WEB MINING

INTRODUCTION

In recent years the World Wide Web has become a very popular medium of publishing. And also World Wide Web is a popular and interactive medium to circulate information today. Although the World Wide Web is rich with information, gathering and making sense of this data is difficult because publications on the web are unorganized. With huge amount of information available online, the World Wide Web is a fertile area for data mining research. Hence based on this fact we can define ***Web Mining*** as the discovery and analysis useful information from the massive collection of documents available in the web. Web mining is a multidisciplinary research methodology, i.e.; web mining is a combination of different disciplines of computer science like database, information retrieval, artificial intelligence, machine learning and natural language processing.

WEB MINING

Web mining makes use of data mining techniques to automatically discover and extract information from web documents and services. Thus web mining is a combination of two active areas of research, the data mining and the World Wide Web. But in generalwe make use of the web in several ways.

PURPOSES OF WEB MINING

According to *Kosalaetal,* we interact with the web for the following purposes:

1. FINDING RELEVANT INFORMATION

We browse the web or use the search service available on the web to find some information. Usually we specify a simple keyword as query and the response from a web search engine is a list of pages, ranked based on their similarity to the query. There are several web search engine's available on the web nowadays. However, search tools available these days have the following problems:

- ***Low Precision:*** This problem occurs due to the irrelevance of many of the search results. We may get many pages of information which are not relevant to our search keyword or query.
- ***Low Recall:*** This problem occurs due to the inability to index all the information available on the web. Because some of the pages with relevant information are not properly indexed, and when we search using search engines, we may not get those pages through any of the search engines.

2. DISCOVERING NEW KNOWLEDGE FROM THE WEB

This is a query-triggered process (retrieval oriented) and also data-triggered process that presumes that we already have a collection of web data and we want toextract potentially useful knowledge out of it i.e., data mining oriented.

3. PERSONALIZED WEB PAGE SYNTHESIS

Synthesize a web page for different individuals from the available set of web pages. Individuals have their own preferences and style of the contents and presentations while interacting with the web. The information providers like to create a system whichresponds to user queries by potentially aggregating information from several sources in a user dependent manner.

4. LEARNING ABOUT INDIVIDUAL USERS

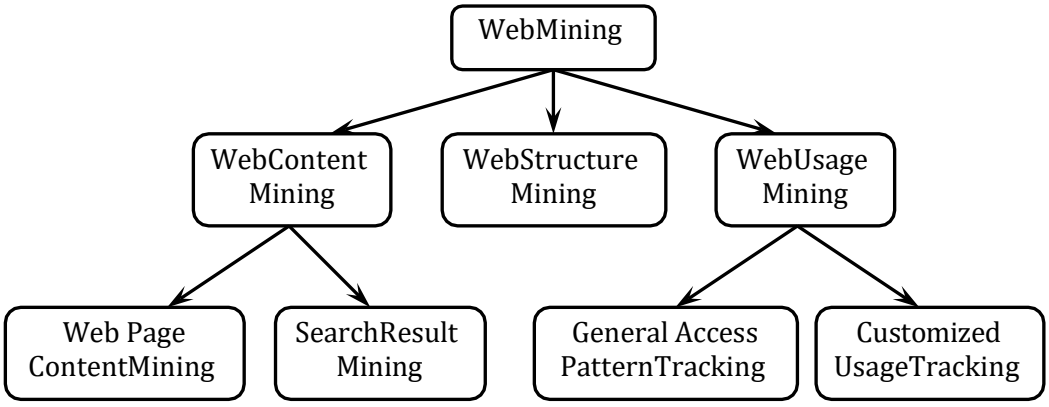
The problem is to know what does the customer do and what they want. This problem can be divided into sub problems, such as mass customizing the information tothe intended consumers or even personalizing it to individual user, problems related to affect a web site design and management, problems related to marketing, etc.

Web mining techniques provide a set of techniques that can be used to solve the above problems. However, there are other tools to handle these problems. Other related techniquesfromdifferentresearchareas,suchasdatabase(DB),informationretrieval(IR), and natural language processing (NLP), can also be used.

WEB MINING CATEGORIES

Web Mining techniques can be categorized into three different categories or types based on which part of the web is to be mined [Madria, 1998]. They are

- WebContentMining.
- WebStructureMining.
- WebUsageMining.



Figure–8.1:Web Mining Categories

WEB CONTENT MINING

- Web content mining describes the discovery of useful information from the web contents. However, what comprise the web contents could encompass a very broad range of data.
- The web contains many kinds of data, like the government information, Digital Libraries; commercial institutions are transforming their businesses and services electronically.
- With the existence of web applications, the users could access the applications through web interfaces. Many applications and systems are being migrated to the web andmany types of applications are emerging in the web environment itself.
- Some of the data contents in the web are hidden data, and some are generated dynamically as a result of queries and reside in the DBMSs. These data are generally not indexed.
- 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 consist of unstructured data such as open texts, semi structured data such as HTML documents, and more structured data such as data in the tables or database generated HTML pages. But, much of the web content data is unstructured, open text data. As a result, the techniques of text mining can be directly employed for web content mining.
- Web content mining techniques are concentrated on the text or a hypertext content which attempts to explore the data content within the structure of the document, i.e.; web content mining attempts to explore the structure within a document (intra-document structure).

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 with or without the description of the links.

This can be used to categorize web pages and is useful to generate information such as the similarity and relationship between different web sites. Web structure mining canbe used to discover authority and hub sites that point to many authorities. Web structure mining studies the structures of documents within the web itself (inter-document structure).

