# Big Data Interactive Visualization

Dissertation in partial fulfillment of the requirements for the degree of

MASTER OF COMPUTER SCIENCE

UPPSALA UNIVERSITET

DEPARTMENT OF COMPUTER SCIENCE

AMMAR ALDHAHYANI

Supervisor & Reviewer
Raazesh Sainudiin

[10 - 06 - 2022]

# Contents

# 1 Abstract

Nowadays, data is getting bigger and bigger and visualizing it statically requires many displays to show all different perspectives of the data. Moreover, every and each visualization would need the data to be preprocessed to provide the right data for that case. This paper brakes down and explains the needed strategies and tools to interactively visualize big data for different scenarios. It starts by explaining the data visualization process and value. Then it demonstrates the dilemma associated with visualizing big data in a static form. Later on, it introduces interactive data visualizations and elucidates how the interactive form of visualization can overcome the big data visualization issues by introducing the interactive visualization strategies and tools that can be used for different visualization types. Finally, an interactive visualization tool [2] was built using D3 with Reactjs which adapt multiple interactive visualization strategies to demonstrate how it enables the user to interact with the data, reveal information, and obtain some insights for a specific use case involving some Twitter data around the BlackLivesMatter and AllLivesMatter movement.

# 2 Introduction

Data visualization is the process of representing data in a graphical or pictorial form which makes the information easy to understand. Human beings have always employed visualizations to make messages or information last in time. What cannot be touched, smelled, or tasted can be represented visually [50]. Although there are multiple goals and objectives for visualizing data the main goal is to make something complex appear simple, especially for the untrained eye. Turning data into knowledge would mean different things to different people and would require a different process for different cases, but in all such cases, visualization offers an indispensable tool. Data visualization is useful for data cleaning, exploring data structure, identifying trends and clusters, detecting outliers and unusual groups, spotting local patterns, evaluating modeling output, and presenting results.

Graphics are a good way to raise questions and suggest ideas that help to stimulate research. In fact, graphics reveal data features that statistics and models may miss. They reveal unusual distributions of data, local patterns, clusterings, gaps, missing values, evidence of rounding or heaping, implicit boundaries, outliers, and so on. New, innovative graphics need instruction and experience to interpret them. Their designers have spent much time developing them and reasonably enough believe that what is obvious to them should be obvious to everyone. To visualize data many questions need to be answered. What are the reasons behind drawing such displays, what is the context, where did the data come from, and how much data is needed?

## 2.1 Interactive Visualization and Big Data

One picture is worth a thousand words. In today's world where data is being recorded from every click on the web to personal records, petabytes of data are being generated and processed every day. It is not enough to process and analyze those data since human brains are incapable of seeing all patterns without the data being visually represented. As big data visualization plays an important role in decision making in various fields, it's, however, challenging to visualize such a huge amount of data in real-time or in static form. Static visualizations are only capable of providing views of the data while a huge data set requires many static views to present a variety of perspectives on the same information. Due to the increased quantity and limited number of pixels in display devices, it is often impossible to visualize all the raw data.

Even if assuming a sufficient number of pixels are available, showing all the data in any single view might not be beneficial as visual perception may be hindered by visual crowding [59]. One way to overcome such a problem is an approach often called the keyhole problem which is to show the full details of a small number of items in the visualization [49]. For example, by showing a few rows at a time of a huge spreadsheet and giving the user the ability to scroll through the spreadsheet. The downside of such an approach is that the user inevitably loses context even if it could help manage a large amount of data.

An enduring design strategy that was introduced by Shneiderman is "overview first, zoom and filter, then details on demand" [53]. This strategy came to avoid the keyhole problem and it works by beginning the analysis with a broad overview of the entire data set. Obviously, some details would be sacrificed by following this approach however interaction techniques are coupled with visualization to allow the user to zoom in on specific information and filter out irrelevant items. More details on this strategy and its approaches are in the next Section.

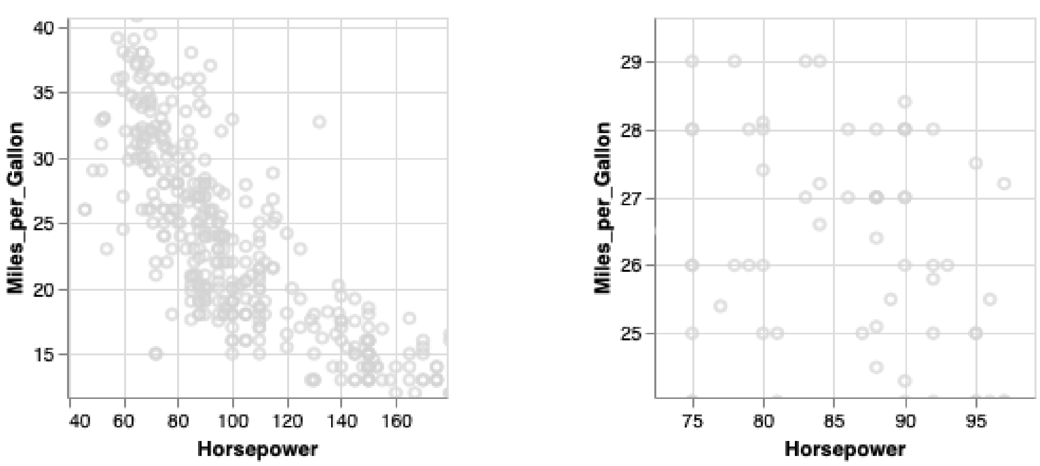
# 3 Interactive Visualization Strategies

## 3.1 Navigation Strategies

Interactive data visualization systems have a basic requirement which is to enable navigation of large information. Navigation techniques enable the user to move facilely between different levels of detail in the data. Zoom & pan, overview & detail, and focus & context are the three main approaches that enable navigation during data visualization. These techniques are applied instead of the detail-only approach that was introduced by the Keyhole problem to overcome the big data visualization complexities which may disorient the user due to the absence of an overview of the large information space [49]. Attempts to compare the three approaches are inconclusive and subject to specific design scenarios, data, and user tasks, however, experimental results demonstrate the superior performance of these navigation strategies over the detail-only strategy [28].

### 3.1.1 Zoom & Pan

Zoom & pan operations when used in data visualization allow the user first to see the overview and then to interactively zoom into the data and pan the viewpoint within the data space to access details of interest. Zooming could be implemented using continuous space navigation provided by the Pad11 system [10] for example or to systematically access different scales using the TreeMap system [30]. The ability to zoom in to details of interest from an overview and zoom out to see the overview of the whole again is the main advantage of this approach. Maps are an example of the use of this approach but however efficiently this approach uses screen space and offers infinite scalability, users may become disoriented and lost when zooming in and panning around the data space. Moreover, since the overview is not shown easily this approach can result in slower navigation because of the intensive use of memory if not used correctly [49].




(a) Before the Zoom (b) After the Zoom

Figure 1: Zoom & pan [60]

### 3.1.2 Overview & Detail

The overview & detail strategy enables multiple views to display both an overview and a detail view simultaneously. Unlike the zoom & pan strategy, the overview & detail strategy preserves the context of the whole data set while allowing the user to examine detailed information about a particular part of interest. This strategy is often used in both map and image viewing systems [43] and different applications combined with zoom & pan strategy like in maps. One issue with this strategy is that even though it enables multiple views, usually the detailed view consumes the display area, especially on small screens, and result in the overview not being fully clear.

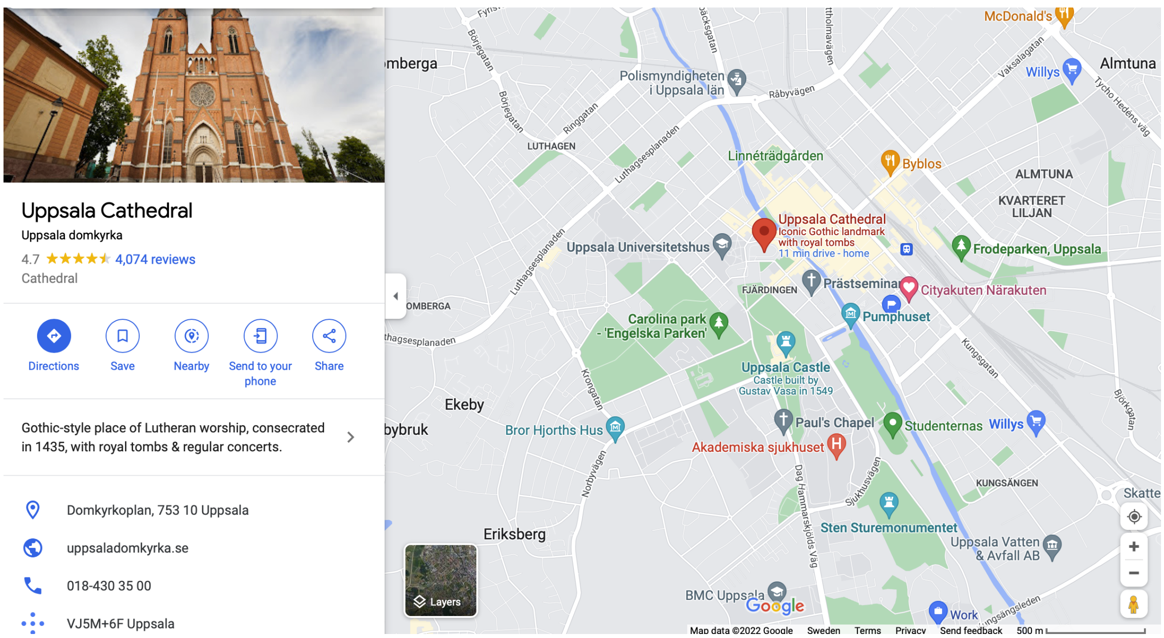
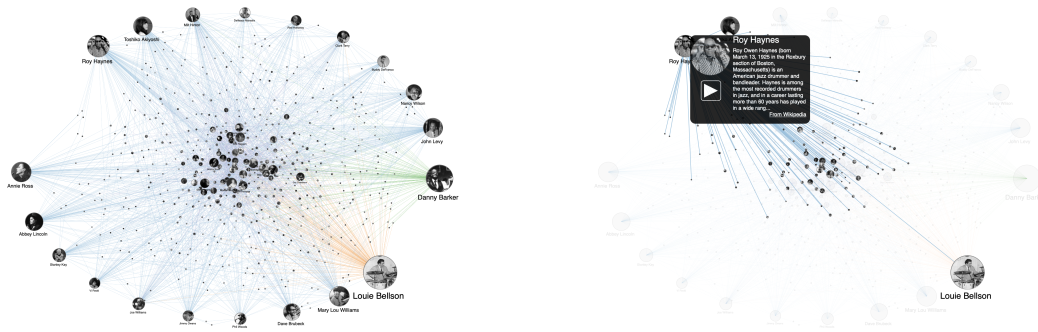



Figure 2: Overview & Detail [26]

### 3.1.3 Focus & Context

Instead of creating a separate view for the intended region of the overview, the focus & context strategy allows the focus region to grow and differentiate itself from the rest of the overview area. The focus region can show additional information by getting expanded and magnified while the overview gets partially compressed to allow for such expansion in the focus area. The bifocal display is one of the variations of the focus & context strategy which uses two levels of magnification and the concept of the bifocal display is used in TableLens [47] and the dock of application icons in desktop operating systems. This strategy provides continuity of detail within the context of the overview however, this strategy applies some distortion to the overview to put more focus on the intended part which might cause some disorientation to the users [7]. Moreover, this technique has limited scalability typically under a 10:1 zoom factor [49].




(a) The Overview Visualisation (b) Focus Region

Figure 3: Focus & Context [41]

## 3.2 Visual Interaction Strategies

While most navigation strategies have issues with scalability, visual interaction strategies support scalability using user-centered techniques. These techniques allow the user to access alternative perspectives and insights since for even modest data volumes it is impossible to show all the data. There are many interaction techniques for data visualizations [57] and the main four categories that should be considered in the design of visualization systems are presented below.

### 3.2.1 Selecting

For a huge data set that can be divided into multiple groups or subsets, The capability to interactively select items of interest in visualization is fundamental. Grouping similar items into a set, selecting items, adding items to a selection, removing selected items, and completely clearing a selection are achievable actions using selection techniques. A user can select items using direct or indirect actions. Direct selection can be implemented in many different ways, such as pointing at the item using the mouse and dragging it to join a group of items [58]. On the other hand, indirect selection criteria are based on a set of constraints that user specifies, such as, the ranges of different values. For example, selecting graph nodes with a user-defined distance from another node [57]. Selection is like stroking visual objects with an artist's brush and that is why it is usually referred to as brushing.

### 3.2.2 Linking

When selection is made in one view, linking is used to dynamically relate information between multiple views [39, 56]. Selecting (brushing) and linking is the most common view coordination strategy and it is usually called linked brushing [8]. With this strategy, selected items in one view are also highlighted in other views which enable users to uncover relationships and construct comprehensive understandings of the data set. Moreover, linked brushing allows users to define complex constraints by optimizing each view to specify constraints on certain data types and degrees of accuracy [57]. For example, by specifying temporal constraints with a timeline visualization and geographic constraints with a map. Linking strategy comes with a diverse range of options for connection between different linked views such as unlinking one view of the data to explore other regions or specifying what type of information is communicated.

