Mimic: Visual Analytics of Online Micro-interactions

Simon Breslav, Azam Khan[1,2]
Autodesk Research[1]
210 King Street East
Toronto, Canada
first.last@autodesk.com

Kasper Hornbæk
Department of Computer Science[2]
University of Copenhagen
Njalsgade 128, 2300 Copenhagen, Denmark
kash@diku.dk

ABSTRACT

We present Mimic, an input capture and visual analytics system that records online user behavior to facilitate the discovery of micro-interactions that may affect problem understanding and decision making. As aggregate statistics and visualizations can mask important behaviors, Mimic can help interaction designers to improve the usability of their designs by going beyond aggregates to examine many individual user sessions in detail. To test Mimic, we replicate a recent crowd-sourcing experiment to better understand why participants consistently perform poorly in answering a canonical conditional probability question called the Mammography Problem. To analyze the micro-interactions, the Mimic web application is used to play back user sessions collected through remote logging of client-side events. We use Mimic to demonstrate the value of using advanced visual interfaces to interactively study interaction data. In the Mammography Problem, issues like user confusion, low confidence, and divided-attention were found based on participants changing their answers, doing repeated scrolling, and overestimating a base rate. Mimic shows how helpful detailed observational data can be and how important the careful design of micro-interactions is in helping users to successfully understand a problem, find a solution, and achieve their goals.

Categories and Subject Descriptors

H5.2. [Information interfaces and presentation]: User Interfaces – Graphical User Interfaces

General Terms

Measurement, Design, Experimentation, Human Factors

Keywords

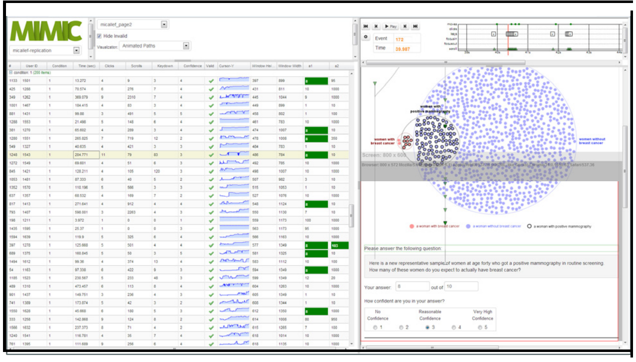
Rich interaction logging; input visualization; crowdsourcing; replication; mammography problem; decision making; modeling

AVI '14, May 27 - 30, 2014, Como, Italy
Copyright 2014 ACM 978-1-4503-2775-6/14/05…$15.00.
http://dx.doi.org/10.1145/2598153.2598168

1. INTRODUCTION

We present a novel open source web application called Mimic to help interaction designers in analyzing micro-interactions [28] to develop hypotheses about underlying causes of observed online behaviors. Statistical characterizations and aggregate visualizations of interaction data, such as correlations and heat maps, can be helpful in understanding high-level usage patterns but it may be impossible to discover the root causes of such patterns without looking at many individual user trials. Mimic is specifically designed to help interaction specialists play back many user sessions by providing sparklines and event timelines to make it easy to find relevant trials, by making it easy to step through selected trials, and by quickly playing back a simulation of the input by using event-based animation instead of real-time playback.

To illustrate the benefits of Mimic, we chose a classic decision problem called the Mammography Problem [6]. As a case study of how Mimic can be used to study micro-interactions to help find otherwise unknown insights, we first closely replicate a crowdsourcing experiment by Micallef et al. [19] that presented the mammography problem as text-only or as text with a visualization. The authors of [19] commented that "subjects' accuracy was remarkably low… inconsistent with previous studies" and that "the reasons for this are unclear." By using Mimic to visualize the micro-interactions, we generate several additional hypotheses of possible sources of user error.


MIMIC
women with breast cancer
women without breast cancer
Screen: 800 x 600
a woman with breast cancer
a woman without breast cancer
a woman with positive mammography

Please answer the following question:

Here is a new representative sample of women at age forty who got a positive mammography in routine screening. How many of these women do you expect to actually have breast cancer?

Your answer: 8 out of 10

How confident are you in your answer?

No Confidence		Reasonable Confidence		Very High Confidence
○ 1	○ 2	● 3	○ 4	○ 5

Figure 1: The Mimic interface with Overview panel of user trials (left), and the event timeline and visualization playback window (right).

2. RELATED WORK

Observational data is a central part of usability investigation and human-computer interaction (HCI) research. While statistical analysis of these data is a mainstay of HCI, we focus here on methods that may provide novel insights for interaction designers and analysts of micro-interactions. To this end, Exploratory

Sequential Data Analysis (ESDA) [29,30] has been proposed as an analysis technique to "encourage analysts to be suspicious of numerical summaries of data, to find powerful visual re-expressions of data that might reveal new structure, and to focus on hypothesis generation rather than hypothesis testing" [7].

2.1 Command-level Capture & Visualization

HCI research has explored interaction command and input histories for decades. Recent relevant work, to help developers understand the real usage of their systems, interactive history tools have been shown examining all commands in an application using Tableau [13]. Event logs, for collection and study of long-term application use has also been demonstrated [14]. Systems to help users to better learn to use large software applications have been proposed visualizing user behavior of command use in Chronicle [11] and widget use in Patina [18].

2.2 The Move to Crowdsourcing

Research has studied crowdsourcing for input capture for user interface evaluation [17], visualization design [12], web site design [34], predicting task performance [27], or even modeling task performance [10]. Search engine providers have seen benefits from capturing low-level cursor data [16] for predicting aspects of the user experience [23]. To debug web applications, researchers have built custom webkits to record and replay webapp states [3], instrumented low-level JavaScript events and callbacks [1,20], and developed remote debugging and live patch tools [21].

