ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

International Journal of Advanced Research in Management, Architecture, Technology and Engineering
(IJARMATE) Vol. 3, Special Issue 11, March 2017

Mining User-Aware Rare Sequential Topic Patterns in Document Streams

M.Sowmiya[1], R.Yasir Abdullah M.E.,[2]

[1]Department of Computer Science, Sri Subramanya College of Engineering and Technology, Palani

[2]Department of Computer Science, Sri Subramanya College of Engineering and Technology, Palani

[1]sowmiya.muruganantha@gmail.com

Abstract : *To characterize and personalized and abnormal behaviors of internet users we propose Sequential Topic Patterns(STP) and formulate the problem of mining of User-Aware Rare Sequential Topic Patterns in document streams on the internet.They are rare on the whole but rela tively frequent for specific users, so can be applied in many real-life scenarios, such as real-time monitoring on abnor mal user behaviors.To solve this data mining problem through the three phases they are :Preprocess of the twitter data set,Topic Extraction and Session Identification of different users and Applying URSTP on the users.Experiments on both real(Twitter) and synthetic dataset show our approach can indeed discover special users and interpretable URSTPs effectively and efficiently.*

1.INTRODUCTION:

The Documentstreams are created and distributed in vari-ous forms on the Internet, such as news streams,emails, micro-blog articles, chatting messages, research paper archives, web forum discussions, and so forth. The contents of these documents generally concentrate on some specific topics, which reflect offline social events and users' characteristics in real life. To mine these pieces of informa-tion, a lot of researches of text mining focused on extracting topics from document collections and document streams through various probabilistic topic models.In order to characterize user behaviors in published doc-ument streams, we study on the correlations among topics extracted from these documents, especially the sequential relations, and specify them as Sequential Topic Patterns (STPs) . Each of them records the complete and repeated behavior of a user when she is publishing a series of docu-ments, and are suitable for inferring users' intrinsic charac-teristics and psychological statuses. First, compared to individual topics, STPs capture both combinations and orders of topics, so can serve well as discriminative units of seman-tic association among documents in ambiguous situations. Second, compared to document-based patterns, topic-based patterns contain abstract information of document contents and are thus beneficial in clustering similar documents and finding some regularities about Internet users. Third, the probabilistic description of topics helps to maintain and accu-mulate the uncertainty degree of individual topics, and can thereby reach high confidence level in pattern matching for uncertain data.

It is worth noting that the ideas above are also applicable for another type of document streams, called browsed docu-ment streams, where Internet users behave as readers of documents instead of authors. In this case, STPs can charac-terize complete browsing behaviors of readers, so compared to statistical methods, mining URSTPs can better discover special interests and browsing habits of Internet users, and is thus capable to give effective and context-aware recom-mendation for them. While, this paper will concentrate on published document streams and leave the applications for recommendation to future work.To solve this innovative and significant problem of mining URSTPs in document streams, many new technical challenges are raised and will be tackled in this paper. First, the input of the task is a textual stream, so existing techniques of sequen-tial pattern mining for probabilistic databases cannot be directly applied to solve this problem. A preprocessing phase is necessary and crucial to get abstract and probabilistic descriptions of documents by topic extraction, and then to rec-ognize complete and repeated activities of Internet users by session identification. Second, in view of the real-time require-ments in many applications, both the accuracy and the effi-ciency of mining algorithms are important and should be taken into account, especially for the probability computation process. Third, different from frequent patterns, the user-aware rare pattern concerned here is a new concept and a for-mal criterion must be well defined, so that it can effectively characterize most of personalized and abnormal behaviors of Internet users, and can adapt to different application scenar-ios. And correspondingly, unsupervised mining algorithms for this kind of rare patterns need to be designed in a manner different from existing frequent pattern mining algorithms.

To sum up, this paper makes the following contributions:

- To the best of our knowledge, this is the first work that gives formal definitions of STPs as well as their rarity measures, and puts forward the problem ofmining URSTPs in document streams, in order tocharacterize and detect personalized and abnormalbehaviors of Internet users;
- We propose a framework to pragmatically solve thisproblem, and design corresponding algorithms tosupport it. At first, we give preprocessing proce-dures with heuristic methods for topic extractionand session identification. Then, borrowing the ideasof pattern-growth in uncertain environment, twoalternative algorithms are designed to discover allthe STP candidates with support values for each user. That provides a trade-off between accuracyand efficiency. At last, we present a user-aware rarityanalysis algorithm according to the formally

definedcriterion to pick out URSTPs and associated users.

