DATA WAREHOUSING AND DATA MINING

Data Warehousing

Overview and concepts:

Definition**: -** A data warehouse is a subject-oriented, integrated, time-varying, non-volatile collection of data in support of the management's decision-making process.

A) Subject-oriented: - A data warehouse is organized around major subjects such as customer, products, sales etc.

B) Non-volatile: - A data warehouse is always a physically separate store of data, which is transformed from the application data found in the appropriate environment.

C) Time-varying: - Data are stored in a data warehouse to provide a historical perspective..

D) Integrated: - A data warehouse is usually constructed by integrating multiple, heterogeneous sources such as relational databases, flat files and OLTP files.

Need for data warehousing**: -** Data warehousing is the process of integrating enterprise-wide corporate data into a single repository. It can be used by the end-users to run queries, make reports and perform analysis.

Basic elements of Data Warehousing: -

Differences between Database System and Data Warehouse: -

1. Database contains records which can be modified, but data warehouse contains only read only data.
2. Data warehouse is maintained separately from an organization's operational database. But operational database (or simply database) is designed for known tasks and workloads, such as indexing and hashing using primary keys, searching etc.

Planning and Requirements:

Project planning and management

Collecting the requirements

Architecture and Infrastructure:

Data Warehouse Architecture and its components: - The architecture of a warehouse consist of three different tiers. The tier-1 is essentially the warehouse server, tier-2 is the OLAP-engine for analytical processing. The tier-3 is a client containing reporting tools, visualization tools, data mining tools, querying tools etc.

There is also the backend process which is concerned with extracting data from multiple operational databases and from external sources, with cleaning, transforming and integrating this data.

Fig-2.12 page-24 (from Data Mining by Arun Pujari)

Infrastructure and metadata: - Meta data serves to identify the contents and location of data in the warehouse. It is a bridge between the data warehouse and the decision support application. It can pin point access to information across the entire data warehouse and can enable the development of applications which automatically update themselves to reflect data warehouse content change.

Type of Metadata: - There are three types of metadata based on their use –

i) Build-Time Metadata: - Whenever the warehouse is designed and built, the metadata that is generated is called build-time metadata. It describes the data's technical structure. It is used extensively by warehouse designers, developers and administrators.

ii) Usage Metadata: - Usage metadata is derived from build-time metadata during production of warehouse. It is an important tool for users and data administrators.

iii) Control Metadata: - This metadata is used by the system programmer to manage the operation of warehouse.

Data design and Data representation:

Principles of dimensional modeling: - Dimension modeling is a special technique for structuring data around business concepts. It structures the numeric measures and the dimensions. The dimension schema can represent the details of the dimensional modeling.

Lattice of Cuboids:- The dimension hierarchy helps us view the multidimensional data in several different data cube representation. Conceptually, multidimensional data can be viewed as a lattice of cuboids. The C [A_1, A_2, ….A_n] at the finest level of granularity is called the base cuboid and it consists of all the data cells. The (n-1)-D cubes are obtained by grouping the cells and computing the combined numeric measures of a given dimension. Finally, the coarsest level consists of one cell with numeric measures of all n dimensions. This is called an apex cuboids.

Advanced Topics

Data warehouse systems use backend tools and utilities to populate and refresh there data. These are

Data extraction: - Data extraction is the process of extracting data for the warehouse from various sources. The data may come from a variety of sources, such as Production data, internal office systems, external systems etc.

Data Transformation: - The sources of data for data warehouse are usually hererogeneous. Data transformation is concerned with transforming hererogeneous data to an uniform structures so that the data can be combined and integrated.

Data loading: - Since a data warehouse integrates time-varying data from multiple sources, the volumes of data to be loaded into a data warehouse can be huge. A loading system should also allow the system administrators to monitor the status, cancel, suspend, resume loading or change the loading rate, and restart loading after failures without any loss of data integrity.

Data quality

Information Access and Delivery:

Matching information to classes of users

OLAP in Data Warehouse: - After modeling the data warehouse, it is necessary to explore the different analytical tools with which to perform the complex analysis of data. These data analysis tools are called On-Line Analytical Processing (OLAP).

OLAP is mainly used to access the live data online and to analyze it. OLAP tools are designed in order to accomplish such analysis on very large databases. It provides a user-friendly environment for interactive data analysis.

