Web User Profiling using Hierarchical Clustering with Improved Similarity Measure

Nilani Algiriyage, Sanath Jayasena and Gihan Dias
Department of Computer Science & Engineering
University of Moratuwa
Sri-Lanka
Email: rangika.nilani@gmail.com, sanath@cse.mrt.ac.lk, gihan@cse.mrt.ac.lk

***Abstract*—Web user profiling targets grouping users in to clusters with similar interests. Web sites are attracted by many visitors and gaining insight to the patterns of access leaves lot of information. Web server access log files record every single request processed by web site visitors. Applying web usage mining techniques allow to identify interesting patterns. In this paper we have improved the similarity measure proposed by Velásquez et al. [1] and used it as the distance measure in an agglomerative hierarchical clustering for a data set from an online banking web site. To generate profiles, frequent item set mining is applied over the clusters. Our results show that proper visitor clustering can be achieved with the improved similarity measure.**

I. Introduction

With the evolution of the Internet and continuous growth of the global information infrastructure, the amount of data collected has been drastically increased. Web server access log files collect substantial data about web visitor access patterns. Data mining techniques can be applied on such data (which is known as Web Mining) to reveal lot of useful information about navigational patterns. Generally a profile, contains facts about someone's interests and behavior. The content of the user profile can be changed based on the context. However most common contents of a user profile can be user's interaction preferences, user's knowledge, user's interests, background skills and knowledge etc. In the web usage domain, user profiling applies to establishing groups of users exhibiting similar browsing behavior. User profiling helps web site owners in multiple ways:personalization, system improvements such as load balancing, data distribution policies, improve web site's structure, develop recommendation systems, business intelligence etc.

Data mining specifically web usage mining has been applied by many researchers to handle the problem of web user profiling. Mobarsher et al. [2] introduced the taxonomy of web mining for the data mining activities performed on web data. There are major two types of web mining which is refereed as web content mining and web usage mining. Web content mining engages in automatically searching information resources in web pages and web usage mining is related to discovering user access patterns from web usage data. We have focused on the later one, web usage mining in the context of web visitor profiling.

In this work we have improved the similarity measure proposed by Velásquez et al. [1] and applied it in a hierarchical clustering. Our similarity measure has two major portions; a measure based on the Lavenshtein distance to find similarity between access sequences and a measure based on normalized time values spent on each web page. We used frequent item set mining to generate profiles from the clustered data. In the results, we show that proper profiling can be achieved for our data set using the improved similarity measure.

The rest of the paper is organized in the following way. Section II presents the related work and in section III we discuss the methodology we followed for web user profiling. In section IV experimental evaluation and results are described and finally we discuss the conclusions in section V.

978-1-4799-1740-2/15/$31.00 ©2015 IEEE

II. Related Work

In the research community user profiling is studied in different context. Among the web mining techniques applied, clustering seems most widely used method for grouping similar users. Xu et al. [3] have tested the feasibility of applying k-means algorithm to cluster web users. Fuzzy clustering has been applied in multiple research works. The idea behind the fuzzy clustering is to enable the generation of overlapping clusters. This has been successfully applied in the web usage mining process by Nasraoui et al. [4] Suryavanshi et al. [5] and Castellano et al. [6].

In order to understand similarity between two visitors, similarity measures are required. Jitian et al. [7] talks about different similarity measures that can be applied for web log data. Xie et al. [8] has proposed a distance measure for clustering based on Dempster-Shafers theory of combining evidence.

Velásquez et al. [1] have proposed a similarity measure based on the content of the respective web pages and the similarity between different page sequences. They have derived the measure in the following way. Let α and β be two visitor behavior vectors of cardinality C^α and C^β respectively and $\Gamma(.)$ is a function that applied over α or β that returns the navigation sequence corresponding to a visitor vector.

$$sm(\alpha,\beta) = dG(\Gamma(\alpha),\Gamma(\beta))\frac{1}{\eta}\sum_{k=1}^{\eta}\tau_k * dp(p_{\alpha,k},p_{\beta,k}) \quad (1)$$

where dG is the similarity between sequences of pages visited. $\eta = min\{C^\alpha, C^\beta\}$ and τ_k is an indicator of visitor's

interest of the web page visited. The term τ_k is defined as $\tau_k = min\{\frac{t_{\alpha,k}}{t_{\beta,k}}\frac{t_{\beta,k}}{t_{\alpha,k}}\}$. The term $dp(p_{\alpha,k}, p_{\beta,k})$ is the similarity of pages visited. This is the angle cosine similarity between two word-page vectors. They have applied it in a Self Organizing Feature Maps (SOFM) clustering for user session clustering.

