A Privacy Preserving Repository For Data Integration Across Data Sharing Services

Smita R. Kapoor
S.A.T.I.College of Engineering, Vidisha.

Prof. Mr. Vismay Jain
S.A.T.I.College of Engineering, Vidisha.

Dr. R. C. Jain
Director, S.A.T.I. College of Engineering, Vidisha.

Abstract**—***Data sources and data warehouses expect to sign business agreements in which the scope of the shared data and corresponding privacy policies are specified. Existing data sharing and integration systems are usually implemented as centralized data warehouses collecting and storing data from various data sources. The most significant problem of existing data sharing and integration solutions is that they give data warehouses too much power, which may not be needed for data sharing. Our repository collects data from data sharing services based on users' integration requirements rather than all the data from the data sharing services as existing central authorities. 2) While existing central authorities have full control of the collected data, the capability of our repository is restricted to computing the integration results required by users and cannot get other information about the data or use it for other purposes. 3) The data collected by our repository cannot be used to generate other results except that of the specified data integration request, and, hence, the compromise of our repository can only reveal the results of the specified data integration request, while the compromise of central authorities will reveal all data.*

Index Terms**—***Context aware data sharing, Data Integration, Multiparty computation, Privacy concerns of service-oriented solutions, privacy management in data collection, privacy preserving data mining, services composition*

IJERT

1. Introduction

Data sources and data warehouses expect to sign business agreements in which the scope of the shared data and corresponding privacy policies are specified. MUCH effort has been devoted to facilitating data sharing and integration among various organizations. However, the development of such systems is hindered by the lack of robust and flexible techniques to protect the privacy of the shared data. Existing data sharing and integration systems are usually implemented as centralized data warehouses collecting and storing data from various data sources. The most significant problem of existing data sharing and integration solutions is that they give data warehouses too much power, which may not be needed for data sharing. For instance, a hospital may be asked to share its patients' social security numbers (SSNs) because they are used to locate patients' records from various hospitals. Unfortunately, SSNs can also be used for other purposes, such as checking patients' credit histories. But, when SSNs are only used as keys to link records from various hospitals, the SSNs can be replaced by their hash values without affecting their functionality as keys [1].

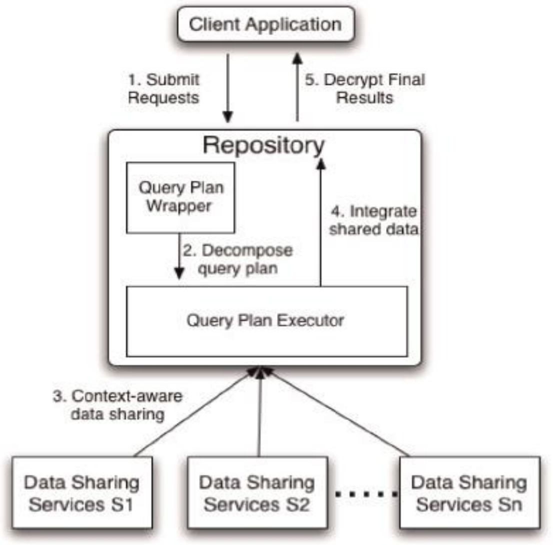

Fig. 1 Our privacy preserving repository for data integration across data sharing services.

Privacy preserving data mining is one of the most demanding research areas within the data mining community. In many cases, multiple parties may wish to share aggregate private data, without leaking any sensitive information at their end. For example, different superstores with sensitive sales data may wish to coordinate among themselves in knowing aggregate trends without leaking the trends of their individual stores. This requires secure protocols for sharing the information across the different parties. The data may be distributed in two ways across different sites: Horizontal partition and Vertical partition. Horizontal partition means, where different sites have different sets of records containing the same attributes. Vertical partition means, where different sites have different attributes of the same sets of records [2].privacy preserving data mining method on the decision tree over horizontally partitioned data using UTP.

A. Decision Tree Classification

Classification is often seen as the most useful form of data mining. Decision trees are the most useful, popular and powerful tools for classification and prediction. This may be because they form rules which are easy to understand, or perhaps because they can be converted easily into SQL. While not as “robust” as neural networks and not as statistically “tidy” as discriminate analysis, decision tree often show very good generalization capability.

Decision trees are built by choosing an attribute and a value for that attribute which splits the dataset. The attribute and value are chosen to minimize diversity of class label in the two resulting sets (an alternative way of looking at this is to maximize information gain or to minimize entropy). The first split is unlikely to be perfect, so we recursively split the sets created until all the sets we have consist of only one class. Creating the decision tree is simply a matter of collating the splits in the correct order. The trick in data mining (where we may be dealing with large datasets; possibly even too big to fit into memory) is to find that attribute and value with the minimum number of passes through the database.

B. Secure Protocols with an Un-trusted Third Party

A straightforward solution for privacy preserving data mining is to use a trusted third party to gather data from all data sources and then send back results after running the desired data mining algorithms. However, the level of trust is not acceptable in this scheme since the privacy of the data sources cannot be protected from the third party. There have been several approaches to support privacy preserving data mining over multiple data bases without using third parties [3, 4]. The existence of an Un-trusted Third Party (UTP) enables efficient protocols without revealing private information. The idea of an UTP is that it is willing to perform some computation for the parties in the protocol. It is not trusted with the data or the results. The trust placed in this party is that it does not join with any of the participating parties to violate information privacy and correctly executes the protocol.

