Modeling Task Deviations as Eccentricity Distribution Peaks

Vagner Figueredo de Santana
IBM Research
São Paulo, Brazil
vagsant@br.ibm.com

Rogério Abreu de Paula
IBM Research
São Paulo, Brazil
ropaula@br.ibm.com

Claudio Santos Pinhanez
IBM Research
São Paulo, Brazil
csantosp@br.ibm.com

ABSTRACT
Detailed usage data is becoming available through different devices (e.g., personal computer, cell phones, tablets, watches, glasses, wrist bands), in huge volumes, and in a speed that requires new models and visualizations to support the understanding of detailed user actions at scale. Without appropriate methods that summarize or provide means of analyzing large usage data sets, a semantic gap between the event-by-event data and the tasks profile remains. In this context, this work proposes a technique to support the analysis of task deviation from the examination of detailed user interface events streams. From the analysis of 427 event-by-event logged sessions (captured under user consent) of a technical reference website, this work presents how to identify task deviations by using eccentricity distribution. The proposed technique is a promising way of identifying task deviations in large log data sets containing information about how users performed real tasks.

Author Keywords
Interaction log analysis; task modeling; task deviation; client-side events; usage logging; usage modeling; usability; accessibility.

ACM Classification Keywords
H.5.2 User Interfaces – Evaluation/methodology.

INTRODUCTION
The literature counts on different tools and models that support the understanding of user actions while interacting with websites by the analysis of server log files. This approach is being considered for long for different reasons (e.g., ease of obtaining such data from Web servers). Examples of such tools are *Descubridor de Conhecimento en la Web* [9], LumberJack [7], Web Utilization Miner [23], WebCANVAS [5], and WebSIFT [8]. However, server logs do not provide details on how users interacted with user interface (UI) elements. More recent initiatives consider client-side data in order to understand user actions in details, for instance, MouseTrack [1], MultimodalWebRemUSINE [19], UsaProxy [2], WELFIT [22], and WUP [6], WebHint [25], and USABILICS [24]. Hence, client-side data emerged as a way of gathering detailed data, allowing a better understanding of user actions while they are interacting with UIs.

One of the invariants regarding the existing systems is that tools focus on providing insights about the usability level of the evaluated UIs. According to the International Organization for Standardization (ISO) [13], usability is the capacity of a product to be used by specific users to realize certain tasks with efficacy, efficiency, and satisfaction, in a certain context of use. Nielsen presents that usability can also be defined in terms of 5 quality components [17]:

- **Learnability:** How easy users accomplish basic tasks the first time they use the design?
- **Efficiency:** Once users have learned the design, how quickly can they perform tasks?
- **Memorability:** When users return to the design after a period, how easily can they reestablish proficiency?
- **Errors:** How many errors occur, how severe are they, and how easily can users recover from them?
- **Satisfaction:** How pleasant is for the user to use the design?

Considering these definitions, task emerges as a key term. According to Lewis and Rieman [15], "*To get a good interface you have to figure out who is going to use it to do what.*" Thus, supporting the understanding of the following questions is fundamental for grasping the overall user behavior during interaction and the usability level of the UI being used:

- *When in the session do users deviate from the task?*
- *Where in the UI do users deviate from the task?*

When dealing with big data sets, answering one of these questions can reduce the amount of data to consider in further analysis. For example, knowing where the task deviation occurs allows filtering data related to a UI or Web page. In addition, knowing when users deviate from tasks supports filtering data considering timestamp. This work focus in this last scenario, highlighting when users deviated from tasks to support further detailed analysis to be performed by Human-Computer Interaction (HCI) practitioners. In addition, previous work [21] presented first results on proxies for usability metrics considering data gathered during a user test (N=10) run in a controlled environment. In this paper we present how to use eccentricity distribution to model task deviation from the usage data logged during real tasks, remotely and asynchronously, totaling 427 sessions.

This paper contributes with a method to model task deviations as eccentricity distributions peaks and proposes a visualization that summarizes multiple eccentricity distributions. This work is organized as follows: next section presents related works; then the next section details the method, data set, and the graph structure considered; the following section shows the results obtained, and; the last section concludes by discussing outcomes, limitations of this study, and future steps.

RELATED WORK

Initiatives on modeling user interface usage and tasks in Human Computer Interaction grown after initiatives as GOMS (Goals, Operators, Methods, and Selection rules) [12] and CTT (Concur Task Trees) [18]. Recent works present approaches and solutions to represent clicks in a bipartite graph to measure association with queries and clicked pages [16], to provide a probabilistic model regarding interaction of users with user interfaces [3], to present a personalized search solution involving a graph-based representation of the user profile [9], to summarize observational data, supporting the identification of patterns and usability problems [20], to model user engagement based on metrics (e.g., number of visitors, number of clicks, number of visits, average number of page views per visit, average time per visit, number of days a user visited the site, number of times a user visited the site, and the average time a user spent on the site) [14], to present a structure called query-flow graph, a weighted directed graph that models queries performed by users, where vertices represent different queries and edges connect two vertices *(u, v)* if query *v* occurred after query *u* in at least one logged session [10], or to use graph metrics as proxies for usability problems [21].

The literature counts on interesting approaches on how to model usage. However, presented works do not provide means of evaluating and visualizing interaction data in the light of task deviation/completion at scale, without depending on task specifications.