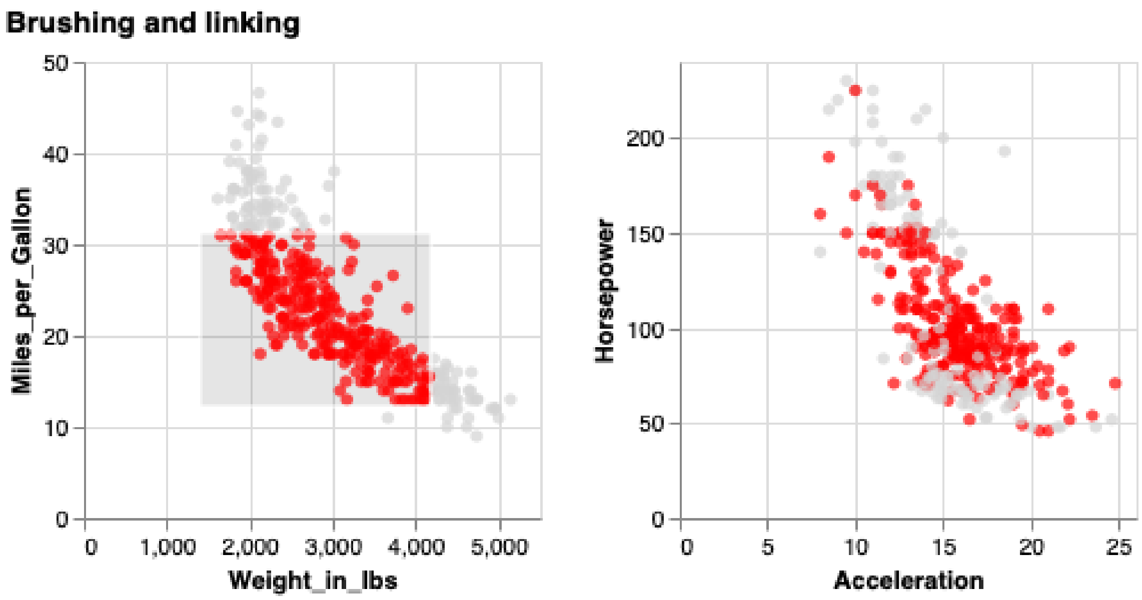



Figure 4: Selecting (brushing) & Linking [14]

#### 3.2.3 Filtering

One of the main strategies to deal with a huge data set is by using interactive filtering operations which reduce the quantity of data visualized and help focus on interesting features. Dynamic query filters provide quick feedback, reduce the quantity of information, and allow exploration of the relationships between attributes. Visual widgets are used to assign a range of interests and view the filtered results in the visualization. There is a diffusion between selecting and filtering however there is a thin but important distinction between them. Filtering is usually achieved through a separate interface and via indirect action. Also, filtering can be executed before viewing a large data set to avoid overwhelming the system. On the other hand, selecting is usually achieved via direct action, such as mouse clicks. Even though selecting and filtering strategies have different mechanisms, the outcome of both of them on the view can be indistinguishable [57].

#### 3.2.4 Rearranging and Remapping

It is important that users have the ability to customize the visual mapping form rather than having a single configuration that might become inadequate. Since the spatial layout is the most important visual mapping, rearranging the spatial layout of the information is very effective for revealing new insight. This simple but important operation gives the user the flexibility to explore relationships between different attributes and finds what best suits their needs.

# 4 Visualization Tools

Visualization tools are supposed to supplement the domain expertise and deliver a big image so that users can frame critical questions and later propose heuristic and insightful answers to these questions [32]. Traditional tools like Excel can not or at least struggle to perform visual analytics on big data involving millions of records so new visualization tools or enhancements in existing tools need to handle the complexity of big data [38]. Dozens, if not hundreds of tools are available to create a visualization of large data sets. Most of them are basic and have a lot of overlapping tools. One way to categorize tools is whether they are drag and drop types or they require coding for creating a visualization. Most of the proprietary visualization tools are of the drag and drop type, which requires no coding skills. Although it is easy to learn and create visualizations in minutes the drawback is that out-of-box visualizations are not possible [38]. In this section, the focus is on the standouts that either have more capability for the types of visualizations they can create and the amount of freedom in customization they can offer or are significantly easier to use than the other options out there.

## 4.1 Tableau

Tableau is a leading data visualization tool used for data analysis and business intelligence. It is capable of delivering interactive visualizations in no time with its drag and drop nature. Gartner's Magic Quadrant classified Tableau as a leader for analytics and business intelligence [40]. Tableau is a user-friendly tool and was built for a diverse number of teams. It is an easy-to-use tool that requires no programming skills and it provides results in a wide variety of formats. It can be connected to the data stored in excel, CSV, and text files and can recognize fields and formats. Moreover, it provides integration with all the major advanced databases, including Teradata, SAP, MySQL, Amazon AWS, and Hadoop. Below are some of the main advantages and disadvantages of using Tableau to visualize big data. To meet the need of diverse users, Tableau software provides selections that include Tableau Desktop, Tableau Server, and Tableau Mobile to choose from. As Tableau Desktop is preferred by individuals and small organizations, Tableau Server is more convenient for big organizations with many users.

On the other hand, it is a very expensive product to scale across huge organizations. Moreover, it provides very basic prepossessing so in most cases data needs to be exported in perfect tables. Finally, no out-of-the-box visualizations can be provided and users have to stick to the offered options.

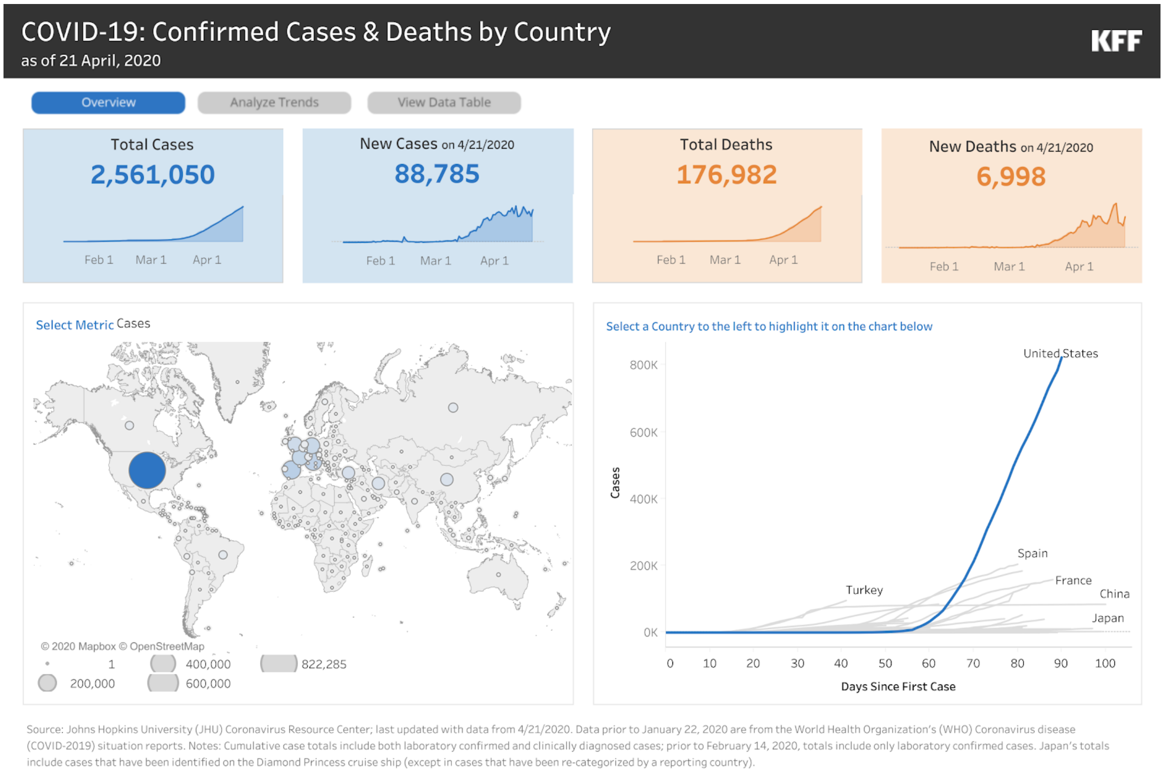



Figure 5: Tableau dashboard example from (JHU) Coronavirus resource center [31]

## 4.2 D3.js

D3 stands for data-driven-documents [11] and is a free open source tool that makes any visualization possible and in fact, many visualization libraries where JavaScript is the coding language are built on top of D3. D3 outperforms all other JavaScript-based tools as it offers versatile functionalities like data manipulation and transformation and makes effective use of the power of new browser and web technologies [38]. It can create very powerful and highly interactive visualizations. As it requires coding skills in JavaScript, CSS, and HTML any basic visualization needs to be coded from scratch. However, D3 has a big community and many built-in reusable functions, and samples of commonly used graphics are available. D3 can create any imaginable visualization and offers excellent interactivity within coding limits.

Not supporting older browsers and the programming experience needed to even create simple visualizations are some of the disadvantages that would hold users from using D3. Usually, easier tools with fewer lines of codes would be used unless customization and performance are priorities, especially with big data.

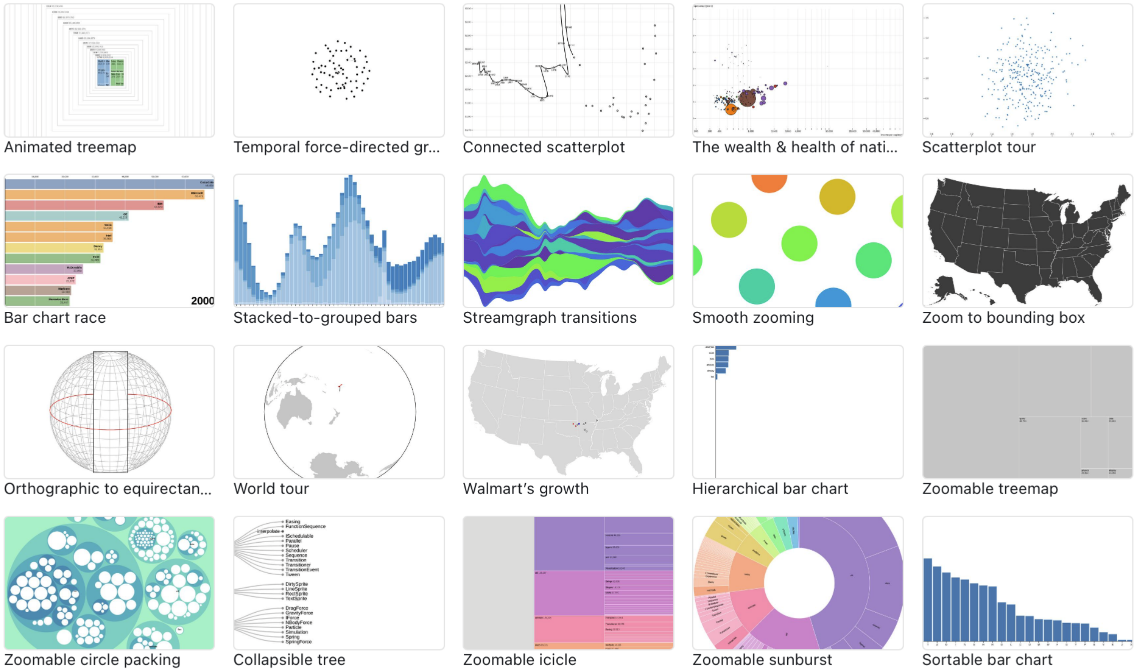


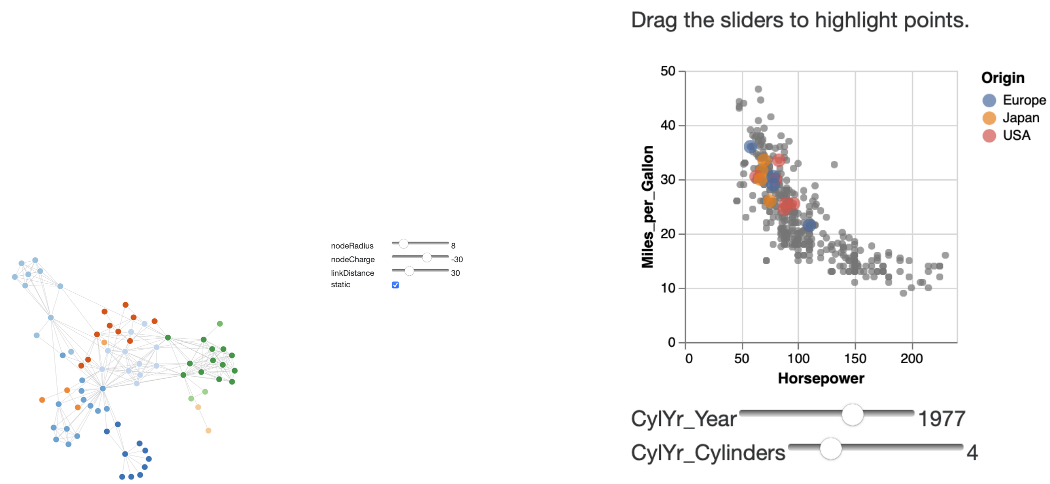

Figure 6: D3 Gallery [12]

| Tableau | D3.js |
|---|---|
| Business Intelligence tool; visualization software package | Java Script Library, not a visualization software package. |
| Proprietary, expensive | Free and Open Source |
| Drag and Drop; Learning very easy | Heavy coding required; learning very difficult |
| Development time of Dashboard is in minutes | Development time is from hours to days |
| Variety of built-in charts and maps to select from, but out-of-box visualizations are not possible | Any imaginable visualization(code-able) is possible, but every chart has to be built from scratch |
| Visualization format is proprietary, but allows export to JPEG, PNG,BMF and EMF formats | Output format is Scalable Vector Graphics(SVG) |
| By applying user filter or row-level security feature, restricted data access can be provided to different users | Concealing data to provide restricted access among different users is difficult to achieve |
| Able to identify dimensions and measures and can easily handle gigabytes of data | Struggle in handling large dataset in gigabytes. |

Figure 7: Tableau vs D3 [38]

## 4.3 Vega & Vega-Lite

Vega is a visualization tool based on D3 [52]. Vega is a declarative language for creating interactive visualizations. The user can describe the visual appearance and the interaction behavior of the visualization in a JSON format. Vega processes the JSON object and produces the visualization on the browser. On the other hand, Vega-Lite is a high-level visualization tool based on Vega [51]. Compared to Vega, Vega-Lite is easier and requires fewer lines of codes however it comes with less customization.




(a) Vega [21]

(b) Vega-Lite [55]

Figure 8: Vega & Vega-Lite Visualizations

## 4.4 Google Chart

Google Chart has a rich gallery and it works across all browsers. The most common way to use Google Chart is with simple JavaScript code that can be embedded into the web page. Users can load Google chart libraries, list all the data to be visualized, and select options to customize the charts. It is an easy-to-use library but mostly used for simple visualizations.