1.1 Privacy preserving data mining

In recent years privacy preserving data mining has emerged as a very active research area in data mining. The application possibilities of data mining, combined with the Internet, have attracted and inspired many scientists from different research areas such as computer science, bioinformatics and economics, to actively participate in this relatively young field. Over the last few years this has naturally lead to a growing interest in security or privacy issues in data mining. More precisely, it became clear that discovering knowledge through a combination of different databases raises important security issues. Despite the fact that a centralized warehouse approach allows discovering knowledge, which would have not emerged when the sites were mined individually, privacy of data cannot be guaranteed in the context of data warehousing [5].

IJERT

Although data mining results usually do not violate privacy of individuals, it cannot be assured that an unauthorized person will not access the centralized warehouse with some malevolent intentions to misuse gathered information for his own purposes during the data mining process. Neither can it be guaranteed that, when data is partitioned over different sites and data is not encrypted, it is impossible to derive new knowledge about the other sites. Data mining techniques try to identify regularities in data, which are unknown and hard to discover by individuals. Regularities or patterns are to be understood as revelations over the entire data, rather than on individuals. However to find such revelations the mining process has to access and use individual information.

More formally, this problem is recognized as the inference problem [6]. Originally this problem dates back to research in database theory during the 70s and early 80s, acknowledged back then as access control. Models were developed offering protection against unauthorized use or access of the database. However, such models seemed unable to sufficiently protect sensitive information. More precisely, indirect accesses (through a different database and metadata) still allowed one to attain information not authorized for. Here, metadata consists, e.g., of dependencies between different databases, integrity constraints, domain knowledge, etc. In other words, the inference problem occurs when one can obtain vital information through metadata violating individuals (or companies) privacy. With the elaboration of different network technologies and growing interest in pattern recognition this problem naturally carries over to data mining. In fact it gets even worse as illustrated by Sweeny in [7]. In her work Sweeny shows that the typical de-identification technique applied on data sets for public use, does not render the result anonymous. More precisely, she demonstrated that combinations of characteristics (or attributes) can construct a unique or near-unique identifier of tuples, which means that information can be gained on individuals even when their identifiers are distorted.

Over the past few years state of the art research in privacy preserving data mining has concentrated itself along two major lines: data which is horizontally distributed and data which is vertically distributed. Horizontally partitioned data is data which is homogeneously distributed, meaning that all data tuples yield over the same item or feature set. Essentially this boils down to different data sites collecting the same kind of information over different individuals. Consider for instance a supermarket chain which gathers information on the buying behavior of its customers. Typically, such a company has different branches, implying data to be horizontally distributed. Vertically distributed data is data which is heterogeneously distributed. Basically this means that data is collected by different sites or parties on the same individuals but with differing item or feature sets. Consider for instance financial institutions as banks and credit card companies, they both collect data on customers having a credit card but with differing item sets.

There has been a tremendous growth in the area of data management during the last few years. The growth seems to be in the direction of data integration for providing accurate, timely, and useful information. Data warehousing is playing a major role in the integration process. A data warehouse is a subject-oriented, integrated, time-variant, and nonvolatile collection of data in support of management’s decision-making process [8]. Data warehousing has become so popular in the industry that it is cited as being the highest-priority post millennium project of more than half of the information technology (IT) executives. A recent study conducted by the Meta Group found that 95% of the companies surveyed intended to build a data warehouse.

Construction of a data warehouse is generally based on a data warehousing process (DWP) methodology [9]. Currently, there are over 30 such methodologies available in the data warehouse market. The main reason for having so many vendor-specific methodologies is the lack of any centralized efforts at creating platform-independent standards for DWP. Such platform independent and vendor-neutral standards depend only on the semantics of DWP and are not influenced by any vendor or by any specific application. The development of such standards is complicated and involves many steps. Those steps include developing an understanding of the existing methodologies, conducting a careful study of semantics of DWP tasks, development of a DWP data model, and development of a vendor-neutral and platform-independent DWP language. The objective of this research is to focus on the first step and review the current DWP standard practices based on an ontological examination of existing data warehousing methodologies.

Ontology is the study of the entities, both abstract and concrete, that make up the world. It is a formal and explicit specification of a shared conceptualization. Ontology has found widespread use in the analysis and modeling of applications, including knowledge management, content management, and knowledge portal technologies [10].

There is a growing body of ontology-based research in a variety of application areas in information systems, such as knowledge engineering, database design and integration, and information retrieval and extraction. Ontology has been used to derive precise rules for the use of entities, relationships, and attributes in entity-relationship (ER) diagrams, and to define precise rules for classes and

IJERT

class hierarchies. Ontology based modeling has been applied to database design and for mapping unified modeling language (UML) constructs to well defined ontological concepts . Ontology has been used with concept theory and speech act theory for conceptual modeling in systems development. It has also been used for corporate web applications and for semantic interoperability of metadata [11].

2. RELATED WORK

In 2008 by Stephen S. Yau, et. al gives a concept about warehouse for integrating data from various data sharing services without central authorities is existing with our warehouse, data sharing services can update and control the access and limit the usage of their shared data, as a substitute of submitting data to establishment, and our repository will support data sharing and addition. The main differences between our storehouse and existing central authorities are: 1) our repository collects data from data sharing services based on users' integration requirements rather than all the data from the data sharing services as existing central establishment. 2) While existing central establishment have full control of the collected data, the capability of our warehouse is controlled to computing the integration results required by users and cannot get other information about the data or use it for other work. 3) The data composed by our warehouse cannot be used to generate other results except that of the specified data addition request, and, hence, the cooperation of our warehouse can only reveal the results of the specified data integration demand, while the compromise of central establishment will reveal all dat and presented a privacy preserving repository to integrate data from various data distribution services. In contrast to existing data allocation techniques, our warehouse only collects the least amount of information from data sharing services based on users' integration requirements, and data distribution services can check our repository to use their shared information only for users' integration requests, but not other purposes [1].

