, an image, a text file, or a complex entity of these simpler entities [4]. Each media object in an MMM is associated with an *augmented transition network (ATN)* which is a model for multimedia presentations, multimedia database searching, and multimedia browsing [8, 9, 10].

In addition, the proposed probabilistic network-based mechanism facilitates the functionality of an MDBMS by three steps. First, a stochastic process performs probabilistic reasoning to derive sets of probability distributions from a set of historical data and build a probabilistic network. The set of historicall data is used as the training traces for finding the probability distributions. Second, a federation of data warehouses is constructed based on the mined probability distributions [37, 39]. Third, a second stochastic process generates a list of possible state sequences with respect to a given query and indicates which particular media objects to query over the constructed data warehouses [38, 39]. After the second stochastic process, multimedia database searching becomes handy. When the required media objects are predicted, the corresponding ATNs are traversed for information retrieval. Moreover, since there might be multiple data warehouses constructed in the first stochastic process, if an MMM cannot provide all the information for a query, then the second stochastic process is applied to other MMMs until all the information for the query is found.

1.2.1 Markov Models

A Markov model consists of a number of states connected by transitions. The states represent the alternatives of the stochastic process and the transitions contain probabilistic and other data used to determine which state should be selected next. All transitions $S_i \rightarrow S_j$ such that $Pr(S_j$ | S;) > 0 are said to be allowed, the rest are prohibited. A discrete-parameter Markov process or Markov sequence is characterized by the fact that each member of the sequence is conditioned by the value of the previous member of the sequence. A Markov Chain is a (dynamicsystem, evolving in time. Since the current member, x_{k+1}, is conditionally independent of x_0, x_1, ..., x_{k-1} given x_k, the branch probabilities are independent of the time index k. Therefore, the Markov Chain is said to be homogeneous. The stochastic behavior of a homogeneous chain is determined completely by the probability distribution for the initial state and the one-step transition probabilities.

Many applications use Markov model as a framework such as Hidden Markov Models (HMMs) which are based on modeling patterns as a sequence of observation vectors derived from a probabilistic function of a non-deterministic first-order Markov process in speech recognition [31], and Markov Random Field Models permit the introduction of spatial context into pixel labeling problems and lead to algorithms for generating textured images, classifying textures, and segmenting textured images [33, 12, 17]. However, no existing research uses Markov models as a framework in designing a database management system.

1.2.2 Terminology

The definitions of the terminology used in this report are introduced below.

- **media object: A** media object is the primitive constructed or manipulated entity in most multimedia systems and it could be a video clip, an image, a text file, or a complex entity of these simpler entities. A media object is represented as a node in an MMM and has a set of attributes/features whose descriptions are available in d;. A media object is denoted by $C_{i,j}$, where the index 'i'indicates the database identification and *'j'* represents the media object identification within d_i.

- **attribute/feature:** A class of attributes or features, $O_{i,j,k}$ where *'k'* denotes the attribute/feature identification, associated to a media object $C_{i,j}$ are to characterize $C_{i,j}$ and to represent the information pertaining to the d_i available to the application queries. The values of i, *j*, ***k*** are unique. Each media object has its own set of attributes/features.

- **relationship links** Two types of relationship links are captured in the MMM mechanism – equivalence and interaction. In a single database, media object equivalence cannot exist in two different media objects since a database schema represents a non-redundant view. As such, only media objects across different databases can have an equivalence relationship.

1. **equivalence links:** The notion of real world states *(RWS's)* [20, 27], i.e., the scope of the real world the two media objects are designed to reflect in the database, is used to compare two media objects. Two media objects are deemed to possess the same real world states if they represent the same sets of instances of the same real world entity. Two media objects $C_{i,j}$ and $C_{m,n}$ are said to be equivalent if $\mathrm{RWS}(C_{i,j})=\mathrm{RWS}(C_{m,n})$. Equivalence links are used to connect the two equivalent media objects across different databases together.

2. **interaction links:** Interaction links are to represent relationships other than the equivalence relationships. All the relationship links for the media objects in a database are considered as the interaction links.

1.2.3 Overview of the Probabilistic Network-Based Mechanism

Our proposed system specifically considers the interrelation and sharing of information units managed by a large scaled network consisting of heterogeneous databases. In other words, some of the databases are relational, some are object-oriented, some are hierarchical, and some are multimedia. However, since a multimedia information system stores and manages not only images, audio, graphics, animation, and full-motion video, but also text as in traditional text-based databases, the network is represented by a set of multimedia information systems. Figure 1.1 illustrates the overview of the system architecture. This system contains six main parts which are the multimedia resource subsystem, the knowledge discovery subsystem, the probabilistic network subsystem, the data warehouse subsystem, the query *processing* subsystem, and the information retrieval subsystem [37, 38, 39]. This report focuses on how the federation of data warehouses is constructed and thus covers only the knowledge discovery subsystem, the probabilistic network subsystem, and the data warehouse subsystem. The complete discussion of the mechanism is in [40].

- the multimedia resource subsystem

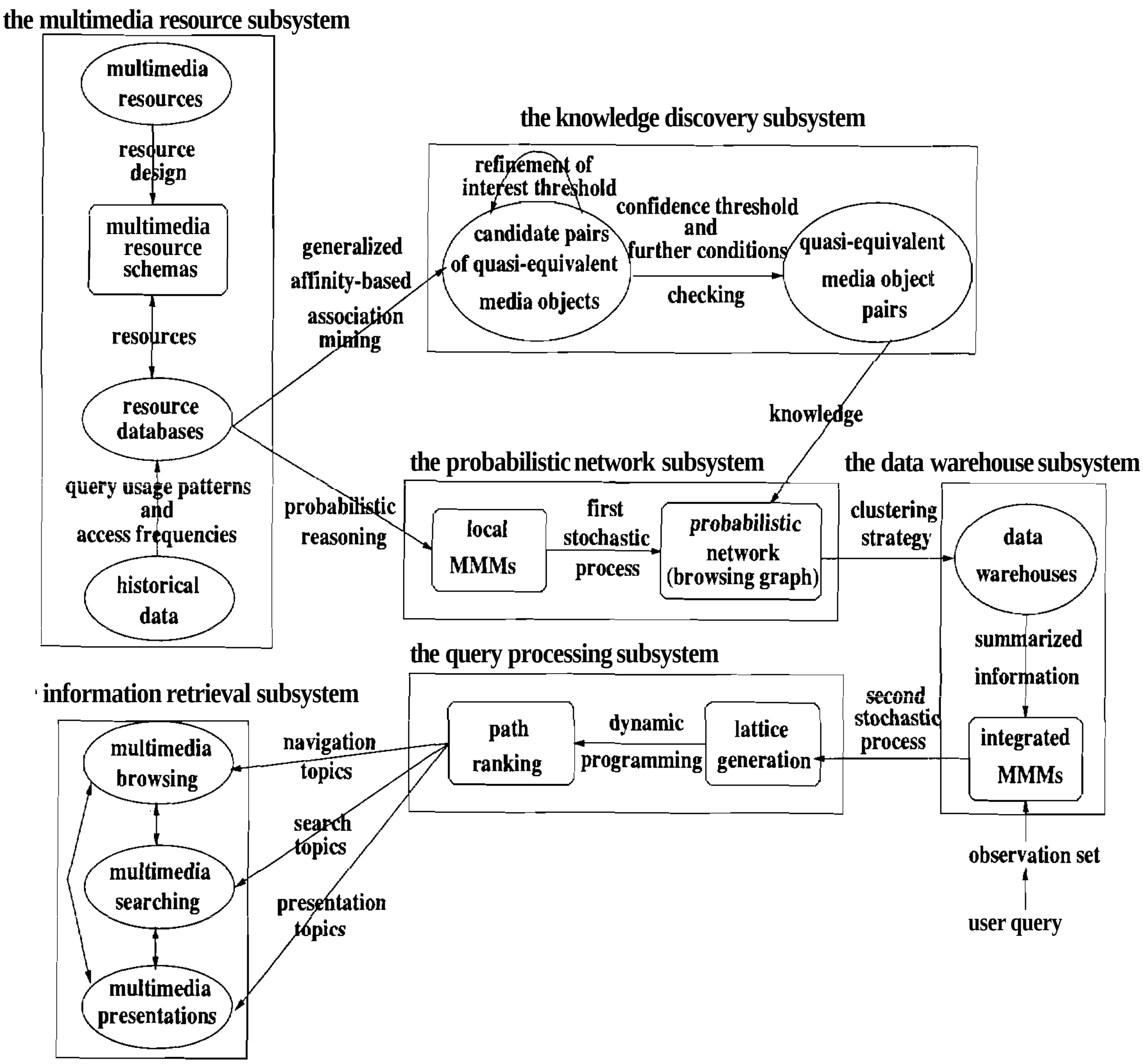
the multimedia resource subsystem
multimedia resources
resource design
multimedia resource schemas
resources
resource databases
query usage patterns and access frequencies
historical data
the knowledge discovery subsystem
refinement of interest threshold
candidate pairs of quasi-equivalent media objects
confidence threshold and further conditions
checking
quasi-equivalent media object pairs
generalized affinity-based association mining
knowledge
the probabilistic network subsystem
probabilistic reasoning
local MMMs
first stochastic process
probabilistic network (browsing graph)
the data warehouse subsystem
clustering strategy
data warehouses
summarized information
integrated MMMs
observation set
user query
the query processing subsystem
path ranking
dynamic programming
lattice generation
second stochastic process
information retrieval subsystem
navigation topics
search topics
presentation topics
multimedia browsing
multimedia searching
multimedia presentations

Fig. 1.1. **The system architecture of the proposed probabilistic network-based mechanism.** The system architecture is composed of six main subsystems which are the *multimedia resource subsystem*, the *knowledge discovery subsystem*, the *probabilistic network subsystem*, the *data warehouse subsystem*, the *query processing subsystem*, and the *information retrieval subsystem*. Each subsystem is enclosed by a box and consists of two or more modules. The control flow is represented by the arrows.

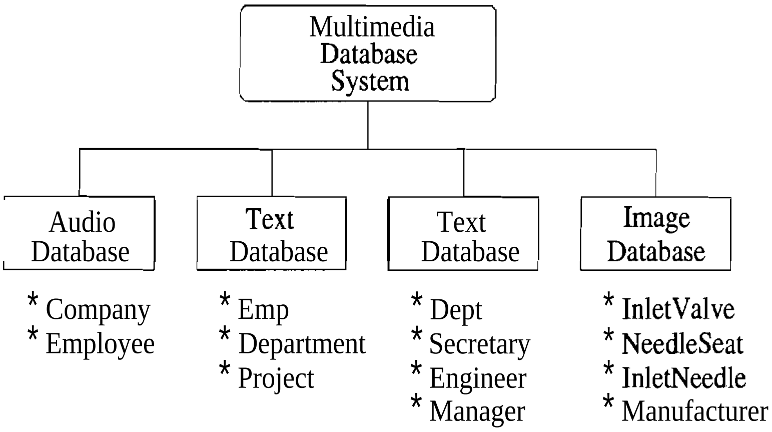

Fig. 1.2. An example multimedia resource subsystem. The *multimedia resource subsystem* has four resource databases – one of them is an audio database, one of them is an image database, and the other two are text databases. Each database has its own defining properties represented by its media objects.

The *multimedia resource subsystem* consists of four modules – multimedia resources, multimedia resource schemas, resource databases, and a set of historical data. Each multimedia resource is associated with a designated resource schema which defines the set of media object definitions with their attributes. The resource database is a set of persistent objects which are instances of the media objects defined in the schema. The set of historical data includes the usage patterns of the media objects with respect to the set of sample queries and the access frequencies of the sample queries. The multimedia resource databases together with the set of historical data provide the prior information. The information in the resource databases and the historical data are the inputs to the *knowledge discovery subsystem* and the *probabilistic network subsystem.*

Throughout this report, an example multimedia database system is used to illustrate how the proposed mechanism works. As shown in Figure 1.2, there are four multimedia resource databases in the *multimedia resource subsystem.* One is an audio database, one is an image database, and two are text databases. The text databases are the traditional databases which can be relational, object-oriented, or hierarchical. Since things in the world around us have a defined physical meaning, each multimedia database has a set of media objects and each media object has its own set of *attributes/features*. Therefore, we think of a database in connection with its media objects and attributeslfeatures. For example, a table and its attributes in a relational database can be mapped to a media object and its attributeslfeatures. As for the audio and image databases, besides the audio/image data, the databases contain several properties that are useful in the context of archiving numerous audio/image entries [24]. For example, in the audio database, the media object Company contains a name attribute, a profile attribute, a president attribute, and a set of *emp_name* attributes. These attributes allow the user applications to query the audio database for certain desired information. The following lists the media objects and part of the attributes/features in the example multimedia resource subsystem.

d_1 = {Company, Employee) $= \{C_{1,1}, C_{1,2}\}$;

$C_{1,1} \Leftrightarrow \{name, profile, president, emp\_name\} = \{O_{1,1,1}, O_{1,1,2}, O_{1,1,3}, O_{1,1,4}\}$;

d_2 = {Emp, Department, *Project*}$=\{C_{2,1}, C_{2,2}, C_{2,3}\}$;

$d_3 = \{Dept$, Secretary, Engineer, Manager) $=\{C_{3,1}, C_{3,2}, C_{3,3}, C_{3,4}\}$;

d_4 = (Inlet Valve, *NeedleSeat*, *InletNeedle*, Manufacturer) $=\{C_{4,1}, C_{4,2}, C_{4,3}, C_{4,4}\}$.

