Implementing Service Oriented Architecture for Data Mining

Vasant Maruti Patil
Department of Computer Engineering,
Savitribai Phule Pune University
SKN Sinhgad Institute of Technology & Science, Lonavala,
Pune, Maharashtra, India
Email: vasantpatil11@gmail.com

Prof. Kalpana Kadam
Department of Computer Engineering,
Savitribai Phule Pune University
SKN Sinhgad Institute of Technology & Science, Lonavala,
Pune, Maharashtra, India
Email: more.kalpana@gmail.com

***Abstract*—**With Web technology, data on internet has become increasingly large and complex. No matter users or internet users needs all this data. Also the data which is available on web not all the time useful information or it is knowledgeable. Hence web data mining is necessary to fulfill this demand. Web data mining can extract unstructured, undiscovered data which is possibly useful information and knowledge, from much incomplete, noisy, ambiguous, random, practical application related data from WWW network. It is a new emerging commercial information/data mining technology. Its main characteristic is to extract key data to support business for decision making from business database through the use of extraction, conversion, analysis and other transaction models. Web service is deployed on the web with an object or component to achieve distributed application software platform through a series of protocols. Web Service platform provides a set of standard types systems, rules, techniques and internet service-oriented applications for communication between the different platforms, different programming languages and different types of systems to achieve interoperability. This paper gives the actual and practical application of web services for data mining, we build a data mining model based on Web services and going forward it is possible to implement the new data mining solution for security configuration. This has been achieved with the use of prototypes of a dynamic web service based data mining systems.

Keywords*-pivoting, Data Mining, SQL, Web services.*

*****

I. INTRODUCTION

With Web technology, data on networks has become increasingly large and complex. No matter managers or network users are required from these complex Web data to get useful information and knowledge, so Web data mining is necessary to adapt this demand. Web data mining can extract undiscovered unexpected possibly useful information and knowledge from much incomplete noise ambiguous random practical application data on WWW network[6]. It is a new commercial information mining technology. Its main characteristic is to extract key data to support business decision-making from business database through extraction, conversion, analysis and other models transaction. Web service is deployed in the Web on an object or component to achieve distributed application software platform through a series of protocols. Web Service platform provides a set of standard types systems and internet service-oriented applications for communication between the different platforms, different programming languages and different types of systems to achieve interoperability[6].

This paper gives the actual application of Web data mining, we build a web data mining model based on Web service and put forward a new web data mining solution according to the prototype of a dynamic Web service based data mining process system. Web mining focus on finding useful data, knowledge or information from the web, web pages and usage log. If we compare traditional data warehouse, information on web is unstructured or semi-structured and dynamic[6]. Most of the time it leads to confusion. So it is not good directly mine data on web pages and need some data pre-processing also web data types are more complex. Data in the web pages are, the data available in web pages and the data created or generated by web operation. Web mining is always as: information selection, patter discovery, pattern analysis, search resource and pre-processing[6].

II. RELATED WORK

"SOAP Processing Performance and Enhancement"[1]. Joe M. Tekli, Member, IEEE, Ernesto Damiani, Senior Member, IEEE, Richard Chbeir, Member, IEEE, Gabriele Gianini. Approach: This survey paper provides a concise, yet comprehensive review of the research efforts aimed at SOAP performance enhancement. A unified view of the problem is provided, covering almost every phase of SOAP message processing and ranging over message parsing, serializing, destabilization, compression, multitasking, security evaluation and data/instruction-level processing. The main idea is to identify the common parts of SOAP messages to be processed only once, avoiding a large amount of overhead. Other approaches investigate nontraditional processor architectures, including micro and macro level parallel processing solutions,

so as to further increase the processing rates of SOAP/XML software tool kits.