Author develop a warehouse to facilitate information integration across data sharing services and gift the method of the information integration via our privacy protective repository REP. the method may be summarized as follows: Step one. The user sends his/her public key pk and also the needs concerning information integration to our repository REP. Step 2. The question set up wrapper of REP analyzes the user's integration needs and converts them to a question set up graph G, then decomposes G to a group of subgraphs fG1;G2; nine nine nine ;Gmg victimisation the Decompose formula and sends the sub graphs to the question set up fiduciary. Each subgraph Gi represents the context of 1 information sharing service for conducting context aware information sharing. Step 3. for each Gi, the question set up fiduciary appearance for the corresponding information sharing service Si and sends Gi to Si, that prepares the information victimisation the Context-Aware information Sharing formula (Algorithm two, Section 7) and returns all randomised information to the question set up fiduciary . Step 4. The question set up fiduciary executes the Integrate formula (Algorithm three, Section 8) on all came back information to execute the G and outputs the results FinalRes of user's request that is encrypted with the user's public key pk. Step 5. REP sends FinalRes to the user United Nations agency then decrypts it with his/her secret key sk [1].

2.1 Searchable Encryption Schemes

In [12] a symmetric searchable encryption scheme and an asymmetric searchable encryption scheme are proposed to store users' data in a third party. These schemes hide users' data from the third party and enable the third party to match data with users' searching requests and return the matched data to users, in the direction of satisfy these two apparently clashing requirements, both introduce additional private information (i.e., a symmetric key and an asymmetric key in) to manipulate the original data or its hash values. All of these approaches only focus on how to control the third party's search capability among two parties. Because some addition applications in our repository require data from more than two data sharing services, our warehouse may need to combine multiple data sets provided by various data sharing services.

2.2 Privacy Preserving Query Processing

The design of efficient privacy preserving query processing techniques [13] There basic idea of these approaches is to execute queries on cryptographically or non cryptography manipulated data. even if the assumptions and goals of these approaches vary very much, all of them bear as of two shortcomings: 1) Existing techniques only include the evaluation for one query and do not consider the role of the query's output in the complete application. 2) They also do not reflect on the inferential relations among different queries in one request. These shortcomings create them inappropriate be used in a complex data integration application that needs to process a set of queries in a given partial order.

IJERT

2.3 Secure Multiparty Computation

In addition obtainable privacy preserving query processing techniques, a technique named secure multiparty calculation [14] can handle any data combination requirements. Generally speaking, any data mixing application can be modeled as a multiparty function that accepts inputs from data distribution services and only releases the final answer to the user. On the other hand, it needs to characterize functions as corrupted circuits, which typically require huge numbers of gates and, hence, introduce excessive transparency.

In 2012 by Alka Gangrade and Ravindra Patel gives the concept about the two layer protocol uses an Un-trusted Third Party (UTP) and explain how to build privacy preserving two-layer decision tree classifier, where database is horizontally partitioned and communicate their intermediate results to the UTP not their private data. In this protocol, an UTP allows well-designed solutions that meet privacy constraint and achieve suitable performance and finally proposed a new classifier using two-layer architecture that enables SMC by hiding the identity of the parties attractive part in the classification process using UTP. Further we may describe that intermediate result is calculated by every party individually and send only intermediate result to UTP not the input data. During the communication among UTP and all party final result is carried out. It requires less memory space. Also provides fast and easy calculations. Using this protocol, classification will almost secure and privacy of individual will be maintained. Additional development of the protocol is estimated in the sense that for joining multi-party attributes using a trusted third party can be used [2].

The performance of privacy preserving techniques should be analysed and compared in terms of both the privacy protection of individual data and the predictive accuracy of the constructed classifiers. Inside this proposed scheme used the different types of parameter they are:

A. Architecture

1) Input Layer*:* Input layer comprises of all the parties that are involved in the computation process. All participating party individually calculates the Information Gain of each attribute and sends Information Gain as an intermediate result form to the UTP. This process is done at every stage of decision tree. *2)* Output Layer*:* The UTP exists at the 2nd layer i.e. the calculation layer of our protocol. UTP collects only midway outcome i.e. in order Gain of all attributes from all parties not data and calculate the total information gain of each feature. Then discover the attribute with maximum information gain and then create the root of decision tree with this attribute and send this attribute to all parties for further calculation. This process is also done at every stage of decision tree. 4. Create the root with largest Information Gain attribute and edges with their values, and then send this attribute to all parties at Input Layer for further development of decision tree.

B. Assumptions

The follow assumptions contain be locate:

1. UTP computes the final result from the intermediate results provided by all parties at every stage of decision tree.
2. UTP computes attribute with highest information gain and send to all party at every stage of decision tree.
3. UTP has the ability to announce the final result of the computation publicly.
4. Each party is not communicating their input data to other party.
5. The communication networks used by the input parties to communicate with the UTP are secure.

In 2008 by Bart Kuijpers et al. [5] proposed the complexity analysis, in which case the earlier evaluation method is the more efficient and give an algorithm for privacy preserving ID3 over horizontally partitioned data involving more than two parties. For grid partitioned data, here discuss two different evaluation methods for preserving privacy ID3, that is, first merging horizontally and increasing vertically or first merging vertically and next developing horizontally with the help of these concept the complexity analysis of both algorithms shows that it is more efficient to first merge data horizontally and further develop it vertically than the other way around.

IJERT

Privacy preserving ID3: Grid partitioned data

Here consider privacy as protecting individual data tuples as well as protecting attributes and values of attributes. So each party will reveal as little as possible about its data while still constructing an applicable distributed assessment tree. The only article that is known with reference to the tree by all parties is its structure and which party is responsible for each decision node. More precisely, which party possesses the attribute used to make the decision, except not which characteristic .Here assume that only a narrow number of parties know the class attribute and no party knows the entire set of attributes, which is obvious as we use grid partitioned data [5].