How ever we see some drawbacks in the proposed similarity measure and in this work, the measure is improved. We have applied it in a hierarchical clustering. Hierarchical clustering [9] builds a hierarchy of clusters decomposing a given set of data objects. Generally there are two types of hierarchical clustering based on the way the hierarchy is formed.

- Agglomerative:Agglomerative which is also known as "bottom-up" merges the objects or groups that are close to one another, until all of the groups are merged into one or until the termination condition is met.

- Divisive:Which is also known as "top-down" approach starts with all of the objects in the same cluster and in each iteration a cluster is split up to smaller clusters until there is one object in each cluster or termination condition is met.

We used the improved similarity measure in an agglomerative hierarchical clustering [10] and visualized it in a dendrogram. Dendrogram [9] is a tree structure which is used to represent the results of a hierarchical clustering. It shows how objects are grouped in step by step. We further used frequent itemset mining in order to generate user profiles from the identified clusters.

III. Data Preparation

We selected a web site of an e-banking service provider in Sri Lanka having following features.

- A static web site with proper page naming.
- Web site has a log-in facility and all registered users have to log-in before performing transactions on the web site.
- Several types of users interact with the web site daily and their interests are different.

Web site has 170 web pages. Details of our data set is discussed in Section IV.

Our methodology and work flow is summarized in Fig. 1. Data cleansing and formatting are the two major preprocessing operations performed. Details of the data preparation steps are described in the following.

A. Data Formatting

Web servers record requests processed by the server in different formats. We have observed that Apache combined log format is very comprehensive. Original log files were converted in to Apache combined log format [11].

B. Data cleansing

During the process of data cleansing we have removed lines of the log file which were not fully recorded and did not satisfy the sufficient length of a log line. Keeping these data in the log file would result in unclear clusters. We observed that some records of the log file were not sorted. We sorted the log file in the ascending order of the recorded time value.

C. User-Session Identification

We have used IP address, "user-agent" string and session timeout period in order to identify user sessions following the methodology proposed by Mobasher et al. [2]. This time period has been considered as 1800 seconds and if the interval is more than the defined period, existing session is closed and new one is initiated.

D. Remove web-crawler sessions

For the purpose of user-profiling we are interested only on requests from human users. Hence we have followed the methodology proposed by Algiriyage et al. [12] to remove web crawler sessions.

E. Low support page filtering

The number of web pages browsed during a user-session had a large variation. For the purpose of profiling we considered only the sessions having browsed more than three pages. We filtered out others as accidentally visited the web site, without having any interest over the contents.

F. Filter Image and Styling Files

For the purpose of human visitor profiling we were not interested in image files. Hence we removed extensions *(cache, css, png, gif, jpeg, js, jpg, ico, axd, ashx, xml)* and considered only web page requests such as *asp, aspx, htm, html, php, pdf* and *doc*.

G. Navigation sequence identification

An example web site with 10 web pages is shown in the Fig 2.

