www.s2is.org
International Journal on Smart Sensing and Intelligent Systems
ISSN 1178-5608

TEMPORAL QUERY PROCESSIG USING SQL SERVER

Mastan Vali Shaik[1]* P Sujatha[2]

[1]Research Scholar, VELS University, Chennai, India

[2]Associate Professor, VELS University, Chennai, India

Email: Shaik.vali@ibrict.edu.om

Submitted: May 27, 2017 Accepted: June 15, 2017 Published: Sep 1, 2017

Abstract- Most data sources in real-life are not static but change their information in time. This evolution of data in time can give valuable insights to business analysts. Temporal data refers to data, where changes over time or temporal aspects play a central role. Temporal data denotes the evaluation of object characteristics over time. One of the main unresolved problems that arise during the data mining process is treating data that contains temporal information. Temporal queries on time evolving data are at the heart of a broad range of business and network intelligence applications ranging from consumer behaviour analysis, trend analysis, temporal pattern mining, and sentiment analysis on social media, cyber security, and network monitoring. Social networks (SN) such as Facebook, twitter, LinkedIn contains huge amount of temporal information. Social media forms a dynamic and evolving environment. Similar to real-world friendships, social media interactions evolve over time. People join or leave groups; groups expand, shrink, dissolve, or split over time. Studying the temporal behaviour of communities is necessary for a deep understanding of communities in social media(SM). In this paper we focus on the use of temporal data and temporal data mining in social networks.

Index terms: temporal data, social networks, temporal database, data mining, Temporal query processing, time stamp using SQL Server.

I. INTRODUCTION

Temporal data refers to data, where changes over time or temporal aspects play a central role. Temporal data denotes the evaluation of object characteristics over time [1]. Temporal data is simply data that represents a state in time, such as the total rainfall in Oman on January, 27, 2017 or number of accidents took between May, 13, 2009 and May, 22, 2009. Temporal Data Mining is a single step in the process of Knowledge Discovery in Temporal Databases that enumerates structures over the temporal data, and any algorithm that enumerates temporal patterns from, or fits models to, temporal data is a Temporal Data Mining Algorithm [1]. Temporal data mining has led to a new way of interacting with a temporal database: specifying queries at a much more abstract level than say, Temporal Structured Query Language permits. Temporal data have a unique structure: High dimensionality and High feature correlation [2]. With the rapid increase of stored data, the interest in the discovery of hidden information has exploded in the last decade. This discovery has mainly been focused on data classification, dataclustering and relationship finding. One important problem that arises during the discovery process is treating data withtemporal dependencies [3]. With the advent of Web2.0/3.0, social networking became more popular and subsequently it is evolved as a part of every human life. As per the statistics of Facebook, everyone interested in having at least one social networking account [4]. Even the mobiles are providing in-built app for social networking sites. The sharing of information via these social networking sites generates a huge temporal data. Subsequently there is a broad range of diverse interpretations related to the usefulness of social media as a powerful source of communication and learning.

Through examining several advantages and disadvantages we will highlight the circumstances for responsible handling of social media. The examination takes place on two levels:

1. The global form of information and expressing opinions via social media and its impact on social systems.

2. The individual form of participating and the specific influence of social media on the own social behaviour.

Temporal data mining tasks include: [5][12]

- Temporal data characterization and comparison,
- Temporal clustering analysis,
- Temporal classification,

• Temporal association rules,

• Temporal pattern analysis, and

• Temporal prediction and trend analysis.

This paper is intended to give an overview of temporal data mining and its significance with social networks such as Facebook, twitter, LinkedIn, Myspace.

II. Temporal Data in Social Networking.

In the past decade, the web has undergone tremendous changes in several aspects such as, its size, content, and usage. After the emergence of Web 2.0 the use of social media and crowdsourcing tools for content generation like WIKI, Social Networks, Blogs and media sharing sites are rapidly increasing. This trending use of these tools has generated a huge amount of temporal information thus posing new set of research challenges for the Information Retrieval (IR) community. These tools have Time as one of its important dimension which is very useful for tasks like document exploration, similarity search, summarization, and clustering. Most of the existing IR systems ignore the Time dimension. However, in the last few years there has been exciting work on analyzing and exploiting temporal information for the presentation, organization, and in particular the exploration of search results [1]. First studies have characterized the evolution of web documents [5][6][7], mostly to improve search engine crawling [8].