- We validate our approach by conducting experi-ments on both real and synthetic datasets

1.1 EXISTING SYSTEM

Topic mining in document collections has been extensively studied in the literature. Topic Detection and Tracking (TDT) t ask a imed to detect and track topics(events) in news streams with clustering-based techniques on keywords. Considering the co-occurrence of words and their semantic associations, a lot of probabilistic generative models for extracting topics from documents w ere also proposed, such as PLSI , LDA and t heir extens ions integrating differen t feature s of documents a s well as model s for s hort texts like Twitter-LDA. In many real applications, document collections gener-ally carry temporal information and can thus be considered as document streams. Various dynamic topic modeling methods have been proposed to discover topics over time in document streams and then to predictoffline social events. However, these methods were designed to construct the evolution model of individual topics from a document stream, rather than to analyze the correlations among multiple topics extracted from suc-cessive documents for specific users.

Sequenti al pattern mining i s an important problem in data mining, and has also b een w ell studied so far. In the context of deter ministic data, a comprehensive survey can be found in project. The c onc ept support is the most popular measure for evaluating the frequency of asequential pattern, a nd is defined as th e number or proportion of data sequences containing the pattern in the target database. Many mining algo ri thms have been proposed b ased on support, such as They discovered frequentsequential patterns whose support values are not lessthan a user-defined threshold, and were extended bySLPM iner to deal with length-decreasing support constraints. Nevertheless, the obtained patterns are notalways interesting for our purpose, because those rarebut significan tp atterns representing personalized and abnormal behaviors are pru ned due to low supports.Furthermore, the algorithms on deterministic databases isnot applicable for document streams, as they failed to handle the uncert ainty in topics.

For uncertain data, most of existing works studied fre-quent itemset mining in probabilistic databases butcomparatively fewer researches addressed the problem ofsequential pattern mining. Muzammal et al. focused onsequence-level uncertainty in sequential databases, and pro-posed methods to evaluate the frequency of a sequentialpattern based on expected support , in the frame of candidategenerate-and-test or pattern-growth. Sinceexpected support would lose the probability distribution ofthe support, a finer measure frequentness probability wasdefined for general itemsets and used in min-ing frequent sequential patterns for sequence-level and ele-ment-level uncertain databases. However,these works did not consider where the uncertain databasescome from and how the probabilities in the original data arecomputed, so cannot be directly employed for our problemwhich takes document streams as input.

1.1.1 DISADVANTAGES

- Difficult to find sequential pattern mining because cannot be applied directly.
- The algorithms on deterministic databases is not applicable for document streams, as they failed to handle the uncertainty in topics.
- Sequential pattern mining is an important problem in data mining. In the context of deterministic data, a comprehensive survey can be found.

1.2 PROPOSED SYSTEM

In our proposed system,Since the problem of mining URSTPs in document streamsproposed in this paper is innovative, there are no other com-plete and comparable approaches for this task as the base-line,but the effectiveness of our approach in discoveringpersonalized and abnormal behaviors, especially the rea-sonability of the URSTP definition, needs to be practicallyvalidated. In this section, we conduct interesting and infor-mative experiments on message streams in Twitter datasets,to show that most of users discovered by our approach areactually special in real life, and the mined URSTPs canindeed capture personalized and abnormal behaviors ofInternet users in an understandable way.

The experiments conducted on both real(Twitter) and synthetic datasets demonstrate that the pro-posed approach is very effective and efficient in discoveringspecial users as well as interesting and interpretableURSTPs from Internet document streams, which can wellcapture users' personalized and abnormal behaviors andcharacteristics.