"Horizontal Aggregations in SQL to Prepare Data Sets for Data Mining Analysis"[2]. Carlos Ordonez and Zhibo Chen. Here they are Preparing a data set for analysis is generally the most time consuming task in a data mining project, requiring many complex SQL queries, joining tables, and aggregating columns. Existing SQL aggregations have limitations to prepare data sets because they return one column per aggregated group. In general, a significant manual effort is required to build data sets, where a horizontal layout is required. This paper propose simple, yet powerful, methods to generate SQL code to return aggregated columns in a horizontal tabular layout, returning a set of numbers instead of one number per row. This new class of functions is called horizontal aggregations. Horizontal aggregations build data sets with a horizontal denormalized layout (e.g., point-dimension, observation variable, instance-feature), which is the standard layout required by most data mining algorithms. This paper gives three fundamental methods to evaluate horizontal aggregations: CASE: Exploiting the programming CASE construct; SPJ: Based on standard relational algebra operators (SPJ queries); PIVOT: Using the PIVOT operator, which is offered by some DBMS's. Experiments with large tables compare the proposed query evaluation methods. The CASE method has similar speed to the PIVOT operator and it is much faster than the SPJ method. In general, the CASE and PIVOT methods exhibit linear scalability, whereas the SPJ method does not.

"A Framework for Composing SOAP, Non-SOAP and Non-Web Services"[3]. Jonathan Lee, Shin-Jie Lee, and Ping-Feng Wang. Published in: IEEE 2013. Recently, there is a trend on developing mobile applications based on service-oriented architecture in numerous application domains, such as telematics and smart home. Although efforts have been made on developing composite SOAP services, little emphasis has been put on invoking and composing a combination of SOAP, non-SOAP, and non-web services into a composite process to execute complex tasks on various mobile devices. Main challenges are two-fold: one is how to invoke and compose heterogeneous web services with various protocols and content types, including SOAP, Restful, and OSGi services; and the other is how to integrate non-web services, like Web contents and mobile applications, into a composite service process. In this work, we propose an approach to invoking and composing SOAP, non-SOAP, and non-web services with two key features: an extended BPEL engine bundled with adapters to enable direct invocation and composition of SOAP, Restful and OSGi services based on Adapter pattern; and two transformation mechanisms devised to enable conversion of Web contents and Android activities into OSGi services. In the experimental evaluations, we demonstrate network traffic and turnaround time of our approach are better than those of the traditional ones.

"Service Selection for Web Services with Probabilistic QoS"[4]. San-Yih Hwang, Chien-Ching Hsu, Chien-Hs iang Lee, Member, IEEE Computer Society. Published in: IEEE 2013. In this paper Web service can be specified from two perspective, namely functional and non-functional properties. Multiple services may possess the same function while very in their non-functional properties, or called quality-of-services(QoS). QoS values are important criteria for service selection or recommendation. Most of the former works in Web service selection and recommendation treat the QoS values as constants. However, QoS values of a service as perceived by a given user are intrinsically random variables because QoS value prediction can never be precise and there are always some unobserved random effects. In this work, author have address the service selection problem by representing services. QoS values as discrete random variables with probability of satisfying constraints imposed on the composite service is high and the execution time is reasonable. This paper gives a method starts with an initial Web service assignment and incrementally adjusts it using simulated annealing. This paper conduct several experiments and the results show the authors approach generally performs better than previous work, such as the integer programming method and the cost-driven method.

"Methodology and Tools for End-to-End SOA Security Configurations"[5]. Fumiko Satoh, Michiaki Tatsubori, Yuichi Nakamura, Nirmal K. Mukhi, Kouichi Ono. The configuration of non-functional requirements, such as security, has become important for SOA applications, but the configuration process has not been discussed comprehensively. In current development processes, the security requirements are not considered in upstream phases and a developer at a downstream phase is responsible for writing the security configuration. However, configuring security requirements properly is quite difficult for developers because the SOA security is cross-domain and all required information is not available in the downstream phase. To resolve this problem, we clarify how to configure security in the SOA application development process, and define the developer's roles in each phase. Additionally, supporting technologies to generate security configurations are proposed: Model-Driven Security and Pattern-based Policy Configuration. This paper gives a methodology for end-to-end security configuration for SOA applications and tools for generating detailed security configurations from the requirements specified in upstream phases model transformations, making it possible to configure security properly without increasing developers' workloads.