Grid Partitioned data

The grid partitioned privacy preserving algorithm by running through the different steps of ID3 easily. It is important to realize that no site knows the complete attribute set S and only a limited number of parties know the class attribute, additional on the whole as much as there are horizontal distributions. make a note of these algorithms only consider the cases for which the parties are denoted by Pij with i = 1, .., v, j = 1, .., h.[5].

2013 by Melanie Herschel and Ioana Manolescu proposed the Data Integration for Digital cities",with the help of these plan is to create, link, combine and develop open data in the Digital Cities data freedom, for the profit of both society and administrations. In that context, Data Bridges addresses many research challenges such as acquiring (or producing) Open City information in RDF, data connecting and combination, data origin, and visualization. Provide a brief introduction to EIT ICT Labs, its goals and structures, and how Data Bridges fits them and focuses on fostering exchange and new result creation in the sphere of Information and Communication tools, transversely the areas of explore higher instruction, and industrial innovation. Author invests further efforts in making these methods scale to large volumes of data. Possible avenues of research include extending our work to the cloud or defining new algorithms and tools, e.g., for fully automatic linking, quality assessment, and visualization [16].

In 2012 by CIHAN VAROL et. al. provides the construct concerning personal info from phone, World Wide net, or email so as to sell or send a poster concerning their product. However, once this info is uninheritable, moved, copied, or edited, the info might lose its quality. Often, the employment of data directors or a tool that has restricted capabilities to correct the mistyped information will cause several issues. Moreover, most of the correction techniques square measure significantly enforced for the words employed in daily conversations. Since personal names have completely different characteristics compared to general text, a hybrid matching formula (PNRS) that employs phonetic coding, string matching and applied math facts to produce a attainable candidate for misspelled names is developed. At the end, the potency of the projected formula is compared with different acknowledge orthography correction techniques [17].

IJERT

PNRS overcame a number of the deficiencies the opposite algorithms have once providing an answer for ill-defined information. First, the rule introduced in RNMS provided fewer inapplicable results (between thirteen and twenty first less) compared to different approximate string matching techniques utilized in dictionary-based search. Second, current techniques heavily accept dictionary-based search wherever they use one among the pattern matching algorithms to supply a suggestion. However, combining the 2 matching techniques improved the general correction rate, since phonetic ways address phonetic kinds of errors additional accurately than string-matching algorithms. during this study, since the phonetic strategy of the rule is especially designed for English-based personal names, different language phonetic structures are going to be addressed within the close to future. Another future goal is to use the techniques not solely to personal names, however additionally to addresses and different personal info still.

3.PERSONAL NAME RECOGNIZING STRATEGY

Personal Name Recognizing Strategy (PNRS) is predicated on the results of variety of methods that ar combined so as to supply the nearest match (Figure 1).

3.1 String Matching Technique—Restricted Near Miss Strategy-

The restricted mischance strategy (RNMS) may be a fairly straightforward thanks to generate suggestions. Two records ar thought-about close to (t = 1), if they will be created identical by inserting a space, interchanging 2 adjacent letters, dynamic one letter, deleting one letter, or adding one letter

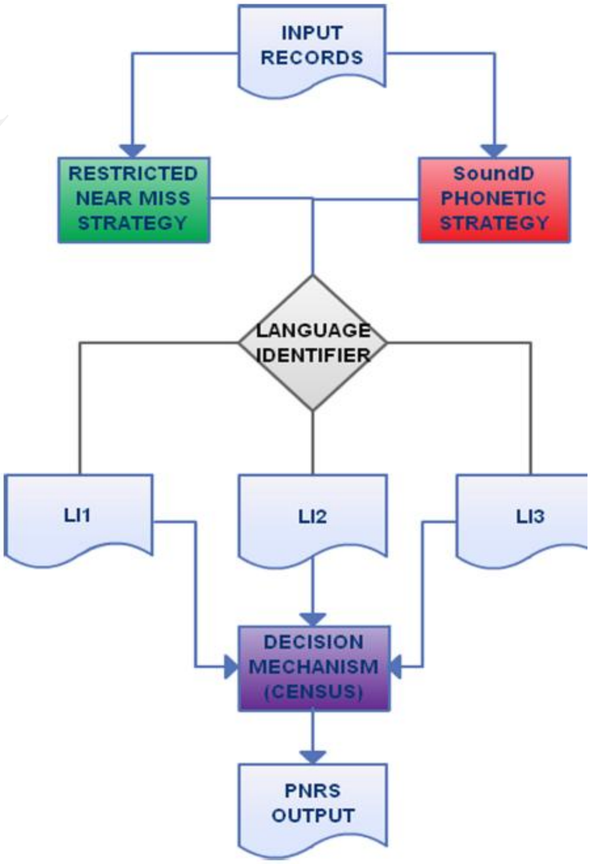

Fig. 1. PNR Sstrategy.

