International Journal of Soft Computing and Engineering (IJSCE)
ISSN: 2231-2307, Volume-1, Issue-6, January 2012

Web Users Session Analysis Using DBSCAN and Two Phase Utility Mining Algorithms

G. Sunil Kumar, C.V.K Sirisha, Kanaka Durga.R, A.Devi

Abstract

One of the important issues in data mining is the interestingness problem. Typically, in a data mining process, the number of patterns discovered can easily exceed the capabilities of a human user to identify interesting results. To address this problem, utility measures have been used to reduce the patterns prior to presenting them to the user. A frequent itemset only reflects the statistical correlation between items, and it does not reflect the semantic significance of the items. This proposed approach uses a utility based itemset mining approach to overcome this limitation. This proposed system first uses Dbscan clustering algorithm which identifies the behavior of the users page visits, order of occurrence of visits. After applying the clustering technique High Two phase utility mining algorithm is applied, aimed at finding itemsets that contribute high utility.Mining web access sequences can discover very useful knowledge from web logs with broad applications. Mining useful Web path traversal patterns is a very important research issue in Web technologies. Knowledge about the frequent Web path traversal patterns enables us to discover the most interesting Websites traversed by the users. However, considering only the binary (presence/absence) occurrences of the Websites in the Web traversal paths, real world scenarios may not be reflected. Therefore, if we consider the time spent by each user as a utility value of a website, more interesting web traversal paths can be discovered using proposed two-phase algorithm. User page visits are sequential in nature. In this paper MSNBC web navigation dataset is used to compare the efficiency and performance in web usage mining is finding the groups which share common interests General Terms Web session mining, log analysis.

Keywords - Webusage Mining, Itemset, DBScan, Association rules.

1. INTRODUCTION

The World Wide Web serves as a huge, widely distributed, global information service center. It contains a rich and dynamic collection of hyperlink information and Web page access and usage information. Data mining, which can automatically discover useful and understandable patterns from massive data sets, has been widely exploited on the Web. Web mining can be broadly divided into three categories, i.e. content mining, usage mining, and link structure mining [1].

Manuscript received December 24, 2011.

G. Sunil Kumar, Asst.Professor, Dept.of Computer Applications, Maris Stella College, Vijayawada, A.P., India(e-mail:grand1sunil@gmail.com).

Weblog mining is a special case of usage mining, which mines Weblog entries to discover user traversal patterns of Web pages.

A Web server usually registers a log entry for every access of a Web page. Each entry includes the URL requested, the IP address where which the request originated, timestamp, etc. popular Websites, such as Web-based e-commerce servers, may register entries in the order of hundreds of megabytes every day. Data mining can be performed on Weblog entries to find association patterns, sequential patterns, and trends of Web accessing. Analyzing and exploring regularities in Weblog entries can identify potential customers for electronic commerce, enhance the quality of Internet information services, improve the performance of Web service system, and optimize the site architecture to cater to the preference of end users. One of the goals of Weblog mining is to find the frequent path traversal patterns in a Web environment. Path traversal pattern mining is to find the paths that frequently co-occurred. It firstly converts the original sequence of log data into a set of traversal subsequences. Each traversal subsequence represents a maximal forward reference from the starting point of a user access. Secondly, a sequence mining algorithm will be applied to determine the frequent traversal patterns, called large reference sequences, from the maximal forward references, where a large reference sequence is a reference sequence that occurs frequently enough in the database.