2.3 Micro-interaction Capture and Visualization

Work most directly related to the goals of Mimic involve the capture and visualization of low-level input data [27] and the development of general tools to support crowdsourced evaluation of web interfaces [24]. Web Usability Probe [4] uses a timeline approach similar to our timeline panel and Nakamichi et al. [22] replay a visualization of mouse movements over the user interface. Stieger and Reips [32] show a static visualization of these data from an online survey. In the context of domain-specific tools, [26] visualizes interactions for geo-analytics and [22] creates visual representations of operation histories of freeform tools.

Previous works have not focused on micro-interaction design and therefore have either not been detailed enough to support interaction debugging or have not examined "in the wild" data enough to realize the importance of scrolling, window resizing, or other contextual actions that users do. Also, by attempting to faithfully visualize these event sequences, we have seen how interrelated they are with respect to how they affect micro-interaction design.

3. MIMIC

The Mimic visual analytics system can help interaction designers to develop a deeper understanding of how users interact with web content. Although we added traditional heat maps as a visualization type as well as the ability to run statistics, through a simple JavaScript scripting interface, Mimic is designed around the goal of making it as easy as possible for a designer to directly look at large numbers of individual user sessions. We believe that this phase of interaction design is often overlooked because of the general cost, in time and effort, to do so.

We observe interactions "in the wild" through remote logging of client-side events. These weblogs contain mouse button and movement events, keyboard typing and virtual keys presses, window and control sizing and focus events, etc. Playing back user sessions in the Mimic web application, effectively mimicking the real-time input from the user, can reveal important details that aggregate visualizations and statistics do not. For example, a user may change their answer several times before moving on to the next field or submitting their answers to the server (see Figure 7). This information can be used to see if the participant is improving their answer or if they are worsening it, and this may be correlated to their level of confidence. If the user's final submitted answers are the only ones that are examined, this kind of analysis would not be possible.

3.1 Client Side Instrumented Web Pages

Crowdsourcing research has typically just recorded the final user response so no further insights can be made about possible causes of error. To disaggregate the final user response, we instrumented the web pages of an online questionnaire to return detailed interaction metrics to a central database, similar to [4,32]. Mimic only requires JavaScript to be enabled on the participant's browser. Each webpage would include mimic.js in the webpage source to capture mouse movement, clicking, double clicking, scrolling, pressing of keys on the keyboard, the cursor entering and leaving the window, the initial size and resize event of the window, as well as focus and blur events (events that happens when the user changes tabs or navigates to different applications). As it is not practical to record every type of browser event and the associated metadata, our workflow has been to (a) pose a specific analytic, (b) consider which data are needed to support such analysis, and (c) determine how to best visualize the data in support of the analysis. This is the iterative methodology we used to determine which events and data are captured and sent to the server-side database.

3.2 Server Side Web Application

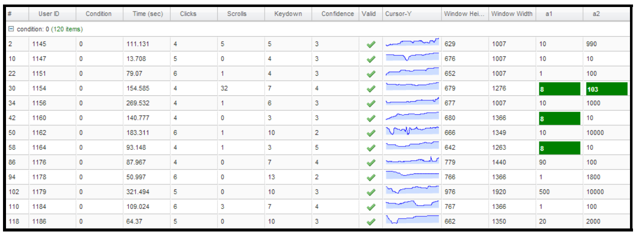
To explore these captured datasets, we developed the Mimic web application that plays back user sessions of interactions with web pages, from a relational database, effectively simulating the users actions, including cursor movement and keyboard presses.

We used Amazon Mechanical Turk to recruit users who would be directed to our website to answer a questionnaire where we would capture user input which was then sent to a database. To access the database, the interaction analyst would log in to the Mimic web interface (see Figure 1). The Mimic panel layout, as well as its features, is designed to help the interaction designer to be able to quickly look at many individual user sessions to develop insights as to why participants respond to the questions in particular ways.

3.2.1 Overview Panel

The left half of the Mimic screen is the Overview panel (see Figure 2) where each row represents a participant together with their key metrics including condition, answer, time, number of clicks, and sparkline showing a plot of the y-position of the cursor, similar to the progression maps in [15]. This interface gives the designer an overview of all the trials. Note that in our example, Mimic was being used from Mechanical Turk so that a between-subjects design could easily be used and therefore each trial is performed by a different Turk user.

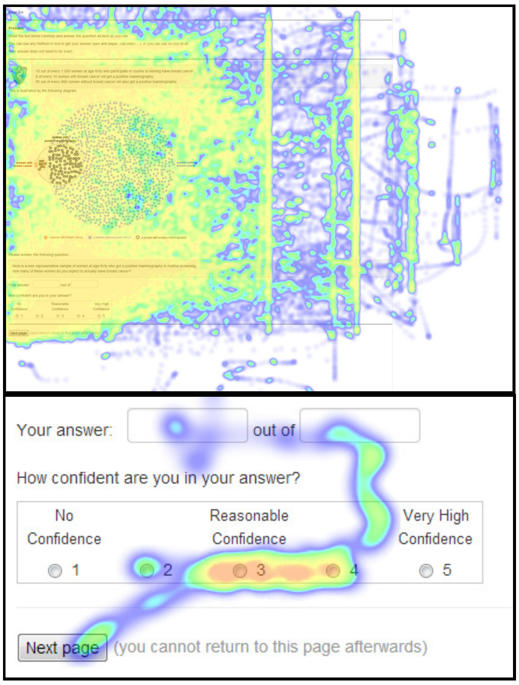
These data can be grouped by experimental condition, thereby allowing all the trials in a condition to be selected (by clicking on the condition row) instead of individual trials or groups of trials. Three types of Visualizations can be chosen in the drop-down list: Heat map, Playback, and Custom. Occasionally, trials may be invalid, likely due to browser issues, indicated by an empty data field or negative time duration, etc. These specific rows can be marked as invalid and will not be included in visualizations or calculations, and can be hidden or shown.