Social media enables us to be connected and interact with each other anywhere and anytime – allowing us to observe human behaviour in an unprecedented scale with a new lens. This social media lens provides us with golden opportunities to understand individuals at scale and to mine human behavioural patterns otherwise impossible.

Social media forms a dynamic and evolving environment. Similar to real-world friendships, social media interactions evolve over time. People join or leave groups; groups expand, shrink, dissolve, or split over time. Studying the temporal behavior of communities is necessary for a deepunderstanding of communities in social media.

SN messages are significantly different from other kinds of text documents, e.g., Web pages. They are created at a high rate and their information content can soon be-come outdated. This is the reason why the timestamp of these messages (e.g.fresh, recent and outdated) can be as important as their content. Several messages may repeat the same piece of information, therefore it is important to recognize when new information is added finally, and SN messages are not independent pieces of text: they can be seen as dynamic contents coming from online platforms where inter-connected users discuss topics and share news with friends. Social media mining is a rapidly growing new field. It is an interdisciplinary field at the crossroad of disparate disciplines deeply rooted in computer science and social sciences. There are an active community and a large body of literature about social media. The fast-growing interests and intensifying need to harness social media data require research and the development of tools for finding insights from big social media data [9].

Social media data is significantly different from the traditional data that we are familiar with in data mining. Apart from enormous size, the mainly user-generated data is noisy and unstructured, with abundant social relations such as friendships and followers-followees. This new type of data mandates new computational data analysis approaches that can combine social theories with statistical and data mining methods. The pressing demand for new techniques guides in and entails a new interdisciplinary field – social media mining.

III. Temporal Database for Social Networking.

Some data may be inherently historical (e.g., medical or judicial records). Temporal databases provide a uniform and systematic way of dealing with historical data. Temporal data mining has led to a new way of interacting with a temporal database: specifying queries at a much more abstract level than say, Temporal Structured Query Language (TSQL) permits. It also facilitates data exploration for problems that, due to multiple and multi-dimensionality, would otherwise be very difficult to explore by humans, regardless of use of, or efficiency issues with, TSQL. [10][11]

Temporal queries on time evolving data are at the heart of a broad range of business and network intelligence applicationsranging from consumer behaviour analysis, trend analysis, temporal pattern mining, and sentiment analysis on social media, cyber security, and network monitoring [3].

Temporal data stored in a temporal database is different from the data stored in non-temporal database in that a time period attached to the data expresses when it was valid or stored in the database. Conventional databases consider the data stored in it to be valid at time instant now, they do not keep track of past or future database states. Temporal in SQL Server is designed to simply the handling of time-varying data. It provides the ability to look at data trends, types of data changes, and the overall data evolution within your database.

By attaching a time period to the data, it becomes possible to store different database states.

There are a few of requirements on temporal tables in SQL Server:

- The main table must have a primary key.
- They must have columns for start time and end time. These must be of type datetime2 (with any precision).
- The history table must be schema-aligned with the main table, meaning that it has the same columns (names, data types, ordering).

The history tables can either be created manually or automatically be SQL Server.

A. Timestamp model

The following table illustrates a simple time stamp model for registration process.

All the tuples in the relation have additional temporal attribute.

Here *Registration (level, venue, **time**)*

A tuple (l, v, t) indicates the fact that registration **l** is in the venue **v** at time **t.**

Here **time** is a temporal attribute. A temporal attribute can be a single dimensional has one temporal attribute or multi-dimensional. In a multi-dimensional each tuple in a relational can have more than one temporal attribute.

Registration		
level	Venue	Time
Bachelor	Lab-1	26-Jan-17 10.00
Bachelor	Lab-1	26-Jan-17 10.10
Bachelor	Lab-1	26-Jan-17 11.15
…..		
……		
Diploma	Lab-2	27-Jan-17 09.30
Bachelor	Lab-1	26-Jan-17 11.45
Diploma	Lab-2	28-Jan-17 09.00
Diploma	Lab-2	28-Jan-17 10.15

Fig 1. sample timestamp model for registration entity

A first step towards a temporal database thus is to timestamp the data. This allows the distinction of different databasestates. One approach is that a temporal database may timestamp entities with time periods. Another approach is the time stamping of the property values of the entities.

In the relational data model, tuples are time stamped, where as in object-oriented data models, objects and/or attribute values may be time stamped.