The problem of clustering has become increasingly important in recent years. The clustering problem has been addressed in many contexts and by researchers in many disciplines; this reflects its broad appeal and usefulness as one of the steps in exploratory data analysis. Clustering approaches aim at partitioning a set of data points in classes such that points that belong to the same class are more alike than points that belong to different classes. These classes are called clusters and their number may be preassigned or can be a parameter to be determined by the algorithm. There exist applications of clustering in such diverse fields as business, pattern recognition, communications, biology, astrophysics and many others. Cluster analysis is the organization of a collection of patterns (usually represented as a vector of measurements, or a point in a multidimensional space) into clusters based on similarity. Usually, distance measures are utilized. Data clustering has its roots in a number of areas; including data mining, machine learning, biology, and statistics. Traditional clustering algorithms can be classified into two main categories: hierarchical and partitional. In hierarchical clustering, the number of clusters need not be specified a priori, and problems due to initialization and local minima do not arise. However, since hierarchical methods consider only local neighbors in each step, they cannot incorporate a priori knowledge regarding the global shape or size of clusters. As a result, they cannot always separate overlapping clusters. In addition, hierarchical clustering is static, and points committed to a given cluster in the early stages cannot move to a different cluster. Clustering is a widely used technique in data mining application for discovering patterns in underlying data. Most traditional clustering algorithms are limited in handling datasets that contain categorical attributes. However, datasets with categorical types of attributes are common in real life data mining problem. In traditional models, all the Web pages in a database are treated equally by only considering if a Web is present in a traversal path or not. We demonstrate the interesting paths we observed in our experiments, as well as their significance to the decision making process. This rest of this paper is organized as follows. Section 2 overviews the related work. In Section 3, we introduce the technical terms in utility mining model. In Section 4, we present our proposed utility-based path traversal pattern mining algorithm. Section 5 presents the experimental results.

2. RELATED WORK

In the past ten years, a lot of research work has been done to discover meaningful information from large scale of Web server access logs. A Web mining system, called WEBMINER is presented in [2]. A knowledge discovery tool, WebLogMiner, is developed to mine Web server log files [4]. By applying OLAP and some data mining technology on Web logs, WebLogMiner can discover frequent Web access patterns and trends. A Web Utilization Miner (WUM) [4] provides a robust mining language in order to specify characteristics of discovered frequent paths that are interesting to the analysts. In this approach, individual navigation paths, called trails, are combined into an aggregated tree structure. Queries can be answered by mapping them into the intermediate nodes of the tree structure. The common assumption of the frequent traversal path mining methods is that each Web page is considered equal in weight. M.-S. Chen et al. [4] proposed two ARM-based (Association Rules Mining) algorithms, full-scan (FS) and selective-scan (SS). Either model does not reflect the semantic significance of different sequences except the statistical correlation of the sequences. Models that can reflect both statistical correlation and semantic significance of different sequences are in demand. A utility mining model is proposed in [4]. It allows a user to express the significance, interests or user preference of itemsets as subjective values. The objective value of an item is defined according to the information stored in a transaction, like the quantity of the item sold in the transaction. The utility of an item/itemset is based on both objective value and subjective value. Intuitively, utility is a quantitative measure of how “useful” (i. e. “profitable”) an itemset is. In practice, it can be profit, cost, or any measure of user preferences. Due to the lack of Apriori property, utility mining is very computational intensive. Y. Liu et al. proposed a Two-Phase algorithm. It substantially reduces the search space and the memory cost, and requires less computation.

3. WEB USAGE TERMINOLOGY

We start with the definition of a set of terms that leads to the formal definition of high utility traversal path mining.

3.1. Traversal Path

The data used for Web log mining is Weblog entry database. Each entry in the database consists of the URL requested, the IP address from which the request originated, timestamp, etc. The database can be stored on Web server, client or agent. The raw Weblog data need to be converted to a set of traversal paths. The goal of frequent traversal pattern mining is to find all the frequent traversal sequences in a given database. We give out the definition of some basic terms.

$X = <i1, i2 \ldots im>$ is a m-sequence of traversal path.

$D = \{T1, T2 \ldots Tn\}$ is a Weblog database, where Ti is a traversal path, $1 \le i \le n$

In sequence mining, a sequence $\alpha = <\alpha_1, \alpha_2, \ldots, \alpha_n>$ is called a subsequence of another sequence $\beta = <\beta_1, \beta_2, \ldots, \beta_m>$ if there exist integers $1 \le j_1 < j_2 < \ldots < j_n \le m$ such that $\alpha_1 \subseteq \beta_{j1}, \alpha_2 \subseteq \beta_{j2}, \ldots, \alpha_n \subseteq \beta_{jn}$. That is, gaps between events are allowed. However, It is not the case in traversal path mining. A traversal path $\alpha = <\alpha_1, \alpha_2, \ldots, \alpha_n>$ is called a subpath of another path $\beta = <\beta_1, \beta_2, \ldots, \beta_m>$ if there exist an integer i such that $\alpha_{i-j} = \beta_j$, for $1 \le j \le m$, $1 \le i+j \le m$. In other words, traversal path dos not allow gaps between different events.