We can view any collection V, of hyperlinked pages as a directed graph $G = (V,E)$, the nodes correspond to the pages, and a directed edge $(p,q) \epsilon E$ indicates the presence of a link from p to$q$. The out-degree of a node p is the number of nodes to which it has links, and the in-degree of p is the number of nodes that have links to it. If$W \subseteq V$is a subset of the pages, we use $G[W]$to denote the graph induces on$W$ – its nodes are the pages in W, and its edges corresponds to all the links between the pages in W.

PAGE RANK

- The importance of a document is measured by counting citations or back links to a givendocument.Thisgivessomeapproximationofadocument'simportanceorquality.
- This concept is extended to web pages. This can be explained as, a page can have a high Page Rank if there are many pages that point to it, or if there are some pages that point to it which have a high Page Rank.
- Thus pages that are well cited from many places around the web are worth looking at. Also, pages that have perhaps only one citation are also generally worth looking at. Page Rank handles both these cases and everything in between, by recursively propagating weights through the link structure of the web.

PAGE RANK IS DEFINED AS FOLLOWS:

We assume page A has pagesT_1,T_2,…,T_n, which point to it (i.e., are citations). The parameter$d$isadampingfactorwhichcanbesetbetween0and1andisusuallysetto 0.85.The$out_deg(A)$denotesthenumberoflinksgoingoutofpageA(outdegreeofA).

DEFINITION:–PAGE RANK

The Page Rank of a page A is given as follows:

$$PR(A) = (1-d) + d\left(\sum_{i=1}^{n} \frac{PR(T_i)}{out\_\deg(T_i)}\right)$$

Here, Let n be the number of documents we have. We define the link matrix M, where the M_{ij}entry is $1/n_j$ifthereisalinkfromdocument$j$todocument$i$,otherwiseM_{ij}is0.And$n_j$isthenumber of the forward link of document j(outdegreeof$j$).Then we can
Compute the Page Rank on the graph which is the domin anteigenvectorofthematrix$A$.

PageRank or$PR(A)$can be calculated using a simple iterative algorithm, and corresponds to the principal eigenvector of the normalized link matrix of the web. Thusthe PageRanks form a sort of a probability distribution over the web pages. PageRank also provides models of user behavior.

Example: Consider a surfer starting from a web page at random and who keeps clickingonlinks,neverhitting"back",buteventuallybegetsboredandswitchestoanother random page. The probability that the surfer visits a page is its PageRank. The damping factor *d* is to model the probability that at each page the surfer would get bored and request another random page.

SOCIAL NETWORK

- Social network analysis is yet another way of studying the web link structure. It uses an exponentially varying damping factor.
- Web structure mining utilizes the hyperlinks structure of the web to apply social network analysis, to model the underlying links structure of the web itself.
- The social network tries to measure the importance of individuals in a network. The same process can be used to study the link structures of the web pages.
- The basic principle here is that, if a web page 'A' points a link to another web page 'B', then 'B' has some undenying importance with respect to 'A'. The links in such a network may have different weights, corresponding to the strength of their relationship.

TRANSVERSE AND INTRINSIC LINKS

- A heuristic method of giving weightage to links was introduced by Kleinberg. According to this method we can identify a links as either transverse or intrinsic.
- A link is said to be a transverse link, if it is between pages with different domain names. And link is said to be an intrinsic link if it is between pages with the same domain name.
- Here, by "domain name", we mean the first level in the URL string associated with a page. Since intrinsic links exist purely to navigate the site, they convey much less information than transverse links of the pages to which they point. Thus, while computing the Page Rank or standing of a page, the intrinsic links need not be takeninto account.
- In this method Kleinberg proposes to delete all intrinsic links from the graph, keeping only the edges corresponding to transverse links. This is a very simple but effective heuristic logic.

REFERENCE NODES AND INDEX NODES

This is another approach for ranking pages. It was proposed by Botafogo. In this approach pages are ranked by defining index nodes and reference nodes.

DEFINITION:–INDEX NODE

An index node is a node whose out–degree is significantly larger than the average out–degree of the graph.

DEFINITION:–REFERENCE NODE

A reference node is a node whose in–degree is significantly larger than the average in–degree of the graph.

CLUSTERING AND DETERMINING SIMILAR PAGES

In this approach for ranking pages, we need to determine the collection of similar pages. To determine the collection of similar pages, we need to define the similarity measure between pages. There can be two basic similarity functions.

DEFINITION:–BIBLIOGRAPHIC COUPLING

For a pair of nodes *p* and *q,* the bibliographic coupling is equal to the number of nodes that have links from both *p* and *q*.

DEFINITION:–CO-CITATION

For a pair of nodes *p*&*q*, theco-citationisthenumberofnodesthatpointtoboth*p*and*q*.

Let us consider a simple graph for clustering the web documents based on the web structure. In the first step; we first identify the influential pages that are referred by substantially large number of pages. The next step is to create a soft cluster around each of the influential pages based on citation count. The pages are assigned to the soft cluster if they are co-cited along with the influential page. After creating those soft clusters, in the next step the similarity between the soft clusters are calculated and based on the values of the similarity measure, certain soft clusters are merged in the hierarchical agglomerative clustering principle.

In order to determine the set of influential pages, a threshold value λ has to be defined to identify the nodes whose degree exceeds this value. In order to build a soft cluster around a node ν, we identify all other nodes *x*, such that there is at least one node *y* that cites both *x* and ν. For this we can use the *bibliographic* and *co-citationcouplings*. The similarity measure between two sub clusters $S_x$and $S_y$is computed as:

$$\frac{|S_x \cap S_y|}{|S_x \cup S_y|}$$

WEB USAGE MINING

- Web usage mining deals with studying the data generated by the web surfer's sessions or behaviors. The web content and structure mining utilize the real or primary data on the web.