- the knowledge discovery subsystem

The generalized *affinity-based* association mining process supports the knowledge discovery subsystem. The association mining process consists of two modules (phases). Phase I checks a set of constraints: **(1)** minimum interest threshold, (2) interest constraint, and **(3)** refinement constraint iteratively to construct a candidate set. In Phase II, a minimum confidence threshold is first used to cut down the size of the candidate set. Then, some further conditions can be imposed to get rid of any unreasonable pairs in the candidate set. After the two phases, the discovered knowledge (i.e., the pairs of quasi-equivalent media objects) is then input into the probabilistic network subsystem to allow the construction of a probabilistic network for all the multimedia resource systems. We use the quasi-equivalent relationship to approximate the structurally equivalent relationship.

- the *probabilistic network subsystem*

  There are two modules in the *probabilistic network subsystem*.. The first module takes the information from the *multimedia resource subsystem* and constructs local MMMs via data mining and probabilistic reasoning. Each resource database is modeled by a local MMM whose three probability distributions – the state transition probability distribution, the observation symbol probability distribution, and the initial state distribution – are mined in the probabilistic reasoning process. After all the local MMMs are constructed, a stochastic process which uses the probability distributions of the local MMMs is applied to measure how similar two local MMMs (resource databases) are. The similarity measures from the first module together with the knowledge (i.e., the quasi-equivalence relationships of the media objects) discovered in the *knowledge discovery subsystem* are used in the second module to construct the probabilistic network. The discovered quasi-equivalent media object pairs determine the topology of the probabilistic network (or the browsing graph).

  Each resource database is associated with a node in the probabilistic network. Two nodes are connected in the probabilistic network only if these two resource databases have quasi-equivalent media objects. Without loss of generality, two nodes are connected by two opposite directed arcs since the structural equivalence relationship is bi-directional. The probability distributions of the local MMMs together with the topology of the probabilistic network are then the inputs to the *data warehouse subsystem* to construct a federation of data warehouses.

- the *data warehouse subsystem*

  The *data warehouse subsystem* consists of two modules. The first module takes the inputs sent by the *probabilistic network subsystem* and generates a federation of data warehouses via invoking a traversal-based greedy database clustering strategy. After the formations of the data warehouses, an integrated MMM for each data warehouse is constructed in the second module.

During the summarized information step, the data warehouse module formulates the three probabilistic distributions for each integrated MMM and discovers summarized/useful knowledge within each data warehouse. Then, all the information mined in the summarized information step is forwarded to the integrated MMM module. In addition to the information forwarded from the first module, the integrated MMM module takes another input – users' queries. The integrated MMM module accepts users' queries, extracts the required attributes/features from them, and submits the attributes/features to the *query processing subsystem*. In other words, the integrated MMMs are the basic units for processing user queries.

- the *query processing subsystem*

  The *query processing subsystem* consists of two modules: the lattice generation module and the path ranking module. Based on the set of the attributes/features extracted from a query and submitted by the integrated MMM module, a stochastic process is applied to generate a lattice (or trellis) structure for the given query in the lattice generation module. This stochastic process performs a proposed dynamic programming approach to compute the edge weights and the cumulative edge weights of the edges on the lattice. The generated lattice yields a list of possible state sequences (media object sequences) to avoid exhaustive search. Searching databases one by one for a query is very time-consuming and expensive especially under a multimedia inforrnation providing environment.

  The complexity of a query depends heavily on the order in which the databases are searched for a successful path. Therefore, if the possible paths can be identified to locate information efficiently, then query processing can be speeded up. As a result, the proposed dynamic programming algorithm is applied to rank the list of possible state sequences yielded by the first module in the path ranking module. Thus, the path ranking module selects the potentially relevant media objects for the query. From the experience, there will be only a limited number of possible paths sent by the first module which reduces the complexity of the path ranking procedure in the second module.

- the *information retrieval subsystem*

  The *information retrieval subsystem* takes the ranked paths identified by the *query processing subsystem* to retrieve the required information for the query. There are three modules in this subsystem – the multimedia browsing module, the multimedia searching module, and the multimedia presentation module. As mentioned previously, information retrieval in the proposed MMM mechanism lies on the ATNs associated with the media objects in the integrated MMMs. Once the required media objects for a query are identified along a path in the *query processing subsystem,* information retrieval simply traverses the ATNs of those identified media objects.

  These three modules are interleaving in the sense that a user might at the same time need to browse through all the possible paths, to traverse the ATNs and/or subnetworks of the media objects along each path, and to watch multimedia presentations to get the information he or she wants for a query. To retrieve information for a query, several steps need to be executed. First, the query is translated into a multimedia input string. Next, a proposed substring matching algorithm is invoked which completely matches or partially matches the multimedia input string of the query to the multimedia input strings of the ATNs or their subnetworks of the media objects along a path. If the query contains topics related to video data, then the multimedia presentation module is invoked. Finally, since all the possible paths are ranked, the top ranked path is first given a try. If the top ranked path cannot provide the information required for the query, then the next ranked path is considered. This is repeated until the information required for the query can be obtained.

1.3 The Markov Model Mediator (MMM)

The essence of a distributed information-providing environment is a large number of databases which are navigated by queries. Many queries in such a distributed information-providing environment require not only the detailed and individual records but also the summary and consolidated information in the databases. In addition, the cost of query processing is very expensive especially in such a large-scaled environment because the workloads are query intensive with mostly complex queries that tend to access millions of records from a set of databases.

As a result, the network of databases is modeled as a probabilistic network with the affinity relations of the databases embedded in some probabilistic models. For this purpose, we have proposed a unified model called *Markov Model Mediators (MMMs)* to serve as the probabilistic models for the probabilistic network. There are two stochastic processes related to MMMs. The first stochastic process performs probabilistic reasoning and discovers the summarized knowledge to construct a federation of data warehouses via the use of a set of historical data, i.e., a probabilistic description of the expected access patterns of the media objects. Since MMMs possess the stochastic property of Markov Models, the conceptual database clustering is based on complex statistical and probabilistic analyses which are best understood by examining the network-like structure in which those statistics are stored. The second stochastic process generates a list of possible state sequences with respect to a query and indicates which particular media objects to query over the constructed data warehouses.

1.31 Components of an MMM

There are two types of MMMs - local MMMs and integrated MMMs. Each multimedia database is modeled as a local MMM and each data warehouse is modeled as an integrated MMM. An MMM (local or integrated) is represented by a 6-tuple $\lambda = (S,\mathcal{F}, A,\mathcal{B},\Pi, \Psi)$ where S is a set of media objects called states; $\mathcal{F}$ is a set of attributes/features; $\mathcal{A}$ is the state transition probability distribution; $\mathcal{B}$ is the observation symbol probability distribution; Π is the initial state distribution; and Ψ is a set of augmented transition networks (ATNs).

An MMM consists of a sequence of states which represent the meclia objects (in S) in the multimedia databases. Since different media objects may have different types of attributes or features, each media object has its own set of attributes/features (in 3) . The states are connected by directed arcs (transitions) which contain probabilistic and other data used to determine which state should be selected next. All transitions $S_i \rightarrow S_j$ such that $Pr(S_j \mid S_i) > 0$ are said to be allowed, the rest are prohibited. $Pr(S_j$ | S;) is greater than 0 when the media objects S_i and S_j have been accessed together by the queries or have a structural equivalence relationship. Essentially, an MMM is a stochastic finite state machine with a stochastic output process attached to each state to describe the probability of occurrence of the output symbols (states). $\mathcal{A}, \mathcal{B}$, and Π are the probability distributions for an MMM and play as the major roles in the stochastic processes. The elements in S and 3 determine the dimensions of A and $\mathcal{B}$. The formulations of A,$\mathcal{B}$, and Π for an MMM and the construction of the data warehouses will be introduced in the next few sections. Since those local MMMs which are accessed frequently are placed in the same data, warehouse, the integrated MMMs are used in the second stochastic process to find the possible list of state sequence for a query.

The augmented transition network (ATN) is a semantic model to model multimedia presentations, multimedia database searching, and multimedia browsing. An ATN can be represented diagrammatically by a labeled directed graph and it consists of a finite set of state nodes connected by the labeled directed arcs. The arcs in an ATN represent the time flow from one state node to another. An arc represents an allowable transition from the state node at its tail to the state node at its head, and the labeled arc represents the transition function. An input string is accepted by an ATN if there is a path of transitions which corresponds to the sequence of symbols in the string and which leads from a specified initial state to one of a set of specified final states. Subnetworks are developed to allow the users to choose the scenarios relative to the spatio-temporal relations of the video or image contents or to specify the criteria based on a keyword or a combination of keywords in the queries in an ATN. Information in text databases can be accessed by keyword: via the text subnetworks. For example, if a text subnetwork contains the keyword "Purdue University Library," then the Purdue University library database is linked via a query with this keyword. The most significant advantage of subnetworks is that a subnetwork can represent an existing media object. In other words, a subnetwork can be shared by multiple multimedia databases, and thus any change in a subnetwork will autoinatically change the contents of the multimedia databases that include the subnetwork. The inputs for ATNs are modeled by multimedia input strings. Also, each subnetwork has its own multimedia input string. Database searching in ATNs is performed via substring matching between the multimedia input string(s) of the ATN (and its subnetworks) and the multimedia input string of a given query. Therefore, each rnedia object has an associated ATN. When the required media objects are predicted: the corresponding ATNs are traversed for information retrieval. For the details of ATNs, please see [8, 9, 10].

1.4 Contributions

Here we briefly state the contributions of this report. A probabilistic network-based mechanism is proposed which incorporates the probabilistic reasoning and data mining techniques into the multimedia database management system (MDBMS).The proposed Markov model mediators (MMMs) serve as the probabilistic model for the nodes in the probabilistic network. No existing research has used a probabilistic network-based approach in MDBMS. Probabilistic reasoning is powerful in a complex probabilistic network with a large number of states. Probabilistic reasoning can be effectively applied in databases to conceptually cluster the databases into a federation of data warehouses. We are not claiming that the clustering method is the best. On the other hand, several empirical studies have been conducted showing that the MMM mechanism performs better than some other chosen clustering methods under the selected database management systems from Purdue University.

The proposed probabilistic network-based mechanism is designed in particular to provide the following four functional capabilities of organizing, mining, managing, and utilizing a structured information space under the designed six subsystems in three steps to facilitate the functionality of an MDBMS.

1. Structural organization concerns how the heterogeneous databases are structured into a probabilistic network.

   - attempts to organize the media objects in a database into the local MMM construct;
   - each database is modeled by a local MMM;
   - attempts to organize the local MMMs into the probabilistic network;
   - the structural organization has a substantial effect on the amount of effort involved in the identification of appropriate information for a user query.

2. Information mining concerns how some useful data patterns are extracted from the data.

   - in order to construct the local and integrated MMMs, a probabilistic reasoning approach that draws upon data mining techniques is employed;
   - in order to determine the topology of the probabilistic network, a generalized affinity-based association mining approach that draws upon data mining techniques is employed;
   - the discovered knowledge has a substantial effect on the construction of the probabilistic network.

3. Database management concerns how the network of databases is partitioned into a federation of beneficial data warehouses.

- in the probabilistic network, databases with structurally quasi-equivalent media objects are interconnected and can be clustered into a collection called a data warehouse;
- each data warehouse is modeled by an integrated MMM;
- the network of databases, therefore, consists of a federation of data warehouses;
- the construction of a federation of data warehouses has a substantial effect on reducing the cost of query processing.

4. Information utilization concerns how relevant information can be identified and retrieved from the resource repositories with respect to a user query.

    - once the federation of data warehouses is constructed, the user can request information by issuing a query;
    - the integrated MMMs are the units for query processing;
    - a dynamic programming algorithm is proposed for identifying possible media objects for a query;
    - each media object has an ATN associated with it;
    - ATNs and the multimedia input strings model multimedia searching, multimedia presentations, and multimedia browsing;
    - the identification of required media objects has a substantial effect on effective and efficient database searching.

The following subsections outline how this report approaches these issues.

1.4.1 The Probabilistic Network: Organizing

The structural organization concerns how the heterogeneous databases are structured into a probabilistic network. The network of databases is modeled as a probabilistic network with affinity relations of the databases embedded in the proposed MMM mechanism. The MMM construct is a mathematically sound framework to model the database access patterns as a sequence of observation vectors derived from the queries issued to the databases. A core set of database constructs that characterize the databases and a set of queries with the probabilistic descriptions of database access patterns are the required prior information. Since the MMMs possess the stochastic property of Markov models, it is used as the probabilistic model embedded in a probabilistic network to manage and utilize the information for a network of databases. An MMM consists of a number of states (media objects) connected by transitions and serves as the probabilistic model for each node (database) in the probabilistic network. Moreover, data mining and probabilistic reasoning techniques are incorporated in the construction of the probabilistic network.

As shown in Figure 1.3, there are four local MMMs in correspondence to the four multimedia resource databases – one audio, one image, and two text databases in Figure 1.2. Each database has its own defining properties represented by a set of media objects, and each media object has a set of attributes/features. Each database is modeled by a local MMM construct where its media objects are organized into the states of the local MMM. A set of historical data regarding the access frequencies and patterns for the queries is used to furthermore organize the local MMMs into a probabilistic network. The structural organization of the probabilistic network has a substantial effect toward conducting the two proposed stochastic processes.

