IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-4, Issue-7 E-ISSN: 2347-2693

Managing and Mining Web Multimedia – B-Tree Indexing Approach

Siddu P. Algur[1], Basavaraj A. Goudannavar[2*]

[1]Department of Computer Science, Rani Channamma University, Belagavi-591156, Karnataka, India
[2]Department of Computer Science, Rani Channamma University, Belagavi-591156, Karnataka, India

Available online at: www.ijcseonline.org

Received:18/Jun/2016 Revised:26/Jun/2016 Accepted:16/Jul/2016 Published:31/Jul/2016

***Abstract*—** Develop a proper method in multimedia data management is important to support multimedia application's domain. Multimedia is defined as the combination of more than one media; they may be of two types - static and dynamic media. Text, graphics, and images are categorized as static media; on the other hand, objects like- animation, music, audio, speech, video are categorized as dynamic media. To manage this multimedia database management system is essential. Multimedia database management system can be defined as software system that manages a collection of multimedia data. Generally, multimedia database contains text, image, animation, video, audio, movie sound etc. But, all data are stored in the database in binary form. The paper presents the effective framework for the multimedia database management in terms of B-tree indexing. The framework is to ensure that the process of data manipulation, store, in a distributed environment can be conducted in an efficient manner.

***Keywords*—** Multimedia database, Multimedia Metadata, Multimedia database management and Indexing

I. INTRODUCTION

A multimedia database system includes a managing web multimedia database system, which manages and also provides support for storing, manipulating, and retrieving multimedia data from the multimedia database, a large collection of multimedia objects, such as image, video, audio, and text data.

Data in web databases are both structured and unstructured. Structured databases include those that have some structure such as relational and object databases. Unstructured databases include those that have very little structure such as text, image, audio, and video databases. In general, multimedia databases are unstructured. Some text databases are semi-structured databases, meaning that they have partial structure. The developments in multimedia database management systems have exploded during the past decade. While numerous papers and some texts have appeared in multimedia databases, more recently these databases are being mined to extract useful information [1]. A multimedia database system is a type of heterogeneous database system, as it manages heterogeneous data types. Heterogeneity is due to the media of the data such as text, video, audio etc. It support and manages multimedia data types. Therefore, all of the issues in designing a DBMS apply for an MM-DBMS. That is, we need framework for MM-DBMSs [2][3].

The rest of the paper is organized as follows: The section 2 represents related works on the association rule on web multimedia datasets, section 3 represents proposed framework, and section 4 represents conclusion and future work.

II. PRIOR WORKS

This paper introduces a framework for multimedia data mining system, to illustrate the general process of multimedia data mining. At the same time, it has proposed several mining methods, but a description is primarily based on association rule. Multimedia data mining is the combination of data mining and multimedia database, it isn't only an emerging research direction, but is also a challenging field of study; in future multimedia database mining is mainly content-based intelligent data retrieval, cross-index of a variety of media will also become a research hotspot, mining for multimedia data and intelligent information retrieval are the needs of future development [4].

In this paper, we present a method to construct such resume and illustrate our framework with current Semantic Web technologies, such as RDF and SPARQL for representing and querying semantic metadata. Some experimental results are provided in order to show the benefits of indexing and retrieving multimedia contents without centralizing multimedia contents or their associated metadata, and to prove the efficiency of a metadata. The proposed a framework for indexing and retrieving distributed multimedia contents [5]. A generic architecture and explained its components by using RDF for the metadata descriptions of multimedia contents.

In this paper started with multimedia database management systems and provided and overview of these systems. In particular, different types of architectures, data models, and functions of these systems are discussed. A multimedia database system includes a managing web multimedia database system, which manages and also provides support for storing, manipulating, and retrieving multimedia data from the multimedia database, a large collection of multimedia objects, such as image, video, audio, and text data. Data in Web databases are both structured and unstructured. Structured databases include those that have some structure such as relational and object databases. Unstructured databases include those that have very little structure such as text, image, audio, and video databases. In general, multimedia databases are unstructured. Some text databases are semi-structured databases, meaning that they have partial structure. The developments in multimedia database management systems have exploded during the past decade. While numerous papers and some texts have appeared in multimedia databases, more recently these databases are being mined to extract useful information [1].

The explosion of multimedia information in diverse kind of data needs efficient operation process in database management such as query retrieval management, classification management, multimedia data processing management and data security management. However, the time stamping based multimedia data manager tool has not been broadly emphasized. This paper addresses the issue of providing data management support for multimedia data using temporal elements. The integration of temporal elements is believed can improve the multimedia data management process. The proposed model provides a new discovery for archiving multimedia data in the more systematic way based on time information data. The concept of temporal data management must be introduced into multimedia data management to ensure that event and transaction of multimedia information record can be managed accurately [6].

The paper describes a metadata model where contextual information, in an archival perspective, is combined with the description of contents, as required for content-based search. The metadata model adopts a flexible multi-level approach to both context and content description, to allow for different granularities in the description of materials. It may be impracticable to provide descriptions at the individual item level in the digitization of an historic photo archive. Instead, a top-down approach to the description of collections can be a good compromise. For the video materials of a broadcast station, on the other hand, there is automatically produced information for each recorded shot that can be also accommodated in the model. The model has been tested in a prototype database system where specialized interfaces are offered for cataloguing and for search. The database contains data from historic, photo and video archives. The prototype also provides tools for filtering information into standard formats for interchange [7].

III. PROPOSED METHODOLOGY

The objective of the proposed framework is to extend the limitations to the existing managing and mining multimedia data aspects. The Fig.1 shows the proposed framework for web multimedia managing and mining approach.