- Web usage mining also 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, registration data, user sessions or transactions, cookies, user queries, bookmark data, mouse clicks and scrolls, and any other data which are the results of these interactions.
- This secondary data can be accumulated by the web server. Analysis of this secondary data can facilitate an understanding of the user behavior and the web structure,thereby improving the design of this huge collection of information.

Therearetwomainapproachesinwebusageminingdrivenbytheapplicationsofthe discoveries.

GENERAL ACCESS PATTERN TRACKING

- This approach of web usage mining is to learn user navigation patterns.
- It analyzes the weblogs to understand access patterns and trends.
- These analyses can provide better information on the structure and grouping of resource providers.

CUSTOMIZED USAGE TRACKING

- This approach of web usage mining is to learn a user profile or user modeling in adaptive interfaces. It analyzes individual user trends. Its purpose is to customize web sites to users.
- The information displayed, the depth of the site structure, and the format of the resources can all be dynamically customized for each user over time, based on their access patterns.
- It is important to note that the success of such applications depends on what and how much valid and reliable knowledge one can discover from the large, raw log data.

The web usage mining techniques can be classified into two approaches:

- The first approachmaps the usage data ofthe web serverinto relational tables before a traditional data mining technique is performed. The typical data mining methods such as clustering and classification could be used to mine the usage data after the data have been pre-processed to the desired form.
- The second approach uses the log data directly by utilizing special preprocessing techniques. The web usage data can also be represented with graphs. Often, the web usage mining uses some background or domain knowledge, such as navigation templates, web content, site topology, concept hierarchies, and syntactic constraints.

TEXT MINING

Due to the continuous growth of the volumes of text data, potentially useful,implicit, previously unknown information from this huge source of knowledge should be properly utilized. And also the automated extraction of information from this huge source of knowledge should be properly updated.

Text mining is the extension of the data mining approach to textual data and is concerned with various tasks, such as extraction of information implicitly contained in collection of documents, or similarity-based structuring.

Text collection lacks the imposed structure of a traditional database. The text expresses a vast range of information, but encodes the information in a form that isdifficult to interpret automatically. Identifying individual items or terms is not easy in a textual data. Thus, unstructured data or free-running text requires specific techniques called text mining techniques to process the unstructured textual data to support in knowledge discovery.

The nature of unstructured textual data motivates the development of separate text mining techniques. These are

- One approach is to impose a structure on the textual database and use any data mining techniques for structured databases.
- The other approach would be to develop a technique for mining that exploits the inherent characteristics of textual databases.

Irrespective of the approach chosen for text mining, there are other relatedsubjects, which also interact with textual data, such as computational linguistics, natural language processing, and information retrieval.

OTHER RELATED AREAS

It is very essential to analyze the relationship between other text data mining relatedareaslikeinformationretrieval(IR),informationextraction(IE)andcomputational linguistics.

INFORMATION RETRIEVAL

IR is concerned with finding and ranking documents that match the user's information needs. A body of text is analyzed by its constituent words, and various techniques are used to build the core words for a document. The major goals of IR are

- Tofinddocumentsthataresimilar,basedonsomespecificationoftheuser.
- To find the right index terms in a collection, so that querying will return theappropriate document.

IR is the automatic retrieval of all relevant documents while at the same time retrieving as few of the non-relevant ones as possible. IR has the primary goals of indexing the text and searching for useful documents in a collection. Recent trends in IR include modeling, document classification and categorization, user interfaces, data visualization, filtering, etc.

The main problem with IR is to know what is currently of interest to the user. It is the process of finding pattern or of exploration.

INFORMATION EXTRACTION

IE has the goal of transforming a collection of documents into information that is more readily digested and analyzed. Usually IE performs this transformation with the help of an IR system. IE extracts relevant facts from the documents, while IR selects relevant documents.

Therefore IE works at a finer granularity level than IR does on the documents. Most IEsystemsusemachinelearningordataminingtechniquestolearntheextractionpatterns or rules for documents semi-automatically or automatically.

The result of the IE process is in the form of a structured database, or acompression or summary of the original text or documents. IE can also be used to improve the indexing process, which is part of the IR process.

COMPUTATIONAL LINGUISTICS

Computational linguistics computes statistics over large text collections in order to discover useful patterns. These patterns are used to inform algorithms for various sub–problems within natural language processing, such as part-Of-speech tagging, word-sense disambiguation, etc.

UNSTRUCTURED TEXT

Unstructured documents are free texts, such as news stories. Usually most of research uses bags of words to represent unstructured documents and extract different features from it.

For an unstructured document, features are extracted to convert it to a structured form. Some of the important features for extraction are listed below:

WORD OCCURRENCES

- Thesetofwordstakesinglewordsfoundinthetrainingcorpusasfeaturesignoringthe sequence in which the words occur.
- This representation is based on the value of single words in isolation. Such a feature is said to be Boolean, if we consider whether a word occurs or not in a document.

- The feature is said to be frequency based if the frequency of the word in a document is taken into consideration.

STOP–WORDS

The feature selection includes removing the case, punctuation, infrequent words, and stop words. Good examples for the set of stop-words are: eg, a, the, an, etc.

LATENT SEMANTIC INDEXING

Latent Semantic Indexing(LSI)transforms the original document vectors to a lower dimensional space by analyzing the correlational structure of terms in the document collection, such that similar documents that do not share terms are placed in the same topic.

STEMMING

Stemming is a process which reduces words to their morphological roots. For example, the words "informing", "information", "informer", and "informed" would be stemmed to their common root "inform'', and only the latter word is used as the feature instead of the former four.

n–GRAM

Other feature representations are also possible, such as using information about word positions in the document, or using n-grams representation (word sequences of length up ton).