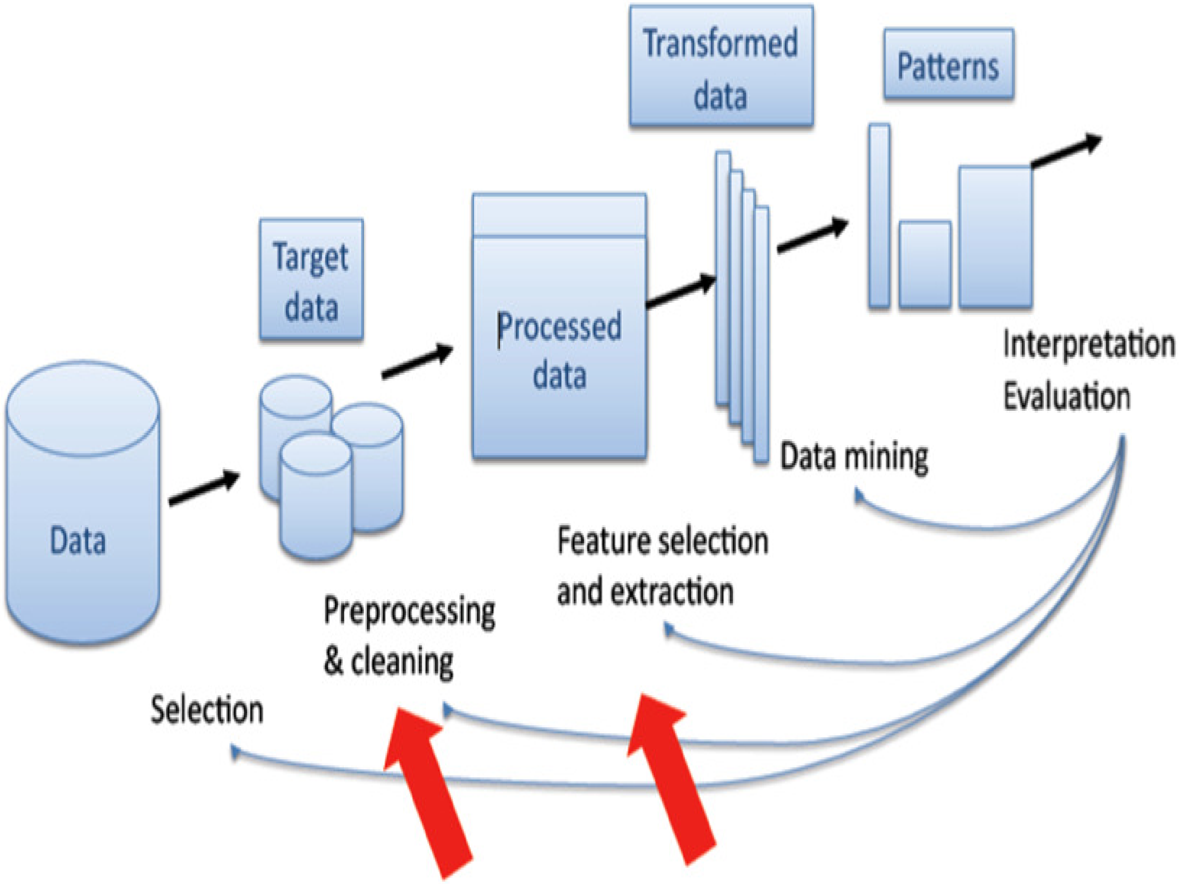
As this paper puts forward an innovative research direc-tion on Web data mining, much work can be built on it in the future. At first, the problem and the approach can alsobe applied in other fields and scenarios. Especially forbrowsed document streams, we can regard readers of docu-ments as personalized users and make contextaware rec-ommendation for them. Also, we will refine the measures ofuser-aware rarity to accommodate different requirements improve the mining algorithms mainly on the degree of par-allelism, an study on-the-fly algorithms aiming at real-time document streams.