1.4.2 Probabilistic Reasoning and Data Mining: Mining

Information mining concerns how some useful data patterns are extracted from the data. Two data mining approaches are proposed to discover the data patterns from the prior information. First, the probability distributions of the MMMs (local or integrated) are mined through a proposed probabilistic reasoning approach. The probabilistic reasoning process starts with the probabilistic descriptions of the queries and the semantic structures of the databases. Then the probability distributions for local and integrated MMMs are formulated [37].

Second, a generalized affinity-based association mining algorithm is developed to discover the set of quasi-equivalent media objects for the network of databases. An

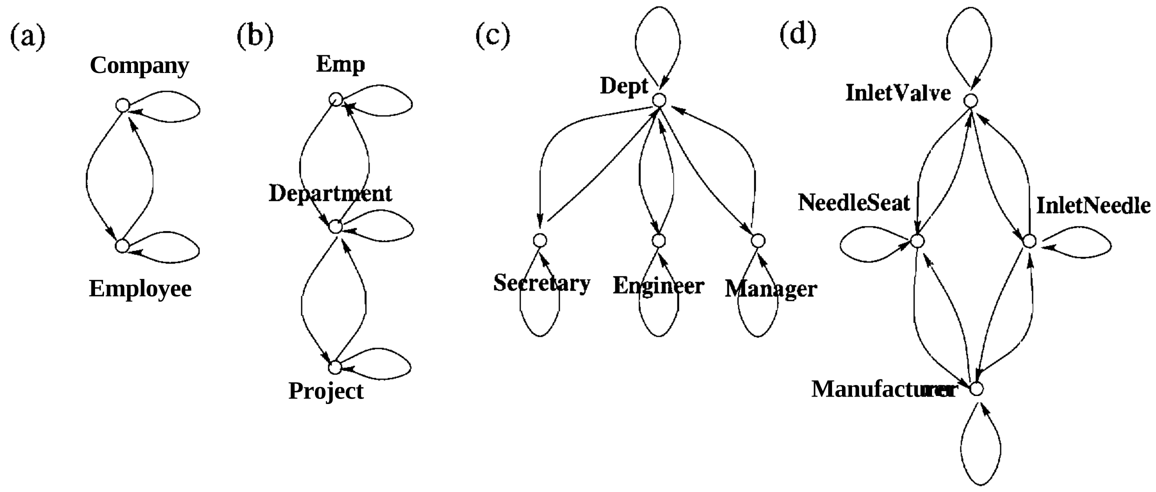

Fig. 1.3. The constructed local MMMs for the four databases in Figure 1.2. Each database is modeled by a local MMM construct and the set of its media objects is organized into the states of the local MMM.

empirical study of the real databases at Purdue University is conducted to illustrate how the generalized affinity-based association mining algorithm discovers the set of quasi-equivalent media objects. Experimental results show that the proposed algorithm discovers the knowledge accurately.

In addition, a stochastic process based on the discovered knowledge, i.e., the probability distributions and the set of quasi-equivalent media objects, is proposed to construct the probabilistic network for the network of databases [39]. Hence, the discovered knowledge has a substantial effect on the construction of the probabilistic network since it is part of the inputs for the *probabilistic network subsystem.*

1.4.3 Data Warehousing: Managing

Database management concerns how the network of databases is partitioned into a federation of beneficial data warehouses. The topology and the branch probabilities of the probabilistic network are used to construct the federation of data warehouses for the network of databases. Two databases are interconnected in the probabilistic network only if they have quasi-equivalent media objects. Therefore, two databases can be clustered into the same data warehouse only if they are interconnected in the probabilistic network. A database clustering strategy based on the topology and the stationary probabilities which are transformed from the branch probabilities of the probabilistic network is proposed. The constructed federation of the data warehouses has a substantial effect on reducing the cost of query processing since the databases in a data warehouse usually belong to a certain application domain and are required consecutively on some query access path. Therefore, the cost of query processing is expected to be reduced.

In addition, an integrated MMM is constructed for each data warehouse and the integrated MMMs are the basic units for information retrieval and database searching. For those users who wish to access only parts of the databases, they can incrementally and dynamically access the information they want without being overwhelmed with all of the unstructured information. In other words, the set of data warehouses provides a flexible means of sharing information. Furthermore, since the integrated MMMs are the units for information retrieval and database searching which are conducted by a proposed stochastic process, the probability distributions for the integrated MMMs need to be constructed.

1.4.4 Querying and Information Retrieval: Utilizing

Information utilization concerns how relevant information can be identified and retrieved from the resource repositories with respect to a user query. Once the federation of data warehouses is constructed and the probability distributions of the integrated MMMs are obtained, user queries can be issued on the integrated MMMs. An integrated NIMM is the basic unit for querying and information, retrieval. However, in some cases, a query may need to access one or more integrated MMMs to get the required information. If a particular integrated MMM cannot find all the information for a query, then another integrated MMM is used.

For each user query, a lattice (or trellis) is generated by a proposed stochastic process. This stochastic process includes a proposed dynamic prograrnming algorithm which computes the edge weights and the cumulative edge weights for the edges on the lattice. A list of possible paths will be identified and ranked based on the edge weights and the cumulative edge weights [38, 39]. The dynamic programming algorithm can effectively and efficiently identify the required media objects with respect to a given query. The identification of the required media objects has a substantial effect on information retrieval and database searching since the prediction of *a* subset of media objects for a query minimizes the effort for searching over the whole information space of the databases.

Moreover, since each media object has an associated ATN with it, not only the text information but also the images, video, etc. can be managed and retrieved. A subnetwork is constructed for each image, video frame, or text. Once the limited subset of media objects is identified, database searching becomes handy. Simply traverse the ATNs of those media objects to find the requested information for the query. If a media object contains images, video frames, or texts, then its subnetworks are traversed. The input for an ATN or a subnetwork is a multimedia input string, and each user query is also translated into a multimedia input string. Thus, multimedia database searching becomes the problem of substring matching between the multimedia input string of the query and the multimedia input strings for the ATNs and/or their subnetworks. Furthermore, multimedia relationships such as the temporal and spatial relationships are supported in ATNs. Functions such as multimedia presentations, multimedia searching, and multimedia browsing are also supported [8, 9, 10].

1.5 Organization of the Report

The rest of this report is organized as follows. Chapter 2 illustrates the *knowledge discovery subsystem* by developing the generalized affinity-based association mining process to discover the set of quasi-equivalent media objects. Chapter 3 shows how the data mining and probabilistic reasoning are used to construct the probabilistic network for the *probabilistic network subsystem.* A federation of data warehouses is constructed via a proposed stochastic database clustering strategy for the *data warehouse subsystem* is presented in Chapter 4. In Chapter 5, the *query processing subsystem* and the *information retrieval subsystem* are discussed. Finally, conclusions are in Chapter 6.

2. KNOWLEDGE DISCOVERY IN DATABASES

2.1 Introduction

In the last decade, the exponential growth of computer networks and data-collection technology, such as bar-code scanners in business domains and sensors in scientific and industrial domains, has generated an incredibly large offering of products and services for the users of computer networks. In business, data captures information such as sales opportunities and quality/cost control to improve corporate profitability. In science, data represents study observations and phenomena. In manufacturing, data helps to identify performance and optimization opportunities and to improve troubleshooting processes. With the explosive growth in the amount arid complexity of data, advanced data storage technology and database management systems have increased our capabilities to collect and store data of all kinds. For instance, enterprises increasingly store the huge amounts of data in data warehouses for decision-support purposes. The growth of data warehousing in organizations has led to the need for data mining: clean and well-documented databases. However, our ability to interpret and analyze the data is still limited, creating an urgent need to accelerate discovery of information in databases. This need has been recognized by researchers in different areas such as database management systems [11, 14], data warehousing [18, 30], machine learning and artificial intelligence [19, 36], statistics [13], and data visualization [21, 41].

Data in row format has little direct benefit. What is of value is the knowledge that can be inferred from the data and put to use. In other words, its true value depends on the ability to extract useful information for decision support or on the exploration and understanding of the data in the databases. Traditional data analysis methods often depend on humans to deal with the data directly. However, as the volume of data increases, it is not realistic to expect human experts to analyze all the data since manual data analysis simply cannot scale to handle it. In addition, knowledge acquisition from experts may be biased and need to be validated with broader tests. Discovering knowledge from data, or data mining, can help to overcome the limitations. In [16], the authors define *knowledge discovery in database (KDD)* to be the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data and *data mining* to be the application of algorithms for extracting patterns from data. In other words, data mining is a component in the KDD process concerned with the means by which patterns are extracted and enumerated from the data. However, in most of the existing articles and documents, *KDD* and data mining carry a similar meaning. Therefore, we USE! KDD and data mining interchangeably in this report.

Data mining is a process to extract nontrivial, implicit, previously unknown and potentially userful information from data in databases. One of the primary goals of data mining is to find human interpretable patterns (clusters, rules, trees, time series patterns) describing the data in a database. The discovered patterns can be valuable information for decision making and be used to form a classificatiori model, identify trends and associations, refine an existing model, or provide summary knowledge of a database. By knowledge discovery in databases, interesting knowledge, regularities, or high-level information can be extracted from the relevant sets of data in databases and be investigated from different angles, and large databases thereby serve as rich and reliable sources for knowledge generation and verification. However; nothing is more likely to undermine the performance and efficiency of a business than inappropriate, misunderstood, or ignored summarized information obtained from the data. The reason is that the potential business contribution of corporate activities depends on the quality of decisions and in turn on the quality of the data used to make them.

2.2 Discovery of Association Rules

Association rule mining has attracted strong attention and proven to be a highly successful technique for extracting useful information from very large databases.

2.2.1 Association Rules

[1] defines an association rule to be an expression $X \rightarrow$ Y, where X and Y are sets of items and X ∩ Y = ∅. The rule implies that the transactions of the database which contain X tend to contain Y. Association rules are a simple and natural class of database regularities. Intuitively, associated items appear together frequently. Discovering associations in a database will uncover the affinities among the collection of data in the database. These affinities between data, are represented by association rules.

There are three measures of the association – support, *confidence* and interest. The support factor indicates the relative occurrence of both X and Y within the overall data set of transactions and is defined as the ratio of the number of tuples satisfying both X and Y over the total number of tuples. The confidence factor is the probability of Y given X and is defined as the ratio of the number of tuples satisfying both X and Y over the number of tuples satisfying X. In other words, the support factor indicates the frequencies of the occurring patterns in the rule and the confidence factor denotes the strength of implication of the rule [7]. Since not all the discovered association rules which pass the minimum support and minimum confidence factors are interesting enough to present, sometimes an interest factor is defined to indicate the usefulness of the rules. The interest factor is a measure of human interest in the rule. For example, a high interest means that if a transaction contains X, then it is much more likely to have Y than the other items.

Let N to be the total number of tuples and | $\boldsymbol{A}$ | to be the number of tuples containing all items in the set $\boldsymbol{A}$. Define

$$\begin{aligned} support(X) &= P(X) = \frac{\mid X \mid}{N} && (2.1) \\ support(X \rightarrow Y) &= P(X \cap Y) = \frac{\mid X \cup Y \mid}{N} && (2.2) \\ confidence(X \rightarrow Y) &= \frac{P(X \cap Y)}{P(X)} = \frac{\mid X \cup Y \mid}{\mid X \mid} && (2.3) \\ interest(X \rightarrow Y) &= \frac{P(X \cap Y)}{P(X)P(Y)}. && (2.4) \end{aligned}$$

The problem is to find all the association rules satisfying user-specified minimum support and minimum confidence constraints that hold in a given database. Rules with high support and confidence factors represent a higher degree of relevance than rules with low support and confidence factors.

2.2.2 Affinity-Based Association Rules

One of the most important problems in data mining is the discovery of association rules for large databases. We use the relative affinity values to measure how frequently two media objects have been accessed together in a set of queries [37]. Here, the set of queries is considered as the set of transactions since, similar to the case that each transaction may contain one or more items, each issued query may request information from one or more media objects from the databases. However, the current definition of support tells only the number of transactions containing an itemset but not the number of items. An item may be purchased in multiples in a transaction such that it should be considered more frequently than the support measure indicates. Similarly, each query could have a distinct frequency, i.e., a query may be activated several times. For example, though the number of outcomes that two media objects are accessed by the same queries is small, if the total access frequency of those queries accessing both of them is high, then the relative affinity between these two media objects is considered to be high. Therefore, the actual access frequency of a query per time period should be taken into account when the relative affinity between two media objects is calculated, and the calculations of support, confidence, and interest for association rules are based on the relative affinity values. Using the relative affinity measures allows more informative feedback because it tells the number of accesses of the queries but not the number of queries.

Assume a set of historical data with eight queries is used to generate the training traces that are the prior information required for the proposed mechanism. The set of historical data contains information such as the usage patterns and the access frequencies of the sample queries. Table 2.1 shows the usage patterns of media objects versus the sample queries. If a media object was accessed by a certain query, then the

Table 2.1
The usage patterns – the entity with value 1 indicates the query accessed the corresponding media object. For example, the media object $C_{1,2}$ (state 2 in d_1) has been accessed by queries q_1, q_2, q_5, and q_7.