PART–OF–SPEECH(POS)

One important feature is the POS. There can be 25 possible values for POS tags.Most common tags are noun, verb, adjective and adverb. Thus, we can assign a number 1, 2,3, 4or 5, dependingonwhetherthewordis anoun,verb,adjective,adverborany other, respectively.

POSITIONAL COLLOCATIONS

The values of this type of feature are the words that occur in one or two position to the right or left of the given word.

HIGHER ORDER FEATURES

Other features include phrases, document concept categories, terms, hypernyms, named entities, dates, email addresses, locations, organizations, or URLs. These features could be reduced further by applying some other feature selection techniques, such as information gain, mutual information, cross entropy, or odds ratio.

Once the features are extracted, the text is represented as structured data, and traditional data mining techniques can be used. The techniques include discovering frequent sets, frequent sequences and episode rules.

EPISODE RULE DISCOVERY FOR TEXTS

Ahonen proposed to apply sequence mining techniques for text data. In this technique, text is considered as sequential data which consists of a sequence of pairs known as feature vector with index. Where the feature vector is an ordered set of features and the index contains information about the position of the word in the sequence. A feature can be any of the textual features described above.

Define a text episode as a pair$\propto = (V, \leq)$, where Vis a collection of feature vectors and ≤ is a partial order on V.

Given a text sequence S, a text episode $\propto = (V, \leq)$ occurs within S if there is a way of satisfying the feature vectors in V, using the feature vectors in S so that the partial order ≤ is respected. In other words, the feature vectors of V can be found within Sin an order that satisfies ≤.

The support of α in S is defined as the number of minimal occurrences of α in S.With this, the episode discovery technique of sequence mining can be used to discover frequent episodes in a text.

HIERARCHY OF CATEGORIES

When a user enters a query into a search engine, the system brings back many different pages. Then we need to organize the documents into meaningful groups. There are many ways in which we can show how a set of documents are related to one another.

One way is to group together all documents written by the same author, or all documents written in the same year, or published by the same publisher. We can group them according to subject matter as well. Libraries organize some of their information this way, using classification systems like the Dewey Decimal.

A problem with assigning documents to single categories within a hierarchy is that, most documents discuss several different topics simultaneously. A better solution is to describe documents by a set of categories as well as attributes such as source, date, genre, and author which provide good interfaces for manipulating these labels.

For this purpose, Feldman proposed an elegant data structure of concept hierarchy. Concept hierarchy is a directed acyclic graph of concepts, where each of the concepts is identified by a unique name. An arc from concept *A* to *B* denotes that *A* is a more general concept than *B*. Each text document is tagged by a set of concepts that correspond to its content.

Tagging a document with a concept implicitly entails its tagging with all the ancestors of the concept hierarchy. Therefore a document should be tagged with thelowest concepts possible. The method to automatically tag the document to the hierarchyis a top-down approach. An evaluation function determines whether a document currently tagged to a node can also be tagged to any of itschild nodes. If so, thenthe tag moves down the hierarchy till it cannot be moved any further.

The outcome of this process is a hierarchy of documents and, at each node, there isa set of documents having a common concept associated with the node. The hierarchy of documents resulting from the tagging process is useful for text mining process. It is assumed that the hierarchy of concepts is known a priori. We can even have such a hierarchy of documents without a concept hierarchy, by using any hierarchical clustering algorithm which results in such a hierarchy.

Popescul posed a related problem of tagging key words to the set of documents arranged in a hierarchy. The method is a two-phase principle. It starts with a bag of key words at the leaf level and moves up the hierarchy. The set of key words for a non–leaf node is obtained by combining all the key words to all its child nodes.After finding the set of key words for the root node, the process starts with a top-down approach. If a key word at any node is also equally probable for all of its child nodes, then the key word is associated with the current (parent) node and not with any of the child nodes. Otherwise,if the key word is more probable for a child node, it is moved down to the most probable set of child nodes.

TEXT CLUSTERING

- Text clustering is another important task of text mining. Once the features of an unstructured text are identified or the structured data of the text is available, text clustering can be done by employing any of the clustering techniques.
- One popular text clustering algorithm is ***Ward's Minimum Variance method***.It is an agglomerative hierarchical clustering technique and it tends to generate very compact clusters.
- In this method, either the Euclidean metric or Hamming distance metric is used as the measure of dissimilarities between feature vectors.
- The clustering method begins with n clusters, one for each text. At any stage, two clusters are merged to generate a new cluster. The clusters $C_k$and $C_l$are merged to geta new cluster C_{kl}based on the following criterion:

$$V_{kl} = MIN_{i,j} V_{ij}$$

$$V_{ij} = \frac{\|x_i - x_j\|^2}{\frac{1}{n_1} + \frac{1}{n_2}}$$

Where $x_i$is the mean value ofthe dissimilarityforthe cluster$C_i$and$n_i$is the number of elements in this cluster.

SCATTER/GATHER

- It is a method of grouping the documents using clustering. The scatter/gather uses text clustering to group documents according to their overall similarities in their content.
- Scatter/gather allows the user to scatter documents into clusters or groups, and then gather a subset of these groups and re-scatter them to form new groups.
- Each cluster in scatter/gather is represented by a list of words from the cluster that attempt to give the user the idea of what the documents in the cluster are about.
- If a cluster has too many documents, the user can re–cluster the documents in the cluster and re-group that subset of documents into still smaller groups.
- This re-grouping process tends to change the kinds of themes of the clusters, because the documents in a sub collection discuss a different set of topics than all thedocuments in the larger collection.