The proposed framework consists of the following components:

i) Web multimedia metadata extraction and Pre-processing
ii) Managing Multimedia data
iii) Mining Techniques
iv) Knowledge Discovery Process

The each of the components of the proposed framework is discussed as follows:

A. Web Multimedia metadata extractions and Pre-processing

There are many tools available to decompose the web multimedia data meaning that, it is possible to retrieve individual components object from multimedia data. The various techniques to extract web multimedia data and pre-processing are discussed in our previous work [8].

B. Managing Multimedia Data

This section provides an overview of the various functions such as Data Manipulation, Indexing Multimedia data, Metadata/ Storage management, security and integrity.

i. Data manipulation

Manipulating data is that process of re-sorting, rearranging and moving data, without fundamentally changing it. One of the key characteristics of a manipulation technique versus related techniques like transformation is that the underlying data remains unchanged.

There are also boundaries in purpose for Data Manipulation:

- **Inference** - creates more value from existing data. One might use induction or deduction to obtain valuable information from less-valuable data (e.g. in response to a query), or attempt to shift anticipated computational costs from a time when they are more expensive to a time when they are less expensive (e.g. in anticipation of a query). The latter could include performing induction or deduction ahead of time, but also includes such things as indexing, which is a form of inference in the sense that it adds a data-object (in particular a meta-data-object, the index) to the database.
- **Maintenance** - where *unneeded* facts are removed, generally with the intent to reduce computational costs associated (usually space and search-time). Data items are *unneeded* when they may be derived from other

facts in the Data space or when they will never be relevant to any future queries.

- **Query** - In a sense, **Inference** always *grows* the data set while **Maintenance** always *shrinks* it.

However, not all inference needs to add to the *persistent* data set. Making a volatile inference logically rolls inference and maintenance into one action - you derive new facts, fire those off to whomever requested them, then immediately remove those facts as unnecessary to future queries. This sort of volatile inference most closely corresponds to the notion of a *query* response.

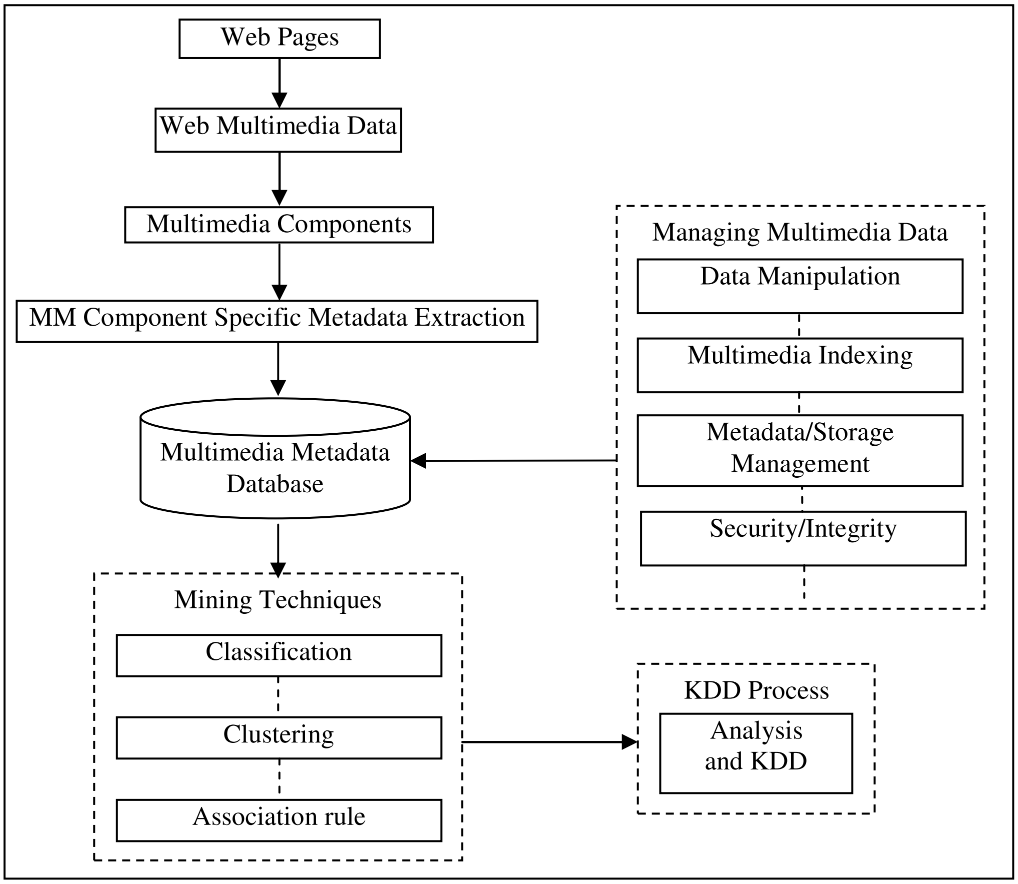

Fig 1: Framework for Managing and Mining Multimedia Database

ii) Multimedia Indexing

The availability of multimedia in the large databases of web multimedia data emerged the imperative need to address the challenge of content-based searching, where users pose multimedia objects as queries, in order to find relevant content. However, exhaustive searching is infeasible for the large scale applications of social networks due to the extensive time consumption it requires. Thus, the large databases of web multimedia data should be supported by indexing schemes which are able to provide: (a) low space requirements for storing the multimedia content within the indexing scheme; (b) efficient search time; and (c) high retrieval accuracy. Nevertheless, multimedia objects like compressed images, video and audio streams are usually described by sequences of descriptor vectors with over than a thousand dimensions. In this high dimensional space, the performance of existing indexing schemes deteriorate significantly, since content-based similarity search in high dimensions is challenging, due the well known problem of Dimensionality Curse [9]. In order to address the aforementioned challenges, the existing indexing schemes are divided into two main categories: those that follow (a) exact and (b) approximate similarity search strategies. The family of exact similarity search, despite achieving identical retrieval accuracy to exhaustive search, fails to support the high dimensionality. Meanwhile, storage space and search time are dramatically increased. The family of indexing schemes that follow the approximate search strategy, despite reducing the space and search time requirements, fails to preserve the retrieval accuracy of the exhaustive search.