1.2.1 ADVANTAGES

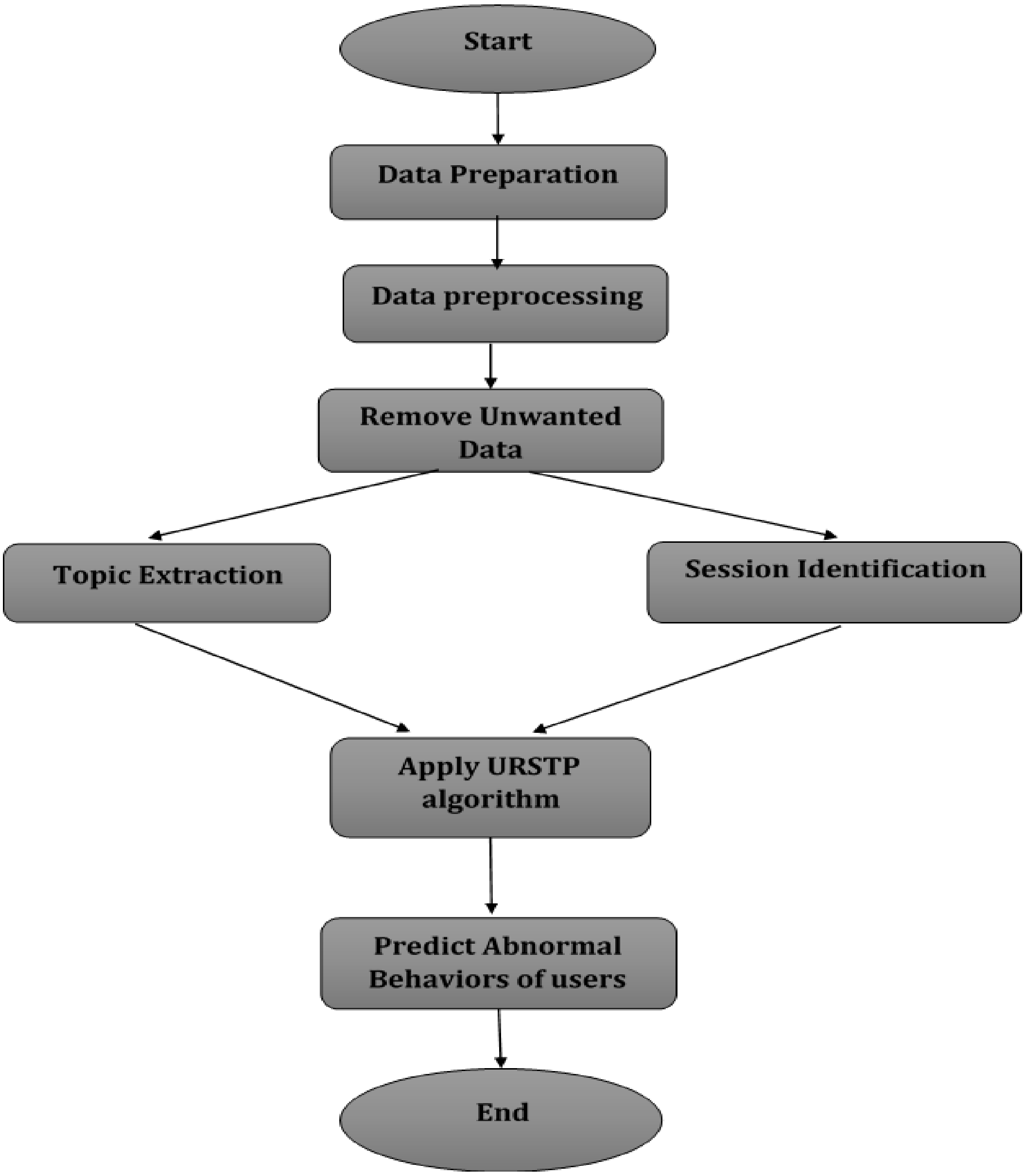
- To discover special users using URSTPs effectively and efficiently.
- Significantly reflect users' characteristics as well as capture users' personalized and abnormal behaviors and characteristics.