3.2. Utility Mining

Following is the formal definition of utility mining model. $I = \{i1, i2, \ldots, im\}$ is a set of items.

- $D = \{T_1, T_2, \ldots, T_n\}$ is a transaction database where each transaction $T_i \in D$ is a subset of I.
- $o(i_p, T_q)$, objective value, represents the value of item i_p in transaction T_q.
- $s(i_p)$, subjective value, is the specific value assigned by a user to express the user’s preference. This value reflects the importance of an item, which is independent of transactions. $s(i_p)$ is greater than $s(i_q)$ if the user prefers item i_p to item i_q.
- $u(x, y)$: $(R, R) \rightarrow R^{+}$, utility function, where R is the set of real numbers, and R^{+} is the set of positive real numbers. Suppose $u(i_p, T_q)$ is defined as $o(i_p, T_q) \times s(i_p)$, where $o(i_p, T_q)$, is the value of item i_p in transaction T_q, and $s(i_p)$ is the unit profit of item i_p.
- $u(X, T_q)$, utility of an itemset X in transaction T_q, is defined as $\sum_{i_p \in X} u(i_p, T_q)$, where $X = \{i_1, i_2, \ldots, i_k\}$

3.3. Utility-based Web Path Traversal Pattern Mining

By introducing the concept of utility into web path traversal pattern mining problem, the subjective value could be the end user’s preference, and the objective value could be the browsing time a user spent on a given page. Thus, utility-based web path traversal pattern mining is to find all the Web traversal sequences that have high utility beyond a minimum threshold. A web page refers to an item, a traversal sequence refers to an itemset, the time a user spent on a given page X in a browsing sequence T is defined as utility, denoted as $u(X, T)$. The more time a user spent on a Web page, the more interesting or important it is to the user. Table 1 is an example of a traversal path database. The number in the bracket represents the time spent on this Web page which can be regarded as the utility of this page in a given sequence. In Table 1, $u(<C>, T1)$ is 2, and $u(<D, E>, T8) = u(D, T8) + u(E, T8) = 7+2 = 9$. From this example, it is easy to observe that utility mining does find different results with frequency based mining. The high utility traversal paths may assist Web service providers to design better web link structures, thus cater to the users’ interests.

TID	User Traversal
T1	A(2),C(3)
T2	B(5), D(1)E(1)
T3	A(1)C(1)E(3)
T4	A(1)D(18)E(5)
T5	C(4),E(2)

4. PROPOSED ALGORITHMS

Algorithm for Utility-based Web Path Traversal Pattern Mining

Utility-based path traversal pattern mining is aimed at finding sequences whose utility exceeds a user specified minimum threshold. The challenge of utility mining is that it does not follow "downward closure property" (anti-monotone property), that is, a high utility itemset may consist of some low utility sub-itemsets. "Downward closure property" contributes to the success of ARM algorithms, such as Apriori [8], where any subset of a frequent itemset must also be frequent. Two-Phase algorithm proposed by Y. Liu et al. [6] is aimed at solving this difficulty. In Phase I, transaction-level utility is proposed and defined as the sum of the utilities of all the transactions containing *X*. (The purpose of introducing this new concept is not to define a new problem, but to utilize its property to prune the search space.) This model maintains a Transaction-level Downward Closure Property: any subset of a high transaction-level utility itemset must also be high in transaction-level utility. In Phase II, only one database scan is performed to filter out the high transaction-level utility itemsets that are actually low utility itemsets. In this paper, we extend Two-Phase algorithm to traversal path mining problem. High transaction-level utility sequences are identified in Phase I. The size of candidate set is reduced by only considering the supersets of high transaction-level utility sequences. In Phase II, only one database scan is performed to filter out the high transaction-level utility sequences that are actually low utility sequences. This algorithm guarantees that the complete set of high utility sequences will be identified.

Algorithm1: Proposed new Dbscan clustering:

Step1:

Construct the similarity matrix using S3M measure(Definition 1).

Step2:

select all points from D that satisfy the Eps and Minpts
C = 0
for each unvisited point P in dataset D
mark P as visited
N = get Neighbors (P, eps)
if sizeof(N) < MinPts
mark P as NOISE
else
begin
C = next cluster
mark P as visited
end
add P to cluster C
for each point P' in N
if P' is not visited
mark P' as visited
N' = getNeighbors(P', eps)
if sizeof(N') >= MinPts
N = N joined with N'
if P' is not yet member of any cluster
add P' to cluster C

Step 3:

Return C

Step 4:

For all Ti ϵ U Compute Si= R(Ti)
Using definition 2 for given threshold d.