TEMPORAL AND SPATIAL DATA MINING

INTRODUCTION

Many applications maintain temporal and spatial features in their databases; these features cannot be treated as any other attributes and need special attention. To put it in another way, so far we have been asking ourselves 'what' knowledge is being mined, but finding 'when' and 'where' knowledge are also equally important, and these cannot be trivially handled by the methods discussed thus far. In this chapter, we shall study the mining techniques of temporal data and spatial data. The widespread use of GIS by local and federal governments and other institutions, necessitate the development of adequate mining tools for geo-referenced data. Perhaps, the second generation of data mining techniques would contribute in a major way to spatial and temporal data mining.

WHAT IS TEMPORAL DATA MINING?

Temporal Data Mining is an important extension of data mining and it can be defined as the non-trivial extraction of implicit, potentially useful and previously unrecordedinformationwithanimplicitorexplicittemporalcontent,fromlargequantities of data. It has the capability to infer causal and temporal proximity relationships, and thisis something that non-temporal data mining cannot do. It may be noted that data mining from temporal data is not temporal data mining, if the temporal component is either ignored or treated as a simple numerical attribute. Also note that temporal rules cannot be mined from a database which is free of temporal components by traditional (non-temporal) data mining techniques. Thus, the underlying database must be a temporal one and specific temporal data mining techniques are also necessary.

Thus, temporal data mining aims at mining new and hitherto unknown knowledge, which takes into account the temporal aspects of the data. Let us first concentrate on the different temporal aspects of the data.

TYPES OF TEMPORAL DATA

There can be four different levels of temporality. Based on these we can say that the datacontains temporal features or not.

STATIC

Staticdataarefreeofanytemporalreferenceandtheinferencesthatcanbederived from this data are also free of any temporality.

SEQUENCES(ORDERED SEQUENCES OF EVENTS)

In this category of data, though there may not be any explicit reference to time, there exists a sort of qualitative temporal relationship between data items. The market-basket transaction is a good example of this category. The entry-sequence of transactions

automatically incorporates a sort of temporality. If a transaction appears in the data- base before another transaction, it implies that the former transaction has occurred before the latter. There may not be any reference to quantitative temporal relationships. While most collections are often limited to the sequence relationships *before* and *after*, this category also includes the richer relationships, such as *during*, *meet*, *overlap*, etc. Such relationships are called qualitative relationships between time events. Sequence mining is one of the major activities in temporal data mining.

TIME STAMPED

In this category the temporal information is explicit. Note that the relationship can be quantitative, in the sense that we can not only say that one transaction occurred before another, but also the exact temporal distance between the data elements. Some examples include census data, land-use data and satellite meteorological data. The inference made here can be temporal or non-temporal.Time series data are a special case of this category, with the events being uniformly spaced on the time scale. Time series data mining is another topic that we shall be discussing in this chapter.

FULLY TEMPORAL

In this category, the validity of the data element is time-dependent.The inferences are necessarily temporal in such cases.

TEMPORAL DATA MINING TASKS

Some of the conventional mining tasks can be extended with some additional temporal information as described below.

TEMPORAL ASSOCIATION

The association rule discovery can be extended to temporal association. In static association rule discovery tasks, we were trying to find static associations between two non-temporal itemsets. In the temporal association discovery, we attempt to discover temporal association between non-temporal itemsets. We can say that: "70% of thereaders who buy a DBMS book also buy a Data Mining book *after* a semester".

TEMPORAL CLASSIFICATION

We can cluster the data items along temporal dimensions. For example, we can identify a set of people who go for a walk in the evening and a set of people who go for a walk in the morning. We can categorize sets of patients based on their visit sequence to different medical experts. We can also categorize sets of net surfers based on the mouse-click sequence.

TEMPORAL CHARACTERIZATION

An interesting experiment would be to extend the concept of decision tree construction on temporal attributes. For example, a rule could be: "The first case of filariais normally reported after the first pre-monsoon rain and during the months of May- August".

TREND ANALYSIS

Theanalysisofoneormoretimeseriesofcontinuousdatamayshowsimilartrends, i.e., similar shapes across the time axis. For example, "The deployment of the Data Mining system is increasingly becoming popular in the banking industry". These types of analyses are of a higher level than the earlier ones. Here, we are trying to find the relationships of change in one or more static attributes, with respect to changes in the temporal attributes.

SEQUENCE ANALYSIS

Events occurring at different points in time may be related by causal relationships, in that an earlier event may appear to cause a later one. To discover such relationships, sequences of events must be analysed to discover common patterns. This categoryincludes the discovery of frequent events and also the problem of event prediction. It may be noted that frequent sequence mining finds the frequent subsequences; while event prediction predicts the occurrences of events which are rare.

TEMPORAL ASSOCIATION RULES

Association rules identify whether a particular subset of items are supported by an adequate number of transactions. They are normally static in character. As we have mentioned earlier, a static association rule discovers the association between any two events-for instance, the purchase of aerated soft drinks and stomach upsets. However, the association may not indicate any causal relationship, unless the temporality in the association is brought out. One can extend the association rule discovery to incorporate temporal aspects too. It should be noted that the presence of a temporal association rule may suggest a number of interpretations, such as

- Theearliereventplayssomeroleincausingthelater event.
- Thereisathirdsetofreasonsthatcausebothevents.
- Theconfluenceofeventsis coincidental.

Temporal association rules are sometimes viewed in the literature as causal rules. Causal rules describe relationships, where changes in one event cause subsequent changes inotherpartsofthedomain.Theyarecommontargetsofscientificinvestigationwithinthe medical domain, where the search for factors that may cause or aggravate particular disease is important. The static properties, such as gender, and the temporal properties, such as medical treatments, are taken into account during mining.

While the concept of association rule discovery is the same for temporal and non-temporal rules, algorithms designed for conventional rules cannot be directly applied to extract temporal rules. The reason is that classical association rules have no notion of order, whilst time implies an ordering.