#	User ID	Condition	Time (sec)	Clicks	Scrolls	Keydown	Confidence	Valid	Cursor-Y	Window Hei...	Window Width	a1	a2
⊟ condition: 0 (120 items)													
2	1145	0	111.131	4	5	5	3	✓		629	1007	10	990
10	1147	0	13.708	5	0	4	3	✓		676	1007	10	10
22	1151	0	79.07	6	1	4	3	✓		652	1007	1	100
30	1154	0	154.585	4	32	7	4	✓		679	1276	8	103
34	1156	0	269.532	4	1	6	3	✓		677	1007	10	1000
42	1160	0	140.777	4	0	3	3	✓		680	1366	8	10
50	1162	0	183.311	6	1	10	2	✓		666	1349	10	10000
58	1164	0	93.148	4	1	3	5	✓		642	1263	8	10
86	1176	0	87.967	4	0	7	4	✓		779	1440	90	100
94	1178	0	50.997	6	0	13	2	✓		766	1366	1	1800
102	1179	0	321.494	5	0	10	3	✓		976	1920	500	10000
110	1184	0	109.024	6	3	7	4	✓		767	1366	1	100
118	1186	0	64.37	5	0	10	3	✓		662	1350	20	2000

Figure 2: The trial Overview Panel, used to select which trials to focus on such as those with correct (green background) or incorrect answers, interesting y-position mouse cursor sparkline, time, clicks, etc.

The interaction designer can select any combination of rows or entire condition blocks. Clicking on the column heads, the table is reordered as one would expect. Once a trial is selected, the designer can use the cursor keys to easily step through trials and see the visualization update (see Figure 3). This mechanism makes it extremely easy to quickly view many sessions to develop a good sense of the user experience that participants are having.



Your answer: out of

How confident are you in your answer?

No Confidence		Reasonable Confidence		Very High Confidence
○ 1	○ 2	○ 3	○ 4	○ 5

Next page (you cannot return to this page afterwards)

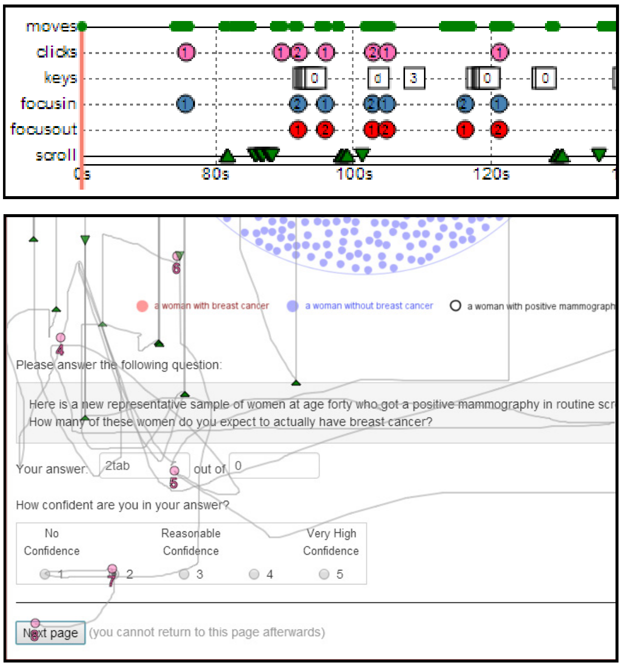
Figure 3: The heat map visualization type show in the Visualization Panel with (top) all 200 participant trials selected and (bottom) a single trial close-up on answer section.

3.2.2 *Visualization Panel*

The Visualization Panel is shown on the right half of the Mimic window. The Heat Map visualization type shows the aggregate behavior of many participants. This can suggest large interaction patterns (such as the evidence of scrolling in different window sizes in Figure 3), but generally, this type of visualization does not help explain why a certain micro-interaction behavior is observed.

The Timeline visualization type has two components: the timeline panel and the playback panel below it (see Figure 4). While static cursor movement heat maps and mouse trails with numbered clicks can be displayed (see Figure 7), the primary benefit of Mimic comes from the playback of the input events together with the event simulation (sending the actual events to the target controls on the page) to convey the user's experience with the webpage (see accompanying video figure). The designer can replay the entire user session or they can click in the timeline, together with zooming and panning, to navigate through a session.

An event-based playback mode presents the recorded events as quickly as possible in the Visualization Panel removing any user delays. A real-time playback mode presents the recorded events in the Visualization Panel at the same time as they happened, relative to the beginning of the recording. This mode truly mimics the user behavior as closely as possible conveying the subtle hesitations and delays that were in the original input stream. While this mode is the most faithful to the original input, it also takes much longer to study a trial in real-time.


moves
clicks
keys
focusin
focusout
scroll
0s
80s
100s
120s
a woman with breast cancer
a woman without breast cancer
a woman with positive mammograph

Please answer the following question:

Here is a new representative sample of women at age forty who got a positive mammography in routine scr
How many of these women do you expect to actually have breast cancer?

Your answer: 2tab out of 0

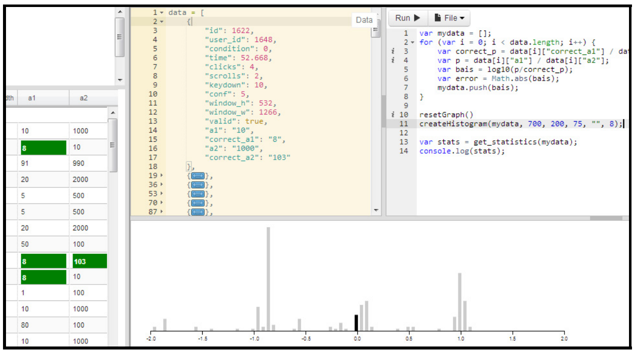
How confident are you in your answer?