iii) *Metadata/Storage Management*

Metadata management can be defined as the end-to-end process and framework for creating, controlling, enhancing, attributing, defining and managing a metadata schema, model or other structured aggregation system, either independently or within a repository and the associated supporting processes for web-based systems, text, audio, image, video etc. may be referenced from a triples table of object, attribute and value. Metadata may include descriptions about the data, the source of the data as well as the quality of the data.

The storage management encompasses the technologies and processes organizations use to maximize or improve the performance of their data storage resources. It is a broad category that includes virtualization, replication, mirroring, security, compression, and process automation, storage provisioning and related techniques. As a result, organizations feel constant pressure to expand their storage capacity. However, doubling a multimedia storage capacity every year is an expensive proposition. In order to reduce some of those costs and improve the capabilities and security of their storage solutions, organizations turn to a variety of storage management solutions. The storage manager has to ensure that access is controlled to the multimedia database. Storage manager may also be responsible for partitioning the data according to the security levels. The security impact of access methods and indexing strategy for multimedia data are yet to be determined. Numerous index strategies have been developed for multimedia data including for text, images, audio and video.

iv) *Maintaining data integrity and security*

Database security, according to Connolly et al. [10], concerns "The protection of the database against intentional or unintentional threats using computer-based or non-computer-based controls." Besides the effect that poor database security can have on the database, it may also threaten other parts of a system and thus an entire organization.

The risks related to database security are:

- Theft and fraud which are activities made intentionally by people. This risk may result in loss of confidentiality or privacy.
- Loss of confidentiality which refers to loss of organizational secrets.
- Loss of privacy which refers to exposure of personal information.
- Loss of integrity which refers to invalid or corrupt data.
- Loss of availability which means that data or system cannot be reached.

Threats that correspond to those risks are such situations or events in which it is likely that an action, event or person will harm an organization. Threats can be tangible, that is, cause loss of hardware or software, or intangible, as in with loss of credibility or confidence. In order to be able to face threats, a risk analysis should be conducted, in which a group of people in an organization tries to identify and gather information about the organization's assets, the risks and threats that may harm the organization and the counter measures that can be used to face those risks. Decisions made using such risk analysis are there after used to implement security measures in the system. These security measures can be computer-based controls or non-computer-based controls.

a) Computer-Based Controls

According to Connolly et al. [10], computer-based controls are used for protecting DBMS through means of authorization, views, backup and recovery, integrity, encryption and associated procedures.

i) Authorization

Authorization is used to define which activities (or privileges) are granted to different users (or subjects), which allows them to manipulate or retrieve information from different database objects. Usually, a simple mechanism of usernames and passwords is used, whether in the DBMS or in combination with the operating system where the DBMS resides. A user has to fill name and password, and the authentication mechanism confirms that comparing the password with the corresponding password in a list it maintains. The DBMS usually maintains a list of privileges that subjects have on certain database objects. A DBMS that operates as a closed system maintains a privileges list in which users are not allowed to operate on any objects except the ones in the list. A DBMS that operates as an open system, on the other hand, allows users to operate on all objects except those that are explicitly removed and listed in the privileges list. Privileges may also be group-based or role-based. Both users and objects may be joined in a group and privileges may be given to a group of users or objects. Certain roles can also be given privileges on objects, and a number of different users may undertake a certain role [11].

ii) Backup and Recovery

In order to be able to recover from a failure, a DBMS must regularly make a copy of the database and log files. Log files are a list of activities made in the database that can be used to recover the database after a failure. Check points made in certain time intervals can assure that the backup and log files are synchronized. This allows for safe recovery since operations that are listed in the log file need only be carried out from the point in time when the last backup was made.

iii) Integrity

Integrity controls can be used to see to that data in database does not get corrupt. Such controls are called relational integrity controls, and are rules that some databases implement internally to maintain data validity. Other databases do not implement those controls and it is up to the application programmer that uses the database to see to that data validity is being maintained.

iv) Encryption

Encryption is a method that is used for encoding the data so that other programs cannot read it. Some DBMSs contain an internal encryption mechanism, while other relays on the operating system or third-party programs.