When considering solutions for visualizing usability information from observational data, the literature counts on different approaches. Next, we present techniques and tools that use such approaches:

- **Timeline-based visualizations** – Representing events of interest (of one or more modalities) in a timeline, highlighting when in the session unnecessary events or deviations occurred, as in WUP [6] and MultimodalWebRemUSINE [19].
- **Graph-based visualizations** – Representing events of different granularities as nodes (e.g., client-side events, as in WELFIT [22], or pageviews, as in WebCANVAS [5]) and the sequence in which they occurred as edges.
- **Tree-based visualizations** – Representing visited pages or steps of a task in a tree, as in WebQuilt [26].
- **Sankey-based visualizations** – Representing low granularity data (e.g., pageviews) summarizing visits and different paths users traverse, highlighting where sessions converge/diverge, as in Google Analytics[1].
- **Gaze plot visualizations** – Representing UI elements that users looked at during the session, in the sequence they occurred, and highlighting points where users performed longer fixations, as in Tobii[2].
- **Mouse traces visualizations** – As in gaze plots, mouse traces are presented over the evaluated UI to present the order and where the mouse clicks and movements were performed, as in MouseTrack [1].
- **Heat map visualizations** – Representing frequency of events of interest using a color scale (e.g., from green to red) in a 2D representation. In the context of UI evaluation, the heat map is often used as an over layer to the UI being evaluated, representing UI elements where users clicked/hovered/touched the most or even where users performed longer fixations, as in Tobii.

The presented visualizations count on pros and cons, next we summarize them considering scalability, detail of information represented, and support for HCI practitioners to identify task deviations/completion.

Timelines support representing detailed events occurred in a session and allow the comparison among few sessions; its main shortcoming is scalability, since comparing tens of timelines is impractical.

Graphs and trees support comparing events of different granularities (i.e., from client-side data to pageview level data) and the analysis of multiple sessions, allowing the identification of patterns and outliers; its main shortcoming is the size that such graphs/trees can reach in datasets representing more than few tens of sessions, since it is harder for HCI practitioners to interpret such huge reports. Note that this shortcoming might be an issue before the interpretation takes place, since that some graph drawing tools (e.g., GraphViz[3]) have restrictions for drawing graphs with hundreds/thousands of nodes.

Sankey diagrams are powerful for representing multiple sessions and points in the session where they converge/diverge; its main shortcoming is related to the granularity level of the data considered, since it usually represents interaction at the pageview level, leaving aside detailed interaction data.

[1] www.google.com/analytics

[2] www.tobii.com

[3] www.graphviz.org

Gaze plots and mouse traces support detailed identification of the elements that users interacted with, in the proper sequence that it occurred. This view is especially valuable for evaluating session-per-session and UI-per-UI cases; its main shortcoming is scalability, since consuming the reports of multiple sessions at the same view might be impractical for HCI practitioners.

Eye tracker heat maps support the identification of UI elements that concentrate higher densities of events of interest for one or few UIs; its main shortcoming relates to the analysis of different sessions related to multiple UIs.

From the analysis of the presented visualization methods, graphs and heat maps stood out: Graph supports the analysis of user detailed actions of multiple sessions across multiple UIs/pages. In addition, Graph Mining offers models, algorithms, metrics, and topology attributes that can be used in the context of analysis of task completion/deviation. Heat maps offer a summarized representation in a constrained 2D area. It can be applied for representing density of specific points in or representing combined distributions of a specific metric.

Thus, in this work we present a visualization to summarize task deviation information from the eccentricity distribution. The proposed report is used to highlight when in the session users deviated from the task, without depending on the number of sessions, and without depending on task specifications.

METHOD

This section details the data set analyzed, the data structure used, and how the analysis was performed.

Data set

The data set considered in this work is composed by 427 logged sessions captured during a two-year period. The website where the sessions occurred is called WARAU (Websites Adaptation to Requirements of Accessibility and Usability)[4]. WARAU is a technical reference and UI evaluations repository. The website supports the development of high quality websites integrating technologies as HTML, CSS, and JavaScript, aiming at Accessibility and Usability.

The tasks that users can perform at the website involve, for instance:

- Viewing a reference page that explains a term or how to code a UI component;
- Signing in;
- Logging in;
- Posting a comment;
- Creating a heuristic evaluation.

The event streams related to the 427 sessions analyzed were captured by the evaluation tool WELFIT (Web Event Logger and Flow Identification Tool) [22]. The logging of user interface events occurred remotely under users consent, after the acceptance of an invitation to be part of this user study. An invite was presented once for every user that accessed the website. The data logged comprises all events triggered at the user interface while users performed real tasks. The data set counts on 241,413 events (mean of 564.4 events per session).

The following list presents descriptive information regarding accesses to the reference website in the period this study took place:

- Total of 220,448 sessions (mean of 9,185 per month);
- New sessions represent 89.03%;
- The average duration of the session is 38 seconds;
- Users view in average 1.21 page per session.

These characteristics highlight the role of the website as a source of technical information, since most of the users land in the website coming from a search engine, interact with the content, and then leave the website. This data set was considered because it counts on details performed during the usage, allowing the present analysis of showing task deviations. Moreover, since the website is commonly used as a reference, it would be interesting to identify task deviations and tasks completion characteristics in order to characterize how users use a technical reference website.