3.2 Phonetic Matching Technique—SoundD Phonetic Strategy

A phonetic code could be a rough approximation of however the word sounds [Cohen et al. 2003]. the English communication could be a actually phonetic code, which suggests every sound during a word is portrayed by a logo or sound image. Since Soundex has fewer phonetic transformation rules (rules area unit significantly designed for the English language) compared to the opposite algorithms, it provides a lot of logical matches for international scope of names among different phonetic ways [Varol and Bayrak 2009]. Therefore, we've got used a variation of the Soundex algorithmic rule jointly of our suggestion algorithms. As mentioned earlier, a serious drawback with Soundex is that it keeps the primary letter, therefore any error at the start of a reputation leads to a distinct Soundex code. This causes valid candidates to be eliminated thanks to the error within the initial letter. Therefore, we've got changed the Soundex algorithmic rule with initial letter parsing rules (Table I). Then, the algorithmic rule converts all the letters into numbers in line with Table I. aside from the primary letter, all zeros (vowels, h, w, and y) area unit removed and sequences of a similar range area unit reduced to at least one (e.g. 222 is replaced with 2). the ultimate code could be a four digit range (longer codes area unit interrupt, and shorter codes area unit extended with zeros). As Associate in Nursing example, the Soundex code for martha is 5630. The applied phonetic strategy compares the phonetic code of the misspelled word to any or all the words within the thesaurus. If the phonetic codes match or area unit inside one edit distance of the first word, then the word is value-added to the temporary suggestion list.

3.3 Language Identifier

Since the input file could contain a global scope of names, it's troublesome to standardize the phonetic equivalents of sure letters. Therefore, the input file is parsed with a language symbol to mirror whether or not the information contains English-based names. Some letters or letter combos in an exceedingly name will enable the language to be determined [Fung and Schultz 2008].
for instance
—tsch, final mann, and witz ar specifically German;
—g¨ ui and TX ar essentially Spanish;
— `u will solely be French.

The process of those rules yields one or many languages that might in theory, be accountable for the orthography entered by the user. Supported the language identifier's results
—LI1. If the input file contains English-based names, the rule moves the results of the restricted mishap Strategy and also the SoundD Phonetic Strategy into the permanent suggestion pool at identical time and provides them of equal weight as long because the threshold t is equal for each algorithms, as shown in rule one.
—LI2. If the input data contains international names, and if there's a minimum of one candidate for a misspelled name at intervals 2 edit distance scores away with RNMS, then the phonetic results ar omitted and also the permanent suggestion pool consists solely of the results from RNMS (Algorithm 1). However, pronunciations of different language names are often kind of like English, for instance, the name city. Therefore, even for international names, if no candidate is provided by RNMS, then the results of the SoundD Phonetic Strategy ar captive to the pool.

—LI3. If a suggestion within the pool has constant precise phonetic cryptography because the misspelled one and it's the sole suggestion that's one edit distance faraway from the misspelled word with constant phonetic cryptography, or there's only 1 candidate name that's only 1 edit distance away with each the Restricted misadventure Strategy and therefore the SoundD Phonetic Strategy.

3.4 Decision Mechanism

At the ultimate stage it's attainable to check many attainable candidate names that have an equivalent weights (_). hoping on solely the edit distances doesn't usually offer the specified result. Therefore, we have a tendency to designed our call mechanism supported the content of the input info and accessorial the U.S agency call mechanism to that. the choice mechanism knowledge may be a compiled list of fashionable initial and last names that area unit scored supported the frequency of these names at intervals the u. s. [Census Bureau 1990]. this permits the tool to decide on a best-fit suggestion to eliminate the requirement for user interaction. The census score portion of the algorithmic program is enforced victimization the subsequent steps.
Step 1. Compare the present prompt names to the census file. If there's a match, store the census score related to the suggestion.
Step 2. select the name with the very best weighted hybrid designation score because the best-fit suggestion.
Step 3. If all suggestions area unit unmatched on the census file, select the suggestion with the very best weight from the weighted hybrid designation algorithmic program [17].

In 2012 by Melanie Herschel et.al.This position paper describes our efforts at intervals knowledge Bridges to integrate town knowledge and offer a quick introduction to EIT ICT Labs, its goals and structures, and the way knowledge Bridges fits them and have a tendency to then detail the activity and a few chosen results from 2011 - 2012, and plans for 2013. we have a tendency to conclude with

IJERT

some analysis challenges we have a tendency to attempt to address within the future “Digital Cities of the Future” is associate degree action line (or chapter) of EIT ICT Labs. at intervals this action line, we have a tendency to coordinated associate degree activity referred to as “Data Bridges: knowledge Integration for Digital cities”, whose aim is to provide, link, integrate and exploit open knowledge within the Digital Cities knowledge area, for the advantage of each voters and administrations. There in context, knowledge Bridges addresses several analysis challenges like exploit (or producing) Open town knowledge in RDF, knowledge linking and integration, knowledge beginning, and mental image. We’ve delineated the information Bridges activity, connecting partners from domain and trade to come up with made and prime quality town knowledge, outline strategies for RDF knowledge deposition and analytics, and build user-centric applications for Smarter Digital Cities. The exploitation of town knowledge continues to be in its infancy in knowledge Bridges, that is natural since we have a tendency to 1st ought to have made, prime quality knowledge sets to use. within the future, we have a tendency to set up on production additional advanced analytics, applications, visualizations, etc. to urge from this raw-data to actual data of interest to the user [18].

In 2012 by Hemant religion this paper provided a comprehensive study on analysis works numerous} authors associated with DW read maintenance considering various parameters and conferred an equivalent in tabular method. knowledge|a knowledge|an info} warehouse chiefly stores integrated information over information from many various remote data sources for question and analysis. The integrated info at the information warehouse is hold on within the kind of materialized views. mistreatment these materialized views, user queries is also answered quickly and with efficiency because the info is also directly accessible. These materialized views should be maintained in answer to actual relation updates within the totally different remote sources. one in {every of} the problems associated with materialized views is that whether or not they ought to be recomputed or they ought to be custom-made incrementally once every amendment within the base relations. read maintenance is that the method of change a materialized read in response to changes to the underlying information is termed read maintenance. There area unit many algorithms developed by totally different authors to ease the matter of read maintenance for information warehouse systems Associate in Nursing conferred an analysis numerous} approaches being projected by various authors to alter the read maintenance in information warehouse. we've got examined these approaches on numerous parameters and provided a comparative study in a very tabular manner. within the future work, we tend to propose a framework of information warehouse materialized read maintenance to beat the higher than issues ascertained by numerous authors [19].