No Confidence		Reasonable Confidence		Very High Confidence
○ 1	○ 2	○ 3	○ 4	○ 5

Next page (you cannot return to this page afterwards)

Figure 4: The Timeline Panel. Input events are displayed indicating timing and correlations (top), while the mouse movement and keyboard presses are displayed in the corresponding playback panel (bottom).

The Custom visualization type (see Figure 5) can be used for generating other visualizations, like histograms or graphs, to further explore the data in an unconstrained way using procedural programming through JavaScript code. Statistics can also be calculated by making server side calls to SciPy library (http://scipy.org/) through the scripting panel to help develop data-driven models by looking for correlations or seeing how a graph changes as different subsets of input data are selected. The data view (top-left subpanel) shows what data values will be considered as input to the script. This view is repopulated when the designer changes their selection of rows in the Overview panel. As shown in Figure 5, a data record can be expanded to show both the names of the data fields available for use in the script but also the values of the fields. The code view (top-right subpanel) can be used to load and save JavaScript programs, to the same database, to re-use scripts. The visualization view (bottom subpanel) displays any visualizations that may be drawn by the script. Note that as the JavaScript program is run normally, designers can also use typical browser debugging facilities to output to the console, display alerts, etc.



a1	a2
10	1000
8	10
91	990
20	2000
5	500
5	500
20	2000
50	100
8	103
8	10
1	100
10	1000
80	100
10	1000

```
data = [
    {
        "id": 1622,
        "user_id": 1648,
        "condition": 0,
        "time": 52.668,
        "clicks": 4,
        "scrolls": 2,
        "keydown": 10,
        "conf": 5,
        "window_h": 532,
        "window_w": 1266,
        "valid": true,
        "a1": "10",
        "correct_a1": "8",
        "a2": "1000",
        "correct_a2": "103"
    },
    {…},
    {…},
    {…},
    {…},
    {…},
```

```
var mydata = [];
for (var i = 0; i < data.length; i++) {
    var correct_p = data[i]["correct_a1"] / da
    var p = data[i]["a1"] / data[i]["a2"];
    var bais = log10(p/correct_p);
    var error = Math.abs(bais);
    mydata.push(bais);
}

resetGraph()
createHistogram(mydata, 700, 200, 75, "", 8);

var stats = get_statistics(mydata);
console.log(stats);
```

Figure 5: The Custom visualization type provides designers with a data view (tan background) and a simple JavaScript window for scripting of custom analysis and visualization.

4. RESEARCH QUESTIONS

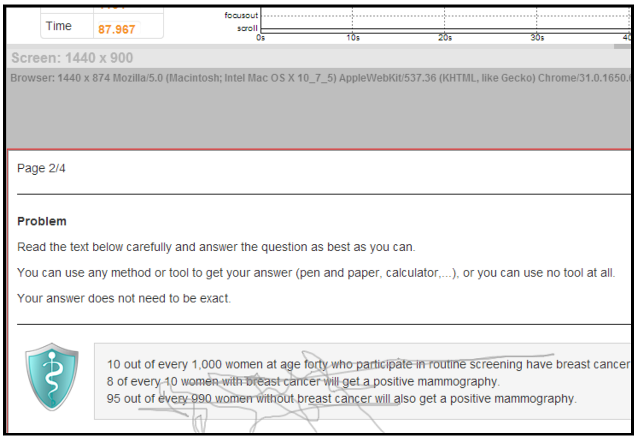
Driven by the replication case study that we performed, we developed a number of research questions which had not been discussed previously.

4.1 Q1 Visibility: What is the user looking at?

The challenging problem of determining what part of the online user interface is being seen and what contents are in that part, in an unobtrusive way, had not been addressed in previous crowdsourcing research.

We used several events to capture and portray the visible sub-region of the current web page. We added the capture of screen resolution, browser window position and size, sub-window home position and dimensions, and focus, blur, enter and exit events. To visualize these data, we rendered shaded regions in the Visualization Panel to indicate the screen size, the browser position, and the sub-window of the portion of the page that was visible to the user (outlined in red, see Figure 6). When animated during session playback, the moving regions clearly convey this subtle but important part of the user experience.

Within the page, the display or hiding of dynamic content must also be handled. Furthermore, right-click context menus, browser dialog boxes, and the selection of items in pop-up windows, such as drop-down lists, all effected the visibility question. To address these issues in a single consistent way, we simulated input events by passing them to the controls on the underlying page in the visualization iFrame. Note that changes in window content or dimensions in the middle of a trial invalidates existing mouse trails and heat maps. During session playback, if this occurs, we clear the trails or heat map buffer and restart their accumulation. Also, cursor movements over pop-ups convey transient actions that are only valid while the pop-up is shown. However, as these are relatively minor and can easily be understood to the analyst watching the trial, we retain these marks in the visualization.



Time 87.967

Screen: 1440 x 900

Browser: 1440 x 874 Mozilla/5.0 (Macintosh; Intel Mac OS X 10_7_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/31.0.1650.

Page 2/4

Problem

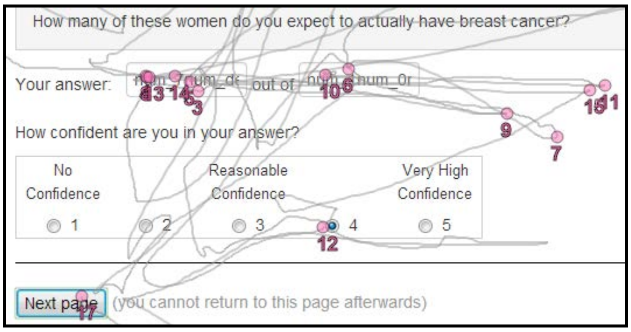
Read the text below carefully and answer the question as best as you can.

You can use any method or tool to get your answer (pen and paper, calculator,...), or you can use no tool at all.

Your answer does not need to be exact.