"An Efficient Data Mining Framework on Hadoop using Java Persistence API"[6]. Yang Li, Shi ZhongZhi. Data

 ISSN: 2321-8169
4818 - 4822

indexing is common in data mining when working with high-dimensional, large-scale data sets. Hadoop, a cloud computing project using the Map Reduce framework in Jave, has become of significant interest in distributed data mining. A feasible distributed data indexing algorithm is proposed for Hadoop data mining, based on ZSCORE binning and inverted indexing and on the Hadoop Sequence File format. A data mining framework on Hadoop using the Java Persistence API(JPA) and MySQL Cluster is proposed. The framework is elaborated in the implementation of a decision tree algorithm on Hadoop. In this paper the data indexing algorithm with Hadoop Map File indexing , which performs a binary search , in a modest cloud environment. The result show the algorithm is more efficient than native Map File indexing. This paper compare the JDBC and JPA implementations of the data mining framework. The performance show the framework is efficient for data mining on Hadoop.

III. PROPOSED SYSTEM

The proposed method presented in this paper uses an encryption techniques. The core technologies for Web services framework include SOAP (simple object access protocol), WSDL (web services description language) and UDDI (universal description discovery and integration), and their expressions are standard XML documents. Service-Oriented Architecture (SOA) of Web services architecture as shows in Figure 1. Web Service Provider describes its provided services through the WSDL and informs Web services register server about the description. Registration server updated directory and release on Internet according to the description of WSDL and UDDI agreement. Users request register server before using Web services to access to Web services provided the address and service interface information, and then communicate through connections set up by the SOAP protocol and Web services providers. The core framework of the proposed web data mining system based on web service is similar to the Web data miming process system, including a Single Integrated User Interface, Web Data Mining Process Designer, and Web Data Mining Process Executor. Each activity in Web data mining process is viewed as a web service provided by Web data mining service providers on Internet. The Web data mining service providers offer common KDD functions for Web data per-processing, Web data mining algorithms, and visualization analysis. The goal of using web services in this research is to achieve universal interoperability between applications by using Web standards. We know Web Services use a loosely coupled integration model to allow flexible integration of heterogeneous systems in a variety of domains including business-to business, business-to-consumer and also enterprise application integration.

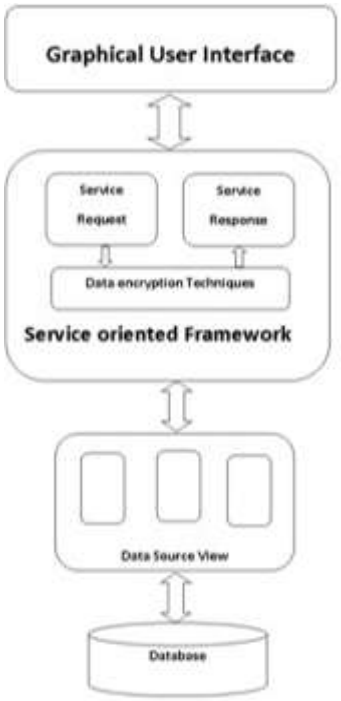

Fig 1. System Architecture

A. Algorithm

```
//Define p as the following
var int[64] c, k

c[ 0..15] := {7, 12, 17, 22,  7, 12, 17, 22,  7, 12, 17, 22,  7, 12, 17, 22}
c[16..31] := {5,  9, 14, 20,  5,  9, 14, 20,  5,  9, 14, 20,  5,  9, 14, 20}
c[32..47] := {4, 11, 16, 23,  4, 11, 16, 23,  4, 11, 16, 23,  4, 11, 16, 23}
c[48..63] := {6, 10, 15, 21,  6, 10, 15, 21,  6, 10, 15, 21,  6, 10, 15, 21}

//Initializing the variables:
var int h0 := 0x67452301
var int h1 := 0xEFCDAB89
var int h2 := 0x98BADCFE
var int h3 := 0x10325476

//Using binary integer part of the sine's of integers as constants:
for i from 0 to 63
    k[i] := floor(abs(sin(i + 1)) × 2^32)

//Pre-processing:

append "1" bit to message
```