SEQUENCE MINING

An efficient approach to mining causal relations is sequence mining. As observed earlier, sequence mining 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, even though there is no explicit temporality in the data. Discovering sequential patterns from a large database of sequences has been recognized as an important problem in the field of knowledge discovery and data mining. To put it briefly, given a set of data sequences, the problem is to discover subsequence's that are frequent, in the sense thatthe percentage of data sequences containing them exceeds a user-specified minimum support.

Mining frequent sequential patterns has found a host of potential application domains, including retailing (i.e., market-basket data), telecommunications and, more recently, the World Wide Web (WWW). In market-basket databases, each data sequence corresponds to items bought by an individual customer over time, and frequent patterns can be useful for predicting future customer behaviour. In telecommunications, frequent sequences of alarms output by network switches capture important relationships between alarm signals that can then be employed for on-line prediction, analysis, and correction of network faults. In the context of the WWW, server sites typically generate huge volumes of daily log data capturing the sequences of page accesses for thousands or millions of users. Let us begin with formally defining the sequence mining problem.

SEQUENCE MINING PROBLEM

The most general form of the sequence mining problem [Zaki, 1998] can be statedas follows:

Let $\sum=\{i_1,i_2,....,i_m\}$beasetofmdistinctitemscomprising thealphabet.

An event is a non-empty, disordered collection of items. Without any loss of generality, we write the items in an event in some predefined order.An event is denoted as$\{ i_1,i_2,....,i_k\}$, where $i_j$isanitemin Σ .Often,wedropthe','andparenthesesfor notational convenience.

Any event that is given as input will also be called a transaction. Thus, transactions and events have the same structure, except that a transaction is known to us prior to the process and an event is generated during the algorithm. We use both terms interchangeably if there is no ambiguity.

DEFINITION:–SEQUENCE

Asequenceisanorderedlistofevents.Asequence,$\propto$isdenotedas($\propto_1 \rightarrow \propto_2 \rightarrow \cdots \rightarrow \propto_q$), where $\propto_i$is an event. A sequence is called a k-sequence, if the sum of the cardinalities of $\propto_i$is k.

A subsequence is a sequence within a sequence, preserving the order.In other words, its items need not be adjacent in time but their ordering in a sequence should not violate the time ordering of the supporting events. A subsequence can be obtained from a sequence by deleting some items and/or events. A formal definition of a subsequence is given below.

DEFINITION:–FREQUENCY

The frequency of a sequence s, with respect to this database D, is the total numberof input sequences in D that support it.

DEFINITION:–FREQUENT SEQUENCE

A frequent sequence is a sequence whose frequency exceeds some user-specified threshold. A frequent set is *maximal* if it is not a subsequence of another frequentsequence.

The rationale behind frequent sequences lies in detecting precedence and causal relationships that make them statistically remarkable.

EXAMPLE11.1

Letusconsiderthefollowingtransactiondatabase.Therearesixitems,that is,$\Sigma = \{A,B,C,D,E,F\}$,andthedatabasedepictsthepurchasesmadeby4customers during a certain period of time. The transactions are ordered chronologically. Table 11.2 extracts one sequence.

Table11.1:ASampleofSequenceData

Customer	A	B	C	D	E	F
C1	1	1	0	1	1	1
C2	0	0	1	1	0	1
C1	0	1	1	0	1	1
C3	0	0	0	1	1	1
C2	1	1	1	1	0	1
C1	0	1	0	0	1	0
C3	1	0	1	1	1	0
C2	0	1	0	0	1	0
C4	1	1	1	1	1	1
C1	0	0	1	1	1	1
C4	0	1	0	0	0	1
C3	1	0	1	1	1	1
C2	0	1	1	0	0	0
C4	1	0	1	1	1	1
C2	0	1	0	0	0	0

Table11.2:OneSequencefromtheSequenceDataofTable11.

Customer	A	B	C	D	E	F
C1	1	1	0	1	1	1
C1	0	1	1	0	1	1
C1	0	1	0	0	1	0
C1	0	0	1	1	1	1

Wewritethetransactionsequencesofallcustomersinthefollowingform,thereby indicating the items a transaction contains.

Sequence1:(A, B, D, E,F)→(B, C,E,F)→(B, E)→(C, D, E,F)

Sequence2:(C, D, F)→(A, B, C, D, F)→(B,E)→(B, C)→(B)

Sequence3:(D, E,F)→(A, C, D, E)→(A, C,D,E,F)

Sequence4:(A, B, C,D,E, F)→(B,F)→(A, C, D, E,F)

Thus,thedatabaseofsequencesDconsistsoffour sequences.

Please note that AC (it is, in fact, an abbreviation of (A, C)) is not a subsequence of Sequence 1, but itis a subsequence of Sequence 3 and also of Sequence 4. But the sequence A →C is a subsequence of Sequence 1, whereas it is not a subsequence of Sequence 3. By AC, we mean that there should be a transaction containing both A and C. By A →C, we mean that there is a transaction containing C which appears after - not necessarily, immediately after - another (different) transaction containing A.

The frequency of AC in D is 3. Note that we do not count multiple occurrences of AC in the same sequence. The total numberofsequences that support ACis onlythree, namely Sequence 2, Sequence 3 and Sequence 4. Similarly, the frequency of B →D is 2. The sequences supporting B →D are Sequence 1 and Sequence 4.

Note that B →BE is not supported by Sequence 4, which supports BE →B. The subsequence B →BE indicates that a transaction containing B and E follows (sometime later) another transaction containing B. On the other hand, BE →B represents a subsequence of transactions, in which a transaction containing B and E occurs before a transaction containing B. Both are 3-sequences.