The basic OLAP operations for a multidimensional model are:

i) Slicing: - This operation is used for reducing the data cube by one or more dimensions. The slice operation performs a selection on one dimension of the given cube, resulting in a subcube.

ii) Dicing: - This process is for selecting a smaller data cube and analyzing it from different perspective. The dice operation defines a subcube by performing a selection on two or more dimensions.

iii) Drilling: - This operation is meant for moving up and down along classification hierarchies. The different instances of drilling operations may be “Drill-up” and “Drill-Down”, “Drill-within”, “Drill-across” and “Pivot”.

Data warehousing and the web

Implementation and Maintenance:

Physical design process

Data warehouse deployment

Growth and maintenance

Introduction:

Basics of Data Mining: - Data mining is the non-trival process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data. It is the extraction of the hidden predictive information from large database. It is a powerful new technology with great potential to analyze important information in the data warehouse.

Different definitions of Data Mining: -

1. Data mining or knowledge discovery in database, as it is also known, is the non-trival extraction of implicit, previously unknown and potentially useful information from the data. This encompasses a number of technical approaches such as clustering, data sumarisation, classification, finding dependency networks, analyzing changes, and detecting anomalies.
2. Data mining is the search for the relationships and global patterns that exist in large databases but are hidden among vast amounts of data, such as the relationship between patient data and their medical diagnosis. This relationship represents valuable knowledge about the database, and the objects in the database, if the database is a faithful mirror of the real world registered by the database.
3. Data mining refers to using a variety of techniques to identify relationship in between information or decision-making knowledge in the database and extracting these in such a way that they can be put to use in areas such as decision support, prediction, forecasting and estimation. The data is often voluminous, but it has low value and no direct use can be made of it. It is the hidden information in the data that is useful.
4. Discovering relations that connect variables in a database is the subject of data mining. The data mining system self-learns from the previous history of the investigated system, formulating and testing hypothesis about rules which systems obey. When concise and valuable knowledge about the system of interest is discovered, it can and should be

interpreted into some decision support system which helps the manager to make wise and informed business decision.

5. Data mining is the process of discovering meaningful, new correlation patterns and trends by sifting through large amount of data stored in repositories, using pattern recognition techniques as well as statistical and mathematical technique

Related concepts

Data Mining process

Data preparation

Data Cleaning

Data Visualization

KDD Process: - Knowledge Discovery in Database (KDD) is the process of identifying a valid, potentially useful and ultimately understandable structure in data. This process involves selecting or sampling data from a data warehouse, cleaning or preprocessing it, transforming or reducing it, applying a data mining component to produce a structure and then evaluating the derived structure.

The stages of KDD are:

1. Selection: - Selecting or segmenting the data that are relevant to some criteria.
2. Preprocessing: - It is the data cleaning stage where unnecessary information is removed
3. Transformation: - Data is transformed in order to be suitable for the task of data mining.
4. Data Mining: - It is the extraction of patterns from the data.
5. Interpretation and evaluation: - The extracted patterns are converted into knowledge
6. Data Visualization: - Visualization makes it possible for the analyst to gain a deeper, more intuitive understanding of the data.

Data Mining Techniques: - There are two fundamental goals of data mining – prediction and description. Prediction makes use of existing variables in the database in order to predict unknown or future values of interest. Description focuses on finding patterns describing the data and the subsequent presentation for user interpretation.

Another approach of the study of data mining techniques is to classify the techniques as User-guided or verification-driven data mining and Discovery-driven or automatic discovery fo rules.

Discovery model is the system automatically discovering important information hidden in the data. The typical discovery driven tasks are

1. Discovery of association rules
2. Discovery of classification rules
3. Clustering
4. Discovery of frequent episodes
5. Deviation detection

Association rules: An association rule is an expression of the form X => Y, where X and Y are the sets of items. The intuitive meaning of such a rule is that the transaction of the database which contains X tends to contain Y.

Let L be a set of items. Let D, the database, be a set of transactions T. T supports an item x, if x is in T. T is said to support a subset of items X, if T supports each item x in X. X=> Y holds with confidence c, if c% of the transactions in D that support X also support Y.

Support means how often X and Y occurs together as a percentage of the total transactions. Confidence measures how much a particular item is dependent on another.

Clustering: - Clustering is a method of grouping data into different groups so that, the data in each group share similar trends and patterns. It is a method in which we make cluster or group of objects that are some how similar in characteristics. The ultimate aim of the clustering is to provide a grouping of similar records.