B. Temporal query on timestamp model using SQL Server

Temporal tables give the possibility to retrieve the data from any point in the past and for every data change (update, delete, merge). With temporal table users can recover data from accidental changes (update/delete) as well as audit all changes to data. Temporal is a database feature that was introduced in ANSI SQL 2011 and is now supported in SQL Server 2016. This feature is also available in Azure SQL Databases. Temporal table is a new built-in feature in SQL Server 2016 that allows you to travel back in time and get the data that is represented at a past state in time rather than the data that is correct at the current moment in time. Timestamp is a data type that exposes automatically generated, unique binary numbers within a database. Timestamp is generally used as a mechanism for version-stamping table rows. The storage size is 8 bytes. The timestamp data type is just an incrementing number and does not preserve a date or a time. To record a date or time, use a date time data type.

The temporal information extracted from documents can directly be used to allow the user of a search engine to constrain his/her query in a temporal manner. That is, in addition to a textual part, a query contains a temporal part. For example, in addition to “bachelor registration” a temporal constraint like “20-Jan-17 to 30-Jan-17” could be specified.

The user would obviously expect documents about bachelor registration as results for his query. The objective when using a combination of a text and a temporal query can thus be formulated in the following way: both parts of the query are satisfied, i.e., the more the textual and the temporal parts fit to a document, the higher should be the rank of this document.

SQL Server 2016 treats a temporal table is a table with a PERIOD definition, contains system columns with a datatype of DATETIME2, has an associated history table into which the system records all prior versions of each record with their period of validity. The value of each record at any point in time can be determined.

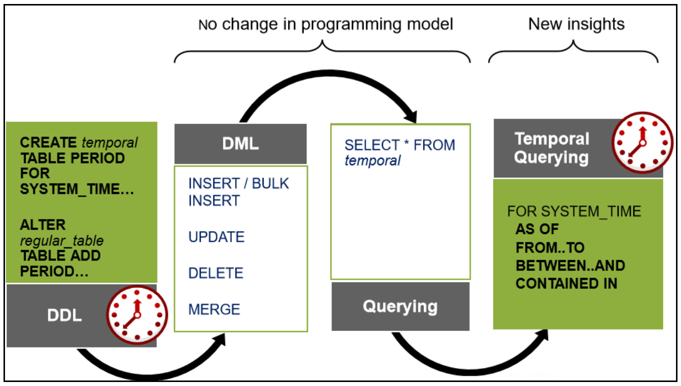

Fig 2: Creating and querying a temporal table in SQL Server2016

Every temporal database has two tables. The main temporal table that contains actual and current data and a history table which contains historical data. When you want to get latest (actual) state of data in a temporal table, you can query completely the same way as you query non-temporal table.

Regular querying process will be done on temporal table where as historical bale is used to query a historical data.

If the PERIOD columns are not hidden, their values will appear in a SELECT * query. If you specified PERIOD columns as hidden, their values won't appear in a SELECT * query. When the PERIOD columns are hidden, reference the PERIOD columns specifically in the SELECT clause to return the values for these columns.

To perform any type of time-based analysis, use the new FOR SYSTEM_TIME clause with four temporal-specific sub-clauses to query data across the current and history tables.

The following syntax is used to query on historical table.

```
AS OF <date_time>
FROM <start_date_time> TO <end_date_time>
BETWEEN <start_date_time> AND <end_date_time>
CONTAINED IN (<start_date_time> , <end_date_time>)
ALL
```

The built-in support in SQL Server for managing temporal data reduces application logic and ensures consistent handling of time-related events across all applications that run against your database, including purchased applications. Through simple declarative SQL statements, administrators can instruct SQL Server to automatically maintain a history of database changes or track effective business dates, eliminating the need for such logic to be hand-coded into triggers, stored procedures, or in-house applications. This, in turn, helps companies adhere more quickly to new compliance initiatives. Furthermore, a consistent approach to managing temporal data reduces query complexity and promotes enhanced analysis of time-dependent events.