$use_{k,m}$	**states (media objects)**												
	1	2	1	2	3	1	2	3	4	1	2	3	4
	$C_{1,1}$	$C_{1,2}$	$C_{2,1}$	$C_{2,2}$	$C_{2,3}$	$C_{3,1}$	$C_{3,2}$	$C_{3,3}$	$C_{3,4}$	$C_{4,1}$	$C_{4,2}$	$C_{4,3}$	$C_{4,4}$
q_1	1	1	1	1	0	0	0	0	0	0	1	0	1
q_2	1	1	0	0	0	1	1	1	1	0	0	1	1
q_3	0	0	1	1	1	1	1	1	1	0	0	0	0
q_4	1	0	0	1	1	1	1	0	0	1	1	0	0
q_5	1	1	1	1	0	1	0	1	0	1	0	1	1
q_6	1	0	0	0	0	1	0	0	1	0	1	0	1
q_7	0	1	0	1	1	0	1	1	0	0	0	1	1
q_8	0	0	1	1	1	0	1	1	1	0	0	1	1

corresponding entity has a value **1**. The access frequencies of the sample queries are shown in Table 2.2. For example, the access frequency of query q_2 is 100. The training traces generated by the set of historical data together with the resource databases play as ***a** priori* for the proposed mechanism.

Let Q = (1, 2, ..., q} be the set of sample queries that run on the multimedia databases $d_1, d_2, \ldots, d_p$ with media object set OC = (1, 2, ..., g) in the multimedia database system. Define the variables:

- n_i = number of media objects in database d;
- $use_{k,m}$ = usage pattern of media object m with respect to query k per time period (available from the historical data)

$$use_{k,m} = \begin{cases} 1 & \text{if media object } m \text{ is accessed by query k} \\ 0 & \text{otherwise} \end{cases}$$

Table 2.2
The access frequencies of the queries. For example, the access frequency of query q_2 is 100.

$access_k$	q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8
access frequency	25	100	30	70	45	35	40	60

- $access_k$ = access frequency of query k per time period (available from the historical data)
- $aff_{m,n}$ = relative affinity measure of media objects m and n

  $= \sum_{k=1}^{q} use_{k,m} \times use_{k,n} \times access_k$
- $support(m) = \frac{\sum_{k=1}^{q} use_{k,n} \times access_k}{\sum_{k=1}^{q} access_k}$
- $support(m \to n) = \frac{aff_{m,n}}{\sum_{k=1}^{q} access_k}$
- $confidence(m \to n) = \frac{support(m \to n)}{support(m)}$
- $interest(m \to n) = \frac{support(m \to n)}{support(m) support(n)}$

Here, *support(m)* indicates the fraction of the number of accesses of the media object m with respect to the total number of accesses for all the queries. The *support* value of the rule $(m \to n)$ shows the probability of accessing both media objects m and n with respect to all the accesses of the queries. The *confidence* value of the rule $(m \to n)$ denotes the probability of accessing media object n given that media object m has been accessed for the queries. The *interest* value of the rule $(m \to n)$ gives the measurement that if media object m is accessed by a query, then media object n is much more (or much less) likely to be accesses by the same query. For example, a high interest value of the rule $(m \to n)$ implies that media object n is much more likely to have a high affinity relationship with m than other media objects. Then, these values are used in the proposed generalized affinity-based association mining algorithm to find the set of quasi-equivalent media objects. The quasi-equivalent relationship is used to approximate the structurally equivalent relationship. Moreover, since we try to discover the quasi-equivalence relationship of two media objects, only the 2-itemsets are considered, hence reducing the overheads such as database scans and large itemset generations.

2.3 Generalized Affinity-Based Association Mining

In this section, we explore a new data mining capability that involves mining quasi-equivalent media objects in a network of databases where queries tend to access information from related or structurally equivalent media objects residing across multiple databases. For example, a given database might contain a media object $EMPLOYEE$, given attributes name, id, address, department, and salary. Another database has a media object EMP file, representing the enrollment of employees in training courses and containing attributes name and courses. Here, the media objects $EMPLOYEE$ and EMP in the two databases should represent the same *RWS's* for the organization, i.e., they are structurally equivalent. Suppose that, in order to carry out the process of training course administration, it is necessary to know the department for each enrolled employee. To answer this type of query, it is required to access information from both media objects. For this purpose, a generalized affinity-based association mining approach is proposed.

2.3.1 Architecture

Figure 2.1 shows the architecture for the knowledge discovery subsystem. The multimedia resource subsystem provides the required information such as the multimedia resource databases and the set of historical data, and the *generalized affinity-based* association mining is the main process of the knowledge discovery subsystem. The main task of this subsystem is to discover the set of quasi-equivalent media objects which can be used to assist in the construction of the probabilistic network. There are two major phases for the generalized *affinity-based* association *mining* process – Phase I and Phase II. Phase I is executed iteratively based on the refinement of the minimum interest threshold to generate the candidate set of quasi-equivalent media

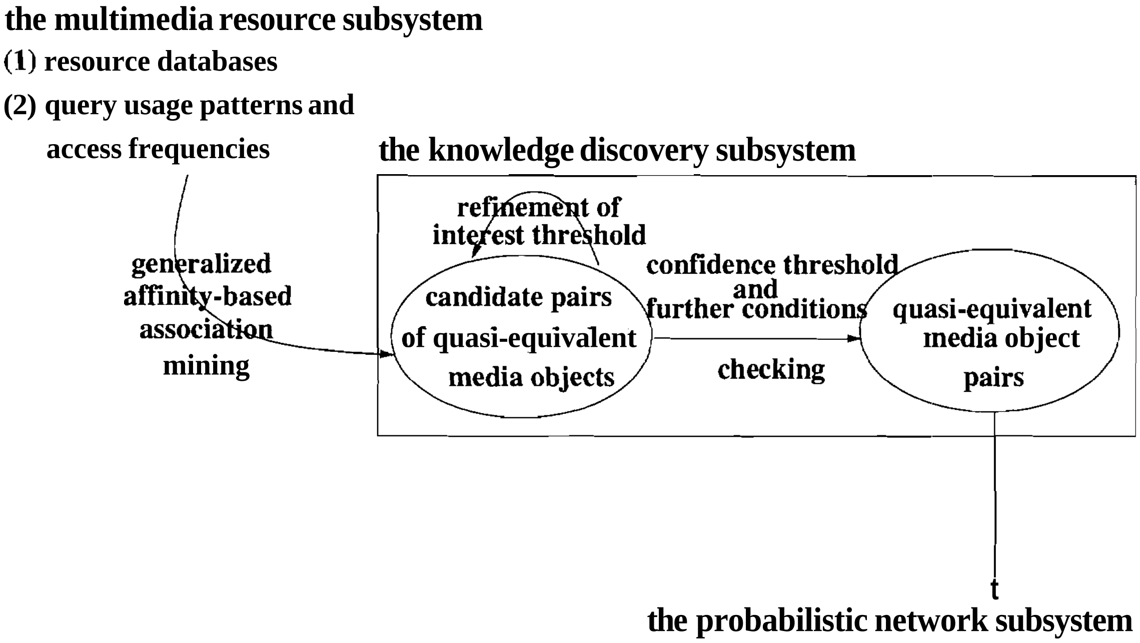

Fig. 2.1. *Architecture for the knowledge discovery subsystem.* The *multimedia resource subsystem* provides information such as the multimedia resource databases and the set of historical data (query usage patterns and access frequencies), and the *generalized afinity-based association mining* is the main process of the subsystem. The *generalized afinity-based association mining* process consists of two phases. The first phase is executed iteratively based on the refinement of the minimum interest threshold to generate the candidate set of quasi-equivalent media objects until a predefined refinement constraint is met. Then, the second phase checks the minimum confidence threshold and further conditions (if any) on the candidate set to get the final set of quasi-equivalent media objects.

objects until a predefined refinement constraint is met. Then. based on the candidate set generated from Phase I, Phase II checks the minimum confidence threshold and further conditions (if any) to get the final set of quasi-equivalent media objects.

We now roughly discuss the steps for the two phases. The detailed algorithm will be introduced in the next subsection. Phase I starts with a set of constraints: (1) minimum interest threshold, (2) interest constraint, and (3) refinement constraint. Any pair whose association rule has an interest value exceeding the interest threshold is first selected into the candidate pool. Next, the interest constraint is imposed to shrink the size of the candidate pool: the pair (m,n) remains in the candidate pool only if both (m,n) and (n,m) are in the candidate pool. That is, both $interest(m \rightarrow$ n) and $interest(n \rightarrow$ m) must satisfy the interest threshold criterion to make sure they are interesting enough in both directions. Then, the output of Phase I consists of a list of pairs of candidates. On seeing the candidates, the refinement constraint is checked to see whether further interest threshold refinement is necessary or not. In this manner, Phase I is iterative.

Once satisfied with the current candidate list, the process proceeds to Phase II, wherein two constraints are set: (1) minimum confidence threshold, and (2) whatever further conditions to be imposed. The minimum confidence threshold is used again to cut down the candidate pool size. The pair (m,n) stays in the candidate pool if either conf$idence(m \rightarrow$ n) or conf$idence(n \rightarrow$ m) reaches the minimum confidence threshold. Upon examining the output, further conditions can be imposed to get rid of unreasonable pairs in the candidate pool.

2.3.2 Algorithm

In this subsection, we describe the algorithm of the proposed generalized *affinity*-based association *mining* process to discover the set of quasi-equivalent media object pairs. This mining process is very useful to explore some semantic relationships from the complicated data structures of the databases automatically, and requires parameters such as the minimum interest threshold, refinement constraint, and minimum confidence threshold to be determined by the users subjectively according to different requirements for different applications. For example, we set the minimum confidence threshold to 99% in this study since we try to approximate the structural equivalence relationship which requires a high confidence factor. This flexibility allows users to set the criteria suitable for different applications. Though the mining process is used to find the set of quasi-equivalent media objects in this study, it can also be used in other applications. For example, in manufacturing, there exist hundreds of assembly-subassembly part relationships [34]. These relationships correspond to the concept of "composition" defined in the OSAM data model in [44] or the aggregation relationships. An aggregation hierarchy expresses part-of relationships between two media objects with 1:M cardinality by definition. Media objects are organized into an aggregation structural hierarchy if one media object is composed by other media objects in a nested or hierarchical fashion. This mining process can be applied to exploit some of the semantic relationships such as the assembly-subassembly part semantic relationships for the applications in the manufacture domain. Of course, the definitions of the affinity, support, confidence, and interest, and the selections of the parameters need to be adjusted accordingly.

Here, the details of the algorithm for the generalized *affinity-based* association mining process for this study are introduced. Starts with all the media objects in the databases. Let L_1 and L_2 represent the sets of 1-itemsets and 2-itemsets, where each 1-itemset has one media object and each 2-itemset has two media objects. Generate L_2 by $L_1 * L_1$ where $*$ is an operation for concatenation. The algorithm needs to make only one pass over the database. While the only pass is made, one record at a time is read and $support(m)$, af $f_{m,n}$, and the summary of $access_k$ are computed. After that, $support(m \to$ n) and $interest(m \to$ 72) can be obtained. There is no need to do multiple database scans?thus reducing the processing overheads. We now discuss how to generate the candidate pool and how to determine the set of quasi-equivalent media objects. Assume the number of media objects in the databases is Nmo. The values for $cria1$, $cria2$, and Conf need to be decided by the users before the algorithm is run. The variable $cria1$ sets the minimum interest threshold for each iteration. Let the maximal interest value for the media object m to be **I,** and the resulting set to be candidate-pool. The minimum interest threshold is defined to be "iteration number ×crial x I,". In this case, the minimum interest threshold increases as the number of iterations increases. The variable *cria2* sets the refinement constraint for Phase I. The refinement constraint threshold is defined to be "*cria2*× the total number of media objects". If the number of media objects which have zero or one pair remaining in the candidate-pool is greater than or equal to the refinement constraint, then Phase I stops and goes to Phase II. Otherwise, go to next iteration with a new minimum interest threshold for Phase I. The variable Conf sets the minimum confidence threshold for Phase II. This value is used to remove the pairs which fail the minimum confidence threshold checking.

⋆ **Steps** for Phase I:

1. For all the l-itemsets, compute *support*(m).

2. For all the 2-itemsets,

   - Compute af $f_{m,n}$.
   - Compute *support*$(m \rightarrow$ n).
   - Compute conf*idence*$(m \rightarrow$ n).
   - Compute *interest*$(m \rightarrow$ n).

3. Initialize candidate-pool $= \emptyset$ and iter = 1; set the values for crial and *cria2*.

4. For m = 1 to Nmo,

   (a) If iter = 1, then find the maximal interest value **I,** from *interest*$(m \rightarrow$ n) where a media object n is in a different database since the equivalence relationship can occur only when two media objects are from different databases.