 ISSN: 2321-8169
4818 - 4822

append "0" bits until message length in bits ≡ 448 (mod 512) append bit length of message as 64-bit little-endian integer to message

```
//Process the message in successive 512-bit chunks:
for each 512-bit chunk of message
break chunk into sixteen 32-bit little-endian words w(i), 0 ≤ i ≤ 15

//Initialize hash value for this chunk:
   var int a := h0
   var int b := h1
   var int c := h2
   var int d := h3


//Main loop:
   for i from 0 to 63
      if 0 ≤ i ≤ 15 then
         f := (b and c) or ((not b) and d)
         g := i
      else if 16 ≤ i ≤ 31
         f := (d and b) or ((not d) and c)
         g := (5×i + 1) mod 16
      else if 32 ≤ i ≤ 47
         f := b xor c xor d
         g := (3×i + 5) mod 16
      else if 48 ≤ i ≤ 63
         f := c xor (b or (not d))
         g := (7×i) mod 16


      temp := d
      d := c
      c := b
      b := ((a + f + k(i) + w(g)) left rotate r(i)) + b
      a := temp


//Add this chunk's hash to result so far:
   v0 := v0 + a
   v1 := v1 + b
   v2 := v2 + c
   v3 := v3 + d.
```

IV. EXPERIMENTAL SETUP

To carry out the functionality of the system we have used JDK1.7,Apache Tomcat, Eclipse IDE, MySQL as a Database.

On the home screen user will enter his details for the processing or as an input for this particular document. Here on home screen we ask to end user to enter his basic details. After this, once the User submitted his basic details on next screen user needs to enter his more details for processing. Internally system encrypts the user details and formulate this data into XML form to send it across the different system to another application.

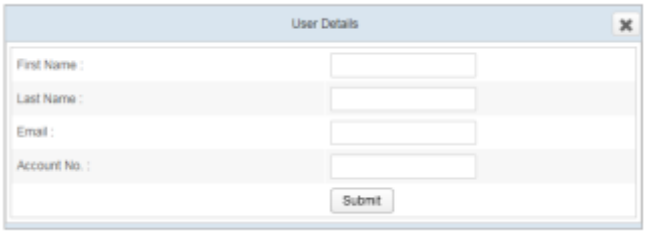

Fig 2. User Details Screen

With User Transaction Details user enters his more details for further processing. After clicking Next button data gets internally encrypted and formed into the appropriate format to send towards the next application.

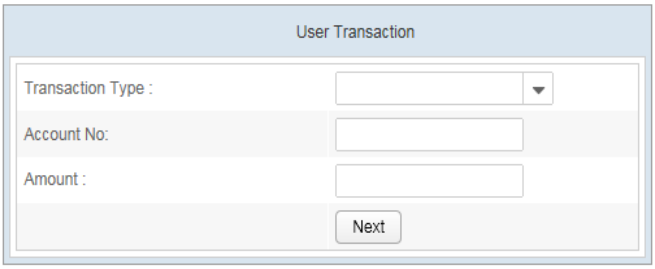

Fig 3. User Transaction Details

N	n	d	SPJ	PIVOT	CASE
12M	1K	7	62	58	59
		12	67	62	64
		25	68	61	61
12M	100K	7	86	81	81
		12	85	77	80
		25	103	89	92
12M	1.5M	7	230	157	155
		12	419	140	141
		25	1086	161	150

Fig 4. Performance analysis/comparison

V. CONCLUSION AND FUTUREWORK

Web data mining is a new upcoming flourishing technology with respect to increased development of internet. The structure of data available on web is complex, so the challenges with

data mining are how to integrate distributed and heterogeneous databases or applications. Handling the requirements through SOA applications have been applied and implemented currently, but the security on such data have not been well defined. With this research paper security configuration processes have been properly defined. Here this paper implements the security configuration and improve the system perform better and extend it to web structure mining.