In database management, data clustering is a technique in which, the information that is logically similar is physically stored together. In order to increase the efficiency of search and the retrieval in database management, the number of disk accesses is to be minimized. In clustering, since the objects of similar properties are placed in one class of objects, a single access to the disk can retrieve the entire class.

Decision Trees: - A decision tree is a classification scheme which generates a tree and a set of rules, representing the model of different classes, from a given data set. The set of records available for developing classification methods is generally divided into two disjoint subsets – a training set and a test set. The training data is used for deriving the classifier, while the test data is used to measure the accuracy of the classifier.

Clustering

Partition Vs Hierarchical clustering: - There are two main approaches to clustering – hierarchical clustering and partitioning clustering.

Partition clustering techniques partition the database into a predefined number of clusters. The hierarchical clustering techniques do a sequence of partitions, in which each partition is nested into the next partition in the sequence. It creates a hierarchy of clusters from small to big or big to small.

Partitional clustering method: -

1. **K-means method**: - K-means clustering method groups data using a ‘‘top-down’’ approach since it starts with a predefined number of clusters and assigns observations to them.

The process of generating clusters starts by defining the number of groups to create (k). The method then allocates an observation to each of these groups, usually randomly. Next, all other observations are compared to each of these allocated observations and placed in the group they are most similar to. The center point for each of these groups is then calculated. The grouping process continues by determining the distance from all observations to these new group centers. If an observation is closer to the center of another group, it is moved to the group it is closest to. The centers of its old and new groups are now recalculated. The process of comparing and moving observations where appropriate is repeated until there is no further need to move any observations.

2. **K-Medoid**: - In this method, each cluster is represented by one of the objects of the cluster located near the center. Different types of K-Medoid methods are:

a) Partition Around Medoids (PAM): - PAM selects k objects arbitrarily from the data as medoids. Each of these k objects are representatives of k classes. Other objects in the database are classified based on their distances to these k-medoids.

The algorithm starts with arbitrarily selected k-medoids. In each step, a swap between a selected object O_i and a non-selected object O_h is made, as long as such a swap results in an improvement in the quality of clustering. To calculate the effect of such a swap between O_i and O_h a cost C_{ih} is computed.

Advantage: - It is a very robust algorithm. The clusters found by this method do not depend on the order in which the objects are examined.

Disadvantage: - It can not handle very large volumes of data.

b) **CLARA**: - This algorithm draws a sample of the data set, and applies PAM algorithm on this sample to determine the optimal set of medoids from the sample. It then classifies the remaining objects using the partitioning principle. If the sample were drawn in a random way, the medoids of the sample would approximate the medoids of the entire data set.

This algorithm try to reduce the computational effort of PAM.

c) **Clustering Large Applications based on RANdomised Search (CLARANS)**: - This algorithm starts with a randomly selected set of k-medoids like PAM, then selects few pairs of objects for swapping at the current state. It checks at most a pre-specified number of pairs for swapping and, if a pair with negative cost is found, it updates the medoid set and continues. Otherwise, it records the current selection of medoids as a local optimum and restarts with a new randomly selected medoid, set to search for another local optimum.

This algorithm stops after the pre-specified number of local optimal medoid sets are determined, and returns the best among these.

Advantage: - It is more efficient than PAM and CLARA.

Disadvantage: - 1. It assumes that all objects fit into the main memory.

2. Result is very sensitive to the input order of objects

3. It may not find a real local minimum due to the trimming of its searching.

Hierarchical clustering methods: - Hierarchical clustering is of two types

i) Bottom-up (Agglomerative) approach and ii) Top-down (Divisive) approach

1. Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH): - It is an agglomerative hierarchical clustering method. In this type of clustering method, at any given stage, sub-clusters are merged based on some criteria.

BIRCH maintains a set of Cluster Features (CF) of the sub-clusters. The criteria for merging two sub-clusters are so defined that decision to merge two sub-clusters can be taken from the information provided solely by the set of CFs of the respective sub-clusters.

The CF of different sub-clusters is maintained in a tree, called CF Tree.

2. **Clustering Using Representatives (CURE)**: - CURE maintains a set of representative points of each sub-cluster. The representative points are well-scattered with the subclusterso as to properly represent the whole sub-cluster.