10 out of every 1,000 women at age forty who participate in routine screening have breast cancer
8 of every 10 women with breast cancer will get a positive mammography.
95 out of every 990 women without breast cancer will also get a positive mammography.

Figure 6: The Visualization Panel showing the position and sizing information for the screen (light grey), browser (dark grey), and page sub-portion (red outline) currently visible.



How many of these women do you expect to actually have breast cancer?

Your answer: num_de out of num_0r

How confident are you in your answer?

No Confidence		Reasonable Confidence		Very High Confidence
○ 1	○ 2	○ 3	● 4	○ 5

Next page (you cannot return to this page afterwards)

Figure 7: The playback panel showing the number of clicks and changing text values in the answer text-fields indicating how many times the user changed their answer.

4.2 Q2 Input Field History: Did the user change their answers before submitting?

As in [32] we found that users would revisit many parts of the page, especially when they reach opinion questions such as the answer and confidence questions (see Figure 7). The capture of virtual keys helps to measure the number of times an input was changed, together with the capture of its current value. Virtual keys include tab, delete, cursor keys, home, end, page up and page down keys. By capturing the current control ID during a key press, we can associate keys with a specific text-field or control. This approach is also needed to correctly visualize trials that have no mouse cursor movement whatsoever.

Click-order numbers are shown next to mouse click positions and event simulation, where the typed text is sent directly into text-field, shows key presses. Together, these techniques show when answer changes occurred before submitting results. To see groups of keys entering a value, the timeline panel (Figure 4 top) can be used to zoom in to more clearly show the values of intermediate answers. When hovering over key icons on the timeline, a tooltip shows the name of the associated control.

4.3 Q3 Re-visitation: Did users return to Areas of Interest?

To detect backtracking patterns, we added capture of the current HTML <div> so that we could highlight the general area of interest during playback and to support some form of automated div re-visitation metric or analysis. To visualize this behavior, the mouse click numbering together with the div grouping could show when this occurred.

It was not uncommon in the case study to see users answer the questions, select an answer for the confidence question (e.g. “How confident are you in your answer?”), then move back up to re-examine the textual problem or the visualization to presumably help them determine if they were as confident as they had reported. In several cases, after this re-examination, users lowered their confidence level before submitting the answers on that page.

5. REPLICATION OF DECISION PROBLEM EXPERIMENT

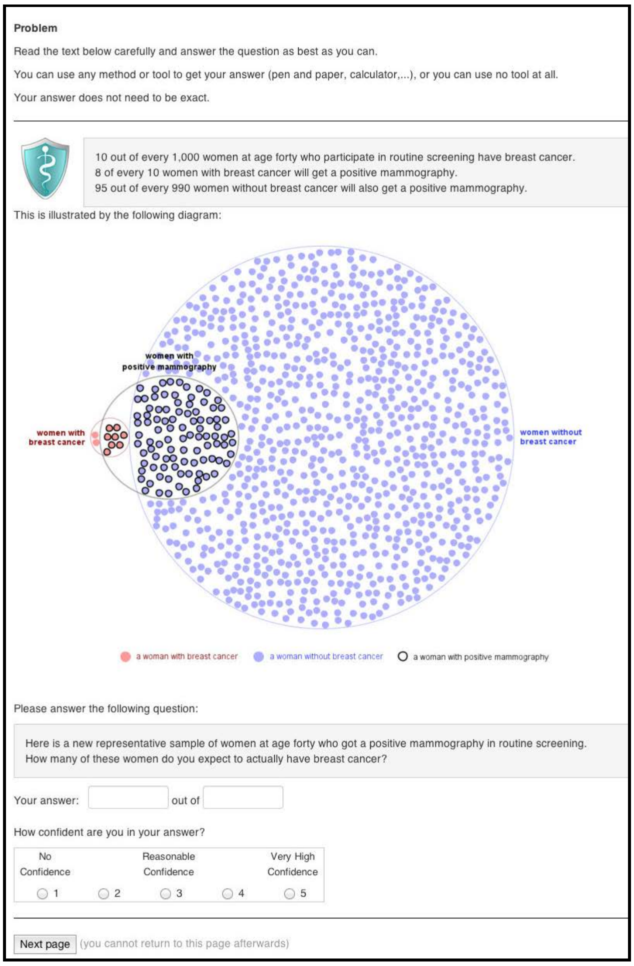
As mentioned earlier, we chose to test Mimic using a classic decision problem called the Mammography Problem [6] recently replicated by Micallef et al. [19] on Mechanical Turk. Decision making under uncertainty is often tested with condition probability problems and researchers have studied several graphical representations of these problems to improve user accuracy, such as icon arrays [5], Euler diagrams [25], and trees [33]. The Mammography Problem itself is given as the text in the boxes with a grey background in Figure 8.

Micallef et al. attempted to use crowdsourcing to improve the ecological validity over previous work for the canonical Mammography Problem where accuracy levels were 48% of 25 participants [31] and 34.7% of 98 participants [2]. However, user performance decreased significantly when done online with participants only achieving accuracy levels of 21% of 24 participants in their first experiment, and an exact answer accuracy of only 5% out of 120 participants in a second experiment. To help us develop insights as to why user performance is so low, and even lower when using crowdsourcing, we used Mimic to understand what micro-interactions may play a role.

5.1 Design, Participants, and Hypothesis

We performed a strict replication [9] of the Micallef et al. experiment using their web pages and image source material[1]. In addition to their pages and images, we replicated many aspects of their experiment including the test name and payment amount on Mechanical Turk. Specifically, we replicated their Experiment 2 with the same dependent variables (Bias, Error, Exact answer, Time and Confidence), the same between-subjects design, and the same four web pages. However, we omitted two conditions where the authors modified the text of the classic Mammography problem and only ran two of the Micallef visualization type conditions where the original text was preserved to allow for comparison to previous work: Text-only (VT, was V0 in [19]) and Euler (VE, was V4 in [19]) where we used the same text as Text-only but also presented an area-proportional Euler Diagram with randomly positioned glyphs, as shown in Figure 8.