Data structure

In order to perform the data analysis and to compare with other techniques summarizing usability information of observational data, the logged data was structured according to the technique presented in [20]. The graph structure considered (also called as usage graph) is a weighted directed graph $G = (V, E, w)$, where:

- $V = A \cup \{start, end\}$ is the set of actions/events triggered at a certain user interface element (e.g., the event *mouseover* triggered in a submit button is represented by one vertex, say v_i, and then a click over the same submit button is represented by a second vertex, say v_j). The vertices *start* and *end* represent the start and the end of the logged session, visit, or any other period being represented in the usage graph.
- Each $v \subset V$ counts on information representing the mean distance in hops from start to v, represented as $d(v)$, the mean timestamp in milliseconds from the start to v, represented as $t(v)$, and the total occurrences of the same event over the same UI element.
- $E \subseteq V \times V$ is the set of directed edges, where e connects two vertices v_i and v_j if v_j occurred immediately after v_i in the logged data, represented as (v_i, v_j).
- w: the total occurrences of (v_i, v_j) in the event stream.
- A vertex v_i is marked as usability problem if $d(v_i) > mean(d(v_j)) + 2\ stdev(d(v_i))$, for v_j representing all outgoing vertices of v_i. The intuition behind this is

[4] http://warau.nied.unicamp.br

to identify cyclic actions, indicating repeated attempts of performing a task or using UI elements.

- Considering time differences, nodes are also marked as usability problems if any of the following is true:
  (1) $t(v_i) > mean(t(v_j)) + 2\ stdev(t(v_i))$; or
  (2) $t(v_i) - t(v_j) > 10$ seconds.
  Where v_j represents all outgoing vertices of v_i. The 10 seconds limit follows Nielsen's 3 Important Limits[5], which presents that 10 seconds is about the time limit for users to keep attention on the task at hand.

Figure 1 presents an example of the usage graph of one of the sessions analyzed and how the usability problems are pointed out by the technique we used. In the figure the ellipses represent UI events; boxes represent UI elements. This example shows how cyclic actions impact in the distances (d) and how usability problems are pointed out in highlighted ellipses.

Data analysis

For the data analysis, usage graphs were built for each session. Then metrics related to diameter, centrality, degree, community detection[6], among others, were computed. For each session, the eccentricity distribution was analyzed and the main characteristics of the distributions were summarized as:

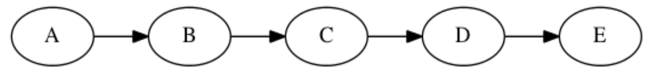
- The presence of peaks in the eccentricity distribution;
- Number of peaks;
- Part in the session that peaks occurred.

The eccentricity of vertex *v* in a connected graph *G* is the maximum graph distance between *v* and any other vertex *u* of G [27]. In the eccentricity distribution, a peak is considered a point in the distribution, say *x*, with a respective count value *f(x)*, surrounded by *x-1* and *x+1*, so that $f(x-1) < f(x)$ and $f(x) > f(x+1)$.

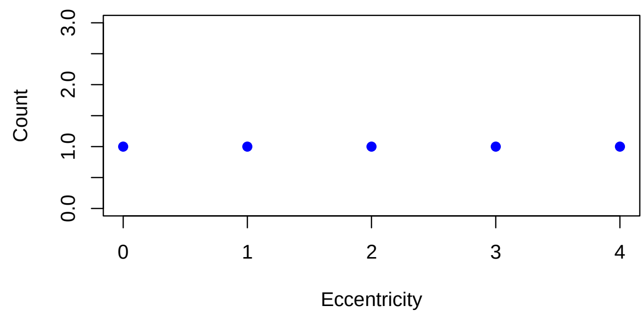
The next examples show how the eccentricity distribution supports insights in relation to task deviations. Moreover, it also allows the comparison of large data sets of detailed actions, supporting the understanding of how users performed tasks and when in the session deviations/cyclic actions occurred.

First, consider a simple event stream resulting in the following usage graph vertices and edges, respectively: *E = {A, B, C, D, E}*; *V={(A, B), (B, C), (C, D), (D, E)}*.

Now consider the eccentricity distribution for the resulting graph (Figure 2 (a) presents the graph and Figure 2 (b) the eccentricity distribution). Note that when analyzing task completion, Figure 2 represents a desirable performance, without deviations and without cyclic actions before reaching the end of the graph.



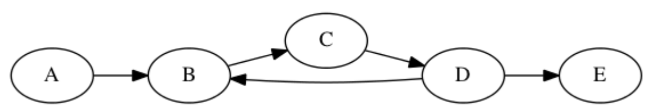
(a)



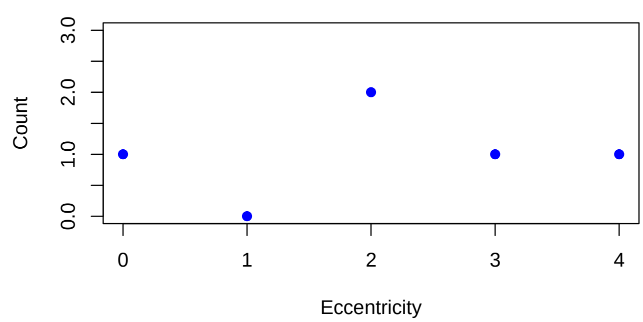
(b)

Figure 2. Simple graph (a) and its corresponding eccentricity distribution (b).