IJERT

4. Basic View Maintenance

In Basic view maintenance approach, source and the data warehouse communicate with each other, when update occurs at source; it sends the notification to warehouse after that warehouse sends the query to source for the corresponding update as source receives the query it sends the reply to warehouse to that corresponding query [20]. This basic algorithm is neither convergent nor weakly consistent in a warehousing environment [21].

In [22] authors have motivated materialized views, their applications, and the problems and techniques for their maintenance. They have also considered new and promising application domains that are likely to drive work in materialized views and view maintenance.

In [23] authors have presented a comparison of three materialized join view maintenance methods in a parallel RDBMS, which they refer to performance at the cost of using more space. The results of this study show that the method of choice depends on the environment, in particular, the update activity on base relations and the amount of available storage space as the naive, auxiliary relation, and global index methods

.

In [24] authors extend the PVM-MED WRAP algorithm to achieve the Complete level of consistency, and also presented a scalable architecture for the proposed algorithm. In this Simulation shows that extending the PVM-MED WRAP algorithm to achieve the complete level of consistency limits the maximal parallelism feature.

In [25] authors have proposed a method to minimize the unnecessary updates for materialized views without increasing of response time. This method, which is named VMOST (View Maintenance based On State Transferring), introduced four states for materialized web views. In this when receiving query/update requests, web views transfer between the four states in accordance with their accessing and changing history. According to the experimental data, VMOST is adaptive to the fast changing web environment and has a good overall performance.

In [26] authors have discussed capabilities of PIVOT and UNPIVOT operators; materialized view maintenance, view maintenance work with PIVOT and UNPIVOT operators and finally they have focused on the research trends in view maintenance.

In [27] authors have presented an efficient technique aimed at updating data and performing view maintenance for real-time data warehouses while still enforcing these two timing requirements for the OLAP transactions. Authors proposed approach aims at addressing the issues of applying updates and performing view maintenance while still effectively serving user OLAP queries in real-time data warehouse environment.

5.Incremental View Maintenance

In Incremental view maintenance approach, only changes in the materialized views of the data warehouse are computed rather than recomputing every view from scratch. ECA is an incremental view maintenance algorithm based on the centralized view maintenance algorithm. It is also the fastest algorithm that will let the data warehouse remain in a consistent state [28].

In [29] authors have described the architecture of the Whips prototype system, which collects, transforms, and integrates data for the warehouse. They have showed how the required functionality can be divided among cooperating distributed CORBA objects, providing both scalability and the flexibility needed for supporting different application needs and heterogeneous sources.

In [30] authors have proposed a new incremental approach to maintaining materialized views both in the data warehouse and in the data marts. This approach uses auxiliary relations derived from the materialized view and the base relations by normalizing the view according to the functional dependencies valid in the view. The motivation for using normalization in this approach is to increase the likelihood of sharing the auxiliary relations across several data marts, as has been shown by the use of normalization in relational database design.

In [31] authors have proposed a new compensation algorithm that is used in removing the anomalies, caused by interfering updates at the base relations, of incremental computation for updating the view. This algorithm does not assume that messages from a data source will reach the view maintenance machinery in the same order as they are generated, and they are also able to detect update notification messages that are lost in their transit to the view, which would otherwise cause the view to be updated incorrectly. Proposed algorithm also does not require that the system be quiescent before the view can be refreshed.

In [32] authors have proposed a maintenance algorithm that does not need the compensation step and applies to general view expressions of the bag algebra, without limit on the number of base relations per data source
.

In [33] authors have proposed an incremental maintenance method for temporal views that allows improvements over the re computation from scratch. They introduce formalism for temporal data warehouse specification that summarizes information needed for its incremental maintenance.

In [34] authors have presented an incremental view maintenance approach based on schema transformation pathways. This approach is not limited to one specific data model or query language, and would be useful in any data transformation or integration framework based on sequences of primitive schema transformations.

In [35] authors have tackled the problem of finding the most efficient batch incremental maintenance strategy under a refresh response time constraint; that is, at any point in time, the system, upon request, must In this authors have also presented a series of analytical results leading to the development of practical algorithms.

In [36] authors have developed the change-table technique for incrementally maintaining general view expressions involving relational and aggregate operators. They show that the change table technique outperforms the previously proposed techniques by orders of magnitude. The developed framework easily extends to efficiently maintaining view expressions containing outer join operators. They have proved that the developed change-table technique is an optimal incremental maintenance scheme for a given view expression tree under some reasonable assumptions. Incremental maintenance technique is accepted in this paper [37]. In this idea and strategy of minimum incremental maintenance is presented. The materialized view definitions and maintenance expressions, as well as algorithms are given. The experiment shows that the maintenance cost of materialized views is decreased and data warehouse processing efficiency is improved.

In [38] authors have proposed an algorithm for incremental view maintenance with the inclusion of some existing approaches. They utilized the concept of version store for older versions of tables that have been updated at the source and they are also able to detect the update notification messages that are lost during updating the view.

In [39] authors have proposed an incremental view maintenance approach based on data source compensation called DSCM, which effectively overcomes the shortcomings of previous compensation solutions through minimizing the extent of compensation to the base relations of data sources and using the precise source states to evaluate the maintenance queries.