Step 5:

Next compute the constrained-similarity upper Approximations Sj for relative similarity r using definition 3
if Si= SJ
end if

Step 6:

Repeat step 3 until U≠ Ø;
Return D.
End

Algorithm2:

4.1. Phase I

Phase I is to find out high transaction level utility web traversal path.

Definition 1. (Transaction Utility) The transaction utility of transaction T_q, denoted as $tu(T_q)$, is the sum of the utilities of all items in T_q: $tu(T_q) = \sum_{i_p \in T_q} u(i_p, T_q)$,

where $u(i_p, T_q)$ is defined the same as in Section 3.2. Table 2 gives the transaction utility for each transaction in our example database in Table 1.

Table 2. Transaction utility of the traversal path Database

TID	Transaction Utility *(tu)*
T1	5
T2	7
T3	4
T4	24
T5	6

Definition 2. (Transaction-level Utility) The transaction-level utility of a sequence *X*, denoted as *tlu(X)*, is the sum of the transaction utilities of all the transactions containing *X*:

$$tlu(X) = \sum_{X \subseteq T_q \in D} tu(T_q)$$

For the example in Table 1, *tlu(<A>) = tu(T1) + tu(T3) + tu(T4)* =5+4+24 and *tlu(<A, C>) = tu(T1) + tu(T3)* = 5+4=9.

Definition 3. (High Transaction-level Utility Sequence) For a given sequence *X*, *X* is a high transaction-level utility sequence if $tlu(X) \geq \varepsilon'$, where **e***'* is the user specified minimum utility threshold. The pseudo code is given in Figure 1. We denote the set of high utility sequences as *Lk* and its candidate set as *Ck*. Let *HTLU* be the collection of all high transaction-level utility sequences in a web traversal path database *Dp*. As pointed out earlier, the difference between traversal path mining and ARM lies in the fact that a sequence of traversal path must be a consecutive sequence whereas a large itemset in ARM is just a set of items in a transaction. **candidate_gen** implements the candidate generation of traversal path[3].

Input: Web traversal path database D_p, subjective value table ST, minimum utility threshold, min_utility.
Output: $HTLU$, high transaction-level utility sequences.

```
Calculate the transaction utility tu_t of each sequence t ∈ D_p;
HTLU_1 = find_high_transaction-level_utility_1-sequence (D_p);
for (k = 2; HTLU_{k-1} ≠ ∅; k++) {
    C_k = candidate_gen (HTLU_{k-1}, min_utility);
    for each sequence t ∈ D_p
        for each c ∈ C_k
            if c ⊑ t then
                c.tlu += tu_t;
    HTLU_k = {c ∈ C_k | c.tlu ≥ min_utility}
}
return HTLU = ∪_k HTLU_k;
```

Figure 1. Pseudo code of Phase I: find high transaction-level utility web traversal paths.

4.2. Phase II

In Phase II, one database scan is required to select the high utility sequences from high transaction-level utility sequences identified in Phase I. The number of the high transaction-level utility sequence is small. Hence, the time saved in Phase I may compensate for the cost incurred by the scan during Phase II. Thus our simple technique can be easily understood for calculations and conceptually as well, though it sounds traditional. Moreover, it performs scalable in terms of execution time under large databases for the reduced temporal data used under the concept of On-shelf utility mining. Hence, the two-phased system is even more promising in our paper, for its mining capability temporal high utility web transactional item sets in web data streams or logs.

5. EXPERIMENTAL RESULTS

The log data used in this paper is extracted from a research website at DePaul CTI (http://www.cs.depaul.edu). The data are randomly sampled from the Weblog data within 2 weeks in April, 2002. We performed preprocessing on the raw data. The original data included 3446 users, 10950 sessions, 105448 browses and 7051 Web pages. Among these Web pages, a large portion is URLs and, thus, data cleaning is needed. After data preprocessing, we carried out two groups of experiments, one for frequent traversal path mining and the other for utility-based traversal path mining. The minimum utility threshold is set at 1% of the total utility. Utility-based traversal path mining obtains 22 high utility sequences including twelve 1-sequences, seven 2-sequences and three 3-sequences. Frequency-based mining obtains 121 frequent sequences, including 1- sequence, 2-sequence, 3-sequence and 4-sequence.