   (b) Set the minimum interest threshold $IntTd$= crial x iter x I,.

```
    (c) For those media objects $n$'s,
        if iter $=$ 1 and $interest(m \rightarrow n) \geq IntTd$,
        then candidate-pool $=$ candidate-pool $\bigcup \{(m,n)\}$.
        else if $interest(m \rightarrow n) < IntTd$,
        then (m,n) is removed from candidate-pool.
5. Check the interest constraint:
    if $(m,n) \in$ candidate-pool and $(n,m) \notin$ candidate-pool,
    then (m,n) is removed from candidate-pool.
6. Check the refinement constraint:
    if the number of media objects which have zero or one pair remaining in the candidate-pool $\geq cria2$ x Nmo,
    then goto Phase II.
    else set iter $=$ iter $+$ 1 and goto step 4.
```

The first step is to compute the *support*(m) for every 1-itemset. Since each query may be activated multiple times, the actual access frequency of each query is taken into account in calculating *support*(m) and *support*$(m \rightarrow n)$ values. That is why this mining process is *affinity-based.* The advantage of using the relative affinity measures is to allow more informative feedback because it tells the number of accesses of the queries but not the number of queries. The second step is to compute the $aff_{m,n}$, *support*$(m \rightarrow n)$, *confidence*$(m \rightarrow n)$, and *interest*$(m \rightarrow n)$ for all the media object pairs. Only the *interest*$(m \rightarrow n)$ and *confidence*$(m \rightarrow n)$ values are needed in determining the set of quasi-equivalent media objects. Initialization of the variables is performed in the third step. The candidate-pool is initially set to be an empty set. Also, the values for the minimum interest threshold $(cria1)$ and the refinement constraint $(cria2)$ need to be defined. Again, these criteria can be adjusted for different applications. Set the number of iteration (iter) to be 1 and go to step 4. Step 4 executes a for-loop for all the media objects. First, find the maximal interest values for all the media objects on the first iteration. Once the maximal interest value I_m for media object m is obtained, the minimum interest threshold can be calculated according to the predefined formula. Similarly, the formula to calculate the minimum interest threshold can be varied for different applications. Then, the corresponding media object pair is put into the candidate-pool or removed from the candidate-pool by comparing its interest value with the minimum interest threshold. The candidate-pool constructed from step 4 goes to step **5** for the interest constraint checking. Since only those media object pairs whose interest values are above the minimum interest threshold on both directions are interesting enough to be considered as quasi-equivalent, the interest constraint is used to cross out the unsatisfied pairs from the candidate-pool in step **5.** In step 6, the refinement constraint is checked to see whether another iteration is required. If the number of the media objects which have zero or one pair remaining in the candidate-pool is equal to or greater than $cria2$ of the total number of media objects, then Phase I stops and goes to Phase II. Otherwise, go to step 4 for another iteration.

⋆ **Steps** for Phase II:

1. Set the minimum confidence threshold $Conf$

2. For each pair (m, n) in candidate-pool,
   if $confidence(m \rightarrow n) < Conf$ and $confidence(n \rightarrow m) < Conf$,
   then (m,n) is removed from candidate-pool.

3. Check if further conditions need to be imposed to remove some unreasonable situations.

The steps for Phase II are used to eliminate those media object pairs which are potentially nonequivalent. First, the minimum confidence threshold needs to be defined. Again, this threshold can be adjusted accordingly for different applications. The second step is to remove those media object pairs whose confidence values are smaller than the minimum confidence threshold on both directions. However, since some

Table 2.3
The maximal interest measure $I,$ for each media object m.

m	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
I_m	1.387	5.863	468.603	2.198	2.479	4.409	4.409	8.835	468.603	23.238	27.879	3.805	8.835	27.879	8.835	8.026	1.837	23.238	2.861	3.805	4.409	2.479

situations cannot be reflected directly by the numbers of accesses from the historical data, human reasoning is required. The last step of Phase II is to check whether there exist some unreasonable situations in the candidate-pool. For example, a media object cannot have equivalent relationships with two or more media objects in the same database at the same time since equivalence can only occur for media objects in different databases. These unreasonable situations need to be examined by humans to get the final set of quasi-equivalent media object pairs.

2.4 Empirical Study

To empirically test the proposed generalized affinity-based association mining approach, we ran the algorithm on the financial database management systems at Purdue University in July, August, and September for the year 1997. We implemented the algorithm with the affinity-based support, confidence, and interest measures reflecting the number of accesses for each media object. The databases represent *22* media objects accessed by *17,222* queries. Let the media objects be numbered from *1* to *22* and the media objects in the same database have consecutive numbers. Set $cria$1 $= 0.2$, $cria2 = 0.5$, and $Conf = 99\%$. Table 2.3 lists the maximal interest values for the *22* media objects and Figure 2.2 shows the candidate pairs in the candidate-pool for each iteration and each phase.

Two iterations were executed in Phase I. At the first iteration, the $I,$ measures for all media objects m's were first found (as shown in Table 2.3). Note that the maximal interest value for a media object may occur on multiple places. This situation occurs when $support(m \rightarrow$ n) is equal to $support(n)$. That is, those queries which access media object n also access media object m. For example, the pairs *(1,9)* and *(1,20)* both have interest value *1.387* which indicates that those queries which access media object *9* also access media object *1*. Similarly, those queries which access media object *20* also access media object *1*. However, the maximal interest for *9* occurs at the pair *(9,3)* and the maximal interest for *20* occurs at the pair *(20,12)*. From the observations, if the I_m measure occurs at $interest(m \rightarrow$ n), the I_n measure occurs at $interest(n \rightarrow$ m), and the I_m and I_n are equal, then m and n are potentially to be quasi-equivalent. Since those queries which access m also access n and those queries which access n also access m, this indicates that m and n are accessed by the same set of queries and thus they are very likely to have the quasi-equivalence relationship. In addition, we observe that when the I_m measure is very large, it converges to one quasi-equivalence pair for the corresponding media object m faster than other media objects since a certain percentage (0.2, 0.4, etc.) of the I_m value is used as the criterion to maintain the candidate-pool. When the I_m value is much larger than other interest values, it is possible that other media objects will be crossed out of the candidate-pool in one or two iterations. As can be seen from Table 3.3, the maximal interest value for media object *3* is *468.603* which occurs at the pair *(3,9)* and at the same time the maximal interest value for media object 9 is *468.603* which occurs at the pair *(9,3)*. Since the value *468.603* is extremely larger than other interest values for media objects *3* and *9*, only the pairs *(3,9)* and *(9,3)* remain in the candidate-pool for media objects 3 and *9* in the first iteration (as shown in Figure 2.2(a)).

When the I_m measures are determined, the $IntTd$ for the first iteration is set to be 0.2 x **I**, and 97 pairs are generated in the candidate-pool. After the interest constraint, 30 pairs are removed and the refinement constraint checking indicates that there is a need to go to the second iteration. The refinement constraint is to check whether the number of the media objects which have zero or one pair remaining in the candidate-pool is equal to or greater than 11 (i.e., 0.5 x 22). The first column in Figure 2.2(a) is an individual media object and the second column lists the candidate media objects corresponding to that individual media object. Those media objects which

PHASE I

(a) candidate–pool: (iteration 1)

media object	media object list
1	7, ~~8~~, ~~9~~, ~~10~~, ~~11~~, 12, ~~13~~, ~~14~~, ~~15~~, ~~16~~, 17, ~~18~~, 19, 20, 21
2	~~9~~, 10
3	9
4	~~8~~. ~~10~~. 11, ~~14~~, 15. 16, 17. ~~18~~, 22
5	7, ~~8~~, ~~10~~, ~~11~~, 12, ~~14~~, 17, ~~18~~, 22
6	7, 17
7	1, ~~3~~, 5, 6, ~~10~~, 17, ~~18~~, 21
8	13, 15, 16, 17
9	3
10	2, ~~3~~, 18
11	14
12	1, 5, 17, 19, 20
13	8, 18
14	11
15	4, 8
16	~~3~~, 4, 8
17	1, 4, 5, 6, 7, 8, ~~10~~, ~~11~~, 12, ~~13~~, 19, 20, 21
18	~~3~~. 10, 13
19	1, ~~6~~, 12, 17
20	1, 12, 17
21	1, 7, 17
22	~~3~~, 4, 5

(b) candidate–pool: (iteration 2)

media object	media object list
1	~~7~~, ~~12~~, 17, 19, ~~20~~, ~~21~~
2	~~10~~
3	9
4	~~11~~, ~~15~~, ~~16~~, 22
5	~~7~~, 17, 22
6	7, 17
7	6, 21
8	13, 15, 16
9	3
10	18
11	14
12	17, 19, 20
13	8, ~~18~~
14	11
15	8
16	8
17	1, 5, 6, ~~7~~, ~~8~~, 12, 19, 20, 21
18	10
19	1, 12, 17
20	12, 17
21	7, 17
22	4, 5

PHASE II

(c) candidate–pool: (confidence checking)

media object	media object list
1	~~17~~, 19
2	
3	9
4	~~22~~
5	~~17~~, ~~22~~
6	7, 17
7	6, 21
8	13, 15, ~~16~~
9	3
10	~~18~~
11	14
12	~~17~~, ~~19~~, 20
13	8
14	11
15	8
16	~~8~~
17	~~1~~, ~~5~~, 6, ~~12~~, 19, 20, 21
18	~~10~~
19	1, ~~12~~, 17
20	12, 17
21	7, 17
22	~~4~~, ~~5~~

(d) candidate–pool: (further checking)

media object	media object list
1	19
3	9
6	7, ~~17~~
7	6, 21
8	13, 15
9	3
11	14
12	20
13	8
14	11
15	8
17	~~6~~, ~~19~~, ~~20~~, ~~21~~
19	1
20	12, ~~17~~
21	7, ~~17~~

Fig. 2.2. The candidate pairs in the candidate-pool.

do not meet the interest constraint in the candidate media object list are crossed out from the candidate media object list. The resulting media object list is then input to the second iteration. At the second iteration, the minimum interest threshold $IntTd$ is incremented to 0.4 × **I**, which makes the pool shrink to 52 pairs. Next, the interest constraint is checked and 12 pairs are removed. (as shown in Figure 2.2(b)). Then, the refinement constraint is satisfied so that Phase I stops and the size of the pool goes from 97 pairs down to 40 pairs. That is, more than half of the pairs have been removed after Phase I is executed. Since the interest measures are based on the affinity relationships of the media objects, saying that the association (m $\rightarrow$ n) has high interest means that if the media object m is accessed by a query, then the media object n is much more likely to be accessed by the same query than other media objects. That is, media object n is much more likely to have a high affinity relationship with m than other media objects. Similarly, if both associations (m $\rightarrow$ n) and (n $\rightarrow$ m) satisfy the minimum interest threshold and interest constraint, then the pairs (m,n) and (n,m) are most likely to be quasi-equivalent.

In Phase II, the minimum confidence threshold Conf is set to be 99%. The reason for such a high confidence threshold is that rules with high confidence factors represent a higher degree of relevance than rules with low confidence factors. Since we try to approximate the structural equivalence relationship, which requires a high confidence factor, the confidence threshold is set high for this purpose. There are 24 pairs left in the candidate-pool after the confidence constraint checking (as shown in Figure 2.2(c)). Finally, it is checked whether some unreasonable situations exist and need to be avoided. In the current candidate-pool, media object numbered 17 appears to have quasi-equivalence relationships with media objects numbered 6, 19, 20, and 21. This is unreasonable because of the following two observations. First, media objects numbered 19, 20, and 21 belong to the same database. As mentioned previously, equivalence relationships exist only in media objects in different databases. Hence, it is impossible for media object numbered *17* to be quasi-equivalent to all three of them. Second, media object numbered 6 is quasi-equivalent to media object numbered 21 and at the same time is in the same database as media object numbered *1* which is quasi-equivalent to media object numbered *19.* Hence, media object numbered *17* cannot have quasi-equivalence relationships to media objects numbered *6, 19,* and *21.* From the above two observations, eight more pairs are removed and the final number of pairs in the candidate-pool is 16 (as shown in Figure 2.2(d)). Since the quasi-equivalence relationship (m, n) is the same as the quasi-equivalence relationship (n, m), there are eight quasi-equivalent pairs when the order is not considered.

3. THE PROBABILISTIC NETWORK

3.1 Data Mining and Probabilistic Reasoning

3.1.1 Architecture

Figure 3.1 shows the architecture for the *probabilistic network subsystem*. The multimedia resource databases and the set of historical data are provided by the *multimedia resource subsystem*. Another input is the set of quasi-equivalent media objects which is obtained from the *knowledge discovery subsystem*. The set of discovered quasi-equivalent media objects is used to determine the topology of the probabilistic network. Two databases are linked in the probabilistic network only if they have quasi-equivalent media objects. Once the required information is available, the three probabilistic distributions for each MMM can be constructed. After that, the first stochastic process formulates a similarity measure method. The similarity values are then transformed into the branch probabilities for the probabilistic network and the branch probabilities are used in partitioning the databases. Database clustering will be discussed in the next chapter.