6. Self maintainable Maintenance

Viewself-maintenance is the process of incrementally refreshing a materialized view using the view

IJERT

instance and the update to some base relation, but without examining any of the base relations, or using only a specified subset of the base relations. When a view together with a set of auxiliary views can be maintained at the warehouse without accessing base data, we say the views are self maintainable [40] [41].

In [42] author has reported on some interesting new results for conjunctive-query views under insertion updates: 1) the CTSM's are extremely simple queries that look for certain tuples in the view to be maintained; 2) these CTSM's can be generated at view definition time using a very simple algorithm based on the concept of Minimal ZPartition; 3) view self-maintenance can also be expressed as simple update queries over the view itself.

In [43] authors have showed that by using key and referential integrity constraints, they often can maintain a select-project-join view without going to the data sources or replicating the base relations in their entirety in the warehouse. They derive a set of auxiliary views such that the warehouse view and the auxiliary views together are selfmaintainable- they can be maintained without going to the data sources or replicating all base data.

In [44] authors have proposed an incremental technique for efficiently maintaining materialized views in these high performance applications by materializing additional relations which are derived from the intermediate results of the view computation. They presented an algorithm which uses the operator tree for the view to determine which additional relations need to be materialized in order to maintain the view. They also give an incremental algorithm for maintaining both the view and the additional relations which has several desirable features.

In [45] author has focused on the problem of determining view in the presence of functional dependencies. Here author, shows (i) SM of a conjunctive-query (CQ) view can be reduced to a problem of query containment, whose solution can be expressed as a (Boolean) query on with no repeated predicates; two useful concepts can be defined: the well-founded derivation DAG and sub goal partitioning. (iii) Derive three simple conditions that each guarantee view SM under general functional dependencies.

In [46] authors gave a preliminary result on self-maintainability of deletions of views over XML data. We give a necessary and sufficient condition of self-maintainability of deletions, and an algorithm to implement self-maintenance of deletions of views for XML data. This paper [47] provided an online view self maintenance method based on source view's increment to keep the materialized view consistent with the data source. In this, authors have used the cooperation between the integrator at warehouse and the monitor at data source to maintenance the view, and the method does not require to query back to data source, it can accelerate view maintenance and lessen the burden of communication between data warehouse and data sources.

IJERT

7.REFERENCES

[1]. Stephen S. Yau, Fellow And Yin Yin **"**A Privacy Preserving Repository For Data Integration Across Data Sharing Services**",** Ieee Transactions On Services Computing, Vol. 1, No. 3, July-September 2008 **.**

[2]. Alka Gangrade, Ravindra Patel," Privacy Preserving Two-Layer Decision Tree Classifier for Multiparty Databases", International Journal of Computer and Information Technology (2277 – 0764) Volume 01– Issue 01, September 2012.

[3]. Yehuda Lindell, Benny Pinkas, "Privacy preserving data mining," Journal of Cryptology vol. 15, no. 3, pp. 177–206. 2002.

[4]. Wenliang Du, Zhijun Zhan, "Building decision tree classifier on private data," In CRPITSpp. 1–8. , 2002,

[5]. Bart Kuijpers, Vanessa Lemmens, Bart Moelans," Privacy Preserving ID3 over Horizontally, Vertically and Grid Partitioned Data",avrxi-0803.155v1[cs.db], 11 march 2008.

[6]. C. Farkas, S. Jajodia, "The inference problem: A survey", SIGKDD Explorations 4 vol. (2) 6–11,2002.

[7]. L. Sweeney, "A primer on data privacyprotection" phd thesis, in: Massachusetts Institute of Technologie, 2001.

[8]. W. H. Inmon, Building the Data Warehouse, 3rd ed. New York: Wiley, 2002.

[9]. R. Kimball, The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouse. New York: Wiley, 1996.

[10]. S. Staab and R. Studer, Eds., "Handbook on Ontologies". NewYork: Springer-Verlag, 2004.

[11] M. Doerr, "The CIDOC conceptual reference module: An ontological approach to semantic interoperability of metadata," AI Mag., vol. 24, no. 3, pp. 75–92, 2003.

[12] D. Boneh, G.D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public Key Encryption with Keyword Search," Advances in Cryptology (EUROCRYPT '04), pp. 506-522, 2004.

[13]L. Xiong, S. Chitti, and L. Liu,"Preserving Data Privacy for Outsourcing Data Aggregation Services," *ACM*

Trans. Internet Technology, vol. 7, no. 3, pp. 17-45, 2007.

[14] B. Pinkas, ”Cryptographic Techniques for Privacy Preserving Data Mining,” SIGKDD Explorations, vol. 4, no. 2, pp. 12-19, 2002.

[15]. Arun Sen And Atish P. Sinha” Toward Developing Data Warehousing Process Standards: An Ontology-Based Review Of Existing Methodologies” Ieee Transactions On Systems, Man, And Cybernetics—Part C: Applications And Reviews, Vol. 37, No. 1, January 2007 .

[16]. Melanie Herschel, Ioana Manolescu” DataBridges:n Data Integration for Digital Cities” 978-1-4503-1709 2/12/10, ACM -2012.

[17]. Cihan Varol, Sam Houston State University Coskun Bayrak, “Hybrid Matching Algorithm For Personal Names” Acm J. Data Inform. Quality 3, 4, Article 8 (September 2012).

[18]. Melanie Herschel, Ioana Manolescu” Databridges: Data Integration For Digital Cities”, Cdmw’12, Maui, Hawaii, Usa. Copyright 2012 Acm 978-1-4503-1709. October 29, 2012,

[19]. Hemant Jain, AnjanaGosain”A Comprehensive Study of View Maintenance Approaches in Data Warehousing Evolution”, ACM SIGSOFT Software Engineering Notes, Volume 37, September 2012 .