Three notable exceptions were (a) we added a browser compatibility script to the page for improved consistency of rendering between various browsers and a Mimic instrumentation script, (b) we restricted Turk users to use desktop browsers only and disallowed workers using mobile devices, and (c) we ran 200 participants per condition instead of 120. The Micallef hypothesis was: the Euler (VE) condition does not lower the mean Error by more than 0.1 points compared to Text-only (V\).



Problem

Read the text below carefully and answer the question as best as you can.

You can use any method or tool to get your answer (pen and paper, calculator,...), or you can use no tool at all.

Your answer does not need to be exact.

10 out of every 1,000 women at age forty who participate in routine screening have breast cancer.
8 of every 10 women with breast cancer will get a positive mammography.
95 out of every 990 women without breast cancer will also get a positive mammography.

This is illustrated by the following diagram:

women with positive mammography
women with breast cancer
women without breast cancer
● a woman with breast cancer
● a woman without breast cancer
○ a woman with positive mammography

Please answer the following question:

Here is a new representative sample of women at age forty who got a positive mammography in routine screening.
How many of these women do you expect to actually have breast cancer?

Your answer: ______ out of ______

How confident are you in your answer?

No Confidence		Reasonable Confidence		Very High Confidence
○ 1	○ 2	○ 3	○ 4	○ 5

(you cannot return to this page afterwards)

Figure 8: The primary question page of the Micallef Replication Experiment showing the problem statement together with a Euler diagram visualization of the problem.

5.2 Results

5.2.1 *Bias*

Figure 9 shows the distribution of Bias (a normalized error metric to better characterize the “distance” of the participant’s answer from the exact answer) for the two conditions suggesting our results closely replicates the previous work.

The median biases were -0.004 and -0.51 for VT and VE. The differences are not statistically significant (Kruskal-Wallis, H = 3.12, p = .08), unlike in Micallef et al.’s study.

5.2.2 *Accuracy*

Exact answers for VT and VE were 3.5% and 2.0% and were 3.3% and 5.0% in Micallef. Median errors for VT and VE were both 0.8902. Mean errors were larger (0.77 and 0.67) confirming the Micallef hypothesis. The differences between conditions, shown in Figure 11, are statistically significant (Kruskal-Wallis, H = 7.66, p = .005).

[1] http://www.aviz.fr/Research/Bayes

5.2.3 *Confidence*

Like Micallef, confidence scores had a median of 3 (“reasonably confident”) and means between 3.34 and 3.29. Differences were not significant (Kruskal-Wallis, H = 0.18, p = .67). Correlation with error was low (r = -0.04).

5.2.4 *Time*

Completion times were similar to Micallef and similar across both conditions (ANOVA, F(1, 399 = 0.067, p = .8). Median for VT was 106.35 sec. (M = 140.1, SD = 112.38) and VE was 115.17 sec (M = 142.98, SD = 109.59).

5.2.5 *Strategies*

Participants reported that they tried to get the exact answer 39% and 42% for VT and VE while 54.5% and 53% did not, and 6.5% and 5.0% were unsure. As for the degree to which participants reported relying on the diagram, the median answer to the 5-point Likert scale was 3 for VE (M = 3.11, SD = 1.51). Participants who were assigned to VT and later asked whether they would have used the diagram gave a median answer of 4 (M = 3.51, SD = 1.43).

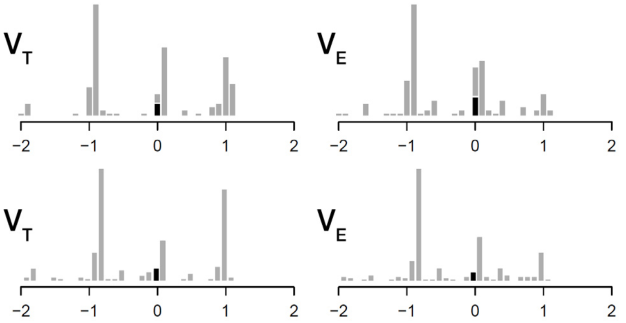


Figure 9: Distribution of biases in answer. Black bars are exact answers. (Top: Micallef results, Bottom: our results).

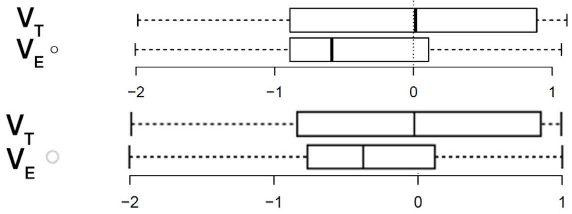


Figure 10: Biases in answers per presentation type. (Top: Micallef results, Bottom: our results).

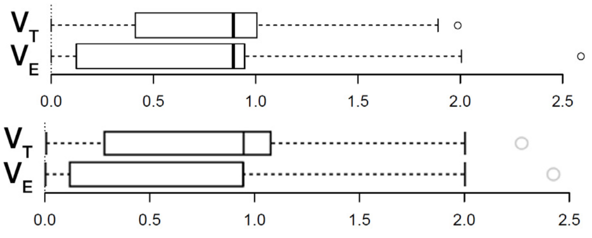


Figure 11: Answer errors per presentation type. (Top: Micallef results, Bottom: our results).

In summary, our study replicated Experiment 2 in [19] with the notable exception of a much lower VE accuracy of 2% down from 5%. But the question remains: why did the visualization not help user performance and why is performance so low overall?

5.3 Insights Gained Using Mimic

Using Mimic to examine many individual recorded sessions, we were able to find a number of exemplary micro-interactions as well as several issues not reported in [19], and we generated several ideas on how to improve the way problems are presented (see Table 1).