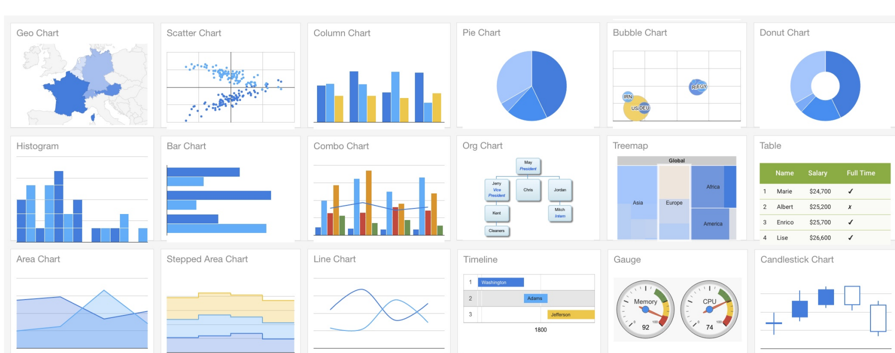



Figure 9: Google Chart Gallery [25]

## 4.5 Datawrapper

Datawrapper is a user-friendly, open-source web tool that can be used to create basic interactive charts. By loading the dataset into Datawrapper, it can be embedded onto a website. It is mostly used to generate pie charts, line charts, bar charts (horizontal and vertical), and maps.

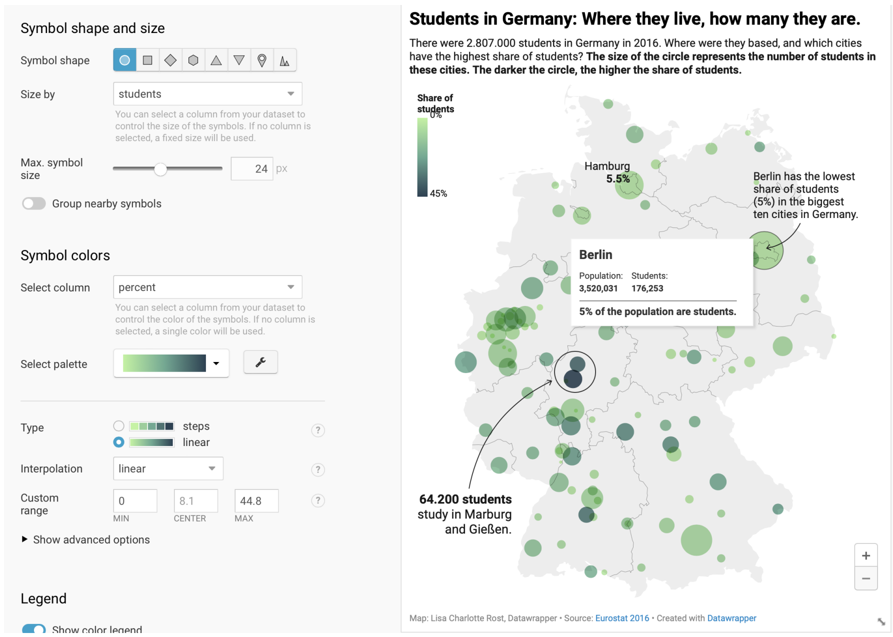



Figure 10: Datawrapper interface [17]

## 4.6 Canvas

Canvas.js is a JavaScript library that can offer 30 different types of interactive charts. It is fast and can run on various devices.

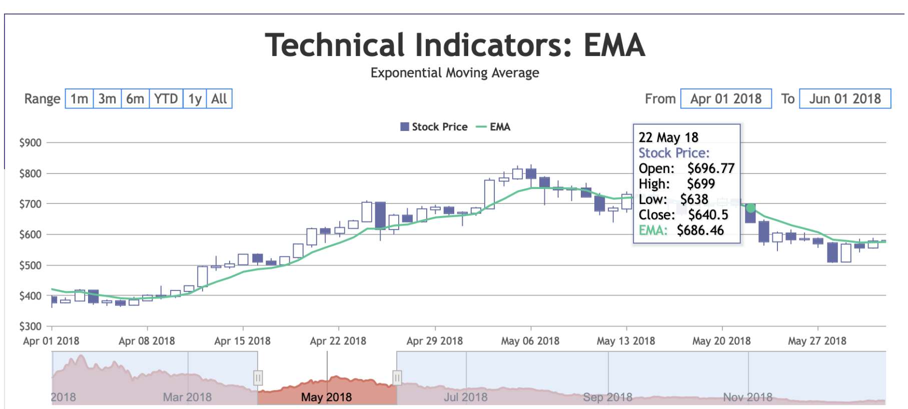



Figure 11: Canvas.js [16]

## 4.7 Other Tools

A few other tools are summarized below:

- **Qlikview:** Qlik is one of the major players in the data analytics space with their Qlikview tool which is also one of the biggest competitors of Tableau [46].
- **Microsoft Power BI:** It is another leading data visualization tool like Tableau and Qlikview. It is used for data analysis and business intelligence. It is a drag and drops tool that is easy to use and it allows the user to connect to multiple data sources [45].
- **Infogram:** It is a web-based data tool that requires no coding skills. Users include newsrooms, marketing teams, governments, and students [29].
- **Plotly:** It is an open-source tool that provides a list of charts with interactive nature [44].
- **Sigmajs:** It is a modern JavaScript library specialized in rendering and interacting with network graphs. It is easy to use and can draw larger graphs faster than D3 but with fewer customization [54].

# 5 Understanding Data via Interactive Visualisation

## 5.1 Network Data

Networks are often the most convenient way to represent interactions among entities in social, biological, infrastructure and other information systems [9, 27]. Examples of network data include interactions among people in social networks, connectivity of web pages in world wide web graphs, and protein-protein interaction (PPI) in biological networks [33, 5]. Network analysis and visualization (also called Network Graph) help discover structures and patterns in a network and thereby reveal useful insights [27, 4, 6, 5, 3].
Network data contains complex relationships between a huge amount of elements. A network visualization displays directed and undirected graph structures which illuminates relationships between entities. Elements or entities are usually displayed as round nodes and lines show the relationships between them. Network visualization is essential in multiple fields to simplify complex systems. For example, it is used in cyber security to better understand cyber threats, reveal network vulnerabilities, detect malware and discover trends. Moreover, in infrastructure management to create interactive visualization that reveal bottlenecks and vulnerabilities in connected critical infrastructure. And many other applications in social networks and biological networks.
Some simple network data can be visualized simply using a static network visualization like the network visualization representing the metro stations in Figure 12.

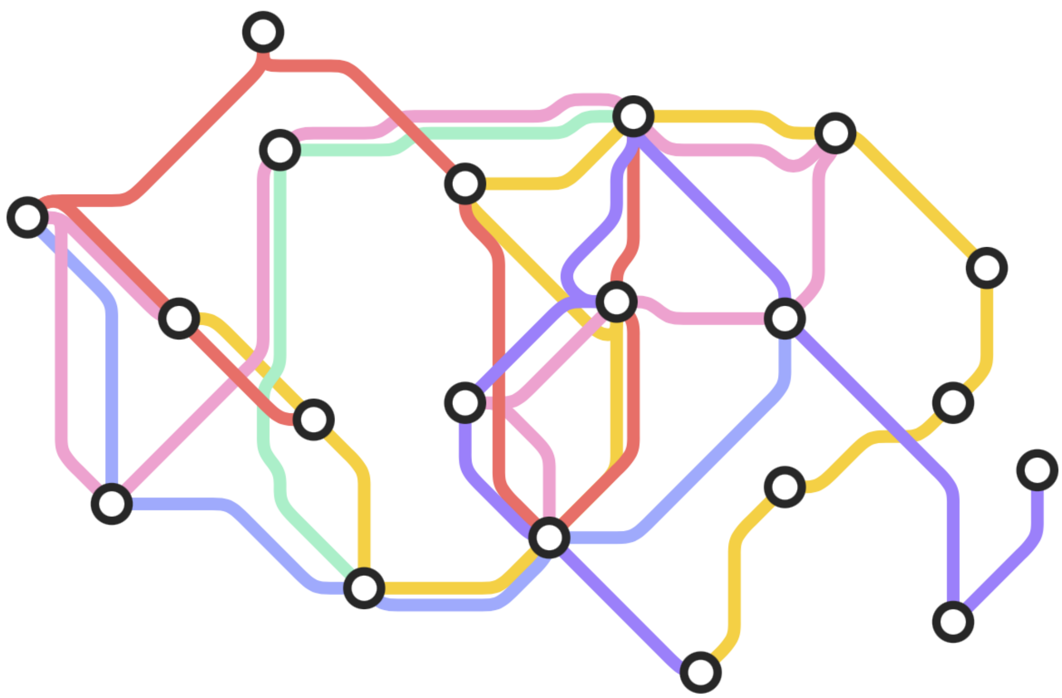
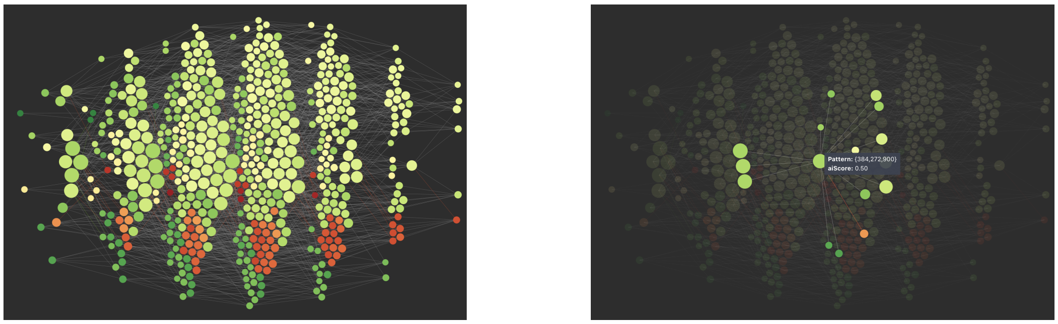


Figure 12: Metro stations as network diagrams [35]

However most network data are big and complex containing multiple layers and many interactions which make it, if displayed naively, unreadable.



(a) The whole network graph (b) Focus on just one pattern

Figure 13: Network of pattern in medical compound dataset [22]

## 5.2 Causal Inference

Not statistical but causal questions motivate most quantitative studies in the health, social and behavioral sciences [42]. Causality is always a tricky subject and it is often mixed with correlation. A correlation is when two variables tend to change together and it does not necessarily indicate causation. For example Figure 14 from the BBC news, shows that the number of crimes in London rises with temperature. This can easily mislead the viewers to conclude that warmer temperature causes violent crimes [36].

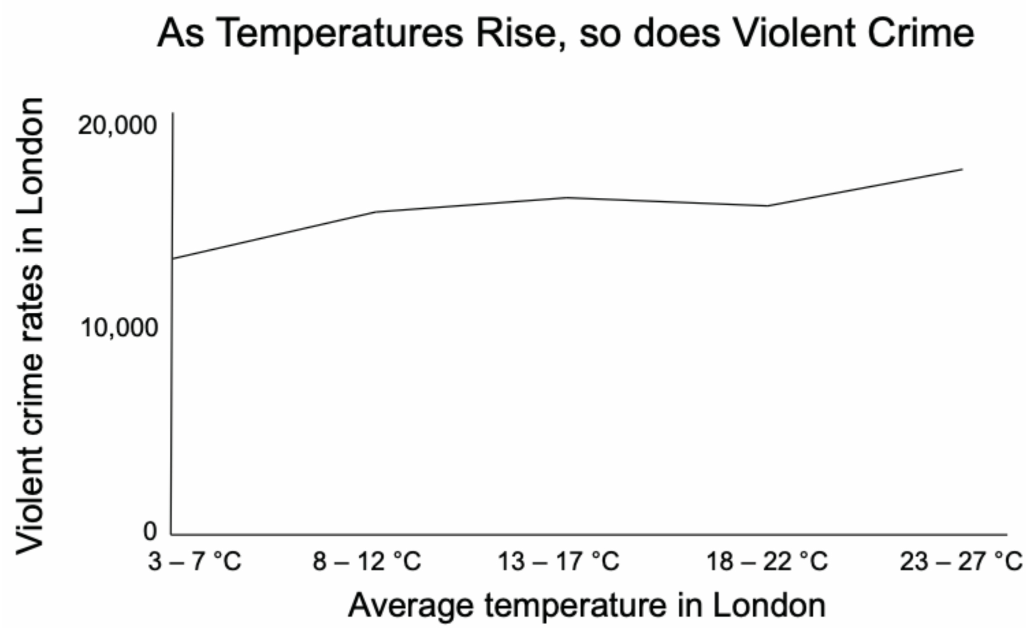



Figure 14: Recreation of BBC news article figure, "Heatwave: Is there more crime in hot weather?"[36]

It is difficult to distinguish causation from correlation [48]. Only a correlation was established between temperature and crime rate in this case because it is likely that there are other factors not shown in this static graph that could influence the number of violent crimes in both cases. Even journalists and researchers can amplify causal implications from such results which in many cases mislead the audience.
As causality is hard to be identified so a static visualization is confusing since it can only show just one to a few factors while there might be many other factors to analyze to reveal hidden insights.
Interactive visualization could help by providing filtering or multiple strategies with all factors presented to help look at all the possible variables together or by customizing the view and look at them in many different ways to facilitate the process and strengthen the judgments.

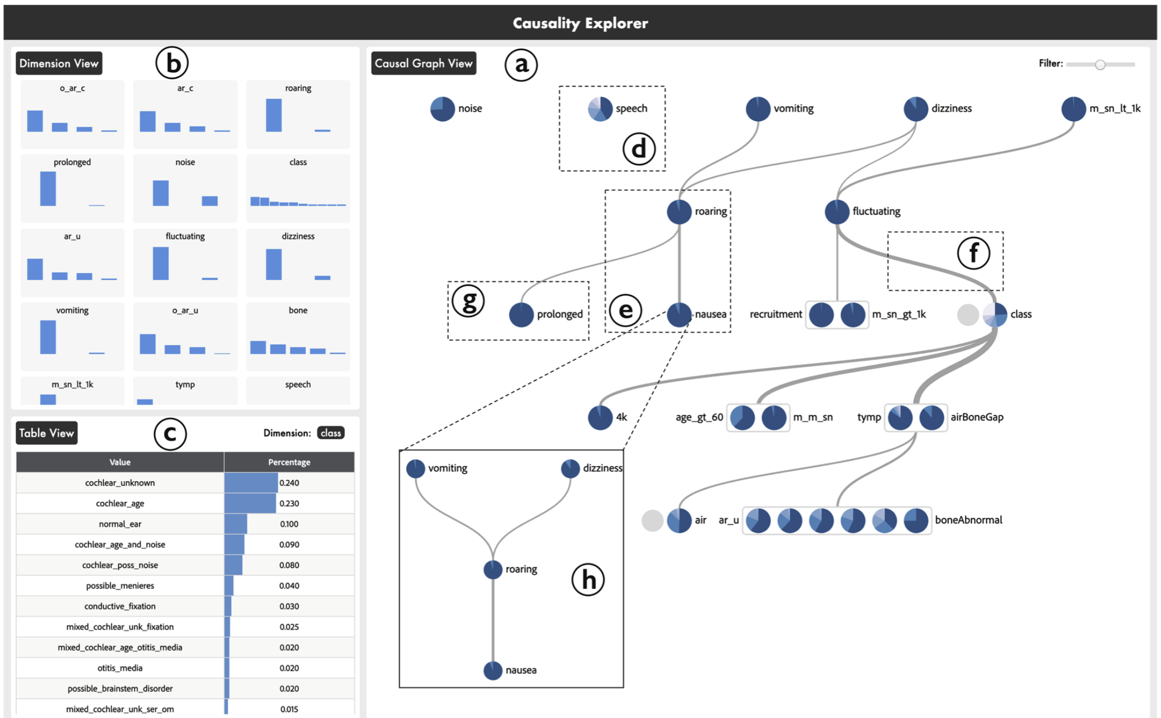



Figure 15: The interactive user interface of Causality Explorer[20]. (a) is the interactive scalable causal graph and (b) & (c) are views for comparative analysis to support what-if analyses.