Table 3 shows the top 10 sequences discovered by the two models. (We don't take 1-sequence into consideration).

DBScan cluster results:

It is clear that traversal patterns discovered by the two algorithms are not always the same. Three out of the top 10 sequences in Table 3(a) are not frequent, that is, *<77,121>*, *<1259, 1266>*, *<1259, 77, 121>*. On the other hand, four out of the top 10 frequent sequences are dropped out by utility mining (in Table 3(b)) because their utility does not reach the minimum utility threshold, that is, *<1259, 1301>*, *<1259, 69>*, *<1301,1647>* and *<1301, 1648>*. The Web page sequence *<1259, 1266>* is referred to *</news/default.asp, /news/news.asp?theid=580>*. It is a high utility traversal path but not a frequent one. It indicates that people who browse *</news/default.asp>* would like to spend a lot of time on reading *</news/news.asp?theid=580>* which seems as a Web page for a certain category of news.

```
Elapsed time: .03

( 0.) 4,2,4,4.254902          --> 0
( 1.) 4,2,3,4.254902          --> 0
( 2.) 4,6,3,2                 --> 2
( 3.) 6,3,7,7                 --> 1
( 4.) 4,2,4,4.254902          --> 0
( 5.) 4,2,3,4.254902          --> 0
( 6.) 4,6,3,2                 --> 2
( 7.) 6,3,7,7                 --> 1
( 8.) 4,2,3,4.254902          --> 0
( 9.) 4,6,3,2                 --> 2
(10.) 6,3,7,7                 --> 1
(11.) 4,2,4,4.254902          --> 0
(12.) 4,2,3,4.254902          --> 0
(13.) 4,6,3,2                 --> 2
(14.) 4,6,3,2                 --> 2
(15.) 6,3,7,7                 --> 1
(16.) 4,2,4,4.254902          --> 0
(17.) 4,2,3,4.254902          --> 0
(18.) 4,6,3,2                 --> 2
(19.) 6,3,7,7                 --> 1
(20.) 4,2,3,4.254902          --> 0
```

(a) Top 10 High utility traversal sequences

Sequence	Utility	Support
1259, 77	86947	872
1259, 77, 126	34489	313
77, 126	33200	392
69, 1173	32097	411
77, 121	***27274***	***132***
1259, 1301	22367	412
1259, 1266	***22111***	***128***
1259, 77, 121	***18970***	***78***
1259, 69, 1173	18676	178
1259, 1682	17786	259

(b) Top 10 frequent traversal sequences

Sequence	Utility	Support
1259, 77	86947	872
1259, 1301	*22367*	*412*
69, 1173	32097	411
77, 126	33200	392
1259, 77, 126	34489	313
1259, 1682	17786	259
1259, 69	*11872*	*219*
1301, 1647	*5699*	*203*
1259, 69, 1173	18676	178
1301, 1648	*6219*	*173*

The Web page 1266 is not a frequent one but people who are interested in it would like to spend a lot of time on it. The high utility path traversal patterns indicate customer behavior patterns, which can assist Web designers to design better web link structures. For example, super links can be added to a "significant" Web page to attract more reading. The Web page sequence *<1301, 1647>* is referred to *</people/, /people/search.asp>*. It is a frequent sequence but not high in utility. Note that these two pages are designed for users to search for faculty members, thus, it is of little meaning to the Web designers. The Web page sequence *<1301, 1648>* is referred to *</people/, /people/search.asp?sort=ft>*, which is similar with the former sequence. The Web page sequence *<1259, 1301>* is referred to *</news/default.asp, /people/>*, which is just a link from the main page to the research website. Overall speaking, observed from our experiments, we realize that high utility traversal sequences are valuable, which can show the customers' hidden behavior patterns to the web service providers, which in turn could be utilized to provide better services.