3.1.2 Formulation of the Probability Distributions

State Transition Probability Distribution A

The relative affinity measurements are calculated to indicate how frequently two media objects are accessed together. When two multimedia databases whose media objects are accessed together more frequently, they are said to have a higher relative affinity relationship. Accordingly, in terms of the state transition probability in a Markov model, if two media objects have a higher relative affinity relationship, the probability that a traversal choice to state (media object) n given the current state (media object) is in m (or vice versa) should be higher. Realistically, the applications

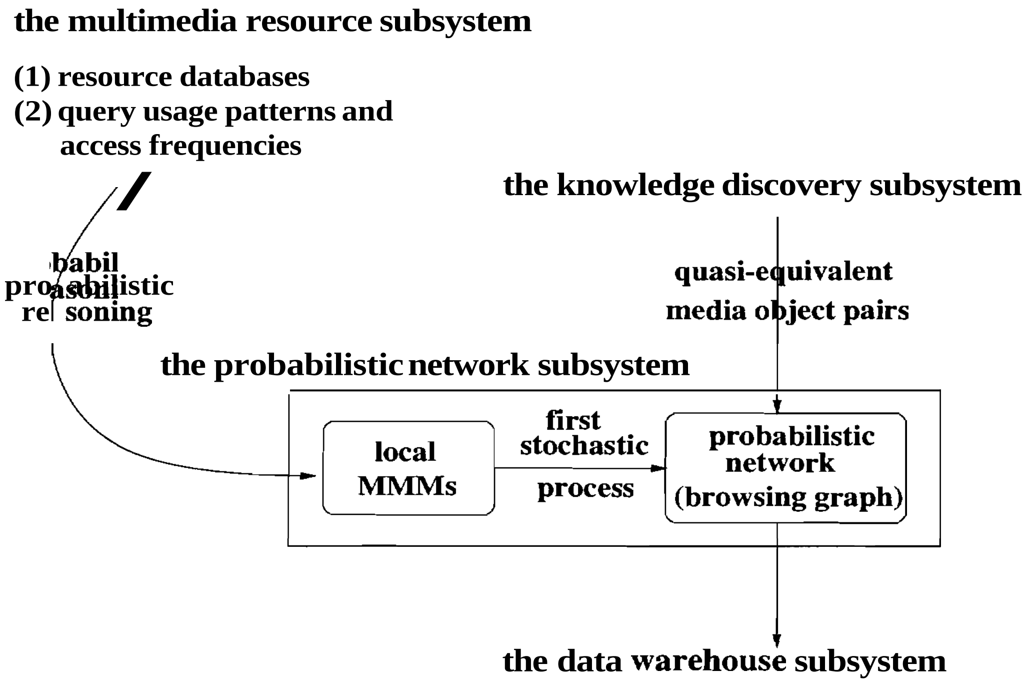

Fig. 3.1. *Architecture for the probabilistic network subsystem.* The *multimedia resource subsystem* and the *knowledge discovery subsystem* provide the information such as the multimedia resource databases, the set of historical data (query usage patterns and access frequencies), and the set of quasi-equivalent media objects to the *probabilistic network subsystem.*

cannot be expected to specify these affinity values. Therefore, formulas to calculate these relative affinity values need to be defined. Define the variables:

- $f_{m,n}$ = the joint probability which refers to the fraction of the relative affinity of media objects m and n in a database (or a warehouse) with respect to the total relative affinity for for all the media objects in a database (or a warehouse)
- f_m = the marginal probability
- $a_{m,n}$ = the conditional probability which refers to the state transition probability for an MMM

We denote A the state transition probability distribution whose elements are $a_{m,n}$.

- For a local MMM A;:

  The state transition probabilities for a local MMM is defined as follows. For any m,n $\in d_i$,

$$aff_{m,n} = \sum_{k=1}^{q} use_{k,m} \times use_{k,n} \times access_k \tag{3.1}$$

$$f_{m,n} = \frac{aff_{m,n}}{\sum_m \sum_n aff_{m,n}} \tag{3.2}$$

$$f_m = \sum_n f_{m,n} \tag{3.3}$$

$$a_{m,n} = \frac{f_{m,n}}{f_m} \tag{3.4}$$

- For an integrated MMM λ_{CC}:

  On the other hand, the state transition probability distribution for an integrated MMM can be defined in the following manner. If a data warehouse CC contains only one database, then no actions need to be taken to re-calculate the state transition probability distribution A for the integrated MMM. However, in most cases, a warehouse contains two or more databases so that the integrated MMM should merge those local MMMs in CC. Therefore, A needs to be re-calculated.

  Merge of two or more local MMMs:

For any media objects s and t in CC if there exists a link from s and t, the relative affinity measure between s and t is computed. Let λ_i and λ_j denote two local MMMs for d_i and d_j, where $j \neq i$ and $\lambda_i, \lambda_j \in CC$. Define the variables:

$aff_{s,t}$ = affinity measure of s and t if there is a link between s and t in an integrated MMM

$a_{s,t}$ = the state transition probability of a local MMM

$a'_{s.t}$ = the state transition probability of an integrated MMM

$p_{s,t}$ = the probability that λ_i goes to λ_j with respect to s and t

$p,$ = the probability that λ_i stays with respect to s

$$aff_{s,t} = \begin{cases} \sum_{k=1}^{q} use_{k,s} \times use_{k,t} \text{ X } access_k, & \text{if a link between } s \text{ and } t \text{ exists} \\ 0 & \text{otherwise} \end{cases} \quad (3.5)$$

$$f_{s,t} = \frac{aff_{s,t}}{\sum_{s \in CC} \sum_{t \in CC} aff_{s,t}} \quad (3.6)$$

$$p_{s,t} = \frac{f_{s,t}}{\sum_{n \in CC} f_{s,n}} \quad (3.7)$$

$$p_s = 1 - \sum_{t \notin \lambda_i} p_{s,t} \quad (3.8)$$

The steps for determining $a'_{s.t}$ where $s, t \in$ CC are:

1. For each local MMM λ_i where i $\in$ CC,
   - Compute the relative affinity $aff_{s,t}$ by using Equation **3.5.**
   - Compute the joint probability $f_{s,t}$ by using Equation **3.6.**
   - Compute the probability $p_{s,t}$ by using Equation 3.7.
   - Compute the probability $p,$ by using Equation 3.8.
2. If both $s, t \in \lambda_i \Longrightarrow a'_{s,t} = p_s a_{s,t}$.
3. If $s \in \lambda_i$ and $t \notin \lambda_i \Longrightarrow a'_{s,t} = p_{s,t}$.
4. Repeat step 1 for all local MMMs in CC.

Table 3.1
The state transition probability distribution A for d_1. For example, the transition goes from state 1 (media object $C_{1,1}$) to state 2 (media object $C_{1,2}$) is 0.3820.

state		
2	0.4474	0.5526

Originally, $\sum_t a_{s,t} = 1$. Now, we need to check whether the new state transition probability distribution satisfies the restriction, too. For any rnedia object $s \in \lambda_i$,

$$
\begin{aligned}
C\ t\ a'_{s,t} &= \textstyle\sum_{t \in \lambda_i} a'_{s,t} + \sum_{t \notin \lambda_i} a'_{s,t} \\
&= \textstyle\sum_{t \in \lambda_i} p_s a_{s,t} + \sum_{t \notin \lambda_i} p_{s,t} \\
&= p_s \textstyle\sum_{t \in \lambda_i} a_{s,t} + \sum_{t \notin \lambda_i} p_{s,t} \\
&= p_s + \textstyle\sum_{t \notin \lambda_i} p_{s,t} \\
&= 1 - \textstyle\sum_{t \notin \lambda_i} p_{s,t} + \sum_{t \notin \lambda_i} p_{s,t} \\
&= 1.
\end{aligned}
$$

A constructed state transition probability distribution for d_1 is shown in Table 3.1. Based on the obtained state transition probability distributions, the local MMMs for d_1 to d_4 in the example multimedia database system can be constructed with the state transition probabilities attached to the arcs to indicate the probabilities that go from one state to another state.

Observation Symbol Probability Distribution $\mathcal{B}$

The observation symbol probability denotes the probability of observing an output symbol (attribute/feature) from a state (media object). Since a media object has its own set of attributes/features and an attribute (a feature) can belong to multiple media objects, the observation symbol probabilities show the probabilities an attribute (a feature) is observed from a set of media objects.

To construct $\mathcal{B}$ for a multimedia database (a warehouse), a temporary matrix BB is first formulated for each multimedia database (a warehouse). The numbers of the media objects in a database (or a warehouse) and the total number of the distinct attributeslfeatures in all the multimedia databases determine the dimension of BB. The appearance order of the distinct attributeslfeatures in BB is defined in the following manner. For each database d; where i=l to the total number of databases, the distinct attributeslfeatures of the first media object of d_i are put in the rows of BB, next are the distinct attributeslfeatures of the second media object of d_i, and so on until the last media object of d;. The process repeats for all the attributeslfeatures of all the media objects in all the databases. The distinct attributelfeature indicates that this particular attributelfeature has not appeared in BB.

- In a local MMM A;:

  Let m_1, m_2, ..., m_{n_i} be the media objects in d; and z_1, z_2, ..., z_{tot} be the attributeslfeatures in all databases, where n; is the number of media objects in d_i and *tot* is the number of distinct attributeslfeatures in all databases. Put m_1, m_2, ..., m_{n_i} in the columns of BB and z_1, z_2, ..., z_{tot} in the rows of BB. Thus, BB is of size *tot* by n;.

- In an integrated MMM λ_{CC}:

  Let m_1, m_2, ..., m_k be the media objects in a warehouse CC and z_1, z_2, ..., z_{tot} be the distinct attributeslfeatures in all the databases, where k is the number of media objects in CC and *tot* is the number of distinct attributeslfeatures in all databases. Put m_1, m_2, ..., m_k in the columns of BB and z_1, z_2, ..., z_{tot} in the rows of BB. Thus, BB is of size *tot* by k.

For each MMM (local or integrated), BB is defined as follows.

$$BB_{p,q} = \begin{cases} 1 & \text{if attributelfeature p belongs to media object q} \\ 0 & \text{otherwise} \end{cases}$$

Table 3.2
$\mathcal{B}$ for d_1.

	1	2		1	2		1	2
1	1/4	0	11	0	0	21	0	0
2	1/4	0	12	0	0	22	0	0
3	1/4	0	13	0	0	23	0	0
4	1/4	1/4	14	0	0	24	0	0
5	0	1/4	15	0	0	25	0	0
6	0	1/4	16	0	0	26	0	0
7	0	1/4	17	0	0	27	0	0
8	0	0	18	0	0	28	0	0
9	0	0	19	0	0	29	0	0
10	0	0	20	0	0	30	0	0

Each entity of BB is assigned a value **1** or **0** to indicate whether the attribute/feature belongs to the corresponding media object. In this example, d_1 has two media objects and there are thirty distinct attributes/features for d_1 to d_4. Hence, the dimension of BB for d_1 is thirty by two. Then the observation symbol probability distribution $\mathcal{B}$ can be obtained via normalizing BB per column. In other words, the sum of the probabilities which the attributes/features are observed from a given media object should be 1.

A constructed observation symbol probability distribution for d_1 is shown in Table 3.2. The rows represent all the distinct attributes/features in the multimedia databases and the columns represent the media objects in the corresponding multimedia database.

Initial State Probability Distribution Π

Since the information from the training traces is available, the preference of the initial states for queries can be obtained. Let m $\in$ OC and CC be a data warehouse. Also, n_i is the number of media objects in $d;$.

- In a local MMM A;:

  For any media object m $\in$ *d;,* the initial state probability is defined as the fraction of the number of occurrences of media object m with respect to the total number of occurrences for all the member media objects in d_i from the training traces.

  $$\Pi_i = \{\pi_{i,m}\} = \frac{\sum_{k=1}^{q} use_{k,m}}{\sum_{l=1}^{n_i} \sum_{k=1}^{q} use_{k,l}} \tag{3.9}$$

  The $\pi_{i,m}$ value is the probability that a state (media object) m in d_i can be the initial state for the incoming queries.

- In an integrated MMM λ_{CC}:

  For any media object m $\in d_i$ and $d_i \in$ CC,

  $$\Pi_{CC} = \{\pi_{CC,m}\} = \frac{\sum_{k=1}^{q} use_{k,m}}{\sum_{d_i \in CC} \sum_{l=1}^{n_i} \sum_{k=1}^{q} use_{k,l}} \tag{3.10}$$

  The $\pi_{CC,m}$ value is the probability that a state (media object) m in CC can be the initial state for the incoming queries.

In this example, using Equation 3.9, the four initial state probability distributions for d_1 to d_4 can be determined.

$\Pi_1 = [5/9\ 4/91$	for database d_1
$\Pi_2 = [4/14\ 6/14\ 4/14]$	for database d_2
$\Pi_3 = [5/19\ 5/19\ 5/19\ 4/19]$	for database d_3
$\Pi_4 = [2/15\ 3/15\ 4/15\ 6/15]$	for database d_4

Once the probabilistic distributions $\mathcal{A}, \mathcal{B}$, Π for our *MMM* mechanism are determined, the two stochastic processes can be executed for data warehouse construction and multimedia database searching.