Now consider an event stream with a repeated action (cycle) in a certain UI element, resulting in the following usage graph vertices and edges: *V={A, B, C, D, E}*; *E={(A, B), (B, C), (C, D), (D, B), (B, C), (D, E)}*. Moreover, consider the eccentricity distribution for the resulting graph after considering that the task deviation occurred (Figure 3). Note that the peak represented in the Figure 3 indicates that a deviation occurred in some of the nodes with eccentricity equal to the value indicated in the peak, in this case, eccentricity equal to 2.



(a)



(b)

Figure 3. Simple graph with a cyclic action (a) and its corresponding eccentricity distribution (b); the point in cords (2,2) represents a peak in the distribution.

[5] www.nngroup.com/articles/response-times-3-important-limits

[6] Community detection algorithm used is detailed in [4] and is composed by two phases that are repeated until no modularity improvement is possible, namely: 1) Modularity is optimized by allowing local changes of communities; 2) Found communities are aggregated to build a network of communities.

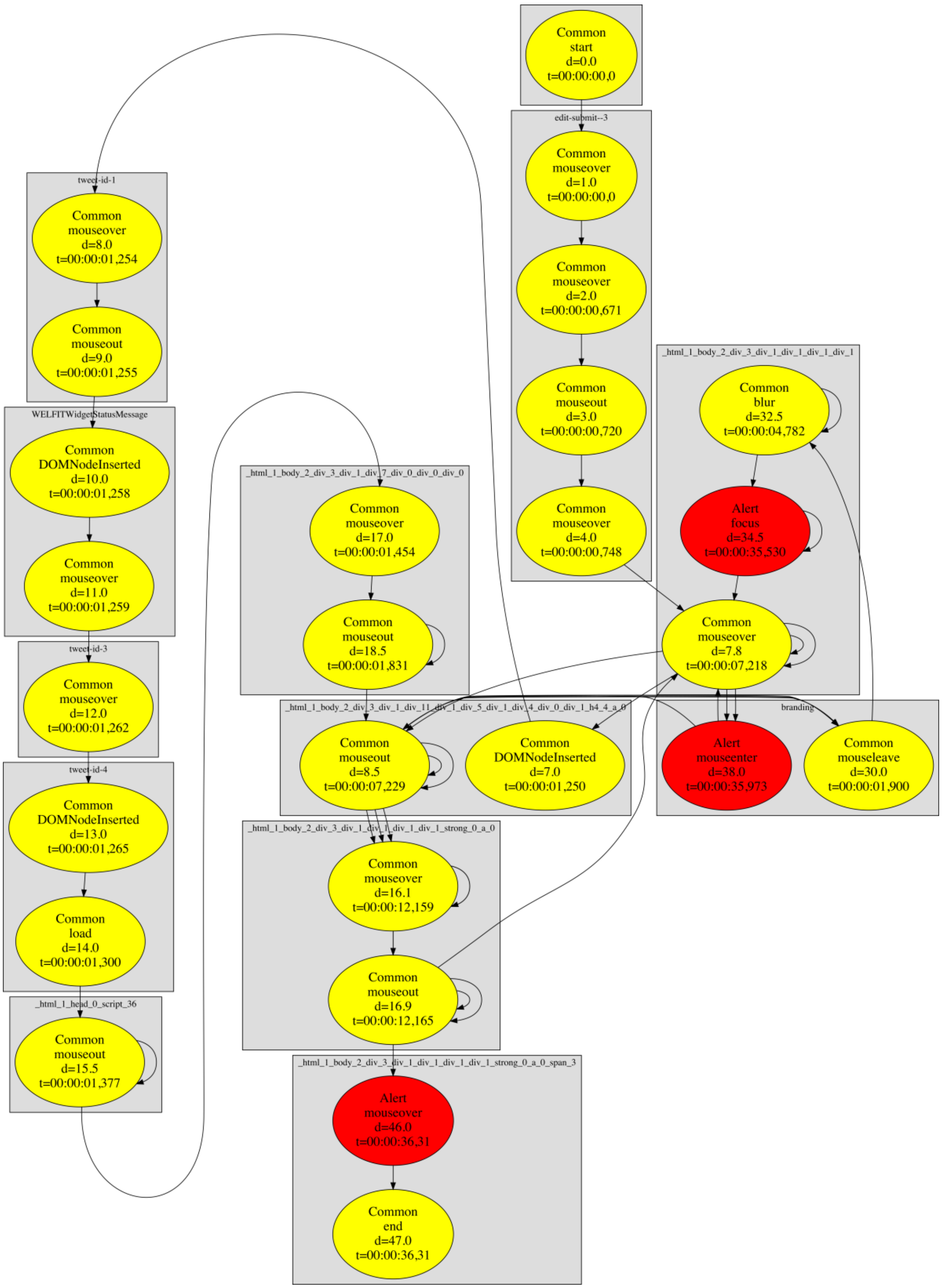
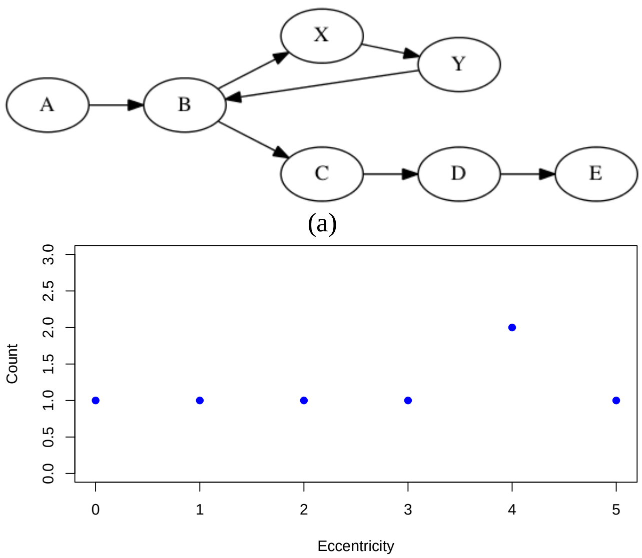


Figure 1. Example of the usage graph of one of the analyzed sessions; highlighted nodes are the usability problems pointed out by the heuristic.