6. CONCLUSION AND FUTURE WORK

Traditional path traversal pattern mining discovers frequent Web accessing sequences from Weblog databases. It is not only useful in improving the website design, but also able to lead to better marketing decisions. However, since it is based on frequency, it fails to reflect the different impacts of different webpages to different users. The difference between webpages makes a strong impact on the decision makings in Internet Information Service Applications. In this paper, we presented "high Two phase web utility mining", which introduced the concept of "utility" into web log. As utility measures the "interesting" or "usefulness" of a webpage, thus satisfies the Web Service Providers in quantifying the user preferences of ease in web data transactions. Hence, we explored a Two-Phase algorithm that discovered high on-shelf utility data on web pages highly efficiently, in which both the phases are carried out with effective algorithms and became responsible in giving us the effective results.

Our proposed algorithm was though a mixed approach of utility mining & two–phase algorithm on web utility mining, both the mining algorithms were individually proved to be the best of many others, in yielding good experimental results when applied on a real-world Msndc Weblog database. Thus, our mixed approach can surely be declared as the best. We also demonstrated the interesting areas we observed, as well as their significance to the decision making process. On-shelf utility mining considered not only individual profit and utility of each item in a web transaction but also common on-shelf time periods of a product combination. In this study, a new on-shelf web utility mining algorithm was preferred in order to speed up the execution efficiency for mining high on-shelf utility web transactional itemsets. The experimental results also showed that the proposed high on-shelf utility approach had good impact when compared to the other traditional utility mining approaches. Finally In this paper we developed a new rough set dbscan clustering algorithm and presented a experimental results on msnbc.com which is useful in finding the user access patterns and the order of visits of the hyperlinks of the each user and the inter cluster similarity among the clusters.

In the future, we will attempt to handle the maintenance problem of high on-shelf utility mining of transactions at the webpage level. Besides, the results from on-shelf utility mining on web transactional log are independent of the order of transactions. Another kind of knowledge called sequential patterns depends on the order of transactions. We will also extend our approach to mining out this kind of knowledge in the future.

A number of interesting problems are open to discuss in the future research. For example, the accuracy, effectiveness and scalability of the proposed idea applied to larger databases need to be evaluated. Other factors can also be explored as utility in web transactional pattern mining. Besides, how to combine frequency and utility together to improve web transactional ease is still a problem need to be studied.

REFERENCES

[1] Agrawal .R and Srikant .R (1994) : Fast algorithm for mining association rules in large databases, The 20th International Conference on Very Large Data Bases, pp. 487-99.

[2] Lan G.C., Hong T.P., and Vincent S. Tseng. (2009): A two-phased mining algorithm for high on-shelf utility itemsets, The 2009 National Computer Symposium, pp. 100-5.

[3] D.N.V.S.L.S. Indira, Jyotsna Supriya .P, and Narayana .S(July 2011): A Modern Approach of On-Shelf Utility Mining with Two-Phase Algorithm on Web Transactions, IJCSNS International Journal of Computer Science and Network Security, VOL.11 No.7

[4] Lan G.C., Hong T.P., and Vincent S. Tseng (2011) : Reducing Database Scans for On-shelf Utility Mining, IETE Tech Rev 2011, vol 28,no. 2, pp.103-12.

[5] Tseng V.S., Chu C.J., Liang T.(2006) : Efficient Mining of Temporal High Utility Itemsets from Data streams Proceedings of Second International Workshop on Utility-Based Data Mining, August 20, 2006.

[6] Yao H., Hamilton H.J., and Butz C.J, (2004) : A foundational approach to mining itemset utilities from databases, Proceedings of the 3rd SIAM International Conference on Data Mining, pp. 482-486.

[7] Yu-Cheng Chen and Jieh- Shan Yeh.(2010) : Preference utility mining of web navigation patterns, IET International Conference on Frontier Computing. Theory, Technologies & Applications (CP568) Taichung, Taiwan, pp.49-54.

[8]. Chang, C.C., Lin, C.Y.: Perfect hashing schemes for mining association rules. The Computer Journal 48(2), 168–179 (2005)

[9]. Grahne, G., Zhu, J.: Fast algorithms for frequent itemset mining using fp trees. IEEE Transactions on Knowledge and Data Engineering 17(10), 1347-1362 (2005).

AUTHORS PROFILE

G. Sunil kumar received his M.C.A from JNTU Kakinada. He is currently working as Lecturer in department of Computer Applications in Maris Stella College and research project developer in GSK research and development solutions. His research areas include Data Mining applications in network environment, Computer Networks, Image Processing, Network Security attacks and its countermeasures. He completed computer science, mathematics, statistics research projects for different universities research scholars.