[20] Abdulaziz S. Almazyad, Mohammad Khubeb Siddiqui, Incremental View Maintenance: An Algorithmic Approach. International Journal of Electrical & Computer Sciences IJECSIJENS Vol: 10 No: 03, June2010.

[21] Y. Zhuge, H. Garcia-Molina, J. Hammer, and J. Widom.View maintenance in a warehousing environment. In *SIGMOD*,pages 316–327, San Jose, California, May 1995.

[22] Ashish Gupta, Inderpal Singh Mumick, Maintenance of Materialized Views: Problems, Techniques, and Applications. IEEE Data Eng. Bull. 18(2): 3-18(1999).

[23] .Gang Luo, Jeffrey F. Naughton, Curt J. Ellmann, Michael Watzke, A Comparison of Three Methods for Join View Maintenance in Parallel RDBMS. ICDE 2003: 177 188.

[24] Fouad, S.A. , Karam, O.H. , El-Sharkawy, M.A. . A parallel view maintenance algorithm for complete consistency. International Conference on Electrical, Electronic and Computer Engineering, 2004.

[25] Yan Zhang, Shiwei Tang, Dongqing Yang, Efficient View Maintenance in a Large-scale Web Warehouse. Fourth International conference on Computer and Information technology, CIT 2004.

[26] A.N.M.B. Rashid and M.S. Islam, Role of Materialized View Maintenance with PIVOT and UNPIVOT Operators. IEEE International Advance Computing Conference (IACC’09), Patiala, India, pp. 951-955, March 6-7, 2009.

[41] N. Huyn, Exploiting Dependencies to Enhance View Self Maintainability. Technical Note, 1997.

[27] Nguyen Hoang Vu, Vivekanand Gopalkrishnan, On Scheduling Data Loading and View Maintenance in Soft Real-time Data Warehouses. Comad, 2009.

[28] Y. Zhuge, H. Garcia-Molina, J. Hammer, and J. Widom.View maintenance in a warehousing environment. In *SIGMOD*,pages 316–327, San Jose, California, May 1995.

[29] J. L. Wiener, H. Gupta, W. J. Labio, Y. Zhuge, H. Garcia- Molina, and J. Widom, A system prototype for view maintenance. Proceedings of the ACM Workshop on Materialized Views , pp. 26-33. June 7, 1996

[30] Mukesh Mohania, Kamalakar Karlapalem and Yahiko Kambayashi, Maintenance of Data Warehouse Views Using Normalisation. Dexa’99, LNCS 1677, pp. 747-750, 1999.

[31] Tok Wang Ling, Eng Koon Sze. Materialized View Maintenance Using Version Numbers. Proceedings of the Sixth International Confer 2001.

[32] Gianulca Moro, Claudio Sartori, Incremental Maintenance of Multi-Source Views. Proceedings of the 12th Australasian database conference 2001.

[33] Sandra de Amo, Mirian Halfeld Ferrari Alves, Incremental Maintenance of Data Warehouses Based on Past Temporal Logic Operators. J. UCS 10(9): 1035-1064 (2004).

[34] Hao Fan, Using Schema Transformation Pathways for Incremental View Maintenance. Proceedings of the 7th international conference on Data Warehousing and Knowledge Discovery (2005).

[35] Hao He, Junyi Xie, Jun Yang, Hai Yu, Asymmetric Batch Incremental View Maintenance. 21st International Conference on Data Engineering, 2005.

[36] Himanshu Gupta, Inderpal Singh Mumick, Incremental maintenance of aggregate and outerjoin expressions. InformationSystems, Vol. 31, Nr. 6 p. 435—464. (2006),

[37] Lijuan Zhou, Qian Shi, Haijun Geng, The Minimum Incremental Maintenance of Materialized Views in Data Warehouse. 2nd International Asia Conference (CAR), March 2010.

[38] Abdulaziz S. Almazyad, Mohammad Khubeb Siddiqui, Incremental View Maintenance: An Algorithmic Approach. International Journal of Electrical & Computer Sciences IJECSIJENS Vol: 10 No: 03, June2010.

[39] Xiaogang Zhang, Luming Yang, De Wang, Incremental View Maintenance Based on Data Source Compensation in Data Warehouses. Changsha, china, October 2010.

[40] D. Quass, A. Gupta, I. S. Mumick, and J. Widom, Making Views Self-Maintainable for Data Warehousing. Proceedings of the Conference on Parallel and Distributed Information Systems, Miami Beach, FL, December 1996.

http://wwwdb.stanford.edu/pub/papers/fdvsm.ps.

IJERT

[42] N. Huyn, Efficient View Self-Maintenance. Proceedings of the ACM Workshop on Materialized Views: Techniques and Applications, Montreal, Canada, June 7, 1996.

[43] D. Quass, A. Gupta, I. S. Mumick, and J. Widom, Making Views Self-Maintainable for Data Warehousing. Proceedings of the Conference on Parallel and Distributed Information Systems, Miami Beach, FL, December 1996.

[44] Vincent, M. Mohania, Y. Kambayashi. A Self Maintainable View Maintenance Technique for Data Warehouses. ACM SIGMOD 7-22, (1997).

[45] N. Huyn, Exploiting Dependencies to Enhance View Self Maintainability. Technical Note, 1997. http://wwwdb.stanford.edu/pub/papers/fdvsm.ps.

[46] Cheng Hua, Ji Gao, Yi Chen, Jian Su, Self maintainability of deletions of materialized views over XML data. International conference on machine learning and cybernetics, 2003.

[47] Hai Liu, Yong Tang, Qimai Chen, The Online Cooperating View Maintenance Based on Source View Increment. CSCWD 11th International conference, pp. 753 756, April 2007.

IJERT