A case study on implementing screen reader accessibility in dynamic visualizations

Rita Costa, Beatriz Malveiro, João Palmeiro, and Pedro Bizarro

Feedzai

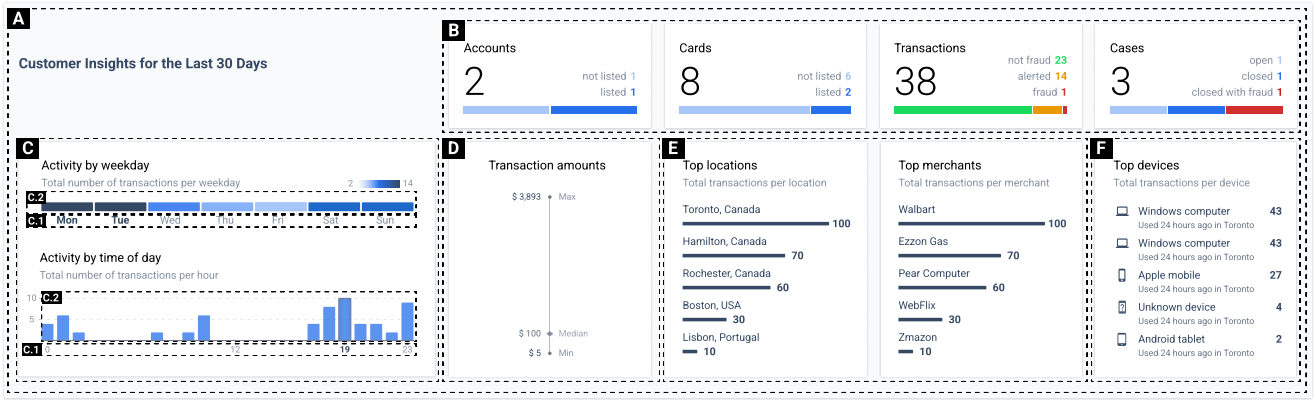

Figure 1: *Set of visualizations summarizing key customer metrics used by fraud analysts during the alert review process. The data is synthetic.*

Abstract

Millions of people worldwide work in jobs where assessing dynamic data presented visually to them is a key part of their tasks. Since the data is only represented in a visual format, these occupations are out of reach for visually impaired people, making them unable to review hundreds of information-heavy cases per day and determine outcomes for each one in just a couple of minutes. In this work, we aim to shrink that gap by detailing the implementation of screen reader accessibility features to real-world visualizations used by fraud detection analysts. We propose a set of features that should be validated with users and, if proved to be useful, transformed into guidelines for creating these types of accessible charts.

CCS Concepts

- ***Human-centered computing → Visualization; Accessibility***

1. Introduction

Mapping tabular data to visual formats brings out patterns that would otherwise be difficult to find or even missed. F. J. Anscombe [Ans73] has demonstrated that without charts, our ability to interpret data becomes limited [MF17] and time-consuming. Therefore, a wide range of tasks — particularly those that need to be completed in a tight time window — benefit significantly from presenting data in charts. If we rely solely on the visual representation of the data, how do we communicate it to people with visual impairments?

In the data visualization community, the topic of accessibility has gathered interest in recent years. For example, Frank Elavsky published Chartability [cha], "a set of heuristics (testable questions) for ensuring that data visualizations, systems, and interfaces are accessible". Sharif et al. [SCWR21] studied how charts are perceived on the web to conclude that screen reader users take 211% more time interacting with a chart on a web page than non-screen reader users and are 61% less accurate. Nevertheless, current research appears to be limited to the passive consumption of charts (e.g., those featured in news articles). It does not yet cover the visualizations that

DOI: 10.2312/evp.20221134

are part of tools that assist decision-making and are, by definition, dynamic and used daily.

In this work, we take existing guidelines regarding web accessibility [wca] and the recent research on this topic for data visualization and apply them to a real-world set of charts built specifically to assist fraud analysts in assessing customer behavior. The result is a case study describing this implementation and a set of features whose usefulness we suggest testing with visually impaired users to create specific guidelines for accessibility for dynamic visualizations.

2. Case study

Being close to the fraud analysis task, we used it as a reference for this case study. Nevertheless, the problem is not limited to this type of analysis. We believe that the conclusions extend to other analytical tasks where people are faced with a repetitive job (going through similarly structured information many times a day, every day), where decisions must be quick (sometimes in seconds) and accurate (making the wrong decision has a cost).

Below, we detail the accessibility features implemented with ARIA (Accessible Rich Internet Applications) [SG22] attributes and specific CSS properties to the set of charts built with visx [SWWA21] that are displayed in **Figure 1**. These plots are inserted at the top of a page with other details regarding the current customer in analysis. They show the first data the user interacts with before diving deeper into further customer details.

[A] Overall organization of the visualizations: "Customer insights for the last 30 days" is the first information that the screen reader announces. It clearly states what the analyst is about to encounter, giving them room to explore it further or move on. Additionally, a hidden paragraph that is only available to the screen reader software is included describing the main trends in the data. The description is dynamic and adapts to the customer data. In the example of **Figure 1**, the screen reader would read: "Customer with two accounts, eight cards, 38 transactions, and three cases. More active on Monday and Tuesday and at 7 pm. Median amount per transaction is 100 dollars. Top location is Toronto, Canada; top merchant is Walbart; top device is a Windows computer."