At any given stage of the algorithm, we have a set of sub-clusters and associated with each sub-cluster, we have a set of representative points. The distance between two sub-clusters is the smallest pair-wise distance between their representative points. For every cub-clusters, C, its nearest sub-clusters $C_{nearest}$ is computer at this stage as followes:

$$D_{closest} (C) = Distance(C,C_{nearest})$$

The sub-cluster C corresponding to the smallest value of $D_{closest}$ (C) is the candidate subclusters\ to be merged with its nearest sub-cluster $C_{nearest}$ for the next stage.

Once the clusters are merged, a new set of representative points are computerd for the merged cluster. The merging process continues till the prespecified number of clusters are obtained.

Advantage: - It is a sampling-based, hierarchical clustering algorithm that is able to discover clusters of arbitrary shapes.

Disadvantage: - It relies on vector operation and therefore can not apparently cluster data in any distance space.

Density based clustering methods

1. Density Based Spatial Clustering of Applications of Noise (DBSCAN): - The main idea of DBSCAN is that, for each object of a cluster, the neighbourhood of a given radius has to contain at least a minimum number of data objects.

This algorithm maintains the set of objects in three different categories. These are : classified, unclassified and noise. Each classified objects has an associated cluster-id. A noise object may also have an associated dummy cluster-id. The unclassified category of objects does not have any cluster-id. For both classified and noise object, the neighbourhoods are already computed. For unclassified object there no neighbourhood is computed. The algorithm gradually converts an unclassified object into a classified or a noise object.

The steps are

i) take an unclassified object and a new cluster-id associated with it.

ii) Compute neighbourhood and see whether it is dense or not.

iii) If neighbourhood does not exceed the pre-specified number, then it is marked as a noise object. Otherwise, all the objects that are within its neighbourhood are retrived and put into a list of candidate object. These objects may be either unclassified or noise.

iv) If the object is noise then the current cluster-id is assigned to it.

v) If the object is unclassified, then the current cluster-id is assigned to it and it is included in the list of candidate objects for which the neighbourhoods are to be obtained.

The algorithm continues till the list of candidate objects is empty.

Categorical clustering

1. **DBSCAN: -**

RULE MINING

Association rule: - Let A be a set of items. Let T be a set of transaction on a database (A). For a given transaction database T, an association rule is an expression of the form X => Y, where X and Y are subsets of A and X => Y holds with confidence τ, if τ% of transactions in D that support X also support Y. The intuitive meaning of such a rule is that a transaction of the database which contains X tends to contain Y.

Mining Association Rules: - The problem of mining association rules can be decomposed into two sub problems:

1. Find all sets of items (itemsets) whose support is greater than the user-specified minimum support, σ. Such itemsets are called frequent itemsets.
2. Use the frequent itemsets to generate the desired rules

Frequent sets: - Let T be the transaction database and σ be the user-specified minimum support. An itemset X $\subseteq$ A is said to be a frequent itemset in T with respect to σ, if $s(X)_T >= \sigma$

Maximal Frequent Set: - A frequent set is a maximal frequent set if it is a frequent set and no superset of this is a frequent set.

Border Set: - An item set is a border set if it is not a frequent set, but all its proper subsets are frequent sets.

Apriori Algorithm: - It is also called the level-wise algorithm. It was proposed by Agrawal and Srikant in 1994.

The first pass of the algorithm simply counts item occurrence to determine the frequent itemsets. A subsequent pass, say pass k, consists of two phases. First, the frequent itemset L_{k-1} found in the $(k-1)^{th}$ pass are used to generate the candidate item set C_k. Next, the database is

scanned and the support of candidates in C_k is counted. The set of candidate itemsets is subjected to a pruning process to ensure that all the subjects of the candidate sets are already known to be frequent itemsets. The candidate generation process and pruning process are important part of this algorithm.

1. Candidate Generation Process: - Given Lk-1, the set of all frequent (k-1) itemsets, we want to enerate a superset of the set of all frequent k-itemsets. The intuition behing the a priori candidate-generation procedure is that if an itemset X has minimum support, so do all subsets of X.
2. Pruning: - The pruning step eliminates the extensions of (k-1) itemsets which are not found to be frequent, from being considered from counting support.

The a priori frequent itemset discovery algorithm uses these two functions at every iteration. It moves upward in the lattice starting from level 1 till levelk, where no candidate set remains after pruning.