Considering the motif that represents tasks deviation [21], note how the eccentricity distribution supports a summarized representation of the same concept of task deviation (Figure 4). Note that the eccentricity distribution peak represents inversely when, in the session, the deviation occurred. Hence, Figure 4 (b) presents that the deviation occurred in the first quarter of the session.



(a)

(b)

Figure 4. (a) Motif identified representing a task deviation and (b) its corresponding eccentricity distribution.

Once that graph metrics were calculated, correlations were computed and the eccentricity distributions were analyzed, highlighting deviations from task and how the summarized results can provide details of how users performed tasks.

The information regarding the number of peaks is used to group sessions in order to point out tasks that users faced difficulties. After clustering sessions based on the number of peaks, then each of the sessions was analyzed in order to identify the tasks they relate to. The rationale here is to cluster sessions that count on similar number of tasks deviations so that similar distributions will correlate similar behaviors related to task deviations across the evaluated website.

The clusters of sessions based on eccentricity distribution peaks were also considered in the creation of the heat maps, summarizing the task profile for these groups. In order to combine the sessions, all eccentricity distributions were normalized according to count and occurrences and according to time. Finally, heat maps summarizing the eccentricity distributions and the task profiles were built.

RESULTS

Table 1 presents a summary of the data set resulting from the 427 logged sessions. It is possible to see that the high standard deviation values are related to the multiplicity of tasks, i.e., some tasks resulting in small usage graphs with few tens of vertices, while other sessions resulted in usage graphs with few thousand vertices. This effect can also be seen in the number of shortest paths, vertices, edges, among others. On the other hand, the eccentricity distribution is proposed as a more valuable metric, highlighting deviations and providing a richer semantic result than a sole number, e.g., deviations occurred mostly during the first quarter of the sessions. 1

Metric	**Mean**	**S.D.**
Log lines	565.37	2,198.29
Vertices	89.25	147.30
Edges	566.37	2,198.29
Degree	2.03	0.54
Weighted degree	3.65	2.69
Diameter	27.76	12.91
Path length	9.58	4.53
Shortest paths	28,587	238,373
Density	0.07	0.08
Modularity	0.66	0.52
Communities	7.14	2.80
Weakly connected components	1.07	1.15
Strongly connected components	13.18	10.09
Average clustering coefficient	0.05	0.22
Page views	2.21	1.35
Usability problems	19.29	36.54
Eccentricity distribution peaks	1.75	1.22

Table 1. Summary of the usage graphs analyzed.

Considering correlations, Spearman test (ρ) shows that the metrics with significant positive correlation with the number of eccentricity distribution peaks are the following:

- Average path length ($\rho = 0.618$, *p-value* < 0.001);
- Modularity ($\rho = 0.601$, *p-value* < 0.001);
- Diameter ($\rho = 0.595$, *p-value* < 0.001);
- Number of shortest paths ($\rho = 0.413$, *p-value* < 0.001);
- Number of communities ($\rho = 0.380$, *p-value* < 0.001);
- Number of usability problems ($\rho = 0.354$, *p-value* < 0.001).

These results are consonant to a previous work reporting a study in a controlled environment [21], indicating that the number of peaks in the eccentricity distribution is also an interesting metric regarding UI evaluation.

Figure 6 shows the heat map of all eccentricity distributions and highlights how the tasks were performed in the studied website. It shows that task deviations usually occur in the first quarter of the session. Recall that the eccentricity distribution peak represents inversely when in the session the deviation occurred.

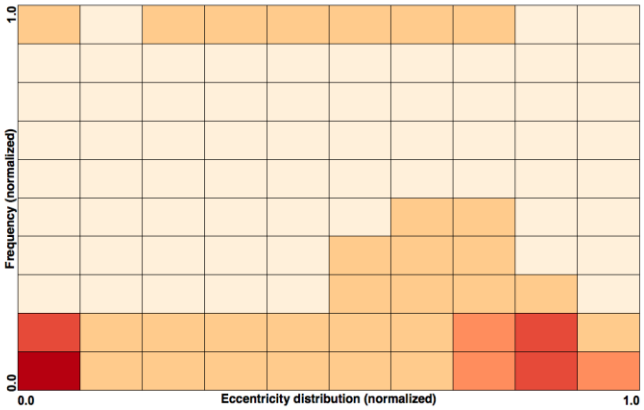


Figure 6. Eccentricity distribution heat map for all the studied sessions; eccentricity and frequency are normalized