C. Mining Techniques and Algorithms

The algorithm and techniques employed to perform multimedia data mining are most important. Data mining techniques are numerous. Many of these techniques may also be applied for multimedia data mining. Within the supervised framework, three data mining methods have been used. These are classification, association modeling. Within the unsupervised learning, clustering is another data mining methodology used.

i) Classification models

Classification is a technique for multimedia data analysis, can learn from every property of a specified set of multimedia. It is divided into a predefined class label, so as to achieve the purpose of classification. Classification is the process of constructing data into categories for its better effective and efficient use, it creates a function that well-planned data item into one of many predefined classes, by inputting a training data set and building a model of the class attribute based on the rest of the attributes. Decision tree classification has a perceptive nature that the users conceptual model without loss of exactness. Hidden Markov Model used for classifying the multimedia data such as images, audio and video data [12].

ii) Clustering models

Cluster analysis or clustering is the task of assigning a set of objects into groups (called clusters) so that the objects in the same cluster are more similar (in some sense or another) to each other than to those in other clusters. Clustering is a main task of explorative data mining, and a common technique for statistical data analysis used in many fields, including machine learning, pattern recognition, image analysis, information retrieval, and bioinformatics. Finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups. Partitional clustering, a division data objects into non-overlapping subsets (clusters) such that each data object is in exactly one subset

iii) Association Rules

Association Rule is one of the most important data mining technique which helps to find relations between data items in huge databases. In this associate rule mining algorithm that searches for approximate association rules [13]. In this article introduce an enhancement to the Apriori association rule algorithm, called ~AR, which generates approximate association rules. The AR algorithm takes into consideration missing values and noisy data. Missing values are replaced by probability distributions over possible values for the missing feature which allows the corresponding transaction to support all itemsets that could possibly match the data.

iv) Predication

Regression technique can be adapted for predication. Regression analysis can be used to model the relationship between one or more independent variables and dependent variables. In data mining independent variables are attributes already known and response variables are what we want to predict. Unfortunately, many real-world problems are not simply prediction. For instance, sales volumes, stock prices, and product failure rates are all very difficult to predict because they may depend on complex interactions of multiple predictor variables. Therefore, more complex techniques (e.g., logistic regression, decision trees, or neural nets) may be necessary to forecast future values. The same model types can often be used for both regression and classification.

v) Statistical Modeling

Statisticians were the first to use the term "data mining." Originally, "data mining" or "data dredging" was a derogatory term referring to attempts to extract information that was not supported by the data. Now, statisticians view data mining as the construction of a statistical model, that is, an underlying distribution from which the visible data is drawn. Suppose our data is a set of numbers. This data is much simpler than data that would be data mined, but it will serve as an example. A statistician might decide that the data comes from a Gaussian distribution and use a formula to compute the most likely parameters of this Gaussian distribution. The mean and standard deviation of this Gaussian distribution completely characterize the distribution and would become the model of the data. Statistical mining models are used to determine the statistical validity of test parameters and can be utilized to test hypothesis, undertake correlation studies and transform and prepare data for further analysis. Pattern matching is used to find hidden characteristics within data and the methods used to find patterns with the data include association rules [14].

D. Indexing with B-Tree

B-trees enable efficient retrieval of records in the native sort order the index because, in a certain sense, B-trees capture and preserve the result of a sort operation. Moreover, they preserve the sort effort in a representation that can accommodate insertions, deletions, and updates. The relationship between B-trees and sorting can be exploited in many ways; the most common ones are that a sort operation can be avoided if an appropriate B-tree exists and that the most efficient algorithm for B-tree creation eschews random "insert" operations and instead pays the cost of an initial sort for the benefit of efficient "append" operations.

B-tree index can preserve or cache the sort effort. With the output of a sort operation, the B-tree with root, leaf nodes, etc. can be created very efficiently. A subsequent scan can retrieve data sorted without additional sort effort. In addition to preserving the sort effort over an arbitrary length of time, B-trees also permit efficient insertions and deletions, retaining their native sort order and enabling efficient scans in sorted order at any time. Ordered retrieval aids many database operations, in particular subsequent join and grouping operations. This is true if the list of sort keys required in the subsequent operation is precisely equal to or a prefix of that in the B-tree. It turns out, however, that B-trees can save a lot of sort effort in many more cases.

E. Knowledge Discovery Process

Knowledge Discovery in Multimedia Databases is the process of searching for hidden knowledge in the massive amounts of multimedia data that are technically capable of generating and storing. Multimedia Data, in its raw form, is simply a collection of elements, from which little knowledge can be gleaned. With the development of multimedia data discovery techniques the value of the data is significantly improved. A variety of methods are available to assist in extracting patterns that when interpreted provide valuable, possibly previously unknown, insight into the stored multimedia data. This information can be predictive or descriptive in nature. Data mining, the pattern extraction phase of KDD, can take on many forms, the choice dependent on the desired results. KDD is a multi-step process that facilitates the conversion of multimedia data to useful information.

IV. RESULTS AND DISCUSSIONS

A. Binning methods:

Binning methods smooth a sorted data value by consulting the neighborhood", or values around it. The sorted values are distributed into a number of 'buckets', or bins. Because binning methods consult the neighborhood of values, they perform local smoothing values around it. The sorted values are distributed into a number of 'buckets', or bins. Because binning methods consult the neighborhood of values, they perform local smoothing. The statistical data binning is a way to group a number of more or less continuous values into a smaller number of "bins". The auto-binner allows to group numeric data in intervals called bins. There are two naming options for the bins and two methods which define the number and the range of values that fall in a bin. The "Numeric binner" node wants to define custom bins. Use fixed number of bins for bins with equal width over the domain range or bins that have an equal frequency of element occurrences. The smallest element corresponds to a probability of 1 and the largest do probability of 5. The different binning methods for data smoothing are shown in Table 1.(a), Table 1.(b) and Table 1.(c).

