# Patterns and Sequences: Interactive Exploration of Clickstreams to Understand Common Visitor Paths

Zhicheng Liu, Yang Wang, Mira Dontcheva, Matthew Hoffman, Seth Walker and Alan Wilson

## Supplemental Material: Algorithms

We use Algorithm 1 to sort the sequences in the Sequence View (Section 9.2), by running SeqSort(1, [$D$]) where $D = \{ S_1, S_2, S_3, ... \}$.

**Algorithm 1** SeqSort($idx$, $g$): $g$ is a two dimensional array of sequences, $N$ is the maximum sequence length

```
if $idx > N$ then
    return CONCATENATE($g$)
$g\prime \leftarrow$ []
for $i \leftarrow 0$ to $g$.length do
    g[i].sort(COMPARATOR)
    $hash \leftarrow$ null
    for $j \leftarrow 0$ to $g[i]$.length do
        $prefix \leftarrow g[i][j]$.events.slice(0,$idx$+1).join("-")
        if $prefix \neq hash$ then
            $g\prime$.push([])
            $hash \leftarrow prefix$
        $g\prime[g\prime$.length-1].push($g$[i][j])
return SeqSort($idx$+1, $g\prime$)

function COMPARATOR($S_1, S_2$)
    return $S_1$.events[$idx$].compareTo($S_2$.events[$idx$])
```

It is possible for a pattern to contain multiple occurrences of the same event, and a sequence in the support set can have many occurrences of that event as well. We use Algorithm 2 to search for a key event in a sequence.

**Algorithm 2** EventSearch($P$, $S$, $e$, $i$): $P$ is the pattern, $S$ is the sequence, $e$ is the event ID, $i$ is the index of $e$ in $P$

```
$f \leftarrow 0$
for $j \leftarrow 1$ to $i$ do
    $f \leftarrow S$.indexOf($P[j]$, $f$) + 1
return $S$.indexOf($e$, $f$)
```

*Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org.*
*Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx/*