## 5.3 Time Series Data

The line graph is the simplest and the most common way to represent time series data. It does help the viewer to observe what has changed over time. Statistically visualizing one or few variables in a small to medium dataset is very useful and has many applications and it is one of the most common visualization type in media. The problem rises when there is a big dataset and tens to hundreds variables to be displayed over a time-period. It is not possible to visualize a hundred attributes over a time-period in just one statistic visualization nor having a hundred separate visualization for each attribute which will be overwhelming and impossible to conclude any useful insights or correlations. For example, as shown in Figure 16, it is confusing and overwhelming to look at the data and come out with useful insights.

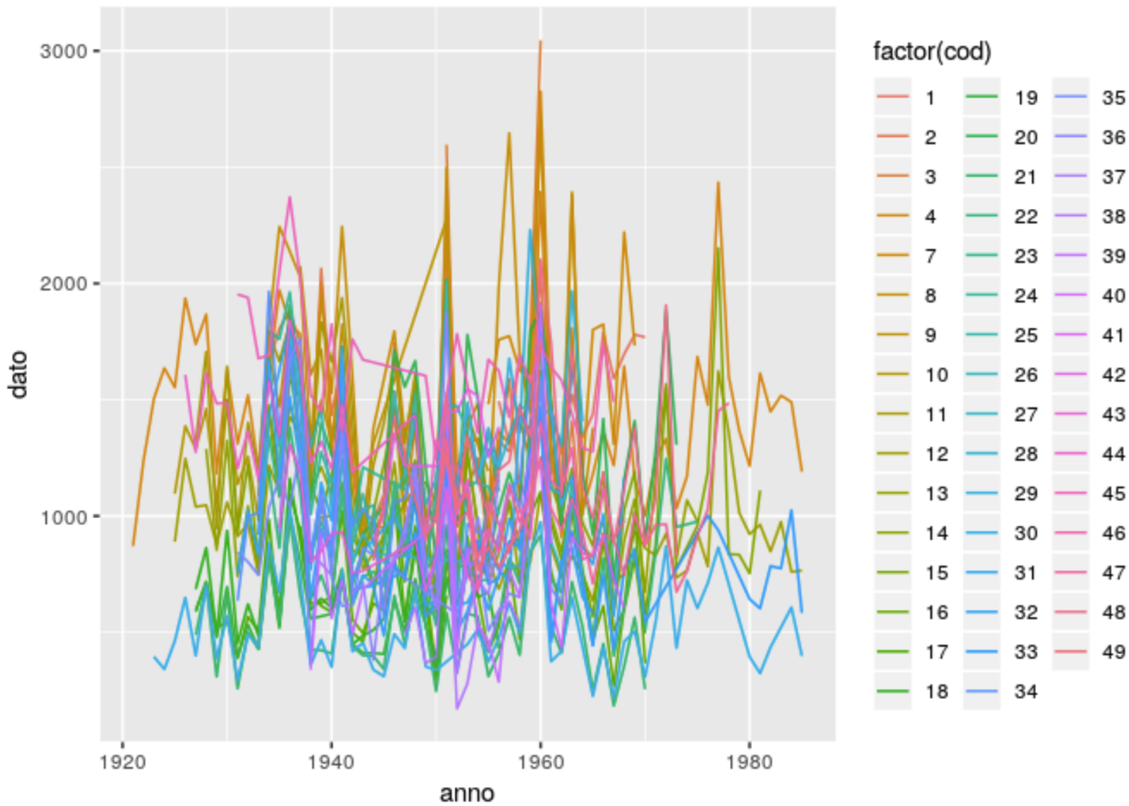
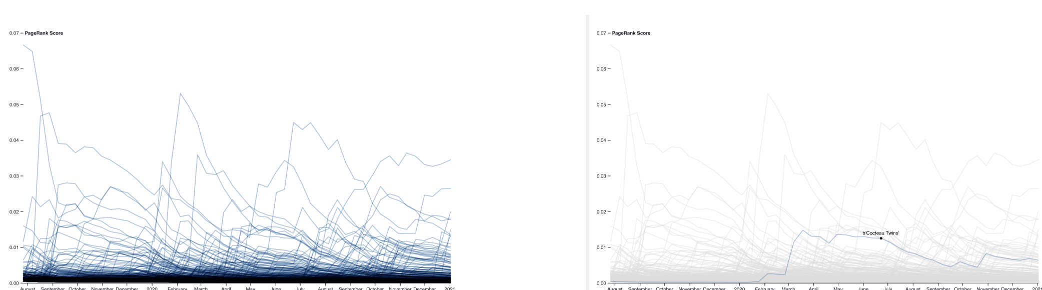



Figure 16: Example of a loaded time series data Visualization [34]

Moreover, a static time-series visualization over a long period can eliminate the details and only show the full picture. Interactive visualization offers the customization and the freedom to explore and customize the visualization to the user's needs. It gives the user the ability to have many different static visualizations in just one display. In Figure 17, Only the focus strategy was introduced to the graph and it made a huge difference by almost eliminating all other attributes.



(a) The whole time series data graph

(b) Focus on just one attribute

Figure 17: Interactive time series data Visualization with focus strategy [15]

In another example like the one shown in Figure 18, both filter and focus strategies were used which gives the user all the possible ways to view the time series data. In other cases, the zoom strategy could help in a long time series graph to zoom in to a short period and examine the data.

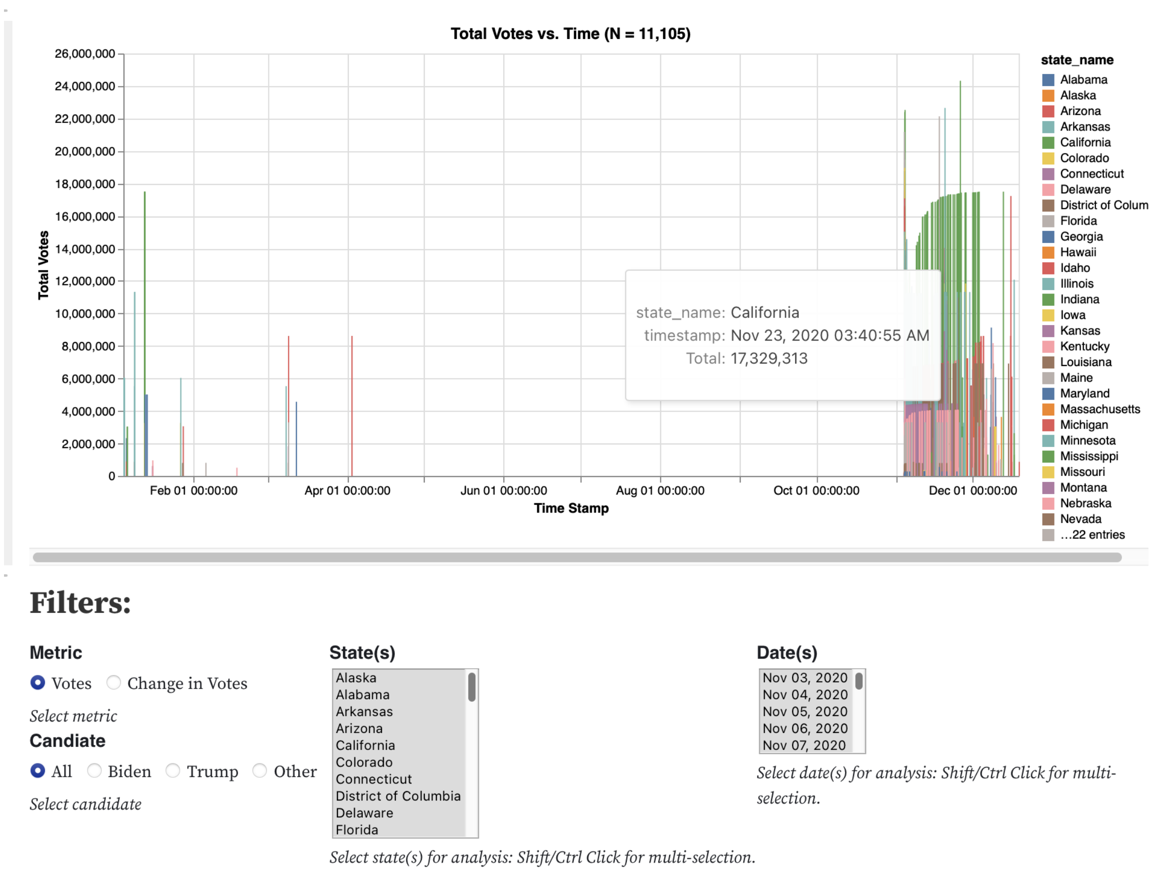



Figure 18: Interactive time series data visualization with focus & filter strategies [18]

# 6 Interactive Visualization [2] : A BLM Use Case

## 6.1 Overview

On May 25 of 2020, George Floyd, a 46-year-old African American man was arrested and treated harshly by three police officers leaving Mr. Floyd unable to breathe which lead to his death [19]. That incident re-initiated a movement known by the name “BlackLivesMatter” which initially became nationally recognized for street demonstrations following the 2014 deaths of two African Americans [23]. BlackLivesMatter movement was also trending in Twitter which is one of the main social media platforms where people discuss political and social matters and millions of people were tweeting about that topic by including hashtags related to the movement and the situation at that moment.

Twitter is a free social networking tool that is widely used and it allows users to share opinions and information by posting brief comments in a real-time news feed [37, 13]. Comments posted by users are called tweets and they are limited to no more than 140 character and it can include links to a website, video, and all other material online. Once a tweet is sent, it appears in the the Twitter’s feed of anyone who is following the user. A user can follow other users in Twitter and that will allow their tweets to appear in the user’s feed. Moreover, the user can like, reply to, and retweet another user’s tweet. Retweeting a tweet means that a user can share somebody else’s tweet that has appeared in the user’s feed. When a user retweet a tweet it goes to all the user’s followers however it shows others that was not originally the user’s tweet. Hashtags are keywords identified by an initial # symbol and they help to locate particular areas of discussion [13]. Hashtags are used within tweets to categorise tweets and to attract more attention to them. Twitter also has other features such as direct message between users, blocking followers, and many others.

## 6.2 Dataset

The data set that was collected [24] has 41.8 million tweets from 10 million users. The collected tweets has one of the keywords regarding BlackLivesMatter movement and AllLivesMatter movement. The data set contains tweets from the beginning of the movement in 2013 to the last of June 2020.

In this thesis, only tweets posted after the death of George Floyed till the last of June 2020 were used. The number of original tweets from this period was 1628162 tweets and below Table 1 shows some of the most popular hashtags between these tweets and the number of occurrences of the hashtags.

Table 1: Hashtags.

| hashTag | hashtag Count |
|---|---|
| BlackLivesMatter | 1215006 |
| blacklivesmatter | 273839 |
| GeorgeFloyd | 67475 |
| BLM | 63920 |
| AllLivesMatter | 61339 |
| JusticeForGeorgeFloyd | 41952 |
| blm | 38580 |
| BlueLivesMatter | 33586 |
| BLACKLIVESMATTER | 22653 |
| Blacklivesmatter | 18392 |
| icantbreathe | 17236 |
| racism | 15992 |
| NoJusticeNoPeace | 14478 |
| georgefloyd | 13393 |

From the Table 1, it can be seen that the majority of the hashtags are aligned with the BlackLivesMatter movement but also there are other hashtags belong to other counter movements like AllLivesMatter and BlueLivesMatter which also emerged as a response to the BlackLivesMatter movement. To make it simpler, BlackLivesMatter movement will be used from now on for all the hashtags aligned with the movement and AllLivesMatter movement will be used to represents both ALLLivesMatter and BlueLivesMatter movements. Some facts about the tweets containing the hashtags:

- Each tweet could include one or more hashtags.
- A tweet that includes a specific hashtag does not necessarily mean the content of the tweet is indeed related to the subject. For example, the hashtag might be used merely for publicity.
- A tweet that includes a specific hashtag does not necessarily mean that the content of the tweet is positive towards the hashtag, it might be neutral or negative.

Each collected tweet in the database has multiple variables and the main ones are shown below in Table 2 :

Table 2: Tweet's variables

| | | | | |
|---|---|---|---|---|
| CurrentTweetDate | CurrentTwID | language | CTweetFavourites | CTweetRetweets |
| CPostUserId | userCreatedAtDate | OPostUserIdinRT | CPostUserName | CPostUserSN |
| favouritesCount | followersCount | friendsCount | isVerified | isGeoEnabled |
| CurrentTweet | hashTags | mediaType | TweetType | etc. |

## 6.3 Delimitations

- The interactive visualization tool was built to adapt the data introduced in this paper and it is not intended for production since this data is private and the app can not accept other data since it is not automated.
- The data presented in the paper was not claimed to represent all the tweets from the same period.
- The observations made when visualizing the data only tend to show the capability of the interactive visualization strategies that were implemented and how they can be useful when visualizing or analyzing the data.