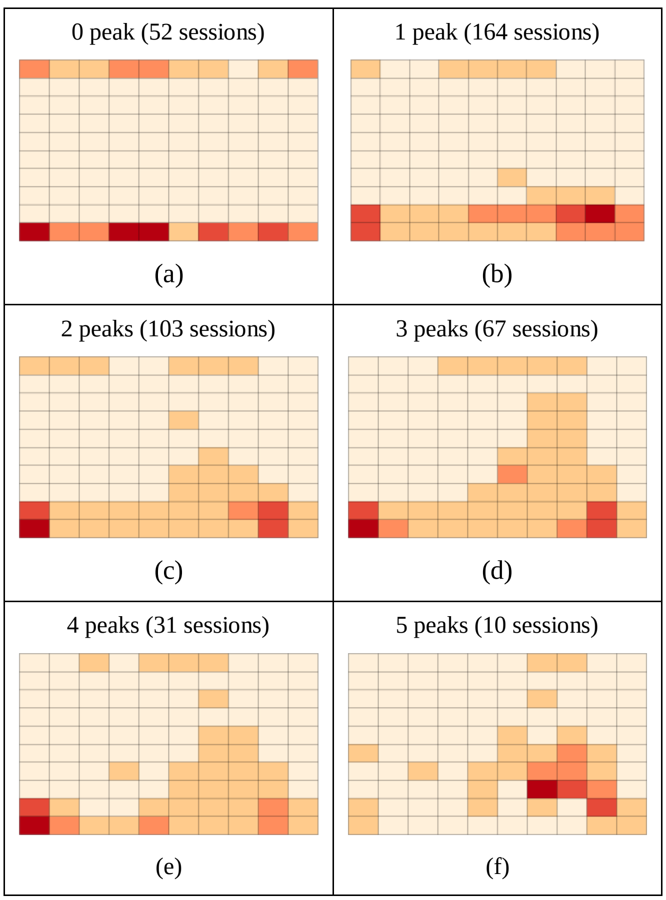


Figure 7. Eccentricity distribution heat map for sessions with 0, 1, 2, 3, 4, and 5 peaks, from (a) to (f), respectively.

Figure 7 presents the different task profiles when considering eccentricity distributions grouped by the number of peaks. Figure 7 (a) presents distributions of usage graphs that have no peak, following the same profile presented in Figure 2. Note that this kind of behavior may be considered as the most effective way of performing a task; it is also worth noting that this is also the behavior expected when the access is automatically performed via scripts/bots. Figure 7 (b) presents the heat map of the eccentricity distributions with 1 peak, related to the 164 sessions. Task deviations occur usually in the middle of the session. Figure 7 (c), (d), and (e) present a deviation concentrated in the first quarter of the session. Figure 7 (f) does not present a clear distribution regarding task deviations; probably because of the small number of sessions with 5 peaks.

In order to relate tasks and eccentricity distribution, each of the sessions was analyzed in detail to identify the tasks the users were performing, then clusters were generated based on the number of peaks of eccentricity distribution to highlight tasks that users faced difficulties. The following tasks were identified in the logged sessions:

1) Perform login;
2) Search for or filter topics;
3) View content;
4) View accessibility evaluation form sample;
5) View the “about page” presentation;
6) View heuristic evaluation form sample;
7) View topics index;
8) Create an accessibility evaluation;
9) View publications page;
10) View references page;
11) Access the administration page;
12) Reset password;
13) View a comment;
14) Delete content;
15) Create a heuristic evaluation;
16) Register user;
17) View the “about page”.

Figure 8 shows tasks occurrence in the clusters generated by considering the number of peaks in the eccentricity distribution. Figure 8 points that sessions in the cluster of 5 peaks require detailed analysis, more specifically how users performed tasks 3, 4, 5, 6, and 7.

Finally, the presence of the same task in different clusters considering eccentricity distribution shows that the same tasks is performed considering different possible paths present in the event streams. Thus, the eccentricity distribution used to model task deviations can be considered as task profile summary of a huge number of sessions that represent detailed interaction data. Moreover, Figure 8 shows that the tasks that are present in the sessions with 4 and 5 peaks are not the most common ones, highlighting that these tasks need review and that related UI components might need improvement.