Table 1.(a): Binned (equal-frequency) data

Binned Data - 2:31 - Auto-Binner (Apply)
File
Table "default" - Rows: 493 | Spec - Columns: 42 | Properties | Flow Variables

Row ID	class	Video D...	Video Bi...	Maximu...	Width P...	Height ...	Display ...	Bits/(Pi...
Row0	sports	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 5	Bin 1
Row1	News	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1	Bin 5	Bin 1
Row2	News	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1
Row3	Entertainment	Bin 5	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1
Row4	Entertainment	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1	Bin 5	Bin 1
Row5	News	Bin 3	Bin 1	Bin 1	Bin 2	Bin 1	Bin 5	Bin 1
Row6	News	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1
Row7	News	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1
Row8	News	Bin 5	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 2
Row9	News	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1	Bin 2	Bin 1
Row10	News	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1
Row11	News	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1
Row12	News	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1
Row13	News	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1
Row14	News	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1
Row15	News	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1
Row16	News	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1
Row17	News	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1
Row18	News	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1
Row19	Entertainment	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1	Bin 1	Bin 1
Row20	News	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1	Bin 5	Bin 1
Row21	News	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1
Row22	sports	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1
Row23	sports	Bin 3	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1
Row24	Entertainment	Bin 1	Bin 1	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1
Row25	News	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1	Bin 5	Bin 1
Row26	News	Bin 1	Bin 1	Bin 1	Bin 2	Bin 1	Bin 2	Bin 1

Equal frequency binning is a binning method that ensures that the bins contain approximately the same number of records. Equal frequency binning divides a continuous field into k bins. Each bin contains approximately N/k records, where N is the total number of records.

Table 1.(b): Bin boundaries

Binned Data - 0:31 - Auto-Binner (Apply)
File
Table "default" - Rows: 493 | Spec - Columns: 42 | Properties | Flow Variables

Row ID	class	Video D...	Video Bit...	Maximum bi...	Width ...	Height ...
Row135	News	[1.01,12.72]	[194,4,179.4]	[421,2,255.6]	(474,676]	[2.35,729.88]
Row136	sports	(12.72,24.43]	[194,4,179.4]	[421,2,255.6]	(474,676]	[2.35,729.88]
Row137	News	[1.01,12.72]	[194,4,179.4]	[421,2,255.6]	(474,676]	[2.35,729.88]
Row138	sports	(12.72,24.43]	[194,4,179.4]	[421,2,255.6]	(474,676]	[2.35,729.88]
Row139	News	[1.01,12.72]	[194,4,179.4]	[421,2,255.6]	(474,676]	[2.35,729.88]
Row140	News	[1.01,12.72]	[194,4,179.4]	(2,255.6,4,090.2]	(474,676]	[2.35,729.88]
Row141	News	(24.43,36.14]	[194,4,179.4]	[421,2,255.6]	(474,676]	[2.35,729.88]
Row142	News	[1.01,12.72]	[194,4,179.4]	[421,2,255.6]	(474,676]	[2.35,729.88]
Row143	sports	[1.01,12.72]	[194,4,179.4]	[421,2,255.6]	(474,676]	[2.35,729.88]
Row144	News	[1.01,12.72]	[194,4,179.4]	[421,2,255.6]	[272,474]	[2.35,729.88]
Row145	News	[1.01,12.72]	[194,4,179.4]	[421,2,255.6]	(474,676]	[2.35,729.88]
Row146	Entertainment	[1.01,12.72]	[194,4,179.4]	[421,2,255.6]	(474,676]	[2.35,729.88]
Row147	News	[1.01,12.72]	[194,4,179.4]	[421,2,255.6]	(474,676]	[2.35,729.88]
Row148	News	[1.01,12.72]	[194,4,179.4]	[421,2,255.6]	(474,676]	[2.35,729.88]
Row149	News	[1.01,12.72]	[194,4,179.4]	[421,2,255.6]	(474,676]	[2.35,729.88]
Row150	Entertainment	[1.01,12.72]	[194,4,179.4]	[421,2,255.6]	(474,676]	[2.35,729.88]
Row151	News	[1.01,12.72]	[194,4,179.4]	[421,2,255.6]	[272,474]	[2.35,729.88]
Row152	News	[1.01,12.72]	[194,4,179.4]	[421,2,255.6]	(474,676]	[2.35,729.88]

Similarly, smoothing by bin medians can be employed, in which each bin value is replaced by the bin median. In smoothing by bin boundaries, the minimum and maximum values in a given bin are identified as the bin boundaries. Each bin value is then replaced by the closest boundary value. In general, the larger the width, the greater the effect of the smoothing. Alternatively, bins may be equal width, where the interval range of values in each bin is constant.

Table 1.(c): Bin Midpoints

Binned Data - 0:31 - Auto-Binner (Apply)

File

Table "default" - Rows: 493 | Spec - Columns: 42 | Properties | Flow Variables

Row ID	class	Video D...	Video Bi...	Maximu...	Width P...	Height ...	Display ...
Row0	sports	6.865	2,186.7	1,338.3	373	366.115	15.273
Row1	News	6.865	2,186.7	1,338.3	575	366.115	15.273
Row2	News	6.865	2,186.7	1,338.3	373	366.115	5.15
Row3	Entertainment	53.705	2,186.7	1,338.3	373	366.115	5.15
Row4	Entertainment	6.865	2,186.7	1,338.3	575	366.115	15.273
Row5	News	30.285	2,186.7	1,338.3	575	366.115	15.273
Row6	News	6.865	2,186.7	1,338.3	373	366.115	5.15
Row7	News	6.865	2,186.7	1,338.3	373	366.115	5.15
Row8	News	53.705	2,186.7	1,338.3	373	366.115	5.15
Row9	News	6.865	2,186.7	1,338.3	575	366.115	5.15
Row10	News	6.865	2,186.7	1,338.3	373	366.115	5.15
Row11	News	6.865	2,186.7	1,338.3	373	366.115	5.15
Row12	News	6.865	2,186.7	1,338.3	373	366.115	5.15
Row13	News	6.865	2,186.7	1,338.3	373	366.115	5.15
Row14	News	6.865	2,186.7	1,338.3	373	366.115	5.15
Row15	News	6.865	2,186.7	1,338.3	373	366.115	5.15
Row16	News	6.865	2,186.7	1,338.3	373	366.115	5.15
Row17	News	6.865	2,186.7	1,338.3	373	366.115	5.15