All the visualizations have headings of varying levels that provide an adequate organization of the different elements on the page to the screen reader technology. The main title has an `<h1>` HTML tag, all other card titles have an `<h2>` tag, and inside the customer behavior cards each chart has an `<h3>` header. In macOS' VoiceOver, for example, the user can go to the rotor menu and immediately be aware of all the titles on the page and navigate to whatever they find most interesting.

[B] Count cards: Figure 1 comprises four similar card-like elements that provide information about the number of accounts, cards, transactions, and cases related to a specific customer. Since these cards have a title visible to the screen reader that announces the element that is being counted, for example, "Transaction statistics", we swap the visible "transactions" for "total". This way, the screen reader will announce the following: "Transactions statistics; total: 38; alerted: 14; fraud: one". The same applies to the other similar cards.

[C] Customer behavior: These two charts have hidden paragraphs only visible to the screen reader that describe the main trends in the data. For example, for the second plot in the **[C]** section this description will read: "The days with most activity are Monday and Tuesday with 14 transactions. Minimum was two, on Friday." When the screen reader finds the chart itself, it will describe the type of visualization the user is interacting with. For example, "an heatmap showing the distribution of transactions over the days of the week". During this interaction, the user can navigate to another plot or skip a certain description via the keyboard shortcuts.

[C.1] Axes: In the axes, we hide labels and ticks from the screen reader using the `aria-hidden` attribute and describe the axis range. In the second plot, the x-axis `aria-label` is "Axis ranging from 12 am to 11 pm"; in the y-axis is "Axis ranging from zero to ten transactions".

[C.2] Chart elements: The details regarding the value of each bar are available after the axes are announced and the user can skip them and go to other chart elements as they wish. In the example of the second plot, the `aria-label` will read the hour in the day and the count for that particular time. With the `aria-roledescription`, we describe the shapes as chart elements.

[D] Transaction amounts: There is no visible gain in forcing the user to go through each element of this minimalist boxplot. Therefore, we completely hide it from the screen reader with an `aria-hidden` attribute and use text to describe what is shown visually. It works because it is a small amount of data.

[E] Top merchants and locations: In this case, the approach is similar to the customer behavior charts **[C]**. We also generate an invisible paragraph stating which is the most common location or merchant and their count. The user can explore it further to know the type of chart, the range of the x-axis, and the count for each particular element.

[F] Devices: In this segment, data on each device is turned into full sentences, so the information is more naturally read by the screen reader. In the first device, it will read: "A Windows computer was used 43 times. Last seen 2 hours ago in Toronto."

3. Conclusions and Future Work

The implementation that is described above was designed taking into consideration general web accessibility recommendations and the limited [KJRK21] literature on the topic. From those references and from our knowledge of these types of tasks, we compiled a group of features that should be part of these systems: **1)** hierarchized; **2)** skippable; **3)** navigable; **4)** natural; and **5)** detailed on request. We did not have the opportunity to test our work with visually impaired users because, in the field of fraud detection, we were unable to find any such users. This is precisely the problem we aim to address by enabling people to perform complex tasks and have access to these jobs.

In the future, we aim to test these charts with actual screen reader users and be able to validate if the features we identified in this implementation can be transformed into actual guidelines.

References

[Ans73] ANSCOMBE F. J.: Graphs in statistical analysis. doi:https://doi.org/10.1080/00031305.1973.10478966. 1

[cha] Chartability. URL: https://chartability.fizz.studio/ [cited 23.02.2022]. 1

[KJRK21] KIM N. W., JOYNER S. C., RIEGELHUTH A., KIM Y.-S.: Accessible visualization: Design space, opportunities, and challenges. *Computer Graphics Forum 40* (2021). doi:https://doi.org/10.1111/cgf.14298. 2

[MF17] MATEJKA J., FITZMAURICE G. W.: Same stats, different graphs: Generating datasets with varied appearance and identical statistics through simulated annealing. *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (2017). doi:https://doi.org/10.1145/3025453.3025912. 1

[SCWR21] SHARIF A., CHINTALAPATI S. S., WOBBROCK J. O., REINECKE K.: *Understanding Screen-Reader Users' Experiences with Online Data Visualizations*. Association for Computing Machinery, New York, NY, USA, 2021. doi:https://doi.org/10.1145/3441852.3471202. 1

[SG22] SIEGMAN T., GARRISH M.: *Digital Publishing WAI-ARIA Module 1.1*. W3C working draft, W3C, Feb. 2022. URL: https://www.w3.org/TR/2022/WD-dpub-aria-1.1-20220215/. 2

[SWWA21] SHOFF H., WILLIAMS C., WONGSUPHASAWAT K., AIRBNB: visx, 11 2021. URL: https://github.com/airbnb/visx. 2

[wca] WCAG 2.1. URL: https://www.w3.org/TR/WCAG21/ [cited 23.02.2022]. 2