Normally, a simple sequence mining problem is concerned with the temporal order of the events within a sequence of transactions. A more general problem is when we also focus on the temporal distance between the events. That is, A →C should indicate the temporal gap between the transaction containing A and the transaction containing C. It is relevant when the correlation between the two events ceases to be effective after some period of time. For example, when we are trying to find out the causal relationshipbetween consuming a beverage and having a stomach upset, it will be irrelevant to correlate two such events occurring within a gap of one year of each other. Thus, we can specify the time distance in terms of a distance threshold, d. So, B →d BE denotes that the event containing B and E occurs in not more than d transactions after the transaction containing B.

Forexample,thereisnosequencewhichsupportsABD→1D,whereasABD→2Dis supported by Sequence 4. When d = 1, we say that the problem is a contiguous sequence mining problem.

A simple sequence mining problem is the sequence mining problem where each transaction contains a single item. There are many applications in which simple sequence mining is relevant.

THEGSPALGORITHM

The algorithms for solving sequence mining problems are mostly based on the Apriori (level-wise) algorithm. One way to use the level-wise paradigm is to first discover all the frequent items in a level-wise fashion. It simply means counting the occurrences of all singleton elements in the database. Then, the transactions are filtered by removing the non-frequent items. At the end of this step, each transaction is a modified transaction consisting of only the frequent elements it contains. We use this modified database as an input to the GSP algorithm. This process requires one pass over the whole database.

GSP makes multiple passes over the database. In the first pass, all single items (1–sequences) 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

Given the set of frequent (*k–1*)–frequent sequences $F_{(k-1)}$, the candidates for thenext pass are generated by joining $F_{(k-1)}$with itself. A pruning phase eliminates any sequence, at least one of whose subsequences is not frequent.

SUPPORT COUNTING

Normally,ahashtree-basedsearchisemployedforefficientsupportcounting. Finallynon-maximalfrequentsequencesare removed.

GSP ALGORITHM PSEUDOCODE

F_1 = the set of frequent 1-sequence
$k = 2$,
do while $F_{k-1} \neq \emptyset$;
    generate candidate sets C_k (Set of candidate k-sequences);
    for all input sequence s in the database ***D do***
    increment count of all a in C_k if s supports a
    $F_k = \{a \in C_k$ such that its frequency exceeds the threshold$\}$
$k = k + 1$
set of all frequent sequences is the union of all F_ks
    end do.

The above algorithm looks like the a priori algorithm. One main difference is however the generation of candidate sets. Let us assume that A →B and A →C are two frequent 2-sequences. The items involved in these sequences are (A, B) and (A, C), respectively. The candidate generation in the usual a priori style would give (A, B, C) as a3-itemset, but in thepresent context we get the following3-sequences as a result ofjoining the above 2-sequences.

A→B→C,A→C→BandA→BC.

Thecandidate-generationphasetakesthisintoaccount.

TheGSPalgorithmdiscoversfrequentsequences, allowingfortimeconstraintssuch as maximum gap and minimum gap, among the sequence elements. Moreover, it supports the notion of a sliding window, i.e., of a time interval within which items are observed as belonging to the same event, even if they originate from different events.

EPISODE DISCOVERY

Another important temporal data mining problem is the discovery of episodes that occur frequently within sequences. Heiki Mannila and his team formulated and devised algorithms for the discovery of frequent episodes. Zaki's formulation of sequence mining, given above, is general enough to have episode mining problem as its special case. Keeping the above definition in mind, we shall now formulate the episode discovery problem. Episode discoveryis similartosequence mining, but forthe following specialassumptions:

- The input sequence is a single long input sequence, unlike in the case of sequencemining where we have a set of data sequences.
- Theevents(inthiscontext,referringtoatransactionasaneventismore appropriate) are typically single item events.
- Anepisodeisasubsequence.

The frequent episode discoveryproblemis to findall episodes that occurfrequently in the event sequence within a time window. Let us define some basic concepts that are necessary in the present context.

DEFINITION:–EVENT

An event is a pair{*A,t*},where Ais a single itemevent, and*t* isan integertimestamp of the occurrence of *A*.

DEFINITION:–EVENT SEQUENCE

An event sequence *Ev_Seq* is a triplet (*Seq,T_start,T_end*), with *T_start* and *T_end* denoting the start and end time of the sequence; and *Seq* = ({A_1, t_1}, {A_2, t_2}, ... {A_k, t_k}) is an ordered sequence of events.

DEFINITION:–TIME WINDOW

A time window W for *Ev_Seq*(*Seq, T_start, T_end*), is an event sequence (*W, t_s,t_e*),where$t_s \geq t_e$start and $t_e \leq$T_end and ($t_e - t_s$) is said to be width of the window.

Weshallrepresentthe dataintheformofa graph,whereeacheventcorrespondsto a node. The precedence relationships among nodes represent the temporal precedence among events. Given a set of nodes and a set of events, it is necessary to specify the mapping to identify the correspondence between nodes and events.

DEFINITION:–EPISODE

An episode *a* is a triple (*V* ≤*g*), where V is a set of nodes, ≤ is a partial order on *V*, and g: *V* →*I* is a mapping associating each node with an event satisfying the partial order.

DEFINITION:–PARALLELANDSERIALEPISODES

Ifthe partialorder ≤isa trivial partial order, the episodeis calledaparallel episode. Ifthepartialorderisatotalordering,thentheepisodeiscalledserial episode.

An episode A →B →C is a serial episode. A serial episode occurs in a given sequence only if A, B and C occur in this order relatively close. There can be other events occurring between these three. In a parallel episode, there are no constraints on the relative order of the events. In general, we can have an episode which has some partial constraints on order too.