YSTEM DESIGNS

1.3.1SYSTEM ARCHITECTURE

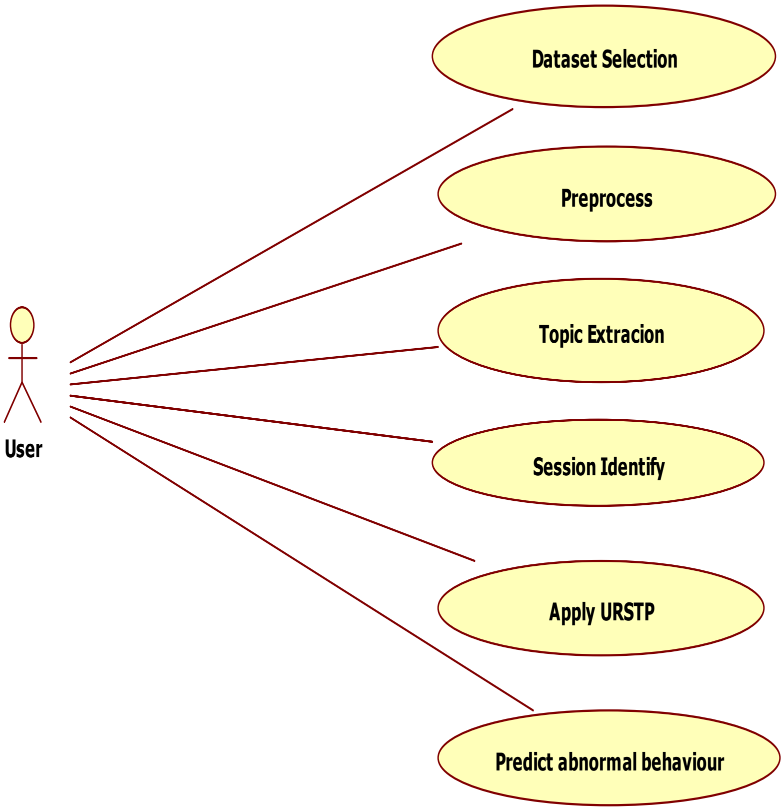


1.3.2 FLOW DIAGRAM



1.3.3UML DIAGRAM

1.3.3.1 USE CASE DIAGRAM

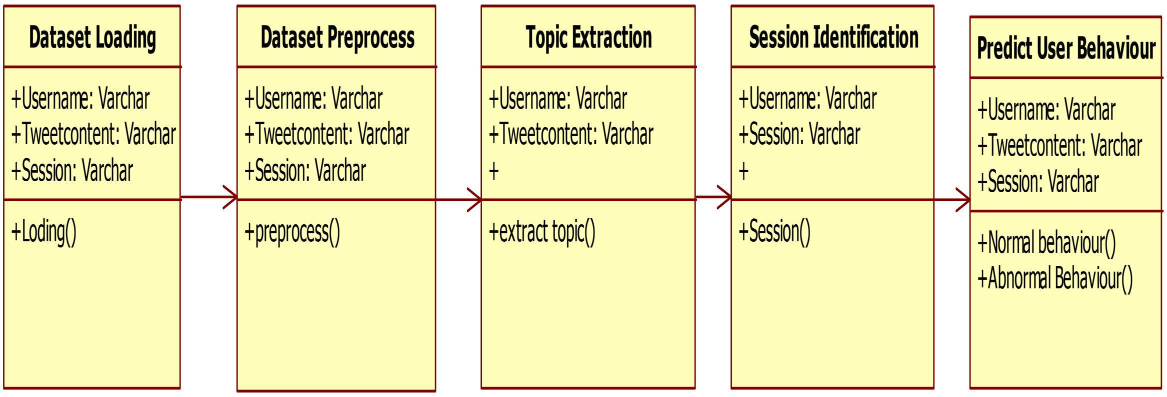


ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

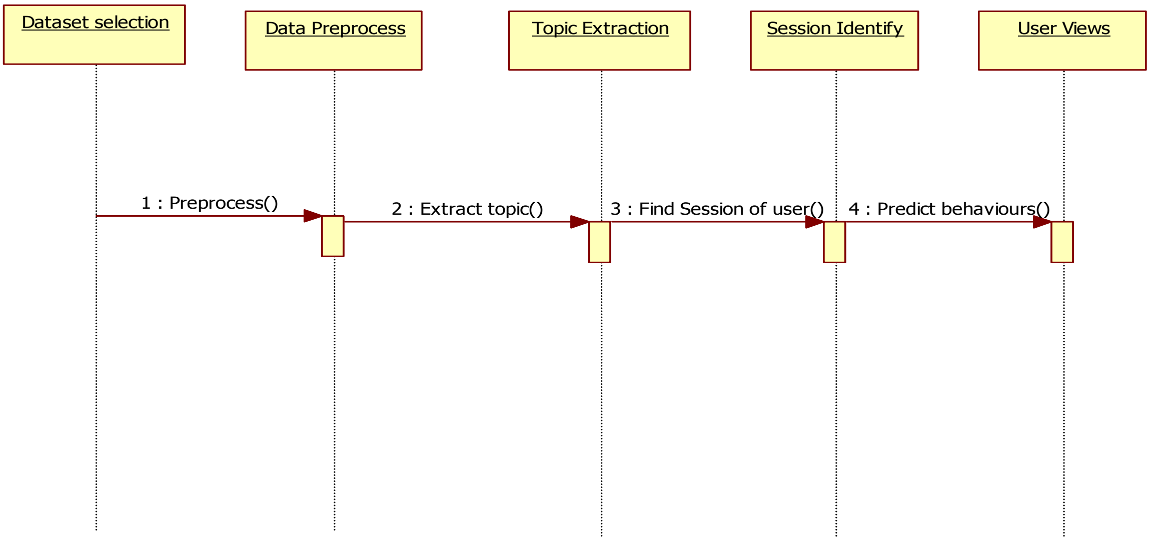
IJARMATE Your ulti-MATE Research Paper !!!

International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE) Vol. 3, Special Issue 11, March 2017

1.3.3.2 CLASS DIAGRAM



1.3.3.3 SEQUENCE DIAGRAM



1.4. TESTING OF PRODUCT

1.4.1SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

1.4.2**TYPES OF TESTS**

1.4.2**.1 Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

1.4.2**.2 Functional test**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

1.4.2**.3 System Test**

System testing ensures that the entire integrated software system meets requirement. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

1.4.2.4 White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

1.4.2.5 Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document.

1.4.2.5.1Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

1.4.2.6 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

1.4.2.7 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

1.4.3Test Results: All the test cases mentioned above passed successfully. No defects encountered.