## 6.4 Data Processing

Creating an interactive visualization for such a huge dataset would require the visualization to be implemented in the cloud or using a high-performance computer since not only the dataset is required to be processed and presented for the first time but also every time there would be any interaction by the user. For this use case, a sample of the dataset was used to create interactive visualization. The below procedures have to be done to the dataset to create a useful visualization with some insights:

- Filtering the top 100 most retweeted tweets that contain BlackLivesMatter or synonymous hashtags. A new column called "Community" is created and its value for each tweet is set to "BlackLivesMatter" if the tweet only contains hashtags of the BlackLivesMatter movement or set to "Mixed" if the tweet also contains hashtags of the AllLivesMatter movement.
- Applying the same method to obtain the top 100 tweets that contain AllLivesMatter or synonymous hashtags but this time the Community column's value for each tweet is set to "AllLivesMatter" if the tweet only contains hashtags of the AllLivesMatter movement or "Mixed" if the tweet also contains hashtags of the BlackLivesMatter movement.
- Binding both groups in decent orders in respect to the tweets' number of retweets but this time selecting only some of the table variables as shown below in Table 3 since not all of them are needed to create the visualization. Moreover, the reason not to directly filter the top 200 tweets is to avoid the domination of The BlackLivesMatter hashtag due to its popularity as shown in Table 1 and also to create multiple groups to be compared.

This process created three communities and assigned each tweet to one community depending on the hashtags included in the tweet. The community in this scenario only represents a group of tweets sharing similar hashtags which would be useful in visualization to observe if the tweets' context of a specific community aligns with the tweets' hashtags of the same community. The three communities created are BlackLivesMatter community, AllLivesMatter community, and Mixed community.

Table 3: New Tweet's Structure

| | | | | | |
|---|---|---|---|---|---|
| twDate | userSn | twId | userId | twRetweet | twFav |
| userN | lang | favouritesCount | followersCount | friendsCount | |
| isVerified | Community | isGeoEnabled | hashTags | CurrentTweet | |

## 6.5 Visualization tool

**D3**

Creating a visualization example that can be shared and accessed by anybody, needs a tool that can be displayed in the browser. D3 which stands for data-driven-documents [11] is one of the most powerful visualisation tools that is also a web visualization tool that can run in the back-end or the front-end. It is a JavaScript library that can run in a plain JavaScript script or combined with one of the front-end frameworks. Processing speed, customisation, and interface are the three main aspects of any visualization tool and D3 gets the best out of the three.

**React.js**

Reactjs is a front-end JavaScript library [1] used for building user interfaces based on UI components and it is created and maintained by Meta and a huge community of developers and companies. Reactjs enables development of complex web based applications and allows the data to change without subsequent page refreshes [1].

Combining D3 with Reactjs is tricky if they are not understood well since both of them need to control the DOM (The document object model) however if they were integrated properly they can create unlimited options and they can take advantage of each other.

## 6.6 Time Series Interactive Visualization [2]

### 6.6.1 Visualization's Overview

The example dataset provided here has multiple variables that can be used in the visualizations such as twDate, followersCount, twRetweet, Community, CurrentTweet, and others. Time series visualization can take advantage of having the dates to display the data over time but also the interactive part of the visualization can play an important role to display other variables that otherwise would be impossible to be displayed.

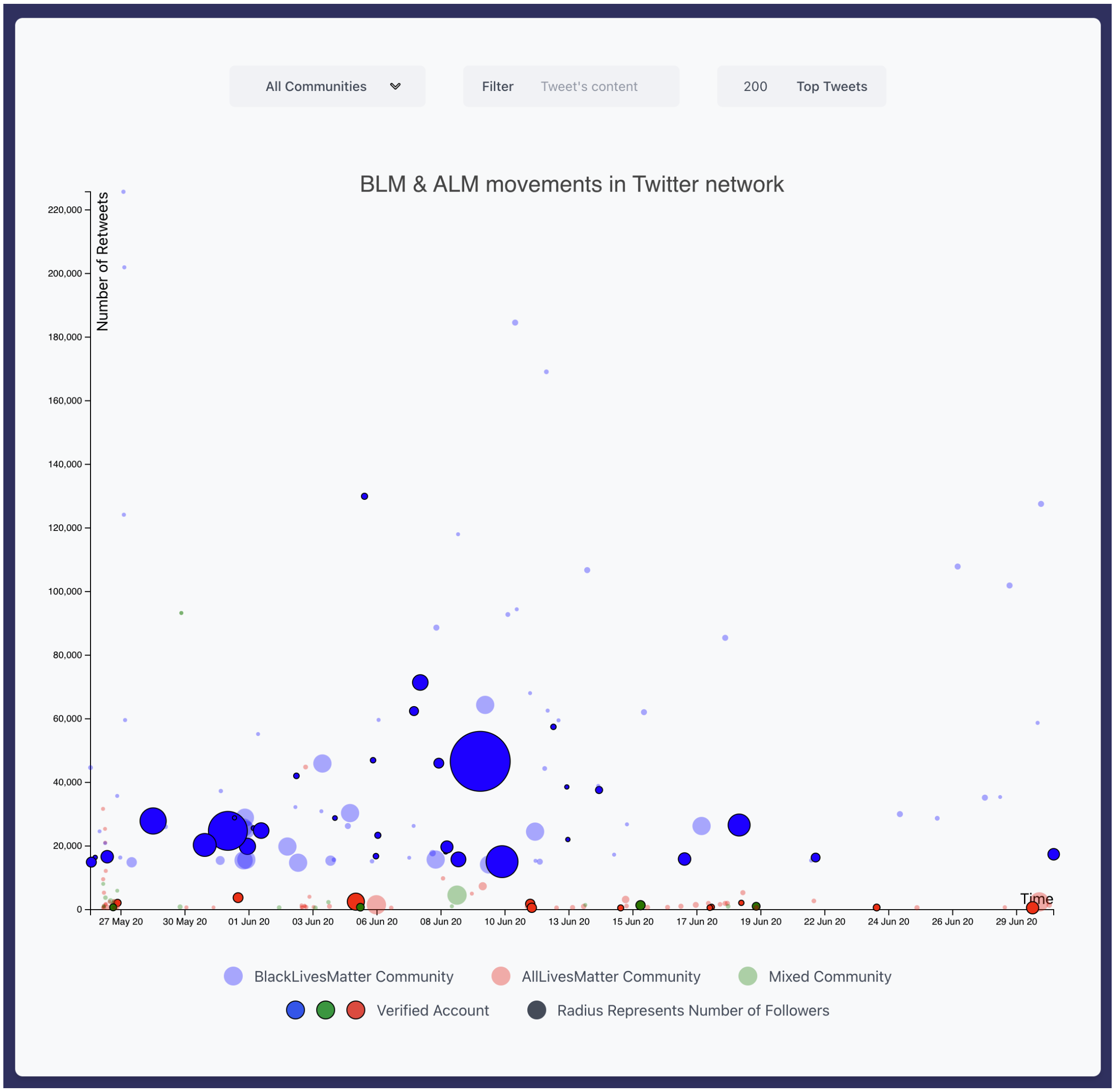



Figure 19: D3 interface for BLM & ALM movements in Twitter

Above in Figure 19 is the initial display of the top tweets of the BLM & ALM movements on Twitter. The display contains multiple components as the visualization itself or the inputs. Below is a detailed explanation of all the components.

- The x-axis represents the date of an original tweet.
- The y-axis represents the number of retweets of the original tweet.
- Blue circles represent tweets that only contain the BlackLivesMatter community hashtags.
- Red circles represent tweets that only contain the AllLivesMatter community hashtags.

- Green circles represent tweets that contain both communities' hashtags.
- Circles with darker colors and a border represent tweets belonging to verified users.
- The radius of the circles represents the number of followers the user has.
- The drop-down menu has multiple options starting with all communities or a specific community by itself.
- The filter input is a search field to type words and filter through tweets that contain them.
- The last input is an integer input which is limited between zero and the length of the whole dataset. It displays the top number of tweets in terms of the number of retweets. Initially, it starts with a number that represents the length of the whole dataset.

As a static visualization, it can reveal some good insights such as the distribution of communities and the popularity among them along with a time frame, however, events, contents, and people behind all of that are not visible. Moreover, In the case of multiple thousands of tweets, the nodes representing tweets would be overlapped and hard to read or differentiate.

### 6.6.2 Navigation Strategies

**Zoom Strategy**

The zoom strategy allows the user initially to see the overview and then zoom in to access details of interest. In this example, the continuous space navigation system is used which allows access to different scales, efficiently uses screen space, and offers infinite scalability [10, 30]. In Figure 20 below, it is the same visualization but after applying zoom functionality whether by using the mouse scroll or the trackpad gestures. After zooming in or out, a constant mouse click on the screen plus moving the mouse to any direction brings out the hidden nodes that disappeared after the application of zoom functionality. One more feature is how the y and x axes adapt to this functionality by changing their values to match only the data being displayed.

The initial view of the zoomed area displayed in Figure 20 was visually showing overlapped data and their features were not quite clear. With such data representing a movement that started at one moment and then followed along the way with critical events, the user would need to observe closely how those events impacted the movements. Zoom strategy in this case enables the user to dive in and have a closer look to determine the outcomes of the events.

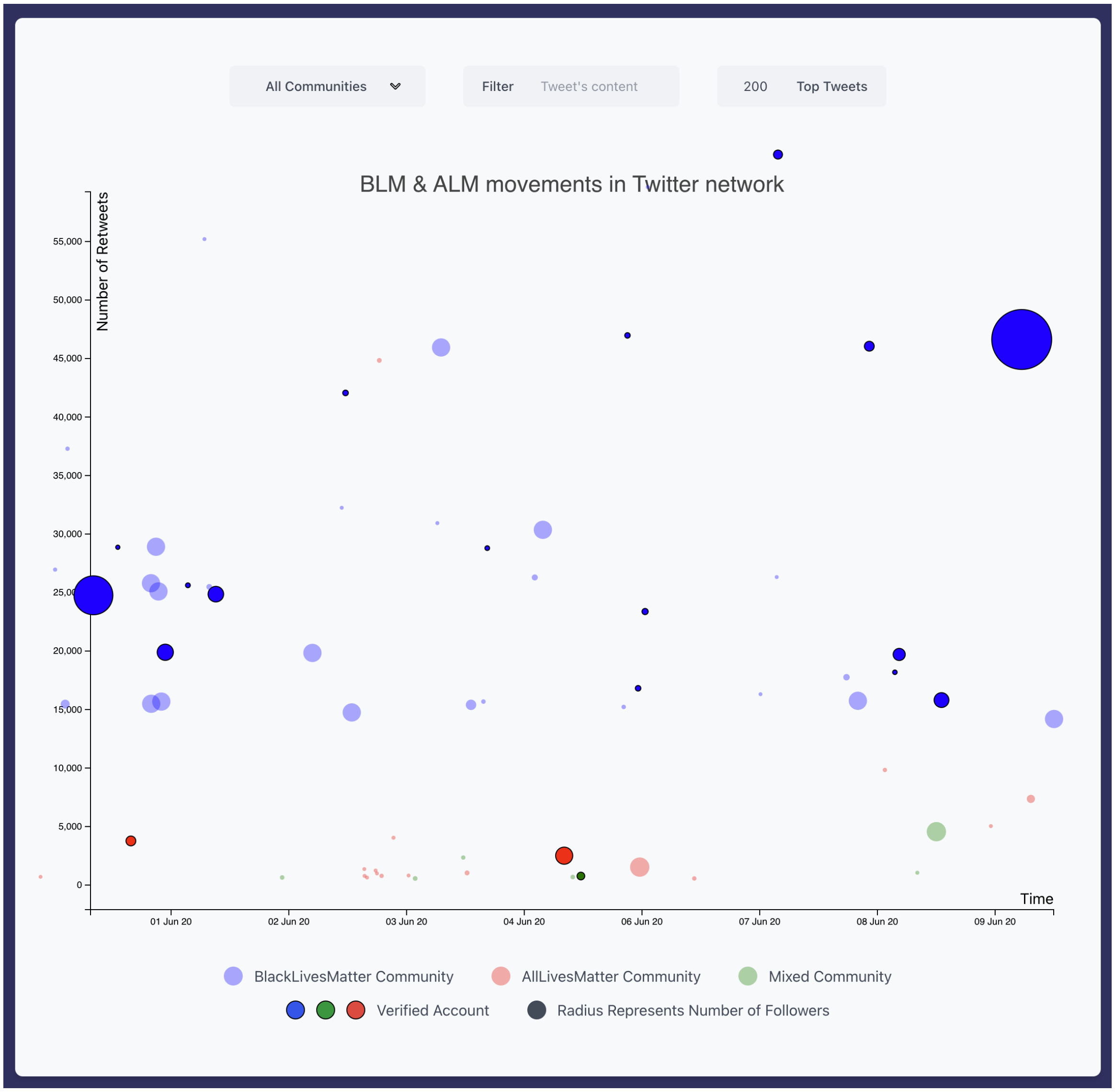



Figure 20: Applying zoom to the interface

**Focus, Context, Overview & Detail**

Looking at the data from the overview or while zooming in and out is helpful as it allows the user to see the distribution of different groups or communities over time and some of their features but to gain more insights, the user needs to see more details about each tweet and user to end up with a full picture of the situation. There needs to be a way to reveal much more detailed information without sacrificing the context of the whole data.

One of the functions which was implemented here uses focus, overview, and detail strategies to uncover more details about the data while preserving the context of the whole data. Figure 21 below, shows a popup window that appears horizontally in the middle of the display. The popup's work mechanism is when the user hovers over one of the data points displayed on the graph in the overview or any zoom mode, the data point distinguishes itself by increasing its radius and the mouse arrow turns to a hand shape to indicate that interactivity happened and is related to the focus strategy. Moreover, as the data point hovers over, a pop-up gets displayed that contains multiple types of information about the tweet and the user such as the number of retweets, the number of favorites, followers, the status of the account being verified or not, the community, the tweets' date and time, and the tweet's content.