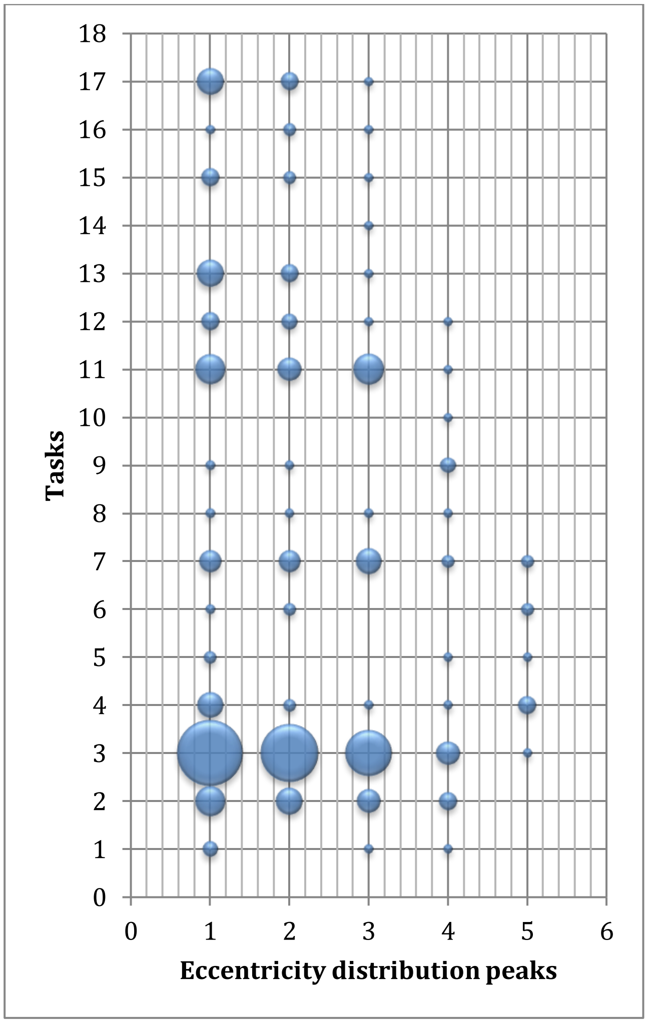


Figure 8. Tasks presence in the clusters of sessions based on the number of eccentricity distribution peaks; size of bubbles represents the number of occurrences found in the sessions.

CONCLUSIONS

This work presented outcomes gathered from an investigation on how graph topology metrics can represent task deviation occurrences. A structure called usage graph was used and a heuristic to point out usability problems were used as base level to evaluate graph metrics and attributes. Moreover, eccentricity distribution was presented as a way of analyzing when and where a task deviation occurred, allowing evaluators to identify task characteristics, task deviations, and, especially, analyze in details the tasks that users face difficulties.

The proposed visualization of the eccentricity distribution supports summarization of multiple sessions represented by event streams of highly detailed interaction data, allowing HCI practitioners to select groups of sessions and specific parts of sessions were task deviations occurred. Moreover, snapshots of such visualization over time support the analysis of learning curve, e.g., by analyzing the different shapes of eccentricity distribution for a certain task or for a certain group of users through time.

Regarding how smooth the tasks were performed, the eccentricity distribution was presented as an interesting proxy, since the lesser the number of deviations from tasks (represented by the peaks), the smoother the eccentricity distribution will be.

Correlations found are consonant to controlled lab studies [21]. The number shortest paths, modularity, and graph diameter are proxies for the number of usability problems pointed by the usage graph structure used. Vertices with high betweenness that are recurrent along the time point out patterns and consistency while users are learning how to use a UI, e.g., an initial page or a dashboard where users start tasks.

The proposed model can be used by HCI practitioners on multiple cases, for instance, in usability tests as a quantitative way of measuring how hard tasks are, in A/B tests comparing two solutions and verifying the eccentricity distributions resulting for the same task and how triggered actions differ. In the case of automated tools, the proposed way of identifying and visualizing eccentricity distributions could be implemented to monitor situations where users are facing difficulties and thus to offer online support, to provide reports depicting detailed actions users perform, or to show the learning curve of a complex computing system.

This work is part of an initiative to build a usage behavior model based on detailed logged data, identifying how graph metrics can be applied to reveal information of how users interacted with UIs. In this work, we contributed with an approach to use the eccentricity distribution to highlight tasks deviations of multiple event streams, pointing out when in the session the deviation occurred. Moreover, we propose a visualization to summarize these distributions, supporting the analysis of task profile at scale.

Regarding limitations of this work, the data set considered is about a technical website. The target audience is composed of developers, content producers, and digital designers. Thus, it does not represent all the Web nor the whole Web audience. Although, the focus of this paper is to present how the eccentricity distribution supports the identification of task deviations. Another point to consider as a limitation is that from 220,448 sessions occurred in the last two years, only 427 (0.19%) were logged. Besides representing a small part of the users of the studied website, this occurred due to the dependency on users accepting to participate in the study, allowing the data logger do capture detailed interaction data. Thus, the number of participants was impacted in favor of privacy and users' choice on providing or not detailed data related on how they perform tasks. Moreover, this is a requirement of the tool used.

Future work involves analyzing event streams to predict actions/errors of a new user based on already logged data and analyzing usage graph topology information regarding UI learning curve involving, for instance:

- Eccentricity distribution changes;
- Number of strongly connected components;
- Mean degree;
- Number of vertices/edges
- Number of shortest paths;
- Edges' weight; and
- Time required to reach vertices with high betweenness.