1.5. MODULES

- Dataset Loading
- Preprocess Phase
- Topic Extraction and session identification of users
- Apply URSTP algorithm
- Predict Abnormal Behaviors of user

1.5.1 MODULES DESCRIPTION

1.5.1.1 Data Loading

In this module, we choosing a twitter dataset in this dataset contains username, twit content that means thoughts of the different users and that session of the particular users also. This dataset contains attributes like username ,twit content and session .These details are load and store in to the database.

1.5.1.2Preproocessing Phase

A preprocessing phase is necessary and crucial to get abstract and probabilistic descriptions of documents from the twitter dataset.The twitter dataset contains any unwanted data's that are removed from this phase.The main aim of this module is detect and remove unwanted data's from twitter dataset.

1.5.1.3 Topic Extraction and Session Identification

Topic extraction and then to recognize complete and repeated activities of Internet users by session identification.This module is used to find what type of topic is talk about the users and the session of the all user is displayed.These details are fetched to every one users has which session and what they are talked about that topic.

1.5.1.4 Apply URSTP Algorithm

The problem of mining URSTPs is defined more formally and systematically, and the application field focuses on published document streams.The preprocessing strategies including topic extraction and session identification are presented in detail, where several heuristic methods are discussed.Experiments are conducted for new algorithms on more real Twitter datasets and more generalized synthetic datasets, and quantitative results for the real case are given to validate our approach.

1.5.1.5 Predict Abnormal Behaviors of users

Next, this modules are categorized in two behaviors normal and abnormal of the tweet content via users emotions.The problem of mining URSTPs in document streams, in order to characterize and detect personalized and abnormal behaviors of Internet users.Given a (published) document stream on the Internet, discover all URSTPs and associated users, which characterize users' personalized and abnormal behaviors.

2. REQUIREMENT SPECIFICATION

2.1 SOFTWARE REQUIREMENT

O/S	:	Windows XP.
Language	:	Java.
Tool	:	Wampserver
IDE	:	NetBeans 7.2.1
Data Base	:	MySql

2.2 HARDWARE REQUIREMENT

System	:	Pentium dual core
Hard Disk	:	160 GB
Monitor	:	15 VGA color
Mouse	:	Logitech.
Keyboard	:	110 keys enhanced

3. HARDWARE REQUIREMENT

3.1 Java

Java is a programming language originally developed by James Gosling at Sun Microsystems (now a subsidiary of Oracle Corporation) and released in 1995 as a core component of Sun Microsystems' Java platform. The language derives much of its syntax from C and C++ but has a simpler object model and fewer low-level facilities. Java applications are typically compiled to byte code (class file) that can run on any Java Virtual Machine (JVM) regardless of computer architecture. Java is a general-purpose, concurrent, class-based, object-oriented language that is specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere." Java is currently one of the most popular programming languages in use, particularly for client-server web applications.

The original and reference implementation Java compilers, virtual machines, and class libraries were developed by Sun from 1995. As of May 2007, in compliance with the specifications of the Java Community Process, Sun relicensed most of its Java technologies under the GNU General Public License. Others have also developed alternative implementations of these Sun technologies, such as the GNU Compiler for Java and GNU Class path.

3.2 Java Platform:

One characteristic of Java is portability, which means that computer programs written in the Java language must run similarly on any hardware/operating-system platform. This is achieved by compiling the Java language code to an intermediate representation called Java byte code, instead of directly to platform-specific machine code. Java byte code instructions are analogous to machine code, but are intended to be interpreted by a virtual machine (VM) written specifically for the host hardware. End-users commonly use a Java Runtime Environment (JRE) installed on their own machine for standalone Java applications, or in a Web browser for Java applets. Standardized libraries provide a generic way to access host-specific features such as graphics, threading, and networking.

A major benefit of using byte code is porting. However, the overhead of interpretation means that interpreted programs almost always run more slowly than programs compiled to native executable would. Just-in-Time compilers were introduced from an early stage that compiles byte codes to machine code during runtime.

Just as application servers such as Glass Fish provide lifecycle services to web applications, the Net Beans runtime container provides them to Swing applications. Application servers understand how to compose web modules, EJB modules, and so on, into a single web application, just as the Net Beans runtime container understands how to compose Net Beans modules into a single Swing application.

3.3 Java a High-level Language:

A high-level programming language developed by Sun Microsystems. Java was originally called OAK, and was designed for handheld devices and set-top boxes. Oak was unsuccessful so in 1995 Sun changed the name to Java and modified the language to take advantage of the burgeoning World Wide Web.

Java is an object-oriented language similar to C++, but simplified to eliminate language features that cause common programming errors. Java source code files (files with a .java extension) are compiled into a format called byte code (files with a .class extension), which can then be executed by a Java interpreter. Compiled Java code can run on most computers because Java interpreters and runtime environments, known as Java Virtual Machines (VMs), exist for most operating systems, including UNIX, the Macintosh OS, and Windows. Byte code can also be converted directly into machine language instructions by a just-in-time compiler (JIT).