This is a good example to highlight the power of combining D3 with React. D3 by itself can create a tool-tip which is a small popup window but the issue with that relating to this example is that D3's tool-tip can't be customized and it normally displays the data point name or a number when hovering over the data point. In the example presented here, there are multiple types of information to be displayed and they need to be styled and represented in a certain way to use the popup space efficiently which can not be done by D3 alone. Another point is that D3's tool-tip is designed to appear above the hovered data point which could create a significant problem here. In this example, the data points are distributed all over the display and some of them are located on the far left or right of the display. Considering the D3 tool-tip mechanism of displaying the tool-tip just above the data point, data points that are located near the borders when are hovered over will result in a big chunk of the popup window being hidden because it crosses the borderline of the visualization display that all D3's elements can only be seen within. On the other hand, React's ability to control the whole DOM –the application programming interface–, obtains the x-y-coordinates of each data point within the D3's display and within the whole interface screen, and thereby allows it to use such information to display the popup window in many different ways. Moreover, React's ability to use many styling libraries allows it to style the popup window in a user-friendly way to display different information. For this example, React uses the x-y-coordinates of the data point and fixes the x-axis position of the popup for all the data points to be in the middle of the screen to overcome the issue when there are data points located near the display's border. On the other hand, it sets the y-axis position of the popup window to be higher by some percentage of that data point's x-axis position so that when a data point hovers over, the popup window will always get displayed above the data point and that solves the issue of overlapping the data point and the popup window. For example, if the data point is located at the center of the display and its popup window's content is big, as the user hovers over the data point, the popup window will tend to show but as the popup window starts to appear and part of it would be over the data point then the mouse arrow will be over the popup window not anymore hovering over the data point which will result in not displaying the popup window. It is a complex procedure but quite powerful and limitless as described in a previous section compared to other tools like Vega and Vega-Lite which are easier libraries that are based on D3 itself but with limited options.

The result of the procedure as shown in Figure 21 is a piece of additional information about the data point without sacrificing the overview and the whole picture of the data.


Figure 21: Popup window with additional information

As the user is able to hover over the data points and reveal the tweet's information and some of the user's information but at some points, the user might need to see more about the users behind the tweets, their bios, and other tweets to gain more insights. Figure 22, shows a Twitter official interface located on the right side of the interface. This interface displays the user's Twitter page and the user can scroll through that user's tweets and activities. For the user to display this interface, instead of hovering over the data point, the user needs to click on the data point and that interface would be displayed on the side of the display revealing the user's Twitter page. Clicking again at the same data point will make that interface disappear. This is another overview & detail strategy implemented in this example to present more information without again sacrificing the overview or distracting the user. Moreover, when the Twitter interface is displayed, the user can click on the

Twitter interface of that Twitter user to open another tab on the browser where Twitter's full official website is presented and directed to that specific user behind the tweet however the user examining this visualization has to be signed in to their Twitter account to go through it. Such a visualization strategy uniforms to Twitter's user and developer rules.

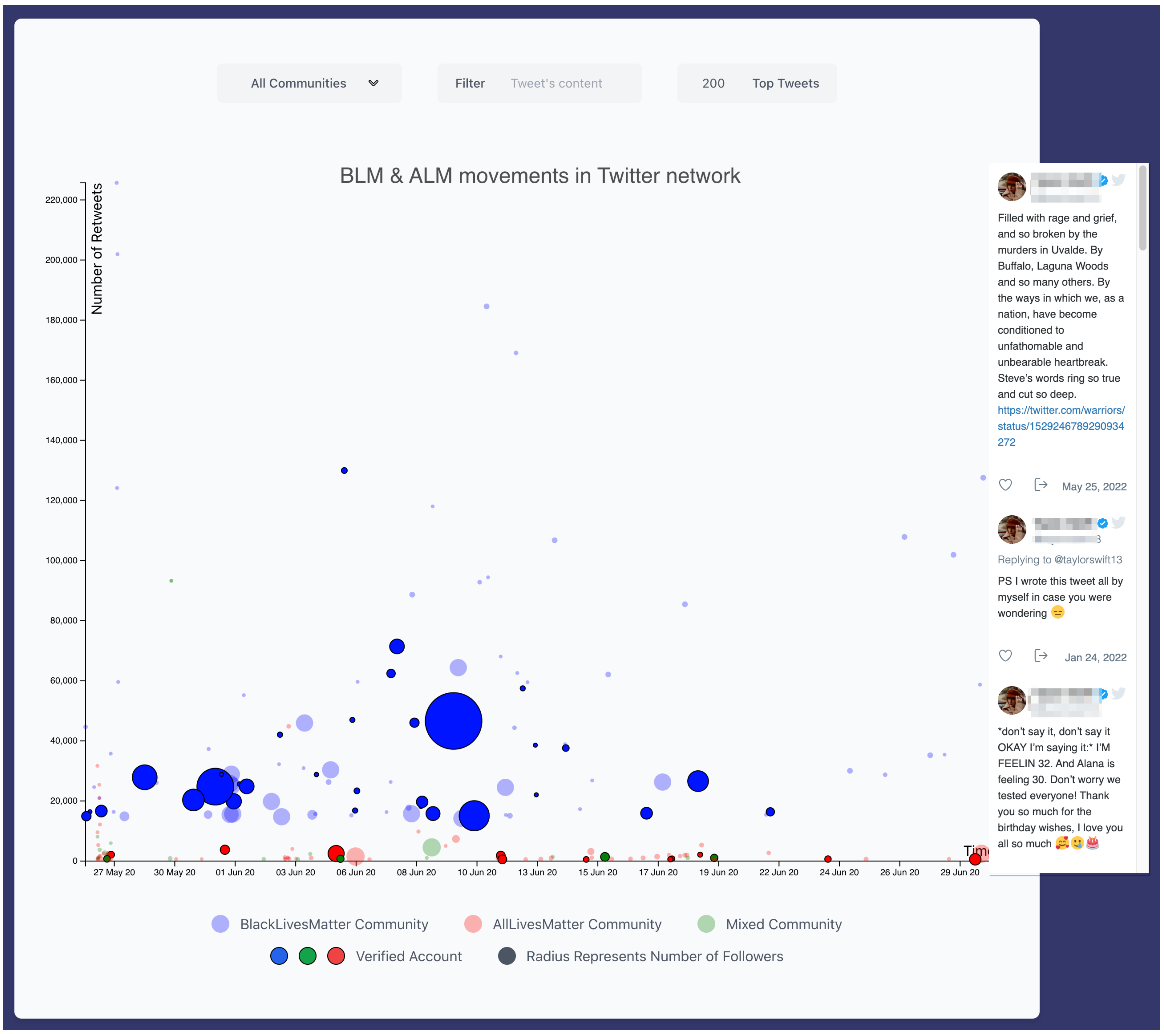



Figure 22: Twitter Interface

### 6.6.3 Visual Interaction Strategies

#### Selecting & Filtering Strategies

This dataset has 200 data points but in other cases, the dataset could have thousands of data points which in many cases, the user would need to filter or select only some data of interest. Enabling the user to interactively select or filter items of interest is fundamental in visualization. Figure 23 shows how the user can filter a certain number of top tweets in terms of the number of retweets. In this case, it is powerful to display the most effective tweets and look through their contents and users behind them and relate them to certain events that happened at that time.

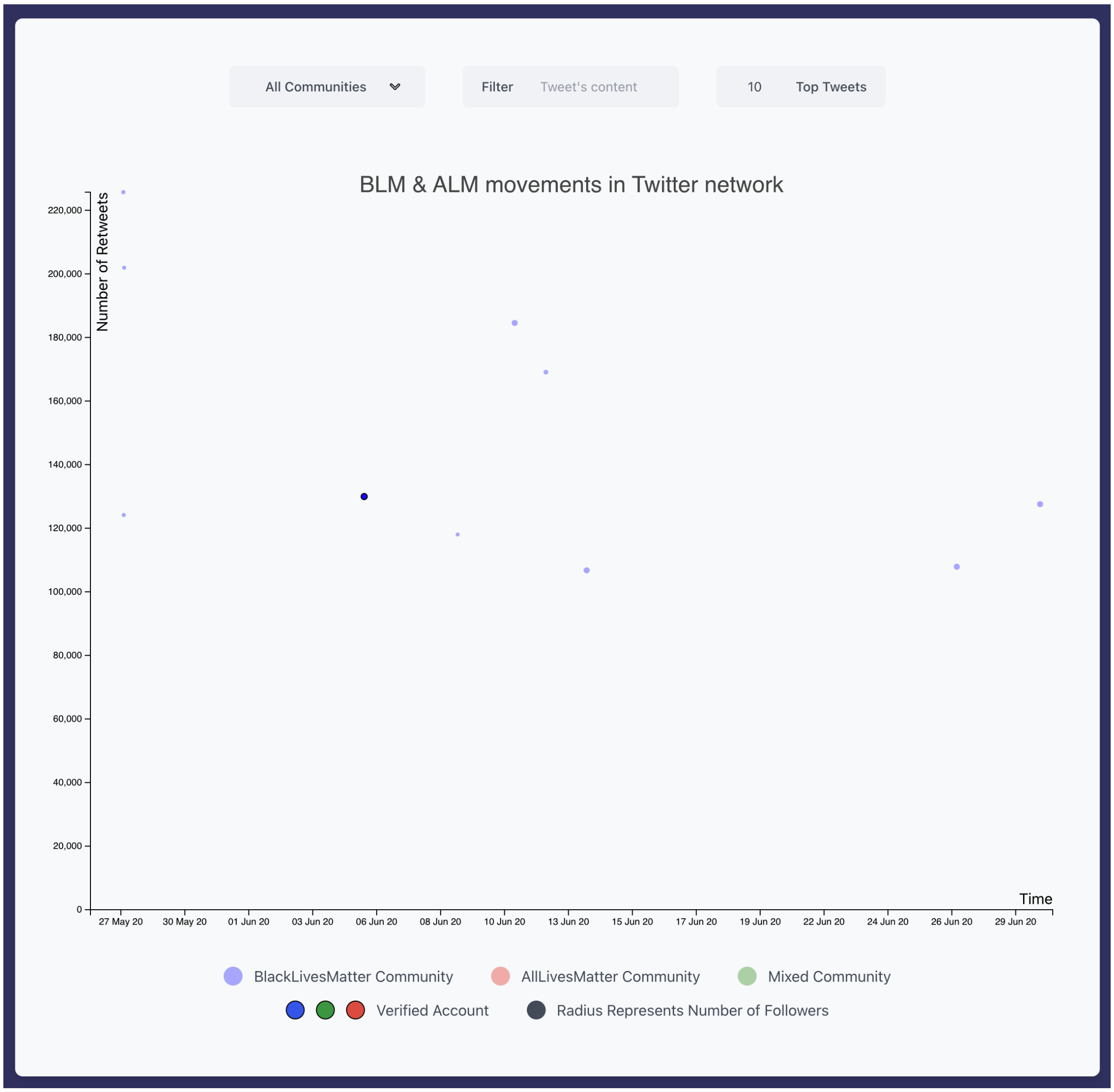
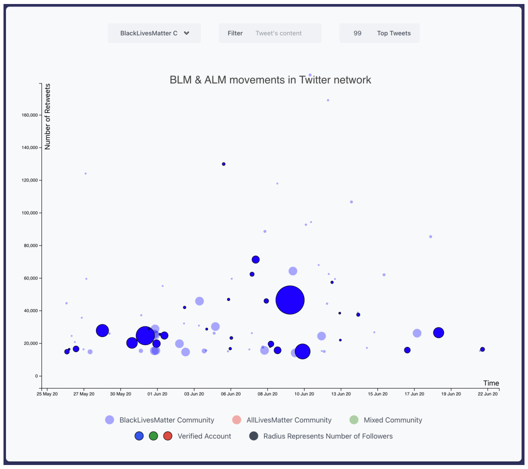
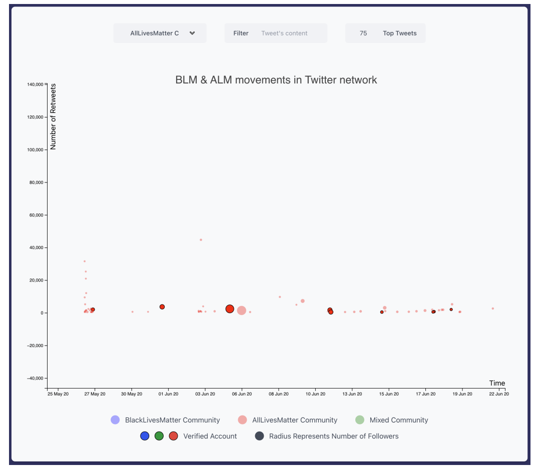



Figure 23: Top 10 Tweets

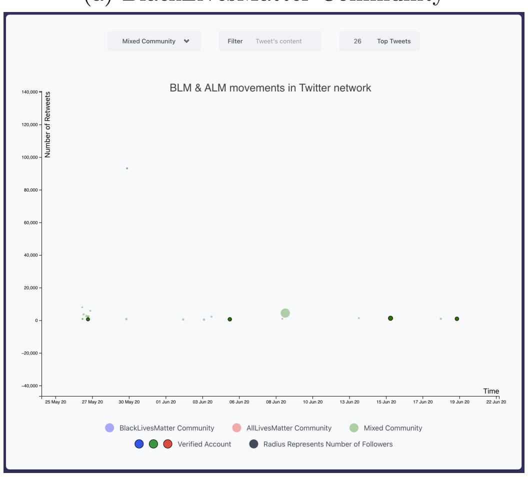
Another use of the selection strategy is to display only a specific group of data. Figure 24 shows multiple visualizations where each one of them displays only the data of one community. It helps the user to see the popularity, distribution, and counts of each community. Furthermore, it helps to focus on one group without getting distracted by other data while going through details using other strategies.




(a) BlackLivesMatter Community




(b) AllLivesMatter Community

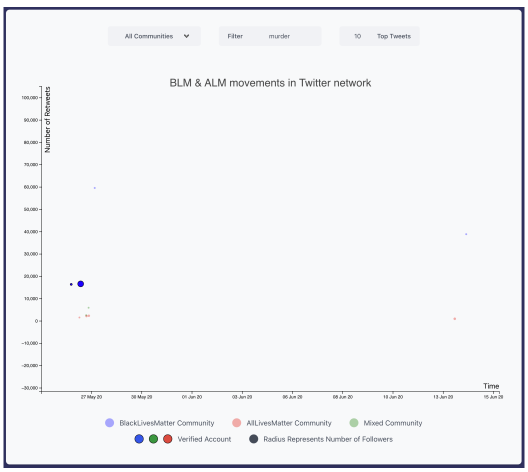



(c) Mixed Community

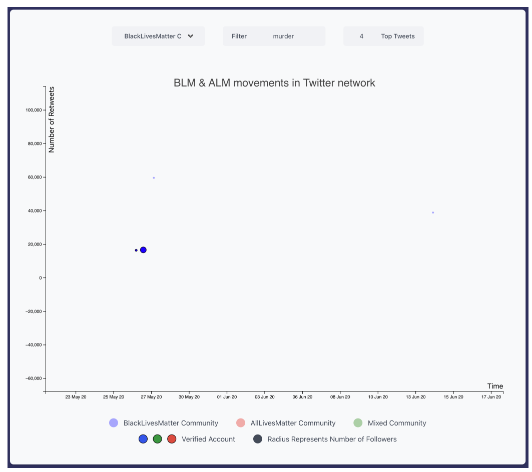
Figure 24: Communities' Tweets

In many cases and this case, specifically, enabling the user to explore the relationships between tweets is powerful. It allows the users to create such a network of data of their choice and based on their vision without overwhelming the display with links between the data or with preprocessed links based on the author's view. Figure 25(a) shows how the user can type a text and then filter only tweets that contain the text. In this case, filtering the tweets using words and names that are related to events that occur through the timeline of the movement could result in creating links between different events and groups.