3.2 A Stochastic Process: Similarity Measure

A stochastic process is proposed for conceptual database clustering. Conceptual modeling allows an abstract representation of the multimedia databases and describes the databases with a set of conceptual schemas at different abstract levels. The objective of conceptual database clustering is to facilitate the understanding of the MDBMS and the speedup of query processing. To perform conceptual database clustering, a similarity measure is computed for each pair of databases which have quasi-equivalent media objects. This calculated similarity value is used to measure how well these two databases together match the observations generated by the sample queries. The similarity measure is partially derived from the *hidden Markov models (HMMs)* [31] and is formulated under the assumptions that the observation set O^k is conditionally independent given X and Y, and the sets $X \in d;$ and $Y \in d_j$ are conditionally independent given d_i and d_j. Let $N_k = kl + k2$. Define

$SM(d_i, d_j)$ = similarity measure between databases d_i and d_j

$\mathcal{OS}$ = set of all observation sets

$O^k = \{o_1, \ldots, o_{N_k}\}$ is an observation set with the attributes/features belonging to d_i and d_j and generated by query k

$X = \{x_1, \ldots, x_{k1}\}$ is a set of media objects belonging to d_i in O^k

$Y = \{y_1, \ldots, y_{k2}\}$ is a set of media objects belonging to d_j in O^k

$P(O^k \mid X, Y; d_i, d_j)$ = the probability of occurrence of O^k given $X \in d;$ and $Y \in d_j$

$F(N_k)$ = an adjusting factor used because the lengths of the observation sets are variable

The similarity measure is defined as follows.

$$SM(d_i, d_j) = (\sum_{O_k \in \mathcal{OS}} P(O^k \mid X, Y; d_i, d_j) P(X, Y; d_i, d_j)) F(N_k), where \tag{3.11}$$

$$P(O^k \mid X, Y; d_i, d_j) = P(o_1, \ldots, o_{k1} \mid X; d_i) P(o_{k1+1}, \cdots, o_{N_k} \mid Y; d_j) \tag{3.12}$$

$$P(X, Y; d;, d_j) = P(X; d_i) P(Y; d_j) \tag{3.13}$$

$$F(X, Y) = 10^{N_k} \tag{3.14}$$

$$P(X; d_i) = P(x_1, \ldots, x_{k1}; d_i) = \prod_{u=2}^{k1} \underbrace{P(x_u \mid x_{u-1})}_{A_i(x_u \mid x_{u-1})} \underbrace{P(x_1)}_{\Pi_i(x_1)}$$

$$P(Y;d_j) = P(y_1,\ldots,y_{k2};d_j) = \prod_{v=k1+2}^{N^k} \underbrace{P(y_{v-k1} \mid y_{v-k1-1})}_{A_j(y_{v-k1}|y_{v-k1-1})} \underbrace{P(y_1)}_{\Pi_j(y_1)}$$

$$P(o_1,\ldots,o_{k1} \mid X;d_i) = P(o_1,\ldots,o_{k1} \mid x_1,\ldots,x_{k1};d_i) = \prod_{u=1}^{k1} \underbrace{P(o_u \mid x_u)}_{B_i(o_u|x_u)}$$

$$P(o_{k1+1},\ldots,o_{N_k} \mid Y;d_j) = P(o_{k1+1},\ldots,o_{N^k} \mid y_1,\ldots,y_{k2};d_j) = \prod_{v=k1+1}^{N^k} \underbrace{P(o_v \mid y_{v-k1})}_{B_j(o_v|y_{v-k1})}$$

The similarity value is computed for two databases i and j which have quasi-equivalent media objects. The reason is that the topology of the probabilistic network is determined by the set of quasi-equivalent media objects input from the *knowledge discovery subsystem.* Therefore, if two databases do not have quasi-equivalent media objects between them, the similarity value between is assigned zero. That is, two databases are connected in the probabilistic network only if their media objects have the quasi-equivalence relationships. The similarity values are then transformed into the branch probability $P_{i,j}$ for nodes i and j (as shown in Table 3.3) in the probabilistic network. The transformation is executed by normalizing the similarity values per row to indicate the branch probabilities from a specific node (database) to all its accessible nodes (databases). Figure 3.2 is the probabilistic network with each node representing a database in the example *multimedia resource subsystem.* Each positive branch probability $P_{i,j}$ is attached to the corresponding arc.

Table 3.3
The branch probabilities transformed from the similarity values. For example, the branch probability from d_1 to d_4 is 0.8067.

$P_{i,j}$	d_1	d_2	d_3	d_4
d_1	-	0.1933	0	0.8067
d_2	0.4555	-	0.5445	0
d_3	0	1	-	0
d_4	1	0	0	-

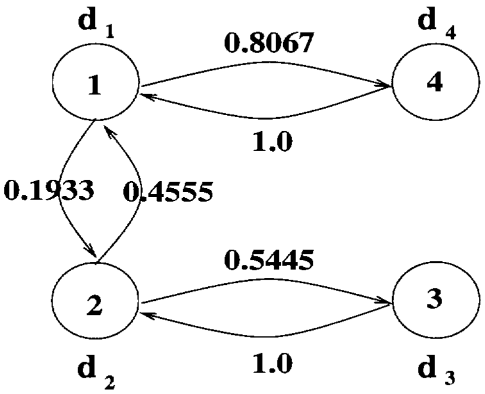

Fig. 3.2. The probabilistic network for the example multimedia database system. Nodes 1 to 4 represent databases d_1 to d_4 in the example, respectively. The positive branch probability is attached to the corresponding arc. The arcs with probabilities 0 are not shown in the figure.

4. DATA WAREHOUSING

4.1 Introduction

Data warehousing employs database technologies for storing and maintaining data. A data warehouse is a subject-oriented, integrated, time-varying, non-volatile collection of data that is used primarily in organizational decision making [18]. Since data warehousing is targeted for decision support, the summarized and consolidated information from data is more important than the detailed and individual records. The summarized and consolidated information may be missing from the data in the database, but can be obtained from the data mining techniques.

In a distributed information-providing environment, the number of databases has increased tremendously and much of the data in each database is structural in nature. Moreover, the workloads are query intensive with mostly complex queries that tend to access millions of records from a set of databases in such an environment. Hence, there is a need to analyze and discover summarized knowledge at the database level, i.e. database clustering. Here, two databases are said to be related in the sense that they either are accessed together frequently or have similar records or objects. Those member databases that are conceptually placed in the same cluster are the data in a data warehouse. A federation of data warehouses is constructed, each with its own decentralized administration.

This study considers conceptual database clustering rather than physical database clustering. Conceptual modeling allows an abstract representation of the participating databases and describes the databases with a set of conceptual schemas at different abstract levels. On the other hand, physical database clustering aims at improving the performance of databases by actually moving around the databases; that is not realistic given the autonomy of the databases. The objective of conceptual database clustering is to facilitate the understanding of the data warehouse schemas/views and the improvement of the query processing performance. An efficient database clustering approach can enhance the performance by placing on the same data warehouse the related set of databases. Query processing, in general, involves the closely interrelated *communication cost* and *processing cost.* Data warehouses may contain large volumes of data. To answer queries efficiently requires a good database clustering strategy. The way the data warehouses are constructed is to put a set of databases belonging to a certain application domain in the same data warehouse since similarity measures of the databases are used for clustering. In addition, since those databases in the same application domain are most likely required consecutively on some query access path, the number of platter switches (communication cost) and the number of nodes traversed (processing cost) for data retrieval with respect to queries can be reduced. Here, when a query is issued in a database, but requires the data in another database, a platter switch or a database switch is counted.

The integrated Markov model mediator (integrated MMM) mechanism is both the schema and view for a data warehouse. The network of databases is clustered into a federation of data warehouses via the proposed stochastic database clustering strategy which uses similarity measures introduced in previous chapter. A similarity measure is calculated only if these two databases have the structurally equivalent or quasi-equivalent media objects. A larger similarity value between two local MMMs implies that they should belong to the same data warehouse. The stochastic strategy is adopted since it provides better performance in object clustering [45]. After the federation of data warehouses is constructed, one integrated MMM serves as the schema and the view for one data warehouse.

4.2 Conceptual Database Clustering

4.2.1 Architecture

Figure **4.1** shows the architecture for the *data warehouse subsystem.* The inputs of the subsystem are the probability distributions of the local MMMs and the probabilistic network which come from the *probabilistic network subsystem.* Another input

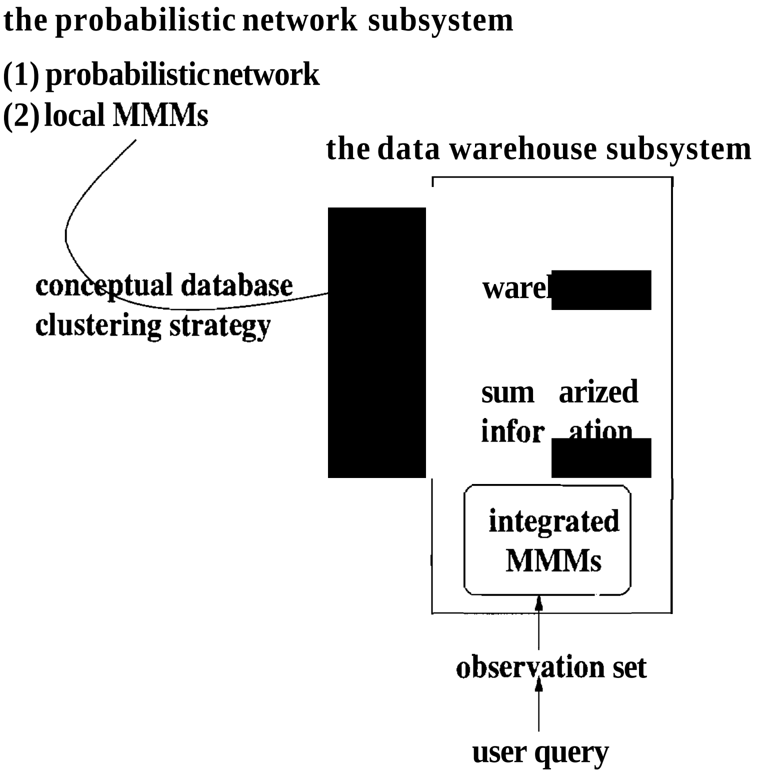

Fig. 4.1. *Architecture for the data warehouse subsystem.* The *probabilistic network subsystem* provides information such as the similarity values and the constructed probabilistic network to the *data warehouse subsystem.* An integrated MMM is the unit for retrieving information for the user queries. An observation set is constructed for each user query and this observation set is used for multimedia database searching.

Table 4.1
The stationary probabilities for databases d_1 to d_4.

	d_1	d_2	d_3	d_4
ϕ_i	0.4062	0.1723	0.0938	0.3277

is the queries issued by the users. Since an integrated MMM consists of the databases which belong to the same application domain, it is the unit for retrieving information for a query. An observation set is constructed for each user query and this observation set is used for multimedia database searching. The branch probabilities of the probabilistic network are transformed into the stationary probabilities. A database clustering strategy based on the stationary probabilities and the topology of the probabilistic network is proposed in this chapter. The database clustering strategy is used to construct a federation of beneficial data warehouses since those databases with high affinity relationships are placed in the same data warehouse. After the data warehouses are formed, an integrated MMM is constructed for each data warehouse and the three probability distributions for the integrated MMM need to be calculated. Information retrieval based on the integrated MMMs is shown in [38, 39].

4.2.2 The Database Clustering Strategy

The proposed stochastic database clustering strategy uses a similarity measure between every pair of databases to measure how well these two databases together match the observations generated by the sample queries. The stationary probability ϕ_i for each node i of the probabilistic network is computed from the $P_{i,j}$ values from the previous chapter. The stationary probability ϕ_i denotes the relative frequency of accessing node i (the ith database) in the long run. Table 4.1 shows the stationary probabilities for the four multimedia databases – d_1, d_2, d_3, and d_4.

$$\sum_i \phi_i = 1 \qquad \phi_j = \sum_i \phi_i P_{i,j} \;\; j = 1, 2, \cdots \tag{4.1}$$

The conceptual database clustering strategy is traversal based and greedy. Data warehouses are created according to the order of the stationary probabilities of the multimedia databases. The database that has the largest stationary probability is selected to start a new data warehouse. While there is room in the current warehouse, all databases accessible in terms of the probabilistic network from the current member databases of the warehouse are considered. The database with the next largest stationary probability is selected and the process continues until the warehouse fills up. At this point, the next unpartitioned database from the sorted list starts a new data warehouse, and the whole process is repeated until no unpartitioned databases remain. The time complexity for this database clustering strategy is $O(p \log \mathrm{p})$, where p is the number of databases.

In this example, assume the size of each data warehouse is two and apply the conceptual database clustering strategy. The size of the data warehouses is predefined and is the same for all data warehouses. The size of a data warehouse means the maximal number of member databases a data warehouse can have. Under this assumption, database d_1 first starts a new warehouse; database d_4 joins as a member of the warehouse since it has the next largest stationary probability and it is accessible from database d_1. Then, database d_2 creates another new warehouse with database d_3 joining the warehouse. Therefore, two data warehouses are constructed. An integrated MMM is formulated to play as the schema/view of each data warehouse.

4.2.3 Construction of Integrated MMMs for Data Warehouses

Once the federation of data warehouses is obtained after the database clustering strategy is applied, it is necessary to find out the probability distributions for the integrated MMMs for the data warehouses since multimedia database searching is based on the probability distributions of the integrated MMMs.

Let's say the data warehouse which contains databases d_1 and d_4 is called CC_1. Tables 4.2 and 4.3 show the constructed A and $\mathcal{B}$ for the warehouse CC_1. The rows and columns of A represent the media objects in the warehouse; while the rows and columns of $\mathcal{B}$ represent the total distinct attributes/features in the system and the media objects in the warehouse, respectively. As for the initial probability distribution

Table 4.2
A for the integrated MMM for the warehouse CC_1.

state	1	2	3	4	5	6
1	0.423	0.262	0	0	0	0.315
2	0.447	0.553	0	0	0	0
3	0	0	0.418	0.254	0.164	0.164
4	0	0	0.269	0.500	0	0.231
5	0	0	0.084	0	0.458	0.458
6	0.238	0	0.052	0.070	0.285	0.355

Table 4.3
$\mathcal{B}$ for the integrated MMM for the warehouse CC_1.