Java is a general purpose programming language with a number of features that make the language well suited for use on the World Wide Web. Small Java applications are called Java applets and can be downloaded from a Web server and run on your computer by a Java-compatible Web browser, such as Netscape Navigator or Microsoft Internet Explorer.

Object-oriented software development matured significantly during the past several years. The convergence of object-oriented modeling techniques and notations, the development of object-oriented frameworks and design patterns, and the evolution of object-oriented programming languages have been essential in the progression of this technology.

Object-Oriented Software Development using Java: Principles, Patterns, and Frameworks contain a much applied focus that develops skills in designing software-particularly in writing well-designed, medium-sized object-oriented programs. It provides a broad and coherent coverage of object-oriented technology, including object-oriented modeling using the Unified Modeling Language (UML) object-oriented design using Design Patterns, and object-oriented programming using Java.

3.4 Net Beans

The **Net Beans Platform** is a reusable framework for simplifying the development of Java Swing desktop applications. The Net Beans IDE bundle for Java SE contains what is needed to start developing Net Beans plug-in and Net Beans Platform based applications; no additional SDK is required.

Applications can install modules dynamically. Any application can include the Update Center module to allow users of the application to download digitally-signed upgrades and new features directly into the running application. Reinstalling an upgrade or a new release does not force users to download the entire application again.

The platform offers reusable services common to desktop applications, allowing developers to focus on the logic specific to their application. Among the features of the platform are:

- User interface management (e.g. menus and toolbars)
- User settings management
- Storage management (saving and loading any kind of data)
- Window management
- Wizard framework (supports step-by-step dialogs)
- Net Beans Visual Library
- Integrated Development Tools

3.5 J2EE

A **Java EE application** or a **Java Platform, Enterprise Edition application** is any deployable unit of Java EE functionality. This can be a single Java EE module or a group of modules packaged into an EAR file along with a Java EE application deployment descriptor. Java EE applications are typically engineered to be distributed across multiple computing tiers.

Enterprise applications can consist of the following:

- EJB modules (packaged in JAR files);
- Web modules (packaged in WAR files);
- connector modules or resource adapters (packaged in RAR files);
- Session Initiation Protocol (SIP) modules (packaged in SAR files);
- application client modules;
- Additional JAR files containing dependent classes or other components required by the application;

Java Platform, Enterprise Edition or **Java EE** is Oracle's enterprise java computing platform. The platform provides an API and runtime environment for developing and running enterprise software, including network and web services, and other large-scale, multi-tiered, scalable, reliable, and secure network applications. Java EE extends the Java Platform, Standard Edition (Java SE), providing an API for fault-tolerance, object-relational mapping, distributed and multi-tier architectures, and web services. The platform incorporates a design based largely on modular components running on an application server. Software for Java EE is primarily developed in the Java programming language and uses XML for configuration.

4. BACK END TECHNOLOGY:

4.1 Wamp Server

WAMPs are packages of independently-created programs installed on computers that use a Microsoft Windows operating system. WAMP is an acronym formed from the initials of the operating system Microsoft Windows and the principal components of the package: Apache, MySQL and one of PHP, Perl or Python.

Apache is a web server. MySQL is an open-source database. PHP is a scripting language that can manipulate information held in a database and generate web pages dynamically each time content is requested by a browser. Other programs may also be included in a package, such as PhpMyAdmin which provides a graphical user interface for the MySQL database manager, or the alternative scripting languages Python or Perl. Equivalent packages are MAMP (for the Apple Mac) and LAMP (for the Linux operating system).

4.2 MySQL

The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation.

Free-software-open source projects that require a full-featured database management system often use MySQL. For commercial use, several paid editions are available, and offer additional functionality. Applications which use MySQL databases include: TYPO3, Joomla, WordPress, phpBB, Drupal and other software built on the LAMP software stack. MySQL is also used in many high-profile, large-scale World Wide Web products, including Wikipedia, Google (though not for searches), Facebook, and Twitter.

4.3 Platforms and interfaces

MySQL works on many different system platforms, including AIX, BSDi, FreeBSD, HP-UX, e ComStation, i5/OS, IRIX, Linux, Mac OS X, Microsoft Windows, Net BSD, Novell NetWare, Open BSD, Open Solaris, OS/2 Warp, QNX, Solaris, Symbian, SunOS, SCO Open Server, SCO UnixWare, Sanos and Tru64. A port of MySQL to OpenVMS also exists.