Pincer-Search Algorithm: - The Pincer-search algorithm is based on a bi-directional search. It attempts to find the frequent itemsets in a bottom-up manner but, at the same time, it maintains a list of maximal frequent itemsets. While making a database pass, it also counts the support of these candidate maximal frequent itemsets to see if any one of these is actually frequent. In that event, it can conclude that all the subsets of these frequent sets are going to be frequent and, hence, they are not verified for the support count in the next pass. Sometime we may discover a very large maximal frequent itemsets very early in the algorithm.

Border Algorithm: - The border algorithm maintains support counters for all frequent sets and all border sets. The algorithm works as follows. Initially, the L_{old} and B_{old} are known along with their respective support counts. The algorithm starts by counting the supports of the itemsets of $L_{old} \bigcup B_{old}$ in T_{new}. This requires one pass over the increment portion of the database. During this phase, the algorithm collects two categories of tiemsets : F and B.

The set F contains the itemset of L_{old} which becomes a frequent set in T_{whole}. Set B contains all the border sets whose support count reach the level σ and hence, are promoted border sets. If there is no promoted border, then F contains all the frequent sets of Twhole. But it there is at least one border set that becomes a promoted border, then the algorithm generates candidate sets which are supersets of the promoted border sets. It makes one pass over the database to count the support of these candidate sets.

Generalized Association Rule: - In the most general form, an association rule is of the form $C_1 \Rightarrow C_2$, where C_1, C_2 are conjunctions of conditions and each condition is either $L_i = v$; or $L_i \in$ [low, upper] for each item L_i in A, where v, low and upper are some values in the domain of L_i.

Database in the real world usually have numeric values. Thus it is important to find association rules for numeric and categorical attributes.

Another feature of association rule generators is an item hierarchy. In an association rule time does not play a part. Contrast this with a sequencing rule in which time is important and that might read.

Association rules with item constraints: - Let B be a Boolean expression over the set of itemsets A. We assume, without loss of generality, that B is in disjunctive normal form. That is B is of the form, $D_1 \vee D_2 \vee \ldots\ldots \vee D_m$, where each disjunct D_i is of the form $\alpha_{i1} \wedge \alpha_{i2} \wedge \ldots\ldots \wedge \alpha_{in}$. Each of the α refers either to the presence of an item or to the absence of an item.

The problem of the mining association rule with item-constraint B is to discover all rules that satisfy B and have support and confidence greater than or equal to the user-specified minimum support and minimum confidence, respectively.

We can split the problem into three phases

Phase 1: - find all frequent itemsets that satisfy the Boolean expression, B.

Phase 2: - To generate rules from these frequent itemsets, we need to find the support of all subsets of frequent itemsets that do not satisfy B

Phase 3: - To generate rules from the frequent itemsets found in Phase 1, using the frequent itemsets found in Phase 1 and 2 to compute confidence.

Decision Trees

Tree Construction Principle : - All the decision tree construction methods are based on the principle of recursively partitioning the data set till homogeneity is achieved.

The construction of the decision tree involves the following three main phases:

1. Construction phase: - The initial decision tree is constructed in this phase, based on the entire training data set. It requires recursively partitioning the training set into two or more sub partitions using a splitting criterion, until a stopping criteria is met.

2. Pruning phase: - The tree constructed in the previous phase may not result in the best possible set of rules due to over-fitting. The pruning phase removes some of the lower branches and nodes to improve its performance.
3. Processing the pruned tree: - it is to improve understandability.

Decision tree generation algorithms: -

1. Classification And Regression Tree (CART): - CART builds a binary decision tree by splitting the records at each node, according to a function of a single attribute.

The initial split produces two nodes, each of which we now attempt to split in the same manner as the root. Once again, we examine all the input fields to find the candidate splitters. If no split can be found that significantly decreases the diversity of a given node, we label it as a leaf node. Eventually, only leaf nodes remain and we have grown the full decision tree.

At the end of the tree-growing process, every record of the training set has been assigned to some leaf of the full decision tree. Each leaf can now be assigned a class and an error rate. The error rate of a leaf node is the percentage of incorrect classification at that node. The error rate of an entire decision tree is a weighted sum of the error rates of all the leaves. Each leaf's contribution to the total is the error rate at that leaf multiplied by the probability that a record will end up in there.