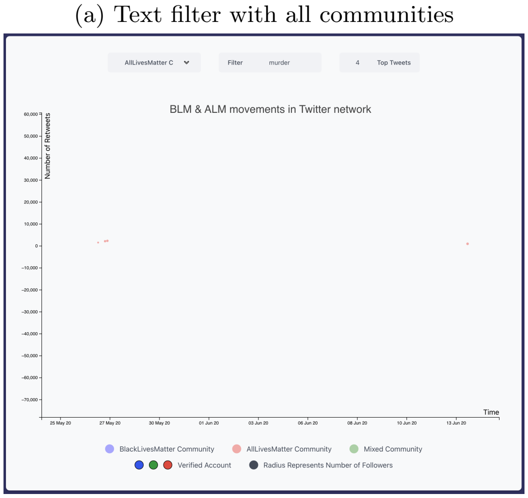
Going further and combining multiple filters as shown in the other visualizations in Figure 25 can create stronger links because it allows the user to create a subgroup of a subgroup that indicate tighter links between them. Such filters can mimic the idea of a network visualization in some cases like the one shown here without any extra analysis the data beforehand and also without devoting another graph to visualize the network data.



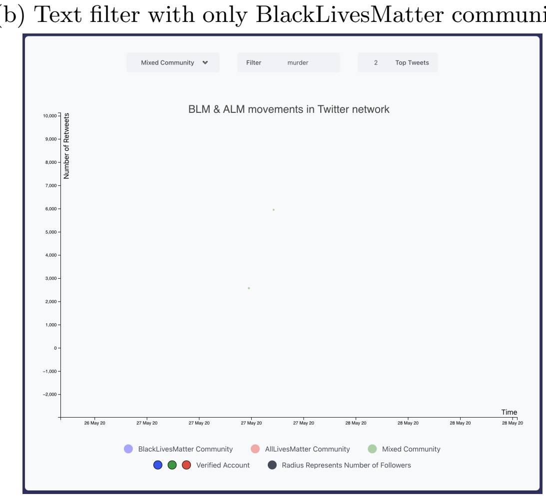

(a) Text filter with all communities




(b) Text filter with only BlackLivesMatter community




(c) Text filter with only AllLivesMatter community




(d) Text filter with only Mixed community

Figure 25: Multiple Filters

## 6.7 Some Data Insights Using Interactive Visualization Strategies

### 6.7.1 Observing AllLivesMatter Community Tweets Over Time

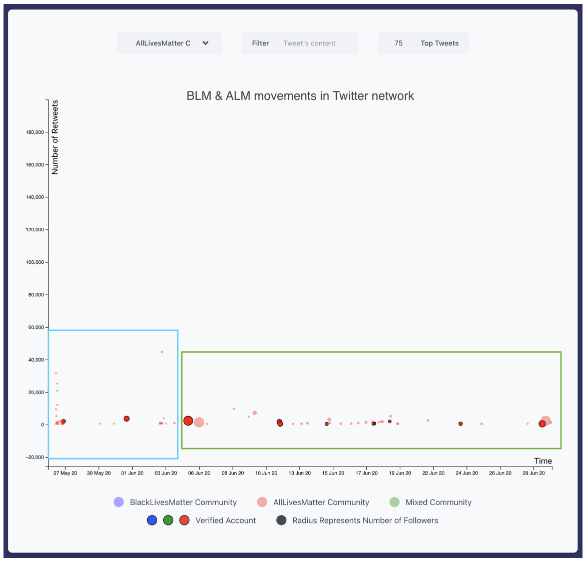
First of all, this data does not represent all tweets that were published after the death of George Floyd nor it is claimed to be. It only represents the top hundred most retweeted tweets of each community of the data collected from a prior study and the insights presented in this section just from the provided data only showcase how the interactive visualization strategies implemented for this thesis project could help in revealing those insights.

By using the community drop-down menu and selecting the AllLivesMatter community, Figure 26 shows that AllLivesMatter Community tweets which are 75 tweets out of the 200 tweets for all communities. Observing all tweets' contents shows that tweets' context changes over time. Tweets inside the blue box in Figure 26 are mostly using the AllLivesMatter hashtag to criticize people behind that hashtag as Figure 26(b) shows some of these tweets. As the tweets inside the blue box represent the first few days following George Floyd's death, it shows how people were reacting to the AllLivesMatters movement and how they thought it was not relevant to the central message of the BlackLivesMatter.

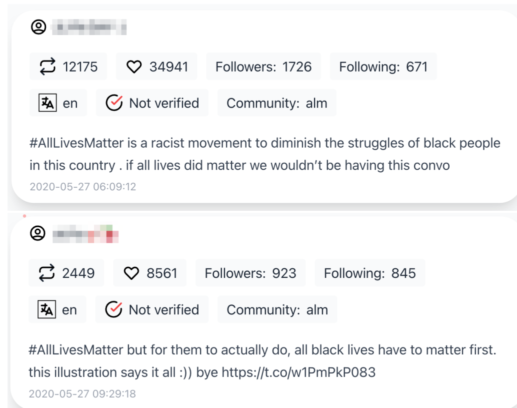
On the other hand, many of the tweets inside the green rectangle which started a few days after George Floyd's death have a different context. Around 60% of the users behind these tweets are from the AllLivesMatters community and the context of their tweets whether complaining about the actions of the people behind the BlackLivesMatter movement and accusing them of violent acts or with modest opinions to ease the conflict.

Perceived insights from the visual exploration of the AllLivesMatter community include:

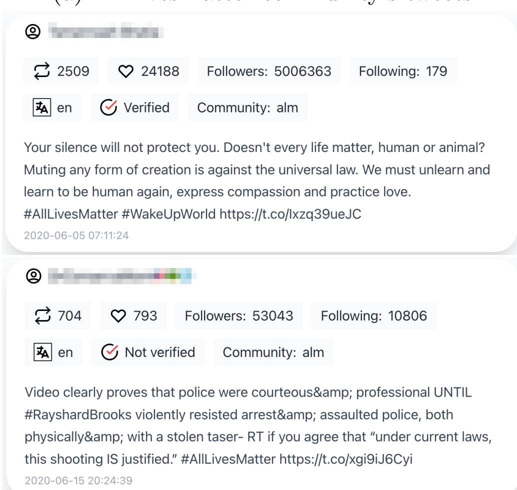
- AllLivesMatter movement is much less in counts and number of retweets compared to the BlackLivesMatter movement.
- Many of the tweets using the AllLivesMatter hashtag are actually by users criticizing the AllLivesMatter movement.
- Tweets during the first few days after the death of George Floyd were the most popular in terms of retweets and most of them were criticizing the AllLivesMater movement.
- With time, many tweets using the AllLivesMatter hashtag started to criticize some subversive actions supposedly committed by some people behind the BlackLivesMatter movement.



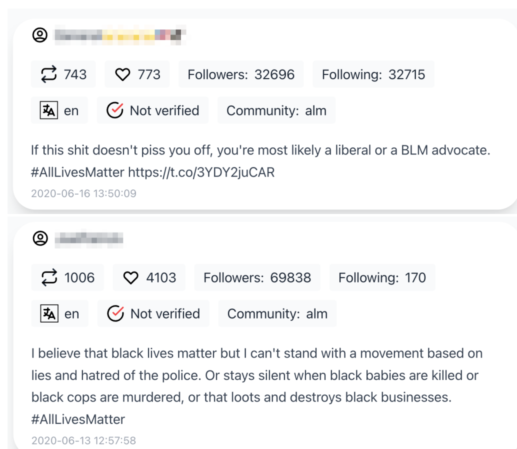

(a) AllLivesMatter community's tweets




(b) Area inside the blue box




(c) Area inside the green rectangle




(d) Area inside the green rectangle

Figure 26: AllLivesMatter Community Overtime

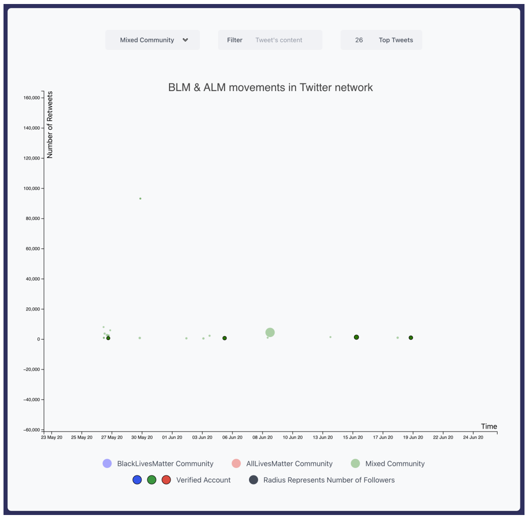
### 6.7.2 Verifying the mixed community

When processing the data before the visualization, a new column called Community was added mainly to categorize tweets that contain both AllLivesMatter and BlackLivesMatter hashtags to observe why was that the case and whom do the users behind the tweets aligned with. Figure 27 shows the 25 tweets which were displayed when the mixed community filter was applied. Moreover, part(b) shows two tweets as an example of the displayed tweets from the Mixed community.

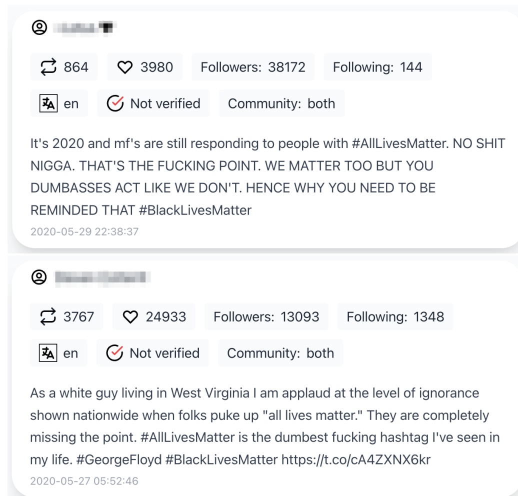
Perceived insights from the visual exploration of tweets containing both hashtags

- It is considered the smallest community out of the three.

- In most cases, including both hashtags was used to criticize the AllLivesMatter movement.




(a) Mixed community's tweets

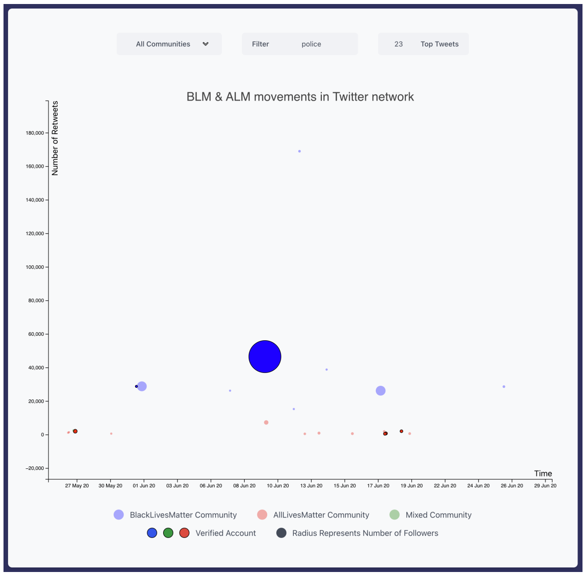



(b) Examples of the tweets

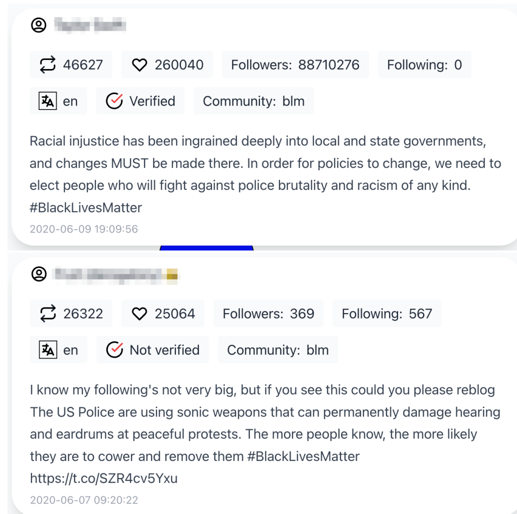
Figure 27: Mixed Community Overtime

#### 6.7.3 Observations from conducting multiple filters

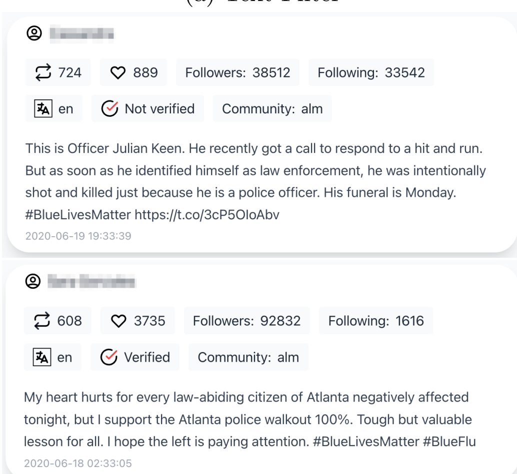
It is quite interesting to see and observe how each community reacts to the same expression. Figure 28(a) shows the counts and distribution of data points after typing "police" in the tweets' content filter. To make sure that only tweets with the "police" word are not to mix that with the "police" word being included in a hashtag, a space was typed before "police". There are 23 tweets in total across all communities, 9 tweets within the BlackLivesMatter community, 13 tweets within the AllLivesMatter community, and only one tweet with mixed hashtags. Figure 28(b) shows tweets from the BlackLivesMatter community with the "police" word in context, while Figure 28(c) shows the ones from the AllLivesMatter community.
It is clear that in most cases each community has a different reaction to the police actions. While the majority of the BlackLivesMatter community believes that police officers and cops are racist and brutal towards black people, the majority of the AllLivesMatter community supports police actions and even sometimes believes that protesters are violent against the cops.