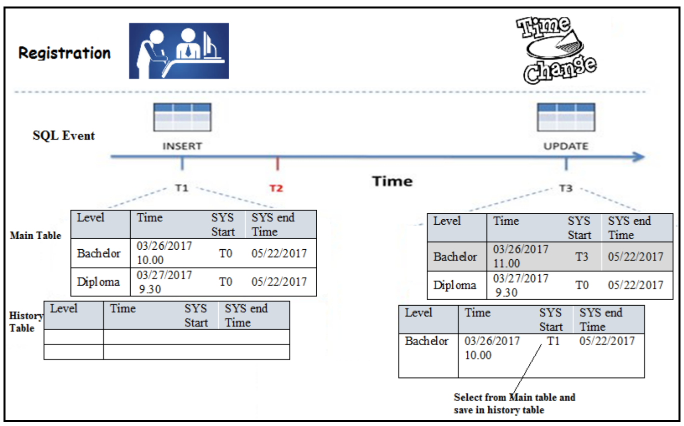

Fig 3: Changes in Main and temporal table.

The above figure illustrates the relationship between Main table and temporal table. The figure shows two tables: Main table and History table. At Time stamp T1, Main table has two records and history table is empty. At time stamp T3, there is a change in first record of Main table. Once a change is occurred, the old record is moved/inserted to History table and new/updated information remains in Main table.

IV. Future Work.

Our research focus on designing a temporal database by using SQL Server or DB2, retrieve similar documents and compare for temporal document similarity in SM or SN.

The main research questions that will be investigated are what make two documents temporally similar? In order to achieve this objective, the following questions will also be investigated.

• Should two documents be considered similar if they cover the same temporal interval? [8]

• Should the temporal focus of the documents be important for their temporal similarity? [8]

• Can two documents be regarded as temporally similar if one contains a small temporal interval of the other document in a detailed way? [8]

In this work we propose to efficiently handle temporal quires by finding temporal document similarity. Instead of comparing a temporal query with the temporal information of a document, two documents can be compared with respect to their temporal similarity.

REFERENCES

[1] Aizat Azmi, Ahmad Amsyar Azman, Sallehuddin Ibrahim, and Mohd Amri Md Yunus, “Techniques In Advancing The Capabilities Of Various Nitrate Detection Methods: A Review”, International Journal on Smart Sensing and Intelligent Systems., VOL. 10, NO. 2, June 2017, pp. 223-261.

[2] Tsugunosuke Sakai, Haruya Tamaki, Yosuke Ota, Ryohei Egusa, Shigenori Inagaki, Fusako Kusunoki, Masanori Sugimoto, Hiroshi Mizoguchi, “Eda-Based Estimation Of Visual Attention By Observation Of Eye Blink Frequency”, International Journal on Smart Sensing and Intelligent Systems., VOL. 10, NO. 2, June 2017, pp. 296-307.

[3] Ismail Ben Abdallah, Yassine Bouteraa, and Chokri Rekik , “Design And Development Of 3d Printed Myoelctric Robotic Exoskeleton For Hand Rehabilitation”, International Journal on Smart Sensing and Intelligent Systems., VOL. 10, NO. 2, June 2017, pp. 341-366.

[4] S. H. Teay, C. Batunlu and A. Albarbar, “Smart Sensing System For Enhanceing The Reliability Of Power Electronic Devices Used In Wind Turbines”, International Journal on Smart Sensing and Intelligent Systems., VOL. 10, NO. 2, June 2017, pp. 407- 424

[5] SCihan Gercek, Djilali Kourtiche, Mustapha Nadi, Isabelle Magne, Pierre Schmitt, Martine Souques and Patrice Roth, “An In Vitro Cost-Effective Test Bench For Active Cardiac Implants, Reproducing Human Exposure To Electric Fields 50/60 Hz”, International Journal on Smart Sensing and Intelligent Systems., VOL. 10, NO. 1, March 2017, pp. 1- 17

[6] P. Visconti, P. Primiceri, R. de Fazio and A. Lay Ekuakille, “A Solar-Powered White Led-Based Uv-Vis Spectrophotometric System Managed By Pc For Air Pollution Detection In Faraway And Unfriendly Locations”, International Journal on Smart Sensing and Intelligent Systems., VOL. 10, NO. 1, March 2017, pp. 18- 49

[7] Samarendra Nath Sur, Rabindranath Bera and Bansibadan Maji, “Feedback Equalizer For Vehicular Channel”, International Journal on Smart Sensing and Intelligent Systems., VOL. 10, NO. 1, March 2017, pp. 50- 68

[8] Yen-Hong A. Chen, Kai-Jan Lin and Yu-Chu M. Li, “Assessment To Effectiveness Of The New Early Streamer Emission Lightning Protection System”, International Journal on Smart Sensing and Intelligent Systems., VOL. 10, NO. 1, March 2017, pp. 108- 123