2. Iterative Dichotomizer 3 (ID3): - Quinlan introduced the ID3 for constructing the decision trees from data. In ID3, each node corresponds to a splitting attribute and each arc is a possible value of that attribute. At each node the splitting attribute is selected to be the most informative among the attributes not yet considered in the path from the root. The algorithm uses the criterion of information gain to determine the goodness of a split. The attribute with the greatest information gain is taken as the splitting attribute, and the data set is split for all distinct values of the attribute.

Advanced Topics

Web Mining: - We use the web for different purposes which are

a) Finding relevant information
b) Discovering new knowledge from the web
c) Personalized web page synthesis
d) Learning about individual users

Web mining techniques provide a set of techniques that can be used to solve the above problems. Along with these techniques, web mining also use other related techniques from different research areas, such as Database, Information Retrieval, and Natural Language Processing.

Web mining has three operations – Clustering, Associations and Sequential Analysis. The mining techniques in the web can be categorized into three areas

a) Web content mining
b) Web structure mining
c) Web usage mining

a) Web content mining: - Web content mining describes the discovery of useful information from the web contents.

The web contains many kinds of data. We can access government information, digital libraries, business data, web applications etc. Some of the web content data are hidden data, and some are generated dynamically as a result of queries and reside in the DBMS.

Basically, the web content consists of several types of data such as textual, image, audio, video, metadata, as well as hyperlinks. Recent research on mining multi-types of data is termed as multimedia data mining. The textual parts of web content data consists of unstructured data such as free texts, semi-structured data such as HTML documents and more structured data such as data in the tables.

b) Web structure mining: - Web structure mining is concerned with discovering the model underlying the link structures of the web. It is used to study the topology of the hyperlinks. This model can be used to categorize web pages and is useful to generate information such as the similarity and relationship between different web sites.

It can be used to discover authority sites for the subjects and overview sites for the subjects that point to many authorities. It studies the structures of documents with the web itself.

c) Web uses mining: - Web uses mining deals with studying the data generated by the web surfer's sessions or behaviours. It mines the secondary data derived from the interactions of the users with the web. The secondary data includes the data from the web server access logs, proxy server logs, browser logs, user profiles etc.

There are two main approaches in web usage mining driven by the applications of the discoveries – General Access Pattern Tracking and Customized Usage Tracking.

Temporal Mining: - Temporal data mining is an important extension of data mining. It can be defined as the non-trivial extraction of implicit, potentially useful and previously unrecorded information with an implicit or explicit temporal content, from large quantities of data. It has the capability to infer casual and temporal nearness relationships.

Types of temporal data: -

i) Static
ii) Sequences
iii) Timestamped
iv) Fully temporal

Temporal data mining tasks: -

i) Temporal Association
ii) Temporal Classification
iii) Temporal Characterization
iv) Trend Analysis
v) Sequence Analysis

Temporal Association Rules: - Temporal association rules are sometimes viewed as casual rules. It describes relationships, where changes in one even cause subsequent changes in other parts of the domain. They are common targets of scientific investigation within the medical domain, where the search for factors that may cause or aggravate particular disease is important.

Sequence Mining: - An efficient approach to mining casual relations is sequence mining. It is a topic in its own right and many application domains such as DNA sequence, signal processing and speech analysis require mining of sequence data.

The GSP Algorithm: - GSP algorithm makes multiple passes over the database. In the first pass, all single items are counted. From the frequent items, a set of candidate 2-sequences are formed and another pass is made to gather their support. The frequent 2-sequences are used to generate the candidate 3-sequences and this process is repeated until no more frequent sequences are found. There are two main steps in the algorithm - Candidate Generation and Support Counting.

Discovery of frequent episode: - Episodes occur frequently within sequences. Heiki Mannila and his team formulated and devised algorithms for the discovery of frequent episodes.

Episode discovery is similar to sequence mining, but for the following special assumptions:

i) The input sequence is a single long input sequence

ii) The events are typically single item events.

iii) An episode is a subsequence.

The frequent episode discovery problem is to find all episodes that occur frequently in the event sequence within a time window.

Spatial Mining: - Spatial data mining deals with spatial (location, or geo-referenced) data. For example, a map of a state or city contains various natural and man-mad geographic feature and a clusters of points. We can mine varieties of information by identifying likely relationships. For example residential area, market area, jungle etc.

The knowledge discovery tasks involving spatial data include finding characteristic rules, discriminate rules, association rules or deviation and evolution rules etc.