(a) Text Filter




(b) Text filter with only BlackLivesMatter community




(c) Text filter with only AllLivesMatter community

Figure 28: Multiple filters Observations

# 7 Conclusion

What can not be touched, smelled, or tasted can be represented visually [50]. With all the usage of data visualization as a tool for data cleaning, data structure exploration, trends and clusters identification, presenting results, and many other usages, it does still struggle with its static form to visualize big data in a way that could help in decision making. Data quantity and the limited number of pixels in display require multiple static visualizations to show variant perspectives of the same data.

Interactive visualization based on the “Overview first, zoom and filter, then details on demand” [53] design strategy by Shneiderman was a way to deal with the snag that emerges when trying to statically visualize big data. Interactive visualization with its twofold strategies, navigation strategies, and visual interaction strategies opens the door to many strategies to be used when visualizing big data. On one hand, navigation strategies such as zoom & pan, overview & detail, and focus & context allow the navigation between levels of detail in the data without sacrificing the whole picture. While on the other hand, visual interaction strategies such as selecting, linking, filtering, and rearranging & remapping are more user-centered techniques that allow access to alternative perspectives to cover what big data can offer without the need for multiple static visualizations to be displayed separately.

Visualizing big data using some or all of the interactive visualization strategies needs a powerful tool. Nowadays, there are plenty of tools and they categorized by whether they are drag and drop types or require coding skills. Drag and drop tools such as Tableau and Microsoft Power BI are powerful visualization tools and they are capable of delivering interactive visualizations in no time. They are user-friendly and they are popular between teams and companies since they can connect to many data stores and can be integrated into all major advanced databases. The drawbacks of using such tools are the limited customization and in many cases, they can not be used as showcases for users outside the tool’s environment to interact with. On the other hand, tools that required coding skills such as D3 and Vega are open source tools. Many of these tools are based on JavaScript so they can be displayed on the browser by anyone around the world. For example, D3 is one of the most powerful and customizable visualization tools that can visualize countless interactive visualizations which can be customized specifically to the user’s needs but on the other hand, it requires a great amount of programming knowledge in D3 and other JavaScript libraries. Other tools like Vega and Vega-Lite are based on D3 and can be programmed with fewer lines of codes compared to using D3 directly but also with less customization.

Interactive visualization can serve preprocessed data to demonstrate the author’s insights or can also serve to visualize more general data and allow the users to obtain their observations by providing them with the right interactive strategies and tools that can help them to display multiple perspectives of the data. However, customised interactive strategies require a custom use case. We have shown the power of D3 by visualizing a specific dataset in Twitter around the BlackLivesMatter and AllLivesMatter movement. We have demonstrated various techniques to shed deeper insights into the data using custom interactive visualizations in the previous Section.

# 8 Future Work

The visualization tool which was built for this project was only built for the data used in this project. A future project would be to automate this tool to accommodate many different datasets. The tool will need to be reconstructed and other libraries will need to be added to manage the projects' variables to adapt the user inputs for each visualization. New other filters will be added as well to give users a variety of options to use when visualizing their data.

The final version of the tool will start to work by first asking the user to upload their data and a configuration file. The configuration file will be a JSON file that the user would initially need to fill up. For example, which column has the x-axis data and what is the type of this data, which column represents the data points, which column represents their radius, which filters to include and to which data's features should be assigned, and other specifications to customize the tool to their needs. There will be a guideline on how to create the configuration file which will be a straightforward procedure and will take just a few minutes to do. When the user uploads both files, the program will read through both files and assign each data feature as described in the configuration file. It will also automatically exclude the unwanted filters to have a clean display. The popup window will be designed to display the requested information in every case by using an adaptive design to work for all scenarios.

# 9 Appendix

## References


[1] Sanchit Aggarwal. Modern web-development using reactjs. *International Journal of Recent Research Aspects*, 5(1):133–137, 2018.

[2] Ammar Aldhahyani. A react and d3 visualization tool for twitter. `https://github.com/ammarhd/Twitter-visualization-thesis`, 2022.

[3] Shaikh Arifuzzaman, Maleq Khan, and Madhav Marathe. A fast parallel algorithm for counting triangles in graphs using dynamic load balancing. In *2015 IEEE International Conference on Big Data (Big Data)*, pages 1839–1847. IEEE, 2015.

[4] Shaikh Arifuzzaman, Maleq Khan, and Madhav Marathe. A space-efficient parallel algorithm for counting exact triangles in massive networks. In *2015 IEEE 17th International Conference on High Performance Computing and Communications, 2015 IEEE 7th International Symposium on Cyberspace Safety and Security, and 2015 IEEE 12th International Conference on Embedded Software and Systems*, pages 527–534. IEEE, 2015.

[5] Shaikh Arifuzzaman and Bikesh Pandey. Scalable mining and analysis of protein-protein interaction networks. In *2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech)*, pages 1098–1105. IEEE, 2017.

[6] SM Arifuzzaman, Maleq Khan, and Madhav Marathe. Patric: A parallel algorithm for counting triangles and computing clustering coefficients in massive networks. 2012.

[7] Patrick Baudisch, Nathaniel Good, Victoria Bellotti, and Pamela Schraedley. Keeping things in context: a comparative evaluation of focus plus context screens, overviews, and zooming. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 259–266, 2002.

[8] Richard A Becker and William S Cleveland. Brushing scatterplots. *Technometrics*, 29(2):127–142, 1987.

[9] Richard A. Becker, Stephen G. Eick, and Allan R. Wilks. Visualizing network data. *IEEE Transactions on visualization and computer graphics*, 1(1):16–28, 1995.

[10] Benjamin B Bederson and James D Hollan. Pad++ a zooming graphical interface for exploring alternate interface physics. In *Proceedings of the 7th annual ACM symposium on User interface software and technology*, pages 17–26, 1994.

[11] Michael Bostock, Vadim Ogievetsky, and Jeffrey Heer. D3: Data-driven documents. *IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis)*, 2011.

[12] Mike Bostock. D3 gallery. `https://observablehq.com/@d3/gallery`, 2020. Accessed: 2022-04-10.

[13] Tim J Bristol. Twitter: consider the possibilities for continuing nursing education. *The Journal of Continuing Education in Nursing*, 41(5):199–200, 2010.

[14] Brushing and linking, interactive visualization. `https://vda-lab.github.io/visualisation-tutorial/vegalite-brushing-and-linking.html`, 2020. Accessed: 2022-04-04.

[15] Dawson Byrd. Line chart, multiple series. `https://observablehq.com/@dawbyrd/page-rank-score-time-series`, 2021. Accessed: 2022-05-20.

[16] Javascript stockchart with ema - exponential moving average indicator. `https://canvasjs.com/javascript-stockcharts/exponential-moving-average-stockchart/`. Accessed: 2022-04-15.

[17] Datawrapper app. `https://app.datawrapper.de/map/xlVE4/visualize?tpl=4SHJQ#refine`, 2022. Accessed: 2022-05-20.

[18] Mario Delgado. 2020 presidential election time series. `https://observablehq.com/@mariodelgadosr/2020-presidential-election-time-series`, 2020. Accessed: 2022-05-20.

[19] Benard P Dreyer, Maria Trent, Ashaunta T Anderson, George L Askew, Rhea Boyd, Tumaini R Coker, Tamera Coyne-Beasley, Elena Fuentes-Afflick, Tiffani Johnson, Fernando Mendoza, et al. The death of george floyd: bending the arc of history toward justice for generations of children. *Pediatrics*, 146(3), 2020.

[20] Dheeru Dua, Casey Graff, et al. Uci machine learning repository. 2017.

[21] Force directed layout. `https://vega.github.io/vega/examples/force-directed-layout/`. Accessed: 2022-04-13.

[22] Raven Gao. Network graph with d3.force grouping. `https://observablehq.com/@ravengao/force-directed-graph-with-cola-grouping`, 2019. Accessed: 2022-05-10.

[23] Alicia Garza. A herstory of the# blacklivesmatter movement. 2014.

[24] Salvatore Giorgi, Sharath Chandra Guntuku, Muhammad Rahman, McKenzie Himelein-Wachowiak, Amy Kwarteng, and Brenda Curtis. Twitter Corpus of the BlackLivesMatter Movement And Counter Protests: 2013 to 2020. 2020.

[25] Google chart gallery. `https://developers.google.com/chart/interactive/docs/gallery`, 2020. Accessed: 2022-04-13.

[26] Google map, detail view. `https://www.google.com/maps/place/Uppsala+Cathedral/@59.8581121,17.6314511,17z/data=!4m12!1m6!3m5!1s0x465fcbf133adc41b:0xf1c543d7d3d0188a!2sUppsala+Cathedral!8m2!3d59.8581094!4d17.6336451!3m4!1s0x465fcbf133adc41b:0xf1c543d7d3d0188a!8m2!3d59.8581094!4d17.6336451`. Accessed: 2022-04-07.

[27] Ivan Herman, Guy Melançon, and M Scott Marshall. Graph visualization and navigation in information visualization: A survey. *IEEE Transactions on visualization and computer graphics*, 6(1):24–43, 2000.

[28] Kasper Hornbæk, Benjamin B Bederson, and Catherine Plaisant. Navigation patterns and usability of zoomable user interfaces with and without an overview. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 9(4):362–389, 2002.

[29] Infogram, visualization tool. `https://infogram.com`. Accessed: 2022-04-15.

[30] Brian Johnson and Ben Shneiderman. Tree-maps: A space-filling approach to the visualization of hierarchical. *Readings in information visualization: Using vision to think*, pages 284–291, 1991.

[31] KFF. Covid19 global tracker. `https://public.tableau.com/app/profile/kffdata/viz/KFF-CoronavirusTracker-Update4-10-20/Overview`, 2020. Accessed: 2022-04-10.

[32] Sun-Yuan Kung. Visualization of big data. In *2015 IEEE 14th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC)*, pages 447–448. IEEE, 2015.

[33] Jure Leskovec and Andrej Krevl. Snap datasets: Stanford large network dataset collection, 2014.

[34] Many time series plots. `https://homepage.divms.uiowa.edu/~luke/classes/STAT4580/timeseries.html`. Accessed: 2022-06-01.

[35] Rick Martin. Metro stations as network diagrams. `https://observablehq.com/@martin-pi/in-progress-metro-stations-as-process-diagrams`, 2021. Accessed: 2022-05-10.

[36] Helena Matute, Fernando Blanco, Ion Yarritu, Marcos Díaz-Lago, Miguel A Vadillo, and Itxaso Barberia. Illusions of causality: how they bias our everyday thinking and how they could be reduced. *Frontiers in psychology*, 6:888, 2015.

[37] Virendra Mistry. Critical care training: using twitter as a teaching tool. *British Journal of Nursing*, 20(20):1292–1296, 2011.

[38] Lekha Nair, Sujala Shetty, and Siddhanth Shetty. Interactive visual analytics on big data: Tableau vs d3. js. *Journal of e-Learning and Knowledge Society*, 12(4), 2016.

[39] Chris North. Multiple views and tight coupling in visualization: A language, taxonomy, and system. In *Proc. CSREA CISST 2001 Workshop of Fundamental Issues in Visualization*, pages 626–632, 2001.

[40] Josh Parenteau, Rita L Sallam, Cindi Howson, Joao Tapadinhas, Kurt Schlegel, and Thomas W Oestreich. Magic quadrant for business intelligence and analytics platforms. *Recuperado de https://www. gartner. com/doc/reprints*, 2016.

[41] Prof. Cristina Pattuelli. Linked jazz. `https://linkedjazz.org/network/`. Accessed: 2022-04-04.

[42] Judea Pearl. The science and ethics of causal modeling. *Handbook of ethics in quantitative methodology*, pages 383–416, 2011.

[43] Catherine Plaisant, David Carr, and Ben Shneiderman. Image-browser taxonomy and guidelines for designers. *Ieee Software*, 12(2):21–32, 1995.

[44] Plotly, dash enterprise. `https://plotly.com`. Accessed: 2022-04-15.

[45] Microsoft power bi. `https://powerbi.microsoft.com/en-us/what-is-power-bi/`. Accessed: 2022-04-15.

[46] Qlik active intelligence platform. `https://www.qlik.com/us`. Accessed: 2022-04-15.

[47] Ramana Rao and Stuart K Card. The table lens: merging graphical and symbolic representations in an interactive focus+ context visualization for tabular information. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 318–322, 1994.

[48] Kenneth J Rothman. *Epidemiology: an introduction*. Oxford university press, 2012.

[49] Gavriel Salvendy. *Handbook of human factors and ergonomics*. John Wiley & Sons, 2012.

[50] JL Valero Sancho. An approach to the taxonomy of data visualisation. *the taxonomy of data visualisation*, 2014.

[51] Arvind Satyanarayan, Dominik Moritz, Kanit Wongsuphasawat, and Jeffrey Heer. Vega-lite: A grammar of interactive graphics. *IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis)*, 2017.

[52] Arvind Satyanarayan, Ryan Russell, Jane Hoffswell, and Jeffrey Heer. Reactive vega: A streaming dataflow architecture for declarative interactive visualization. *IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis)*, 2016.

[53] Ben Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *The craft of information visualization*, pages 336–343. Elsevier, 2003.

[54] Sigma, network visualization. `https://www.sigmajs.org`. Accessed: 2022-04-15.

[55] Query widgets. `https://vega.github.io/vega-lite/examples/interactive_query_widgets.html`. Accessed: 2022-04-13.

[56] Michelle Q Wang Baldonado, Allison Woodruff, and Allan Kuchinsky. Guidelines for using multiple views in information visualization. In *Proceedings of the working conference on Advanced visual interfaces*, pages 110–119, 2000.

[57] Matthew O Ward, Georges Grinstein, and Daniel Keim. *Interactive data visualization: foundations, techniques, and applications, second ed.* A K Peters Publishers, Natick, MA, 2015.

[58] Graham J Wills. Selection: 524,288 ways to say" this is interesting". In *Proceedings IEEE Symposium on Information Visualization'96*, pages 54–60. IEEE, 1996.

[59] Jeremy M Wolfe, Keith R Kluender, Dennis M Levi, Linda M Bartoshuk, Rachel S Herz, Roberta L Klatzky, Susan J Lederman, and Daniel M Merfeld. Sensation and perception. *Sensation and perception*, 2015.

[60] Zooming and panning, interactive visualization. `https://vda-lab.github.io/visualisation-tutorial/vegalite-zooming-and-panning.html`, 2020. Accessed: 2022-04-04.