REFERENCES

1. E. Arroyo, T. Selker, W. Wei 2006. Usability tool for analysis of web designs using mouse tracks. Proc. of ACM CHI 2006 Conf. on Human Factors in Computing Systems, 2 of Work-in-progress, 484-489.
2. R. Atterer, A. Schmidt 2007. Tracking the Interaction of Users with AJAX Applications for Usability Testing. Proc. of ACM SIGCHI 2007 Conf. on Human Factors in Computing Systems, 1347-1350.
3. B. Beckert and M. Wagner 2009. Probabilistic models for the verification of human-computer interaction. In Proceedings of the 32nd annual German conference on Advances in artificial intelligence (KI'09). Springer-Verlag, Berlin, Heidelberg, 687-694.
4. V. D. Blondel, J.L. Guillaume, R. Lambiotte & E. Lefebvre 2008. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment.
5. I.V. Cadez, D. Heckerman, P. Smyth, C. Meek, S. White 2003. Model-Based Clustering and Visualization of Navigation Patterns on a Web Site. Data Mining and Knowledge Discovery.
6. T. Carta, F. Paternò, V.F. Santana 2011 Web Usability Probe: A Tool for Supporting Remote Usability Evaluation of Web Sites. Proceedings of the 13th IFIP TC 13 International Conference on Human-Computer Interaction (INTERACT'11) - Volume Part IV (2011).
7. E. H. Chi, A. Rosien, and J. Heer 2002. Lumberjack: Intelligent discovery and analysis of web user traffic composition. WEBKDD, LNCS, 2703, 1--16, Springer.
8. R. Cooley, P.N. Tan, and J. Srivastava 2000. Discovery of interesting usage patterns from Web data. LNCS, vol. 1836, 163-182, Springer.
9. M. Daoud, L. Lechani, and M. Boughanem 2009. Towards a graph-based user profile modeling for a session-based personalized search. Knowl. Inf. Syst. 21, 3. http://dx.doi.org/10.1007/s10115-009-0232-0
10. D. Donato 2010. Graph structures and algorithms for query-log analysis. In Proceedings of the Programs, proofs, process and 6th international conference on Computability in Europe (CiE'10), Springer-Verlag, Berlin, Heidelberg, 126-131.
11. J. M. Domenech, J. Lorenzo 2007. A tool for web usage mining. IDEAL, LNCS 4881, 695-704. Springer.
12. B. E. John, and D. E. Kieras 1996. The GOMS family of user interface analysis tech-niques: comparison and contrast. ACM Trans. Comput. Hum. Interact. 3, 4, 320-351. http://doi.acm.org/10.1145/235833.236054
13. ISO - International Standard Organization. 1998. ISO-9241 - Ergonomic requirements for office work with display terminals (VDTs), Part 11: Guidance on usability.
14. J. Lehmann, M. Lalmas, E. Yom-Tov, G. Dupret 2012. Models of user engagement. In Proceedings of the 20th international conference on User Modeling, Adaptation, and Personalization (UMAP'12). Springer-Verlag, Berlin, Heidelberg, 164-175. http://dx.doi.org/10.1007/978-3-642-31454-4_14
15. C. Lewis, J. Rieman 1993. Task-centered user interface design - A practical introduction. University of Colorado, Boulder.
16. P. N. Mendes, P. Mika, H. Zaragoza, and R. Blanco 2012. Measuring website similarity using an entity-aware click graph. In Proceedings of the 21st ACM international conference on Information and knowledge management (CIKM '12). ACM, New York, NY, USA, 1697-1701. http://doi.acm.org/10.1145/2396761.2398500
17. J. Nielsen Usability 101: Introduction to Usability. Jakob Nielsen's Alertbox. (2012) http://www.nngroup.com/articles/usability-101-introduction-to-usability/
18. F. Paternò 2003. ConcurTaskTrees: An Engineered Notation for Task Models. The Hand-book of Task Analysis for Human-Computer Interaction. pp. 483-503. Mahwah: Lawrence Erlbaum Associates.
19. F. Paternò, C. Santoro 2008. Remote Usability Evaluation: Discussion of a General Framework and Experiences from Research with a Specific Tool. Maturing Usability. 197-221. Springer-Verlag.
20. V. F. Santana, M. C. C. Baranauskas 2010. Summarizing observational client-side data to reveal web usage patterns. Proceedings of the 2010 ACM Symposium on Applied Computing (ACM SAC 2010).
21. V. F. Santana, R. A. Paula, C. S. Pinhanez 2014 Insights on User Actions from Log-based Graph Topology. Workshop on Log-based Personalization (4th WSCD Workshop). Proceedings of 7th ACM WSDM Conference (WSDM2014), New York.
22. V. F. Santana, M. C. C. Baranauskas 2015. WELFIT: A remote evaluation tool for identifying Web usage patterns through client-side logging. *International Journal of Human-Computer Studies*, *76*, 40-49.
23. M. Spiliopoulou, and L. Faulstich 1998. WUM - A Tool for WWW Ulitization Analysis. In Selected

papers from the International Workshop on The World Wide Web and Data-bases (WebDB '98). Springer-Verlag, London, UK, UK, 184-103.

24. L. G. Vasconcelos, L. A. Baldochi Júnior 2011. USABILICS: avaliação remota de usabilidade e métricas baseadas na análise de tarefas. In Proceedings of the 10th Brazilian Symposium on Human Factors in Computing Systems and the 5th Latin American Conference on Human-Computer Interaction (IHC+CLIHC '11). Brazilian Computer Society, Porto Alegre, Brazil, Brazil, 303-312.

25. A. Vargas, H. Weffers, H. V. Rocha 2010. A method for remote and semi-automatic usability evaluation of web-based applications through users behavior analysis. In MB '10 Proceedings of the 7th International Conference on Methods and Techniques in Behavioral Research. ACM New York, NY, USA.

26. S. J. Waterson, J. I. Hong, T. Sohn, J. A. Landay, J. Heer, and T. Matthews 2002. What did they do? Understanding clickstreams with the WebQuilt visualization system. In Proceedings of the Working Conference on Advanced Visual Interfaces (AVI '02). ACM, New York, NY, USA, 94-102. DOI=10.1145/1556262.1556276 http://doi.acm.org/10.1145/1556262.1556276

27. Eric W. Weisstein. Graph Eccentricity. From MathWorld--A Wolfram Web Resource. http://mathworld.wolfram.com/GraphEccentricity.html.