Many programming languages with language-specific APIs include libraries for accessing MySQL databases. These include MySQL Connector/Net for integration with Microsoft's Visual Studio (languages such as C# and VB are most commonly used) and the JDBC driver for Java. In addition, an ODBC interface called MyODBC allows additional programming languages that support the ODBC interface to communicate with a MySQL database, such as ASP or ColdFusion. The HTSQL - URL-based query method also ships with a MySQL adapter, allowing direct interaction between a MySQL database and any web client via structured URLs. The MySQL server and official libraries are mostly implemented in ANSI C/ANSI C++.

5. CONCLUSION:

Mining URSTPs in published document streams on theInternet is a significant and challenging problem. It formu-lates a new kind of complex event patterns based on docu-ment topics, and has wide potential application scenarios,such as real-time monitoring on abnormal behaviors ofInternet users. In this paper, several new concepts and themining problem are formally defined, and a group of algorithms are designed and combined to systematically solvethis problem. The experiments conducted on both real(Twitter) and synthetic datasets demonstrate that the proposed approach is very effective and efficient in discoveringspecial users as well as interesting and interpretableURSTPs from Internet document streams, which can wellcapture users' personalized and abnormal behaviors andcharacteristics.

7.REFERENCES:

[1] C. C. Aggarwal, Y. Li, J. Wang, and J. Wang, "Frequent patternmining with uncertain data," in Proc. ACM SIGKDD , 2009, pp. 29–38.

[2] R. Agrawal and R. Srikant, "Mining sequential patterns," in Proc.IEEE Int. Conf. Data Eng. , 1995, pp. 3–14.

[3] J. Allan, R. Papka, and V. Lavrenko, "On-line new event detectionand tracking," in Proc. 21st Annu. Int. ACM SIGIR Conf. Res.Develop. Inf. Retrieval, 1998, pp. 37–45.

[4] T. Bernecker, H.-P. Kriegel, M. Renz, F. Verhein, and A. Zuefle,"Probabilistic frequent itemset mining in uncertain databases," inProc. ACM SIGKDD , 2009, pp. 119–128.

[5] D. Blei and J. Lafferty, "Correlated topic models," Adv. Neural Inf.Process. Syst., vol. 18, pp. 147–154, 2006.

[6] D. M. Blei and J. D. Lafferty, "Dynamic topic models," in Proc.ACM Int. Conf. Mach. Learn., 2006, pp. 113–120.

[7] D. Blei, A. Ng, and M. Jordan, "Latent dirichlet allocation," J.Mach. Learn. Res., vol. 3, pp. 993–1022, 2003.

[8] J. Chae, D. Thom, H. Bosch, Y. Jang, R. Maciejewski, D. S. Ebert,and T. Ertl, "Spatiotemporal social media analytics for abnormalevent detection and examination using seasonal-trend decom-position," in Proc. IEEE Conf. Vis. Anal. Sci. Technol. , 2012, pp. 143–152.

[9] K. Chen, L. Luesukprasert, and S. T. Chou, "Hot topic extractionbased on timeline analysis and multidimensional sentence mod-eling," IEEE Trans. Knowl. Data Eng. , vol. 19, no. 8, pp. 1016–1025,Aug. 2007.

[10] C. K. Chui and B. Kao, "A decremental approach for mining fre-quentitemsets from uncertain data," in Proc. 12th Pacific-AsiaConf. Adv. Knowl. Discovery Data Mining, 2008, pp. 64–75.

[11] W. Dou, X. Wang, D. Skau, W. Ribarsky, and M. X. Zhou,"LeadLine: Interactive visual analysis of text data through eventidentification and exploration," in Proc. IEEE Conf. Vis. Anal. Sci.Technol., 2012, pp. 93–102.

[12] G. P. C. Fung, J. X. Yu, P. S. Yu, and H. Lu, "Parameter free burstyevents detection in text streams," in Proc. 31st Int. Conf. Very LargeData Bases, 2005, pp. 181–192.

[13] J. Han, J. Pei, B. Mortazavi-Asl, Q. Chen, U. Dayal, and M. Hsu,"FreeSpan: frequent pattern-projected sequential pattern mining,"in Proc. 6th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining ,2000, pp. 355–359.

[14] N. Hariri, B. Mobasher, and R. Burke, "Context-aware music rec-ommendation based on latent topic sequential patterns," in Proc.6th ACM Conf. Recommender Syst., 2012, pp. 131–138.

[15] T. Hofmann, "Probabilistic latent semantic indexing," in Proc.22nd Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 1999,pp. 50–57.