Table 1: Micro-interaction examples, issues found using Mimic, and ideas based on Mimic findings for improvement to problem presentation

Examples of Micro-interactions
changing answer
scrolling to expose specific part of the page
revisiting an area of interest
value entered for a partial answer (numerator or denominator)
reading (text or visualization)
Issues found using Mimic
excessive scrolling due to large size of visualization
instructive to measure partial answers independently
possible ambiguity of base rate sub-question
answer quality declines with number of changes
confidence declines with number of changes
Ideas for Improvement to Problem Presentation
do not separate problem description and question text
make page contents size-aware, resize visualization to fit
show count of items in legend to avoid explicit counting
visually emphasize base rate

5.3.1 *Disaggregate Analysis on Numerator and Denominator*

While the normalized probability metric, bias, and error were computed, this practice aggregates the two values entered by participants in the natural frequency format: numerator and denominator. To further analyze exactly correct responses, we examined exactly correct numerators (20%, 80/400) independently from exactly correct denominators (4.3%, 17/400) because we saw that many trials in the Overview panel (see cells with green background in Figure 1) had one or the other correct but not both. While this could have been discovered without Mimic, [8,19,25] did not report this finding suggesting that an ESDA approach is beneficial.

If looking at the difference between the correct numerator for VT (43/200) and VE (37/200), more participants got the correct answer with less information i.e. no visualization. This may be caused by the large size of the Euler diagram which seemed to incur a large scrolling cost for participants which we observed in many trials. A post-hoc Spearman's rank correlation coefficient (rS= -0.13, p= .008) between numerator bias and scrolling events, indicates a weak correlation. However, there was no correlation with the denominator bias and scrolling (rS= -0.06, p= .19). Again, looking at individual answers, we see that the denominator is heavily dominated by an answer of 1000 (42.5% = 170/400 with VT = 86/200, VE = 84/200). This may suggest that, independent of the problem visualization, participants interpreted the denominator as being the superset, that is, the entire population.

5.3.2 *What part of the screen is the user looking at?*

In examining the trials using Mimic, we quickly found that many participants performed large amounts of repeated scrolling up and down on the primary question page when the visualization was present (see Figure 8). The very large size of the graphic prevented the two important parts of the text, the problem statement and the specific question (the two text boxes in Figure 8 with a grey background) from both appearing in the participant's window at the same time. This divided-attention problem was not mentioned in [19] and may not have been recognized as a potential factor in user performance. This suggests that the visualization condition could have been much more effective if the Euler diagram could be scaled to be fully visible together with the relevant text when the participant's window is too small.

5.3.3 Did the user change their answers before submitting?

Using the Timeline panel, it could be readily seen when a participant changed their answer before submitting their entries. Again, using Mimic's visualizations, we observed that participants would often return to the problem text or question text after entering an answer in the text-field, not necessarily improving the result. This is confirmed by a post-hoc Spearman's rank correlation coefficient (rS =-0.24, p < .001) between overall bias and key presses indicating that as the number of modifications increases, the more the correct answer is underestimated. Looking at the individual numerator and denominator, the correlation between bias and key presses is even stronger (numerator rS=0.27, p < .001 and denominator rS =0.4, p < .001).

5.3.4 Did user confidence change during the trial?

In [19], confidence was reported as a single value. By looking at individual trials, we see that most participants did not change their confidence level (272/400, M=3.41). However, 78/400 changed answers once (M=3.19), 14/400 changed twice (M=2.5), 11 three times (M=3.18), 1 four times (M=3), and 1 five times (M=3). The remaining 23 participants had no click data (M=3.17).

5.3.5 Reading Text and the Visualization

We observed participants moving the mouse cursor carefully over lines of text, indicating reading (see Figure 6). Also, in some cases, we saw participants carefully counting dots in the Euler diagram indicating that they are "reading" the visualization. In [19], a separate questionnaire was done where one participant stated they counted dots in the diagram. However, this is directly evident in Mimic.

6. CONCLUSION & FUTURE WORK

As more HCI researchers begin to include crowdsourcing approaches in their work, new opportunities for broader and larger studies call for more advanced visual analytics. By reducing the barriers between interaction designers and the direct examination of many trials, we have shown how detailed instrumentation together with multiple levels of visualization and interactive review can reveal previously unknown micro-interaction design issues.

Mimic does not restrict interaction analysts to a single analytics strategy but allows them to explore several overview methods, a number of visualization types, and direct scripting for extensibility. From a design perspective, the review of individual trials can be a source of insight revealing problems but also suggesting opportunities.

By adopting ESDA principles in Mimic, we have demonstrated clear benefits of this approach to reveal a number of micro-interaction design problems in a replication case study of a conditional probability problem. The development of Mimic itself has been an exercise in micro-interaction design ensuring that subtle event sequences are visualized in a way that informs analysts in answering specific research questions.

The database and server-side infrastructure of Mimic also represent a contribution as hundreds of trials can generate large datasets. The 400 trials presented captured very detailed dataset with about 5 to 10 MB of data each –roughly the size of a photo from a digital camera. However, this created a 2.2GB dataset of event time-series best left on the cloud making remote analytics more practical than locally working with the data.

Future development efforts could make it easier to direct instrumentation data from surveys and web-based experiments hosted elsewhere to the Mimic server for analysis so the general HCI community can see how Mimic could be used in their work. By making Mimic an open source cloud-based project, we hope to foster the active design of micro-interactions for the benefit of end-users.