The decimal bounds will be converted so that the lower bound of the first interval will be the floor of the lowest value and the upper bound of the last interval will be the ceiling of the highest value. The edges that separate the intervals will be the ceiling of the decimal edges. Duplicates of edges will be removed. The numbered for bins labeled by an integer with prefix "Bin", borders for labels using '"(a, b]"' interval notation or Midpoints for labels that show the midpoint of the interval.

B. Multimedia indexing:

Multimedia indexing has become a general label to designate a large domain of activities ranging from multimedia description to description languages, from speech recognition to ontology definition of course, these fields existed before the expression 'multimedia indexing' became popular, and most continue to have an independent existence. However, the rise of multimedia has forced people to try to mix them together in order to manage properly big collections of multimedia documents. The goal of multimedia indexing is to describe documents automatically, especially those containing images, Audios or videos, allowing users to retrieve them from large collections, or to navigate these collections easily. Each document contains at least one index field which contains the row id. The index might contain more fields depending on the selected table columns to index. The generated index can be searched using the Index query.

A database index is a data structure that improves the speed of data retrieval operations on a database table at the cost of additional writes and storage space to maintain the index data structure. Indexes are used to quickly locate data without having to search every row in a database table every time a database table is accessed. Indexes can be created using one or more columns of a database table, providing the basis for both rapid random lookups and efficient access of ordered records.

Non-index increase the disk storage space as requirement of the database (increases with the number of fields used and the length of the fields). Indexes slow down Insert, Update and Delete, but will speed up Update if the Where condition has an indexed field. Insert, Update and Delete becomes slower because on each operation the indexes must also be updated. Some databases will mono case values in fields that are indexed.

The Index query creates a new data table or filters an optional input table based on the result of the query. If no input table is available the data table is created based on the information that is available in the given index. If the index contains the original data the table will contain a column for each index field that supports the storing of original data. Whereas the table contains only the row id of the matching document if the index does not contain the original data. The table 6 shows various advantages of indexed scheme over non-indexed scheme with respect to various data manipulation operations.

Table.6 advantage of indexed scheme over non-indexed scheme

Sl. No.	Query Type	Query	Actual Data	Indexed Scheme *Record Found*	Non-Indexed Scheme *Record Found*
1	Search	*Video\ Duration :< 1.49> OR NOT Video\ Duration :< 6.13>*	*7*	*7*	*1*
2		*Audio\ Duration:<1.51> OR NOT Audio\ Duration:<1.6>*	*17*	*17*	*2*
5		*Video\ Bit\ rate\ kbps:<194> OR Video\ Bit\ rate\ kbps:<609>*	*108*	*108*	*55*
6		*Maximum\ bit\ rate\ kbps:<586> OR NOT Maximum\ bit\ rate\ kbps:<691>*	*15*	*15*	*1*
12		*Audio\ Bit\ rate\ kbps:<72> OR NOT Audio\ Bit\ rate\ kbps:<86>*	*24*	*24*	*18*

13		*Maximum\ bit\ ratekbps:<75> OR NOT Maximum\ bit\ ratekbps:<77>*	*23*	*23*	*11*
3	*Time efficiency*	*Video\ Duration :< 1.49> OR NOT Video\ Duration :< 6.13>*		*0.5 Sec*	*0.4 Sec*
4		*Audio\ Duration:<1.51> OR NOT Audio\ Duration:<1.6>*		*0.3 Sec*	*0.5 Sec*
7	*Fetch*	*Width\ Pixels:<272> OR NOT Width\ Pixels:<340>*	*25*	*25*	*10*
8		*Height\ Pixels:<256> OR NOT Height\ Pixels:<338>*	*17*	*17*	*15*
9		*Display\ aspect\ ratio:<1.22> OR NOT Display\ aspect\ ratio:<3.2>*	*20*	*20*	*2*
10		*Bits\/\(Pixel*Frame\):<0.028> OR NOT Bits\/\(Pixel*Frame\):<0.057>*	*43*	*43*	*12*
11		*Stream\ size\ \ MiB:<2.19> OR NOT Stream\ size\ \ MiB:<4.91>*	*32*	*32*	*15*
14		*Stream\ size\ MiB:<1.08> OR NOT Stream\ size\ MiB:<1.24>*	*14*	*14*	*1*
15		*Word\ Count:<2> OR NOT Word\ Count:<5>*	*59*	*59*	*62*
16		*Character\ count:<22> OR NOT Character\ count:<34>*	*37*	*38*	*36*
17		*Size\ in\ kbps:<21> OR NOT Size\ in\ kbps:<22>*	*155*	*155*	*68*
18	Sort	*Image\ Resulation:<320x180> OR Image\ Resulation:<640x360>*	*172*	*172*	*88*
19		*class :< entrainment>*	*62*	*62*	*60*
20		*class :< news>*	*100*	*100*	*92*
21		*class :< sports>*	*85*	*85*	*80*