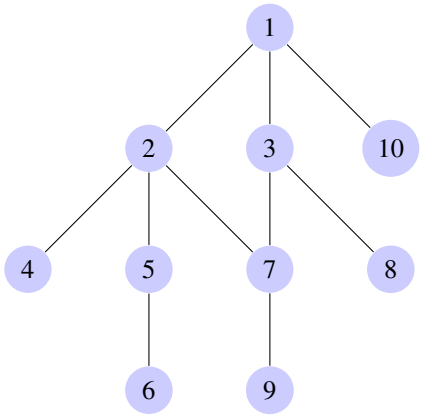

Fig. 2. Web site with navigational paths

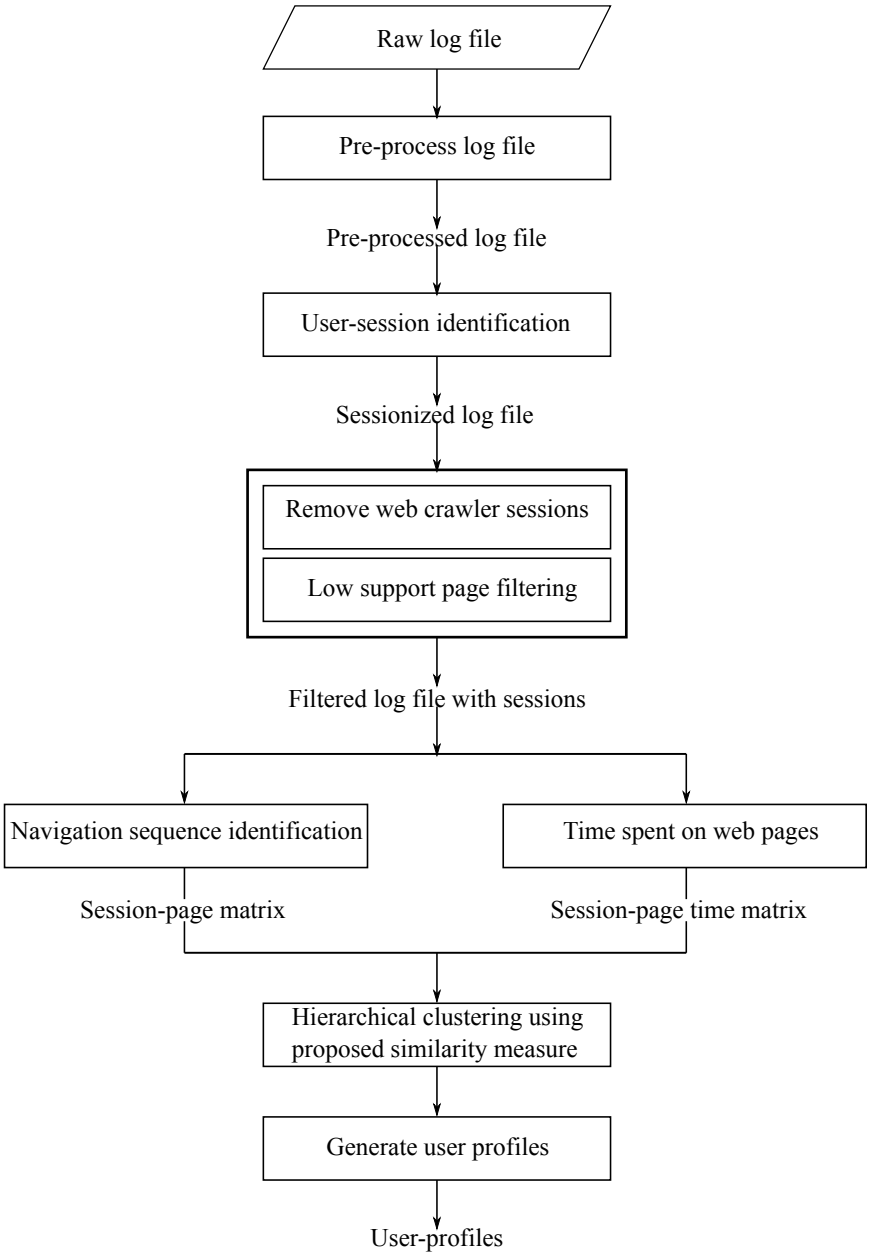

Fig. 1. Methodology for human visitor profiling.

Suppose visitors in two visitor sessions browsed the web site in the following way.

$$S1 = 1 \rightarrow 2, 2 \rightarrow 7, 2 \rightarrow 5, 5 \rightarrow 6$$

$$S2 = 1 \rightarrow 3, 3 \rightarrow 7, 7 \rightarrow 9$$

Based on the traversal patterns in two sessions we can derive the navigation sequences as:

$$Seq(S1) = [1, 2, 7, 5, 6]$$

$$Seq(S2) = [1, 3, 7, 9]$$

In this case we have not considered backward traversals. Hence the cardinality(C) of the two sequences can be given as:

$$C(S1) = 5$$

$$C(S2) = 4$$

H. Prepare Session-page Matrix

Once the navigation sequence is identified we prepared the session-page matrix (Table I) to show the web page sequences visited in each session. As previously described, our methodology for session identification was grouping IP address, "user-agent" within 30 minute time period. Due to this methodology there can be some sessions from the middle of their activities. To avoid the confusions of such sessions we considered only the sessions starting with "sign-in" and "home

pages" in the test data set. Further in the session-page matrix, we did Scipy encoding [13] to map string page names into integer ones.

