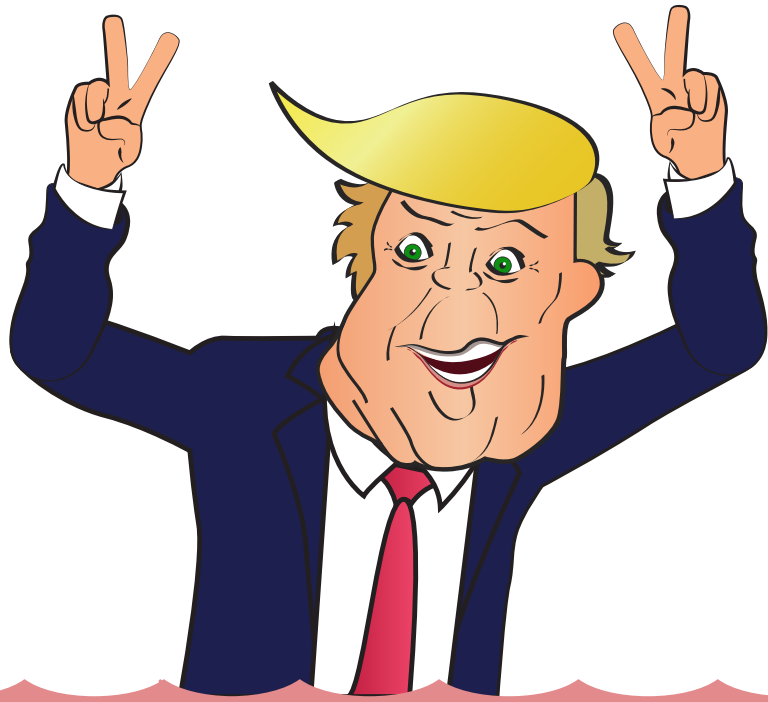
Interactive Visualisation and Sonification of the 2016 United States presidential election



Olena Mikhanosha
Connor Guy Meehan
Jessica D'Ali
Xian Keng Feng

Table of contents

Link to the video: https://vimeo.com/mikhanosha/us-election

Link to GitHub: https://github.com/Daymeo/sea-of-tweets-js

Interactive visualisation and sonification

The primary purpose of this report is to examine in detail the design criteria that guided the development of our group's interactive media project. Each criterion will be explained by addressing the following points about it:

- What is it and what does it mean to our group?
- Why did we choose it?
- How did we fulfill it in our final version?

Next, the report will conclude with a reflection of our experiences as a group throughout the project. The lessons learned will be discussed, as well as how these learnings will be applied in future projects.

Link to the video: https://vimeo.com/mikhanosha/us-election

Link to GitHub: https://github.com/Daymeo/sea-of-tweets-js

Design Criteria

The application should be unique and fun

Our group chose to create an interactive visualisation and sonification of the 2016 United States presidential election for assignment 1. We reached a unanimous decision that, regardless of what type of data, we wanted to present it in a unique way. To elaborate, we wanted to stay away from things like bar graphs, pie charts and infographics. The reason for this is because everyone in the group felt that those sorts of data representations, although in many instances very helpful and informative, are too common. We wanted to create something different.

The first idea to help fulfill this goal came almost instantly. A recurring theme throughout the history of the United States presidential elections is the caricaturisation of candidates in news articles and tabloid comic strips. We felt that employing caricaturisation in our piece would allow us to establish relevance to the topic in a fun way.

Our group member, Olena, has a background in design and is very experienced with Adobe Illustrator. We capitalised on her expertise and had her design and create a set of cartoony images with exaggerated features for each candidate. Initially, we wanted each candidate to only have a different expression based on their state, i.e. whether they were talking or had just won or lost a battle. As we progressed, we felt that we could communicate the events in a more entertaining way. We kept the idea of changing states, but the candidates would become animated as opposed to just changing expressions. For each state, Olena drew three frames that would be played in quick succession to further emphasise the theme of caricaturisation we wanted to display. Moreover, we used a lively colour scheme to highlight the playfulness of our project. As you will see in our final version, the colours are bright and well-contrasted, all the while following the actual colours of each party. In regards to the audio aspect, we cut phrases from each candidate's YouTube interviews and speeches to play depending on whether they had just won or lost. The targeted outrageous yet characteristic sound-bites to further bolster the theme.