VI. ACKNOWLEDGMENT

We would like to thank IJRITCC for giving such wonderful and useful platform for the PG students to publish their research work. Also would like to thanks to our guide & respected teachers for their constant support and motivation for us. Our sincere thanks to SKN Sinhgad Institute of Technology and Science for providing a strong platform to develop our skill and capabilities.

REFERENCES

[1] Joe M. Tekli, Member, Ernesto Damiani, Richard Chbeir, Gabriele Gianini, "SOAP Processing Performance and Enhancement", IEEE Transactions on Services Computing, Vol. 5, No.3, July-September 2012.

[2] Carlos Ordonez and Zhibo Chen, "Horizontal Aggregations in SQL to Prepare Data Sets for Data Mining Analysis", IEEE Transactions on Knowledge and Data Engineering, 2012.

[3] Jonathan Lee, Shin-Jie Lee, and Ping-Feng Wang, "A Framework for Composing SOAP, Non-SOAP and Non-Web Services". DOI 10.1109/TSC.2014.2310213, IEEE Transactions on Services Computing.

[4] San-Yih Hwang, Chien-Ching Hsu, Chien-Hs iang Lee, "Service Selection for Web Services with Probabilistic QoS", DOI 10.1109/TSC.2014.2338851, IEEE Transactions on Services Computing.

[5] Fumiko Satoh, Michiaki Tatsubori, Yuichi Nakamura, Nirmal K. Mukhi, Kouichi Ono, "Methodology and Tools for End-to-End SOA Security Configurations", 2008 IEEE Congress on Service 2008-Part I.

[6] Yang Li, Shi ZhongZhi, "An Efficient Data Mining Framework on Hadoop using Java Persistence API", 2010 10th IEEE International Conference on Computer and Information Technology (CIT 2010).

[7] N. Abu-Ghazaleh and M.J. Lewis, Differential Deserialization for Optimized SOAP Performance, Proc. ACM/IEEE Conf. Super- computing, pp. 21-31, 2005.

[8] N.Abu-Ghazaleh, M.J.Lewis, and M.Govindaraju, "Performance of Dynamic Resizing of Message Fields for Differentisl Serialization of SOAP Message," Proc. Int'l Symp. Web Service and Applications. Pp. 783-789.2004

[9] N. Abu-Ghazaleh, M. Govindaraju, and M.J. Lewis, "Optimizing Performance of Web Services with Chunk-Overlaying and Pipelined-Send," Proc. Int'l Conf. Internet Computing (ICIC), pp. 482-485,2004.

[10] T. Bray, J. Paoli, C. Sperberg-McQueen, Y Mailer, and F. Yergeau, Extensible Markup Language(XML) 1.0, fifth ed. W3C, http://www.w3.org/TR/REC-xml, Nov.2008.

[11] R. Chinnici, J.J. Moreau, A Ryman, and S. Weerawarana, Web Service Description Language (WSDL) Version 2.0 Part 1:Core Language, W3C recommendation, http://www.w3.org/TR/wsdl20, Aug.2009

[12] M.G. Kostoulas, M. Matsa, N. Mendelsohn, E. Perkins, A. Heifets, and M. Mercaldi, "XML Screamer: An Integrated Approach to High Performance XML Parsing, Validation and Deserialization," Proc. 15th Int'l Conf. World Wide Web (WWW '06), pp. 93-102, 2006.

[13] Web Services Security: SOAP Message Security 1.1, http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf.

[14] IBM and Microsoft, "Security in a Web Services World: A Proposal Architecture and Roadmap", http://www.ibm.com/developerworks/webservices/library/specification/ws-secmap/, 2002.