C. Indexing and Non-Indexing Scheme with Search Queries

The OR operator links two terms and finds a matching multimedia data if either of the terms exist in a multimedia data. This is equivalent to union using sets. The symbol ‖ can be used in place of the word OR. To search for multimedia 'video duration' that contains either "1.49" or "6.13" with the query: *Video\ Duration :< 1.49> OR NOT Video\ Duration :< 6.13>, the* B-tree index scheme found 7 out of 7 actual tuples, whereas , without indexing (normal scheme) it has been found only one tuple correctly. The same to search for multimedia 'audio duration' that contains either "1.51" or "1.6" with the query: *Audio\ Duration :< 1.51> OR NOT Audio\ Duration :< 1.6>,* the B-tree index scheme found 17 out of 17 actual tuples, whereas , without indexing it has been found only two tuple. To search for multimedia 'video bit rate kbps' that contains either "194" or "609" with the query: *Video\ Bit\ rate\ kbps:<194> OR Video\ Bit\ rate\ kbps:<609>,* the B-tree index scheme found 108 out of 108 actual tuples, whereas , without indexing (normal scheme) it has been found only 55 tuple correctly. Search for multimedia 'Maximum bit rate kbps' that contains either "586" or "691" with the query: *Maximum\ bit\ rate\ kbps:<586> OR NOT Maximum\ bit\ rate\ kbps:<691>* the B-tree index scheme found 15 out of 15 actual tuples, whereas , without indexing (normal scheme) it has been found only one tuple correctly. In 'Audio bit rate kbps' that contains either "72" or "86" with the query: *Audio\ Bit\ rate\ kbps:<72> OR NOT Audio\ Bit\ rate\ kbps:<86>,* the B-tree index scheme found 24 out of 24 actual tuples, whereas , without indexing it has been found only 18 tuple correctly. Search for multimedia 'Maximum bit rate kbps' that contains either "75" or "77" with the query: *Maximum\ bit\ ratekbps:<75> OR NOT Maximum\ bit\ ratekbps:<77>,* the B-tree index scheme found 15 out of 15 actual tuples, whereas , without indexing it has been found only one tuple correctly.

D. Time efficiency

To search for multimedia 'video duration' that contains either "1.49" or "6.13" with the query: *Video\ Duration :< 1.49> OR NOT Video\ Duration :< 6.13>, the* B-tree index scheme found 7 out of 7 actual tuples found in 0.5 Sec, whereas , without indexing it has been found only one tuple correctly in 0.4 Sec. The same to search for multimedia 'audio duration' that contains either "1.51" or "1.6" with the query: *Audio\ Duration :< 1.51> OR NOT Audio\ Duration :< 1.6>,* the B-tree index scheme found 17 out of 17 actual tuples found in 0.3 Sec, whereas , without indexing it has been found only two tuples correctly found in 0.3 Sec.

E. Indexing and Non-Indexing Schema with Fetch Queries

To fetch for multimedia 'width pixels' that contains either "272" or "340" with the query: *Width\ Pixels:<272> OR NOT Width\ Pixels:<340>* the B-tree index scheme found 25 out of 25 actual tuples, whereas , without indexing it has been found only 10 tuple. To search for multimedia 'height pixels' that contains either "256" or "338" with the query: *Height\ Pixels:<256> OR NOT Height\ Pixels:<338>,* the B-tree index scheme found 17 out of 17 actual tuples, whereas , without indexing it has been found only 15 tuples. In 'display aspect ratio' that contains either "1.22" or "3.2" with the query: *Display\ aspect\ ratio:<1.22> OR NOT Display\ aspect\ ratio:<3.2>,* the B-tree index scheme found 20 out of 20 actual tuples, whereas , without indexing it has been found only two tuple. The same to fetch for multimedia 'bits per pixel' that contains either "0.028" or "0.057" with the query: *Bits\/\(Pixel*Frame\):<0.028> OR NOT Bits\/\(Pixel*Frame\):<0.057>,* the B-tree index scheme found 43 out of 43 actual tuples, whereas , without indexing it has been found only 12 tuples.

In 'stream size in MiB' that contains either "2.19" or "4.91" with the query: *Stream\ size\ \ MiB:<2.19> OR NOT Stream\ size\ \ MiB:<4.91>,* the B-tree index scheme found 32 out of 32 actual tuples, whereas , without indexing it has been found only 15 tuples. In 'word count' that contains either "2" or "5" with the query: *Word\ Count:<2> OR NOT Word\ Count:<5>,* the B-tree index scheme found 59 out of 59 actual tuples, whereas , without indexing it has been found only 62 tuples. In 'character count' that contains either "22" or "34" with the query: *Character\ count:<22> OR NOT Character\ count:<34>,* the B-tree index scheme found 37 out of 38 actual tuples, whereas , without indexing it has been found only 36 tuples. In 'character count' that contains either "21" or "22" with the query: *Size\ in\ kbps:<21> OR NOT Size\ in\ kbps:<22>,* the B-tree index scheme found 155 out of 155 actual tuples, whereas , without indexing it has been found only 68 tuples.

F. Indexing and Non-Indexing Schema with Sort Queries

To sort for multimedia 'image resulation' that contains either "*320x180*" or "*640x360*" with the query: *Image\ Resulation:<320x180> OR Image\ Resulation:<640x360>,* the B-tree index scheme found 172 out of 172 actual tuples, whereas , without indexing it has been found only 88 tuples. To sort for multimedia 'entrainment' that contains either "256" or "338" with the query: *Height\ Pixels:<256> OR NOT Height\ Pixels:<338>,* the B-tree index scheme found 17 out of 17 actual tuples, whereas , without indexing it has been found only 15 tuples. In 'class labels' that contains either "entrainment" use the query: *class :< entrainment >* the B-tree index scheme found 62 out of 62 actual tuples, whereas, without indexing it has been found only 60 tuples. In 'class labels' that contains either "sports" use the query: *class :< sports >* the B-tree index scheme found 85 out of 85 actual tuples, whereas, without indexing it has been found only 80 tuples. The same to search for multimedia 'class labels' that contains either "news" use the query: *class :< news >* the B-tree index scheme found 100 out of 100 actual tuples, whereas, without indexing it has been found only 92 tuples.