[9] Iman Heidarpour Shahrezaei, Morteza Kazerooni and Mohsen Fallah, “A Total Quality Assessment Solution For Synthetic Aperture Radar Nlfm Waveform Generation And Evaluation In A Complex Random Media”, International Journal on Smart Sensing and Intelligent Systems., VOL. 10, NO. 1, March 2017, pp. 174- 198

[10] P. Visconti ,R.Ferri, M.Pucciarelli and E.Venere, “Development And Characterization Of A Solar-Based Energy Harvesting And Power Management System For A Wsn Node Applied To Optimized Goods Transport And Storage”, International Journal on Smart Sensing and Intelligent Systems., VOL. 9, NO. 4, December 2016 , pp. 1637- 1667

[11] YoumeiSong,Jianbo Li, Chenglong Li, Fushu Wang, “Social Popularity Based Routing In Delay Tolerant Networks”, International Journal on Smart Sensing and Intelligent Systems., VOL. 9, NO. 4, December 2016 , pp. 1687- 1709

[12] Seifeddine Ben Warrad and OlfaBoubaker, “Full Order Unknown Inputs Observer For Multiple Time-Delay Systems”, International Journal on Smart Sensing and Intelligent Systems., VOL. 9, NO. 4, December 2016 , pp. 1750- 1775

[13] Rajesh, M., and J. M. Gnanasekar. "Path observation-based physical routing protocol for wireless ad hoc networks." International Journal of Wireless and Mobile Computing 11.3 (2016): 244-257.

[14]. Rajesh, M., and J. M. Gnanasekar. "Congestion control in heterogeneous wireless ad hoc network using FRCC." Australian Journal of Basic and Applied Sciences 9.7 (2015): 698-702.

[15]. Rajesh, M., and J. M. Gnanasekar. "GCCover Heterogeneous Wireless Ad hoc Networks." Journal of Chemical and Pharmaceutical Sciences (2015): 195-200.

[16]. Rajesh, M., and J. M. Gnanasekar. "CONGESTION CONTROL USING AODV PROTOCOL SCHEME FOR WIRELESS AD-HOC NETWORK." Advances in Computer Science and Engineering 16.1/2 (2016): 19.

[17]. Rajesh, M., and J. M. Gnanasekar. "An optimized congestion control and error management system for OCCEM." International Journal of Advanced Research in IT and Engineering 4.4 (2015): 1-10.

[18]. Rajesh, M., and J. M. Gnanasekar. "Constructing Well-Organized Wireless Sensor Networks with Low-Level Identification." World Engineering & Applied Sciences Journal 7.1 (2016).

[19] L. Jamal, M. Shamsujjoha, and H. M. Hasan Babu, "Design of optimal reversible carry look-ahead adder with optimal garbage and quantum cost," International Journal of Engineering and Technology, vol. 2, pp. 44–50, 2012.

[20] S. N. Mahammad and K. Veezhinathan, "Constructing online testable circuits using reversible logic," IEEE Transactions on Instrumentation and Measurement, vol. 59, pp. 101–109, 2010.

[21] W. N. N. Hung, X. Song, G. Yang, J. Yang, and M. A. Perkowski, "Optimal synthesis of multiple output boolean functions using a set of quantum gates by symbolic reachability analysis," IEEE Trans. on CAD of Integrated Circuits and Systems, vol. 25, no. 9, pp. 1652–1663, 2006.

[22] F. Sharmin, M. M. A. Polash, M. Shamsujjoha, L. Jamal, and H. M. Hasan Babu, "Design of a compact reversible random access memory," in 4th IEEE International Conference on Computer Science and Information Technology, vol. 10, june 2011, pp. 103–107.

[23] Dr. AntoBennet, M, Sankar Babu G, Suresh R, Mohammed Sulaiman S, Sheriff M, Janakiraman G ,Natarajan S, "Design & Testing of Tcam Faults Using T_H Algorithm", Middle-East Journal of Scientific Research 23(08): 1921-1929, August 2015 .

[24] Dr. AntoBennet, M "Power Optimization Techniques for sequential elements using pulse triggered flipflops", International Journal of Computer & Modern Technology , Issue 01 ,Volume01 ,pp 29-40, June 2015.