To conceive the main concept, we started by thinking about the key points of a presidential election. We agreed that it was essentially a popularity contest whereby two parties headed by their respective leaders battle through campaigning to win. We liked the idea of Clinton and Trump 'battling', so we brainstormed the different ways in which we could portray them doing that. Ultimately, we selected the idea of a 'tweet battle' where we would pit tweets by each candidate against each other and declare a victor based on the popularity of each tweet. We felt that although this was not an entirely unique idea, it allowed us the freedom to add fun elements throughout the application that would help differentiate our project. For example, as a candidate is tweeting, we were able to animate their supporters jump up and down excitedly.

The data used should be interactive and updated in real time

We chose this criterion after conceiving the main idea for the application. When we decided to make it about 'tweet battles', we had two options. The first was to compile a small database of tweets by both candidates and then selecting tweets from that pool at random. The alternative solution was to pull the data directly from the Twitter API into our program. We chose to use real-time data because we felt that it would make our project more interactive and dynamic. The result is that the user gets a different experience on each runthrough. Our application updates the data in real-time through the Twitter REST API and the Huffington Post Pollster API (Huffington Post 2016). Both of these retrievals happen on an hourly basis. One call to the Twitter API will return the most recent 100 tweets. Further expanding on the interactive element, we found that through the Twitter API we were able to provide links in our application in each tweet that allows users to either like or retweet that particular tweet with their own account.

The application should be easy to follow and understand

In the early stages of development, our group spent a lot of time brainstorming ideas. We did not take into account our time constraints and accepted most ideas as long as they did not clash with others. As we progressed, the scope of the project became increasingly bloated and the idea started to lose focus. We realised that this would become problematic not only for ourselves during development, but also more importantly for our audience. We felt that clarity and ease of understanding were critical to the success of our piece. To ensure that these two points would be achieved, we split our project into the three categories of core concept, visual and interactive. By breaking our project down into smaller and more manageable aspects, we were able to reflect more effectively on the clarity of each and therefore the project as a whole.

As previously mentioned, our core idea was highly convoluted at the beginning of the project as we kept adding more elements to it. Some of these elements were so subtle that only members of our group would understand it as we had spent so much time on it. This problem was highlighted to us we found it difficult to communicate our ideas to family and friends. For instance, we wanted to portray the popularity of each tweet against the respective candidates' number of followers. Our initial idea to demonstrate this was to depict every 100,000 Twitter followers with 1 boat, and then every 1000 retweets would be shown as 1 person on these boats. It was hard to explain this concept succinctly, let alone having people decipher this on their own. We simplified the representations in order to work towards our criterion of keeping things simple. In the final version, each boat depicts 100,000 followers. The total number of retweets for a particular tweet is shown as a score on the hot air balloon. The Twitter presence, i.e. the number of retweets per 100,000 followers is shown as a score located on each boat. This method allows users to quickly see who has a higher total score as well as the strength of each candidate's following.

In regards to the interaction aspect, we were going to implement a mini-game that the user could play between tweet battles. We wanted to add this game to serve as an extra layer of interaction on top of the relationship between our application and the two APIs. Although we all agreed to go ahead with it, the feedback we received from others was that they were confused about what purpose the mini-game served in the big picture. We realised that we were adding features for the sake of adding features, instead of meaningfully enhancing our application. This helped remind us that we were making this application for an audience other than ourselves and that ease of understanding was a key goal. We decided to stop developing the mini-game for this reason and understood that the interaction with Twitter and Huffington Post's APIs was sufficient for the purpose of our project.

Finally, we made several changes to the visual side of our project with the goal of making things easy to follow in consideration. Although our group members understood the meanings of all the different representations and numbers, it was not always clear for a first time viewer. For example in earlier versions, the total num-

ber of retweets was just shown as a number on the hot air balloon. While it could be inferred that it represented a score, viewers would not know how the score was calculated. The score is solely based on the number of retweets of the particular tweet and should serve as an indication of the popularity of each tweet. In order for our audience to easily understand this, we simply added the iconic retweet symbol next to the score. Another example of us altering the project's visual component to improve clarity is how we chose to display each tweet. We were initially going to take the text from each tweet and rewrite them in a stylised font that matched the rest of Olena's design. However, we ultimately decided against that because the source of each tweet would become unclear. Another visual clarification that we made to improve usability was increasing the delay between the sequence of events. The application previously played each tweet message too fast for the audience to read and digest what was happening.