TABLE I. Session-page Matrix

Session	Pages
1	6,61,70,76,101,110,112,134,161
2	6,61,70,72,76,81,112,136,143
3	6,61,101,102,112,161,162
...	6,61,70,73,76,101,110,112,161,168
n	8,28,29,30,35,43,44,45,46,49,59,61,114,121

I. Prepare Session-time Matrix

Session-page matrix was prepared to show the web page sequences browsed in each session. Another matrix is prepared to show the time spent on each page within the session. This matrix which can be called as Session-time matrix is shown in Table II.

TABLE II. Session-time Matrix

Session	p1,p2,p3,p4,p5,p6,p7
1	0.5,1.0,2.1,1.2,2.0,1.5,0.7
2	1.0,1.2,1.5,0.0,0.0,0.0,0.0
3	2.3,1.0,1.6,2.3,0.0,0.0,0.0
..	0.9,2.0,2.4,2.5,2.0,1.5,0.0
n	0.0,0.0,0.9,1.4,0.0,0.0,0.0

J. Calculate Lavenshtein Distance

To compare the similarity between two sequences we need to use a dissimilarity measure. Lavenshtein distance which is also referred as edit distance, measures the similarity between two strings.

For two sequences $a = (a_1, ..., a_x)\ and\ b = (b_1, ..., b_y)$ Lavenshtein distance is defined as:

$$LD(a_x, b_y) \begin{cases} max(x,y) & if\ min(x,y) = 0 \\ min \begin{cases} L(a,b)(x-1,y)+1 \\ L(a,b)(x,y-1,j)+1 \\ L(a,b)(x-1,y-1)+1(a_x \neq b_y) \end{cases} & Otherwise \end{cases} \tag{2}$$

For example in the sequences that we discussed earlier [1,2,7,5,6] and [1,3,7,9], the Lavenshtein distance is 3. That is 3 insert/update/delete operations are required to transform [1,2,7,5,6] to [1,3,7,9]. We calculated the similarity between the two visiting sequences using Equation 3:

$$sim(S1, S2) = 1 - \{(LD(seq(S1), seq(S2))) / max(C(S1), C(S2))\} \tag{3}$$

We used a bit different measure from the Velásquez's [1] approach in calculation of the similarity using Levenshtein distance. For the discussed example, similarity of the sequences is 0.4 based on the Equation 3. We refer this similarity which is based on Lavenshtein distance as dLD.

K. Comparison of Visitor Sessions

To compare the visitor sessions, a similarity measure is required. We propose following measure based on the similarity of navigation sequences and time spent on web pages. To derive the equation we followed the notations described.

1) Similarity Between Navigational Sequences: Let S_x and S_y be two visitor sessions and F is a function applied over S_x and S_y which returns navigation sequences. Cardinality of the two sequences are C_x and C_y.The distance between two sequences is calculated using Levenshtein distance based measure described in Equation 3.

2) Similarity Based on Time Spent: Web visitors spend time on web pages based on the interest and relative importance of the content to them. Preparation of session-time matrix was described in an earlier section. In the Velásquez et al. [1] approach, described in equation 1 for the term τ_k they have considered that the time spent on web page is proportional to the interest the visitor has in its contents. If the times spent by visitor α and β on k^{th} page that they have visited are close to each other the value(τ_k) becomes close to 1 and otherwise it will be close to 0.

The problem of this approach for τ_k is that time is not normalized. For example the value will be 0.5 for following two cases. But there is a huge difference between the times.

$$t_k^{\alpha} = 2$$

$$t_k^{\beta} = 4$$

$$T_k = 2/4 = 0.5$$

$$t_k^{\alpha} = 50$$

$$t_k^{\beta} = 100$$

$$T_k = 50/100 = 0.5$$

To eliminate the problem, we normalized the time value using standard score(Z-score). Suppose time spent on page n is t_n and the mean of time spent on page n by all visitors is μ and standard deviation is σ.

$$z = \frac{t_n - \mu}{\sigma} \tag{4}$$

Then we calculate the euclidean distance between the Z-values.

$$ED(z_x, z_y) = \sum_{i}^{n} (z_{xi} - z_{yi})^2 \tag{5}$$

We refer this similarity which is based on Euclidean distance as dED.

3) Proposed Similarity Measure: We propose a similarity measure based on the web page access sequence and time spent on each page. Let s_α and s_β are two visitor sessions and S_α and S_β are the related resource request pattern sequences.

$$sim(s_\alpha, s_\beta) = dLD(S_\alpha, S_\beta) * dED(z_x, z_y) \quad (6)$$

Formulation of dLD and dED is discussed in Equations 3 and 5 respectively.