7. REFERENCES

[1] Andrica, S., Candea, G. WaRR: A tool for high-fidelity web application record and replay. *2011 IEEE/IFIP 41st International Conference on Dependable Systems & Networks (DSN)*, IEEE (2011), 403–410. [DOI]

[2] Brase, G.L. Pictorial representations in statistical reasoning. *Applied Cognitive Psychology 23*, 3 (2009), 369–381. [DOI]

[3] Burg, B., Bailey, R., Ko, A.J., and Ernst, M.D. Interactive record/replay for web application debugging. *UIST'13*, ACM Press (2013), 473–484. [DOI]

[4] Carta, T., Paternò, F., and Santana, V. de. Web usability probe: a tool for supporting remote usability evaluation of web sites. *Human-Computer Interaction* (2011) 349–357. [DOI]

[5] Cole, W.G. Understanding Bayesian reasoning via graphical displays. *ACM SIGCHI Bulletin 20*, SI (1989), 381–386. [DOI]

[6] Eddy, D. Probabilistic reasoning in clinical medicine: Problems and opportunities. *Judgment under uncertainty: Heuristics and biases*, (1982), 249–267.

[7] Fisher, C., Sanderson, P. Exploratory sequential data analysis: exploring continuous observational data. *interactions 3*, 2 (1996), 25–34. [DOI]

[8] Gigerenzer, G. and Hoffrage, U. How to improve Bayesian reasoning without instruction: Frequency formats. *Psychological Review 102*, 4 (1995), 684–704. [DOI]

[9] Gómez, O.S., Juristo, N., and Vegas, S. Replications types in experimental disciplines. *Symposium on Empirical Software Engineering and Measurement - ESEM'10*, ACM Press (2010), 1–10. [DOI]

[10] Gomez, S. and Laidlaw, D. Modeling task performance for a crowd of users from interaction histories. *CHI'12*, ACM Press (2012), 2465. [DOI]

[11] Grossman, T., Matejka, J., and Fitzmaurice, G. Chronicle: capture, exploration, and playback of document workflow histories. *UIST'10*, ACM Press (2010), 143–152. [DOI]

[12] Heer, J. and Bostock, M. Crowdsourcing graphical perception. *CHI'10*, ACM Press (2010), 203–212. [DOI]

[13] Heer, J., Mackinlay, J., Stolte, C., and Agrawala, M. Graphical histories for visualization: supporting analysis, communication, and evaluation. *IEEE transactions on visualization and computer graphics 14*, 6, 1189–96. [DOI]

[14] Hilbert, D.M., Redmiles, D.F. Extracting usability information from user interface events. *ACM Computing Surveys 32*, 4 (2000), 384–421. [DOI]

[15] Hornbæk, K., Frøkjær, E. Reading patterns and usability in visualizations of electronic documents. *ACM TOCHI 10*, 2 (2003), 119–149. [DOI]

[16] Huang, J., White, R.W., and Dumais, S. No clicks, no problem. *CHI'11*, ACM Press (2011), 1225–1234. [DOI]

[17] Komarov, S., Reinecke, K., and Gajos, K.Z. Crowdsourcing performance evaluations of user interfaces. *CHI'13*, (2013), 207. [DOI]

[18] Matejka, J., Grossman, T., Fitzmaurice, G. Patina: dynamic heatmaps for visualizing application usage. *CHI'13*, ACM (2013), 3227–3236. [DOI]

[19] Micallef, L., Dragicevic, P., and Fekete, J.-D. Assessing the Effect of Visualizations on Bayesian Reasoning through Crowdsourcing. *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (2012), 2536–2545. [DOI]

[20] Mickens, J., Elson, J., and Howell, J. Mugshot: Deterministic Capture and Replay for JavaScript Applications. *NSDI*, (2010).

[21] Mickens, J. Rivet: browser-agnostic remote debugging for web applications. *Proc. USENIX ATC*, (2012).

[22] Nakamura, T. and Igarashi, T. An application-independent system for visualizing user operation history. *UIST'08*, (2008), 23. [DOI]

[23] Navalpakkam, V. and Churchill, E.F. Mouse Tracking : Measuring and Predicting Users' Experience of Web-based Content. *CHI'12*, (2012), 2963–2972. [DOI]

[24] Nebeling, M., Speicher, M., and Norrie, M. CrowdStudy: General Toolkit for Crowdsourced Evaluation of Web Interfaces. *EICS'13*, (2013), 255–264.

[25] Ottley, A., Metevier, B., Han, P., and Chang, R. *Visually Communicating Bayesian Statistics to Laypersons*. Tufts University, (2012), TR-2012-02.

[26] Robinson, A.C. and Weaver, C. Re-Visualization : Interactive Visualization of the Process of Visual Analysis. *Workshop on Visualization, Analytics & Spatial Decision Support,* GIScience. (2006).

[27] Rzeszotarski, J. and Kittur, A. CrowdScape. *UIST'12*, ACM Press (2012), 55. [DOI]

[28] Saffer, D. *Microinteractions: Designing with Details*. O'Reilly, Sebastopol, CA, 2013.

[29] Sanderson, P. and Fisher, C. Exploratory Sequential Data Analysis: Foundations. *Human-Computer Interaction 9*, 3 (1994), 251–317.

[30] Sanderson, P., Scott, J., and Johnston, T. MacSHAPA and the enterprise of exploratory sequential data analysis (ESDA). *International Journal of Human Computer Studies 41*, 5 (1994), 633–681. [DOI]

[31] Sloman, S. a., Over, D., Slovak, L., and Stibel, J.M. Frequency illusions and other fallacies. *Organizational Behavior and Human Decision Processes 91*, 2 (2003), 296–309. [DOI]

[32] Stieger, S. and Reips, U.-D. What are participants doing while filling in an online questionnaire: A paradata collection tool and an empirical study. *Computers in Human Behavior 26*, 6 (2010), 1488–1495. [DOI]

[33] Wassner, C., Martignon, L., and Biehler, R. Bayesianisches Denken in der Schule. *Unterrichtswissenschaft 32*, 1 (2004), 58–96.

[34] Waterson, S.J., Hong, J.I., Sohn, T., Landay, J.A., Heer, J., and Matthews, T. What did they do? understanding clickstreams with the WebQuilt visualization system. *AVI'02*, ACM Press (2002), 94. [DOI]