	1	2	3	4	5	6		1	2	3	4	5	6		1	2	3	4	5	6
1	1/4	0	0	0	0	0	11	0	0	0	0	0	0	21	0	0	0	0	0	0
2	1/4	0	0	0	0	1/2	12	0	0	0	0	0	0	22	0	0	0	0	0	0
3	1/4	0	0	0	0	0	13	0	0	0	0	0	0	23	0	0	1/3	0	0	0
4	1/4	1/4	0	0	0	0	14	0	0	0	0	0	0	24	0	0	1/3	1/4	0	0
5	0	1/4	0	0	0	0	15	0	0	0	0	0	0	25	0	0	1/3	0	1/5	0
6	0	1/4	0	0	0	0	16	0	0	0	0	0	0	26	0	0	0	1/4	0	0
7	0	1/4	0	0	0	0	17	0	0	0	0	0	0	27	0	0	0	1/4	1/5	0
8	0	0	0	0	0	0	18	0	0	0	0	0	0	28	0	0	0	1/4	1/5	1/2
9	0	0	0	0	0	0	19	0	0	0	0	0	0	29	0	0	0	0	1/5	0
10	0	0	0	0	0	0	20	0	0	0	0	0	0	30	0	0	0	0	1/5	0

for the warehouse, Equation 3.10 is used and the values are as follows.

$$\Pi_{CC_1} = [5/24\ 4/24\ 2/24\ 3/24\ 4/24\ 6/24] \qquad \text{for data warehouse } CC_1$$

5. CONCLUSIONS

With the advanced progress in high-speed communication networks, large capacity storage devices, digitalized media, and data compression mechanisms, multimedia applications have become popular. The emergence of networks of databases and the explosive growth in the sizes of networks and data have motivated the need for a *multimedia database management system (MDBMS).* A good MDBMS should not only store, manage, and retrieve multimedia information, but also discover useful information from the databases. In addition, the time and cost for query processing increase as the size of the network increases. The cost for query processing is very high especially when hundreds of multimedia databases are involved. Because of the navigational characteristics of queries, queries tend to access information from those databases in a certain application domain. If those related databases are conceptually grouped together as one unit, the time for query processing can be accelerated and the cost for query processing can be reduced. Toward this end, a probabilistic network-based mechanism is proposed for managing and utilizing the data in such an information-providing multimedia database environment. The probabilistic network-based mechanism systematically incorporates the probabilistic reasoning and data mining techniques into an MDBMS.

With the help of a probabilistic network, the affinity relations of the databases in the network can be embedded in some probabilistic model. For this purpose, we presented a probabilistic based construct called *Markov model mediator* (MMM) to serve as the probabilistic model for each node in the probabilistic network. The MMM construct employs the principle of Markov models and the concept of mediators. An MMM is a stochastic finite state machine with a stochastic output process attached to each state to describe the probability of occurrence of the output symbols (states). Thus, we have two concurrent stochastic processes: the sequence of the local/integrated MMM states modeling the structure of the media objects; and a set of state output processes modeling the predicting/identifying the required media objects for retrieving the information for an incoming query.

Our proposed system provides four functional capabilities: structural organization, information mining, database management, and information utilization under six subsystems in three steps to facilitate the functionality of an MDBMS. The six subsystems are the *multimedia resource subsystem, knowledge discovery subsystem, probabilistic network subsystem, data warehouse subsystem, query processing subsystem,* and *information retrieval subsystem.* The *multimedia resource subsystem* provides the probabilistic descriptions of a set of historical data and the structures of the databases as the prior information for the rest of the subsystems. The *knowledge discovery subsystem* and the *probabilistic network subsystem* provide the facilities to implement the structural organization and information mining capabilities based on the prior information from the *multimedia resource subsystem.* The network of databases is conceptually clustered into a federation of beneficial data warehouses in the *data warehouse subsystem* to perform database management. Finally, the information utilization capability can be achieved from the *query processing subsystem* and the *information retrieval subsystem.* These four functional capabilities are implemented in three steps. First, the probabilistic network is built by reasoning the probability distributions and mining the generalized affinity-based associations. Second, a federation of data warehouses is constructed by deriving the similarity measurement via a stochastic process. Third, a list of possible paths with respect to a query is identified via the second stochastic process.

LIST OF REFERENCES

[1] R. Agrawal, T. Imielinski, A. Swami, "Mining association rules between sets of items in large databases," in Proc. 1993 ACM *SIGMOD* Conference on Management of Data, pp. 207-216, 1993.

[2] E. J. Bellegrada et al., "Discrete Parameter HMM Approach to On-Line Handwriting Recognition," in Proceedings of the 1995 *20th* International Conference on Acoustics, Speech, and Signal Processing, Part 5 (of 5), Detroit, MI, USA, May 1995, pp. 2631-2634.

[3] P. Cabena, P. Hadjinian, R. Stadler, J. Verhees, and A. Zanasi, Discovering data mining from concept to implementation. Prentice-Hall Inc., 1998.

[4] K.S. Candan, P.V. Rangan, and V.S. Subrahmanian, "Collaborative multimedia systems: synthesis of media objects," IEEE Transactions on *Knowledge* and Data Engineering, Vol. 10, No. 3, pp. 433-457, May/June 1998.

[5] S. Chaudhuri and U. Dayal, "An overview of data warehousing and OLAP technology," *SIGMOD* Record, pp. 65-74, Vol. 26, No. 1, March 1997.

[6] P. Cheeseman and J. Stutz, "Bayesian classification (AutoClass): theory and results," in U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, Advances in Ii'nowledge Discovery and Data *Mining*, pp. 153-180, AAAI/MIT Press, 1996.

[7] M.S. Chen, J. Han, and P.S. Yu, "Data mining: An overview from a database perspective," IEEE Transactions on *Knowledge* and Data Engineering, pp. 866-883, Vol. 8, No. 6, December 1996.

[8] Shu-Ching Chen and R.L. Kashyap, "Temporal and spatial semantic models for multimedia presentations," in 1997 International Symposium on Multimedia Information Processing, pp. 441-446, Dec. 11-13, 1997.

[9] Shu-Ching Chen and R.L. Kashyap, "Empirical studies of multimedia semantic models for multimedia presentations," in 13th International Conference on Computer and Their Applications, pp. 226-229, March 25-27, 1998.

[10] Shu-Ching Chen and R.L. Kashyap, "A spatio-temporal semantic model for multimedia presentations and multimedia database systems," accepted for publication in IEEE Transactions on Ii'nowledge and Data Engineering, 1999.

[11] C.J. Date, An Introduction to Database Systems. 6th Ed. Reading, MA: Addison-Wesley, 1995.

[12] P.J. Diggle, Statistical analysis of spatial point patterns. Academic Press, New York, 1983.

[13] J.F. Elder IV and D. Pregibon, "A statistical perspective on knowledge discovery in databases," in U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, Advances in Knowledge Discovery and Data Mining, pp. 83-113, AAAI/MIT Press, 1996.

[14] R. Elmasri and S.B. Navathe, Fundamentals of Database Systems. 2nd Ed. The Benjamin/Cummings, 1994.

[15] M. Ester, H.P. Kriegel, and X. Xu, "Knowledge discovery in large spatial databases: Focusing techniques for efficient class identification," in Proc. Fourth Int*'l* Symp. Large Spatial Databases *(SSD'95)*, pp. 67-82, August 1995.

[16] U.M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery: An overview," in U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, Advances in Knowledge Discovery *and* Data Mining, pp. 1-34, AAAI/MIT Press, 1996.

[17] O. Frank and D. Strauss, "Markov graphs," Journal of the American Statistical Association, 81, 1986, pp. 832-842.

[18] W.H. Inmon, Building the data warehouse. Wellesley, MA: QED Technical Publishing Group, 1992.

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

[20] J.A. Larson, S.B. Navathe, and R. Elmasri, "A theory of attribute equivalence in databases with application to schema integration,?' IEEE Transaction on Software Engineering, vol. 15, no. 4, Apr. 1989.

[21] H-Y. Lee, H-L. Ong, and L-H. Quek, "Exploiting visualization in knowledge discovery," in Proc. 1st Int'l Conf. Knowledge Discovery and Data Mining *(KDD-95)*, pp. 198-203, 1995.

[22] H.C. Lin, L.L. Wang, and S.N. Yang, "Color image retrieval based on hidden Markov models," in IEEE International Conference on Image Processing, Vol. 1, Washington DC., Oct. 23-26, 1995.

[23] Y. Ling, W. Sun, N.D. Rishe, and X. Xiang, "A hybrid estimator for selectivity estimation," IEEE Transactions on *Knowledge* and Data Engineering, Vol. 11, No. 2, pp. 338-354, March/April 1999.

[24] M. Lohr and T.C. Rakow, "Audio support for an object-oriented database-management system," ACM Multimedia Systems Journal, vol. 3, pp. 286-297, November 1995.

[25] H. Lu, R. Setiono, and H. Liu, "NeuroRule: A connectionist approach to data mining," in Proc. 21th Int'l Conf. Very Large Data Bases, pp. 478-489, September 1995.

[26] E. Mittendorf and P. Schauble, "Document and passage retrieval based on Hidden Markov Models," in ACM-SIGIR Conference on Research and *Development* in Information Retrieval, pp. 318-327, 1994.

[27] S.B. Navathe, R. Elmasri, and J.A. Larson, "Integration user views in database design," Comput., vol. 19, Jan. 1986.

[28] YK. Park and CK. Un, "Word Recognition by a Parallel-Branch subunit model based on misrecognized Data in Semi-Continuous HMM," IEEE Signal Processing Letters, v3 n3 1995 Mar., pp. 66-68.

[29] G. Piatetsky-Shapiro, "Knowledge discovery in real databases: A report on the IJCAI-89 Workshop," *AI* Magazine, vol. 11, no. 5, Special issue, pp. 69-70, Jan. 1991.

[30] V. Poe, Building a Data Warehouse for Decision Support. Prentice-Hall, 1996.

[31] L.R. Rabiner and B.H. Juang, "An introduction to hidden markov models," IEEE ASSP Magazine, 3(1), pp. 4-16, January 1986.

[32] M. G. Rahim et al., "Signal Conditioning Techniques for Robust Speech Recognition," IEEE Signal Processing Letters, v 3 n 4, pp. 107-109, April 1996.

[33] B.D. Ripley, Spatial statistics. John Wiley, Chichester, 1981.

[34] A. Rosenthal and S. Heiler, "Querying part hierarchies : A knowledge-based approach," Proc. *ACM/IEEE* Design Automation Conference, 1987.

[35] A. Sen and V.S. Jacob, '(Industrial-strength data warehousing," Communication of the ACM, Vol. 41, No. 9, September 1998.

[36] J.W. Shavlik and T.G. Dietterich. Editors, Readings in Maching Learning. San Mateo, CA: Morgan Kaufmann, 1990.

[37] Mei-Ling Shyu, Shu-Ching Chen, and R. L. Kashyap, "Database Clustering and Data Warehousing," in 1998 ICS *Workshop* on Software Engineering and Database Systems, pp. 30-37, Dec. 17-19, 1998.

[38] Mei-Ling Shyu, Shu-Ching Chen, and R. L. Kashyap, "Information Retrieval Using Markov Model Mediators in Multimedia Database Systems," in 1998 International Symposium on Multimedia Information Processing, pp. 237-242, Dec. 14-16, 1998.

[39] Mei-Ling Shyu and Shu-Ching Chen, "Probabilistic Networks for Data Warehouses and Multimedia Information Systems," Submitted to IEEE Transactions on *Knowledge* and Data Engineering.

[40] Mei-Ling Shyu, A probabilistic network-based mechanism for multimedia database searching and data warehousing, Ph.D. thesis, School of Electrical and Computer Engineering, Purdue University, 1999.

[41] E. Simoudis, B. Livezey, and R. Kerber, "Integrating inductive and deductive reasoning for data mining, " in U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, Advances in *Knowledge* Discovery *and* Data *Mining*, pp. 353-373, AAAI/MIT Press, 1996.

[42] R. Srikant and R. Agrawal, "Mining generalized association rules," in Proc. 21th *Int'l* Conf, *Very* Large Data Bases, pp. 407-419, September 1995.

[43] R. Srikant and R. Agrawal, "Mining quantitative association rules in large relational tables," in Proc. 1996 *ACM* SIGMOD *Int'l* Conf. Management Data, pp. 1-12, June 1996.

[44] S.Y. Su, V. Krishnamurthy, and H. Lam, "An object oriented semantic association model (OSAM)," in Artificial Intelligence : Manufacturing Theory and Practice, ed. S. Kumara, R.L. Kashyap and A.L. Soyster, American Institute of Industrial Engineers, Norcross, Ga, 1988.

[45] M.M. Tsangaris and J.F. Naughton, "On the performance of object clustering techniques," in Proc. ACM SIGMOD Int'l Conf. on *Management* of Data, pp. 144-153, June 1992.

[46] H.J. Watson and B.J. Haley, "Managerial considerations," Communications of the ACM, pp. 32-37, vol. 41, no. 9, September 1998.

[47] J. Widom, "Research problems in data warehousing," in Proc. Fourth *Int'l Conf.* Information and Knowledge Management, pp. 25-30, November 1995.

[48] G. Wiederhold, "Mediators in the architecture of future information systems," IEEE Computer, pp. 38-49, March 1992.

[49] G. Wiederhold, "Intelligent integration of information," in ACM SIGMOD Conference, pp. 434-437, May 1993.

[50] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: An efficient data clustering method for very large databases," in Proc. 1996 ACM SIGMOD Int'l Conference on *Management* of Data, pp. 103-114, June 1996.