IV. Experimental Evaluation

We obtained web server access log files of a online banking service provider in Sri Lanka for a period of two weeks. Size of the log file is 243.6MB and it contained 841,196 HTTP requests before the preprocessing stages. But for the hierarchical clustering we considered a portion of the data set containing 71,256 HTTP requests before and 71,238 requests after pre-processing. Summary of the data set is presented in Table III.

TABLE III. Summary of the log file.

Number of HTTP requests before pre-processing	71,256
Number of HTTP requests after pre-processing	71,238
Total number of visitor sessions	2,033
Number of possible web crawler sessions	185
Number of possible Human Visitor sessions	1,848
Number of visitor sessions with initiating sessions	1,212
Number of visitor sessions after low click page filtering	1,169

The complexity of agglomerative clustering for a large dataset is faster than divisive clustering. Hence we applied the proposed similarity measure in an agglomerative hierarchical clustering. Fig. 3 shows the dendrogram generated. We have used a color threshold value to show different clusters in the dendrogrm. Dendrogram is a tree-structured graph used to visualize the results of a hierarchical clustering. A dendrogram can be pruned at any level to generate clusters. In this experiment, we pruned the dendrogram at level 10 to obtain 10 visitor clusters.

Generating profiles, or gaining more insight to the clusters generated, require some additional work. We performed a frequent item set mining task in order to understand the behavior patterns in each cluster. With the highest confidence and support values, following are the clusters generated in Table IV.

To understand and describe the clusters, we logged-in to the online banking web site as different types of users and performed multiple transactions. Based on our browsing patterns in the access log file, generated clusters were described in the Table IV.

According to the clustering results, some users logged-in to the system and has perform nothing afterward and the others performed different types of transactions. In some clusters, there were random browsing behavior which was hard to understand. The following is an example for such random browsing behavior.

Ex: 6,61,70,72,236,237,241,6,61,70,72,242,243,244,6,61,70,72, 250,251,253,6,61,70,72

TABLE IV. Cluster results

No	Pages Visited	Description	No of sessions
1	6,61,112	successfully logged-in, but nothing was done after the loging	52
1	8,16,24,34,...	successfully logged-in as corporate users, and performed transactions	261
3	6,61,70,76,...	successfully logged-in as personal users, and performed transactions	302
4	6,61,101,112,161,...	successfully logged-in as personal users, and performed transactions	210
5	6,61,70,72,...	Some strange behavior	2
6	6,61,70,76,...	successfully logged-in as personal users, and performed transactions	207
7	8,28,43,44,...	successfully logged-in as corporate users, and performed transactions	96
8	6,61,70,76,...	Some strange and random behavior	1
9	5,15,17,...	New users, who do not log-in to the web site	34
10	6,61,8,...	Some random browsing behavior	4

In one cluster, which contained a single user session, the behavior was very strange that the visitor traversed a long path without successfully doing any transaction.

V. Conclusion

We introduced a new similarity measure to group human users based on their browsing behavior. The similarity measure was included in a agglomerative hierarchical clustering algorithm to identify user clusters. Our results show that there are clear clusters among the visitors/users of the web site. Derived user profiles can be used by the banking organization to get better knowledge about their customer base. And also this can be used as an intrusion detection tool, where we found some clusters having strange browsing patterns. Any way further research has to be carried out to understand how well this clustering helps in intrusion detection.

As future works we can consider the content of web pages and more advanced features for the proposed similarity measure. We have applied the methodology on online banking web server logs. To get a better idea this can be applied in multiple web log files. In our approach the similarity measure is used in a agglomerative hierarchical clustering algorithm. This can be used other clustering algorithms and can test the generated user profiles for further improvements.

Acknowledgment

The authors would like to thank LK domain registry and Techcert for providing necessary data sets and facilities to conduct this research successfully.

References

[1] J. D. Velásquez, H. Yasuda, and R. Weber, "A new similarity measure to understand visitor behavior in a web site," *IEICE transactions on Information and Systems*, pp. 389–396, 2004.

[2] R. Cooley, B. Mobasher, and J. Srivastava, "Information and pattern discovery on the world wide web," in *Proc.* 9^{th} *Intl. Conf. on Tools with Artificial Intelligence*, p. 558, 1997.

[3] J. Xu and H. Liu, "Web user clustering analysis based on kmeans algorithm," in *International Conference on Information Networking and Automation (ICINA)*, vol. 2, pp. V2–6, 2010.