V. CONCLUSION

The web multimedia information in diverse kind of data needs efficient operation process in database management such as classification management, multimedia data processing management and data security management. This paper resolves the issue of providing data management support for multimedia data using metadata. The concept of proposed framework, data management can be introduced into multimedia data management such as B-tree indexing, data manipulation, searching and sorting etc, to ensure that event and transaction of multimedia information record can be managed accurately.

REFERENCES

[1] Bhavani Thuraisingham, "Managing and Mining multimedia Database",International Journals on Artificial Intelligence Tools,Vol.13,No.3,739-759,20 March 2004.

[2] Samir Kumar Jalal, "Multimedia Database: Content and Structure", Workshop on Multimedia and Internet Technologies 26th to 28th February, 2001.

[3] Siddu P. Algur[1], Basavaraj A. Goudannavar[2*], Prashant Bhat[3], "Metadata Based Approach for Web Multimedia Model", International Journal of Computer Science Trends and Technology (IJCST) – 4(10): October, 2015, ISSN: 2277-9655

[4] YANG Changchun, YI Li, "A Data Mining Model and Methods Based on Multimedia Database", 978-1-4244-5143-2/10/$26.00 ©2010 IEEE.

[5] S´ebastien Laborie, Ana-Maria Manzat and Florence Sedes, "Managing and querying efficiently distributed semantic multimedia metadata collections", Digital

Object Identifier 10.1109/MMUL.2009.68 1070-986X/$26.00 © 2009 IEEE

[6] Farham Mohamed, M Nordin A Rahman, Yuzarimi M Lazim and Saiful Bahri Mohamed, "Managing Multimedia Data: A Temporal-Based Approach", International Journal of Multimedia and Ubiquitous Engineering Vol. 7, No. 4, October, 2012.

[7] Cristina Ribeiro, Gabriel David, "A Metadata Model for Multimedia Databases", International Journal of Multimedia and Ubiquitous Engineering Vol. 7, No. 4, October, 2012.

[8] Siddu P. Algur[1], Basavaraj A. Goudannavar[2*],"Web Multimedia Mining: Metadata Based Classification and Analysis of Web Multimedia", *International Journal of Advanced Research in Computer Science and Software Engineering* (IJARCSSE) November-2015, pp. 324-330 ISSN: 2277-128X

[9] E. Chavez, J.Marroquin, and G. Navarro, "Overcoming the curse of dimensionality", in Proc. European Workshop on Content-Based Multimedia Indexing (CBMI), 1999, pp. 57-64.

[10] Database Management System And Security Chapter – 2, http://shodhganga.inflibnet.ac.in/bitstream/10603/24846/3/ch.%202%20dbms%20and%20security.pdf

[11] Bhavani Thuraisingham, "Security and privacy for multimedia database management systems", Multimed Tools Appl (2007) 33:13–29, Springer Science + Business Media, LLC 2007.

[12] Dr.S.Vijayarani and Ms. A. Sakila, "Multimedia Mining Research – An Overview Metadata Model for Multimedia Databases", International Journal of Computer Graphics and Animation (IJCGA) Vol. 5, No. 1, January, 2015.

[13] D Jyothsna R. Nayak and Diane J. Cook, "Approximate Association Rule Mining", From: FLAIRS-01 Proceedings 2001, AAAI (www.aaai.org).

[14] S Xinchen, MihaelaVorvoreanu, and Krishna Madhvan, "Mining Social Media Data for Understanding Students' Learning Experiences", IEEE computer society.1939-1382(c)2013 IEEE.

Authors

Dr. Siddu P. Algur is working as Professor, Dept. of Computer Science, Rani Channamma University, Belagavi, Karnataka, India. He received B.E. degree in Electrical and Electronics from Mysore University, Karnataka, India, in 1986. He received his M.E. degree in from NIT, Allahabad, India, in 1991.He obtained Ph.D. degree from the Department of P.G. Studies and Research in Computer Science at Gulbarga University, Gulbarga.

He worked as Lecturer at KLE Society's College of Engineering and Technology and worked as Assistant Professor in the Department of Computer Science and Engineering at SDM College of Engineering and Technology, Dharwad. He was Professor, Dept. of Information Science and Engineering, BVBCET, Hubli, before holding the present position. He was also Director, School of Mathematics and Computing Sciences, RCU, Belagavi. He was also Director, PG Programmes, RCU, Belagavi. His research interest includes Data Mining, Web Mining, Big Data and Information Retrieval from the web and Knowledge discovery techniques. He published more than 65 research papers in peer reviewed International Journals and chaired the sessions in many International conferences.

Mr. Basavaraj A. Goudannavar is pursuing Ph.D programme in Computer Science at Rani Channamma University Belagavi, Karnataka, India. He received BCA and MCA degrees from Karnatak University, Dharwad, Karnataka, India, in 2005 and 2008 respectively. His research interest includes Data Mining, Web Mining, Web multimedia mining, and Knowledge discovery techniques. He published 11 research papers in

International Journals and International conferences He has attended and participated in International and National