DEFINITION:–SUB EPISODE

An episode is to be a subepisode if it is obtained by deleting some events from an episode.

DEFINITION:–OCCURRENCE OF AN EPISODE IN AN EVENT SEQUENCE

An episode is said to be occurring in a sequence if the events corresponding to the nodes of the episode appear in the sequencing, preserving the partial order of the episode.

DEFINITION:–FREQUENCY OF AN EPISODE

The frequencyofan episode, withrespect to a given windowwidthin a sequence, is the fraction of all the windows in the sequence of the specified width in which the episode occurs. Let us assume that$W(\omega)$ is the set of total number of time windows of width ω in thegivensequence,and$W(\omega,\alpha)$ isthesetofwindowsin $W(\omega)$ inwhichtheepisode αoccurs.Then,the frequency is theratioof$W(\omega,\alpha)$to $W(\omega)$.

Definition: A frequent episode is the episode that has a frequency above a user-specifiedthreshold.

The episode discovery problem can be stated as follows. Given a sequence *Ev_seq*, a class *C* of episodes, a window width ω, and a frequency thresholdσ it is to find all frequent episodes with respect to ω and σ in *Ev_seq*.

EVENT PREDICTION PROBLEM

The discovery of frequent sequences is inappropriate for many applications where sequence pattern discovery is relevant. Consider, for instance, error discovery which is an importantapplicationdomain.Sinceerrorsare rareevents,thestatisticalsupportofsucha sequence is low. Hence, if we restrict the search space to frequent sequences, it will behard to find rare events. On the other 'hand , if we reduce the threshold above which a sequence is considered frequent, it will be practically impossible to inspect the result and distinguish interesting patterns from any trivial sequence. The problem of predicting failure(oranyotherevent)differs fromsequencepredictionproblemsinthatthe data(i.e., events) are timestamped, and differs from time-series prediction problems in thatthe data consists mainly of categorical, non-numerical transactions.

THE EVENT PREDICTION PROBLEM

The event prediction problem is to predict a type of future event, the *target* event, based on past events. More specifically, the problem is to find a prediction rule that successfully predicts future target events by taking the input as timestamped records with categorical items. This problem is particularly interesting in situations where the target event occurs infrequently in the event sequence. When we try to predict an event, we also keep in mind the target event's *prediction period*, which means it must occur at least '*warningtime*' time units before the target event and no more than '*monitoringtime*' time unit s before the target event. It is interesting to note that in event prediction problems a target event is said to be correctly predicted if at least one prediction is made within its predict ion period, regardless of any subsequent negative predictions. Thus, the reliability of a positive prediction is not affected by the presence of negative predictions.

Every event is represented by a tuple consisting of a timestamp field and a number of features. We introduce a wildcard in the event as '?'. Each feature in an event is permitted to take on any of a predefined list of valid feature values, as well as the wildcard ("?") value, which matches any feature value. For example, if the events in a domain are described by three features, then the event <*?, ?, b*> would match any event in which the third feature has the value "*b*".

TIME-SERIES ANALYSIS

Time series are an important class of complex data objects; they arise in many applications. For example, stock price indices, volume of product sales, telecommunication data, one-dimensional medical signals, audio data, and environmental measurement sequences are all time-series databases. Time-series data are sequences of real numbers representing measurements at uniformly-spaced temporal instances. Time-series analysis, like all other forms of data analysis, is used to characterize or explain the reasons for the behaviour of a system and/or to predict its future behaviour.

Time-series analysis is a well-studied subject in statistics and signal processing. We shall focus on the basic time-series analyses in the context of data mining. Note that the most important aspects of time-series data are that these are sequence data but uniformly spacedon the temporalattribute. Analysis tasks oftime series include feature extraction of time-series data; computation of similarity measure among time series data set; segmentation of data set; matching two time series data; clustering and classifying time-series data. We shall introduce the related concepts first.

DEFINITION:–N-SERIES

An n-series X is a sequence$\{x_1,x_2,....,x_n\}$ of real numbers. Each n-series X has an average $\alpha(X)$ and a deviation $\sigma(X)$:

$$\alpha(X)=\frac{1}{n}\sum_{i=1}^{n}x_i \quad and \sigma(X)=\left(\frac{1}{n}\sum_{i=1}^{n}(x_i-\alpha(X))^2\right)^{\frac{1}{2}}$$

SPATIAL MINING

The immense explosion in geographically referenced data occasioned by developments in IT, digital mapping, remote sensing, and the global diffusion of GIS, emphasizes the importance ofdeveloping data miningapproaches to geographical analysis and modelling to aid the processes of scientific discovery. Indeed a number of data mining tools are beingdevelopedto assist the processofexploringlarge amounts ofdata in search of recurrent patterns and relationships.

Spatial data mining is the branch of data mining that deals with spatial (location, or geo-referenced) data. Consider a map of the city of Hyderabad containing various natural and man-made geographic features, and clusters of points (where each point marks the location of a particular house). The houses might be noteworthy because of their size, historical interest, or their current market value. Clustering algorithms exist to assign each point to exactly one cluster, with the number of clusters being defined by the user. We can

mine varieties of information by identifying likely relationships. For example, "the land-value of the cluster of residential area to the east of 'Cyber-Tower' is high" or, "70% the Banjara migrants settle in the city around the market area''. Such information could be of value to realtors, investors, or prospectivehome buyers, andalso to other domains such as satellite images, photographs, and oil and gas explorations. This problem is not trivial-there may be a large number of features to consider. We need to be able to detect relationships among large numbers of geo-referenced objects without incurring significant overheads.

As in the case of temporal data mining, conventional data mining techniques cannot fullyexploitthespatialcharacteristicsofdata.Itisnecessarytodevisealgorithmsthattake this aspect into consideration.