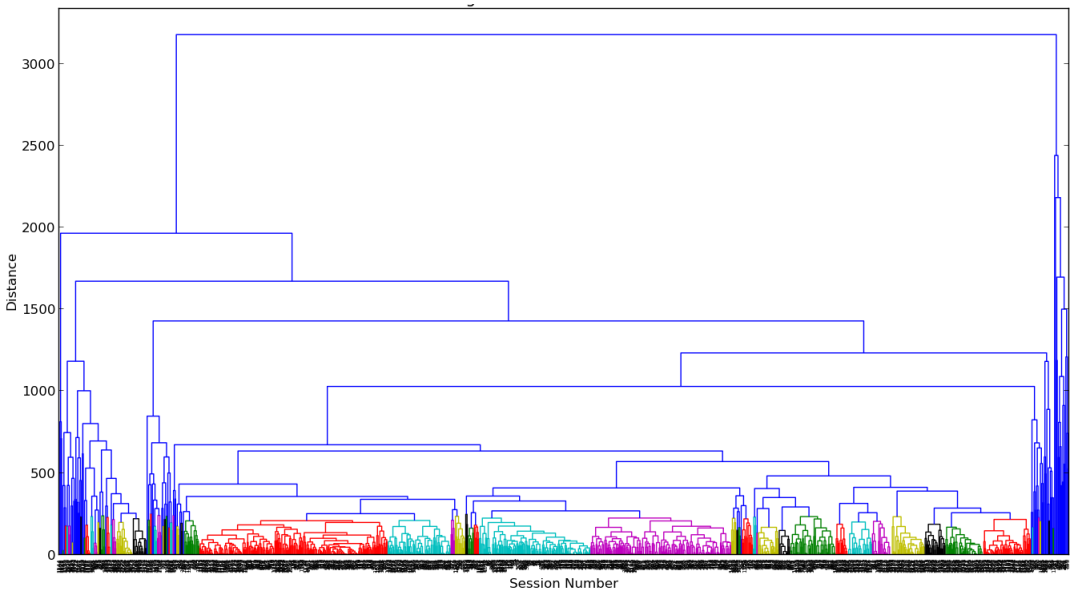

Fig. 3. Dendrogram for the hierarchical clustering.

[4] O. Nasraoui, H. Frigui, A. Joshi, and R. Krishnapuram, "Mining web access logs using relational competitive fuzzy clustering," in *Proc.* 8^{th} *Intl. Conf. on Fuzzy Systems Association World Congress*, pp. 195–204, 1999.

[5] B. S. Suryavanshi, N. Shiri, and S. P. Mudur, "An efficient technique for mining usage profiles using relational fuzzy subtractive clustering," in *Proceedings of the International Workshop on Challenges in Web Information Retrieval and Integration*, pp. 23–29, 2005.

[6] G. Castellano, F. Mesto, M. Minunno, and M. A. Torsello, "Web user profiling using fuzzy clustering," in *Applications of Fuzzy Sets Theory*, pp. 94–101, 2007.

[7] J. Xiao, Y. Zhang, X. Jia, and T. Li, "Measuring similarity of interests for clustering web-users," in *Proc.* 12^{th} *Australasian database Conf.*, pp. 107–114, 2001.

[8] Y. Xie and V. V. Phoha, "Web user clustering from access log using belief function," in *Proc.* 1^{st} *Intl. Conf. on Knowledge Capture*, pp. 202–208, 2001.

[9] H. Jiawei and M. Kamber, "Data mining: concepts and techniques," *San Francisco, CA, itd: Morgan Kaufmann*, vol. 5, 2001.

[10] "scipy.cluster.hierarchy.dendrogram-numpy and scipy documentation (2014,may 11)." [Online]. Available:http://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.cluster.hierarchy.dendrogram.html.

[11] "Apache http server version 2.2." [Online]. Available:http://httpd.apache.org/docs/2.2/logs.html.

[12] N. Algiriyage, S. Jayasena, G. Dias, A. Perera, and K. Dayananda, "Identification and characterization of crawlers through analysis of web logs," in *Proc.* 8^{th} *IEEE Intl. Conf. on Industrial and Information Systems (ICIIS)*, pp. 150–155, 2013.

[13] "Encoding categorical features." [Online]. Available:http://scikit-learn.org/stable/modules/preprocessing.html.