The application should be web-based and run smoothly

We decided that the best place to display our project would be on a website for various reasons. The main reason is that our application is mainly observed as opposed to something like a game that users play. Because of this, it would be difficult to convince people to download it. Building on this point, developing our application to be web-based would greatly improve the accessibility of it with the high availability of smartphones and usage to browse the web nowadays. Furthermore, our application is relevant to the 2016 US presidential elections and so ideally it would be referenced in web articles and reports. Once again, this would be easier to achieve with a weblink.

We switched to p5 JavaScript because it suited our needs better than Processing. JavaScript is quicker and performs better for integrating our client closely with the web, which is what we wanted to do. In addition to this, JavaScript compliments the interaction with Twitter that our application requires. It is easier to render tweets as they look on Twitter. JavaScript also allows us to input like and retweet buttons within each tweet in our application more efficiently. Finally, we chose to use a Node server to make requests to the Twitter and Huffington Post APIs because of its synergy with JavaScript.

Development criteria

The most important requirement for the development of this project was that it must be a modular, class based. We wanted to design a modular engine that controlled the scene of the application so that elements can easily be added and removed. To achieve this we created a function that allows us to specify how far forward something is and add it to the scene in the correct position so the elements are correctly layered. The function then returns a pointer to that object to allow us to run functions and create a behaviour driven application. This method also means that, to render each element, we can simply loop through an array, rendering each element confident that they will remain in order.

Lessons learned

Working on this assignment has provided us with various valuable insights as to what it is like to work in a group on a digital media project. From these insights, we are able to draw strategies and techniques that we could apply to similar jobs in the future.

The predominant issue to be highlighted would definitely be related to our project scope management as a group. The first mistake we made was that we did not clearly define the scope as a group. In addition to this, we did not determine a system to prevent scope creep or prioritise tasks. The result is that for the first few weeks we kept adding additional ideas without verifying the necessity or relevance of them to our piece. It finally became evident that we had created too much work for ourselves as assignment deadlines approached and we struggled to meet the workload demand.

Luckily, everyone was very communicative and the group got along well from the start of the project. We held a meeting and made the decision to cull unnecessary elements or to put them on a backlog as a 'nice to have' if time permitted. In the process of unbloating the scope, we followed a system where each person was able to choose one thing they absolutely wanted to keep in the final version. This served to keep everyone on the same level of contribution in terms of ideas and so that no one felt left in the dark. This demonstrated the importance of great communication among the group and the use of fair procedures to boost team morale and participation.

Another thing that we felt worked out very well for us was the way that we distributed work. We did so based on each group member's strengths and weaknesses. This allowed our group to work as efficiently and effectively as possible. We also made sure to support and help each other where possible.

The final point to note that we feel contributed to the success of our project is that we actively sought feedback from friends and family. Hearing the opinions of non-group members helped us identify areas of uncertainty and therefore chances to improve our application. It also reminded us that we were creating an application for people other than ourselves, as we often assumed others had the same knowledge and understanding of the work as ourselves.

We will take our learnings from participating in this digital media group project into future works. We all agree that defining the scope early on is highly important, as well as ranking the importance of tasks to establish work priority. Making an effort to communicate with future group members is also something we will all take with us, as we felt it made the project easier and much more enjoyable. Finally, we will always try to allocate jobs based on strengths and weaknesses where possible as we have found it helps to produce the best work as possible.

Bibliography

Greenberg, I., Xu, D., Kumar, D. 2013, Processing: Creative Coding & Generative Art in Processing, Springer Science+Business Media, New York

Huffington Post 2016, 'POLL CHART 2016 General Election: Trump vs. Clinton', viewed 13/9/2016, <http://elections.huffingtonpost.com/pollster/2016-general-election-trump-vs-clinton>

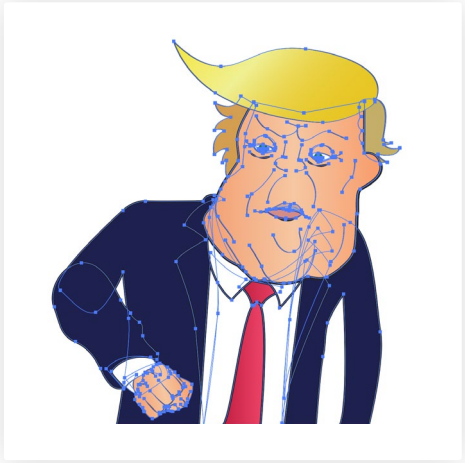
MB, Best of Donald Trump (funny) ,video recording, YouTube, viewed 30 October 2016, <https://www.youtube.com/watch?v=C-yIEfG-WbRk>

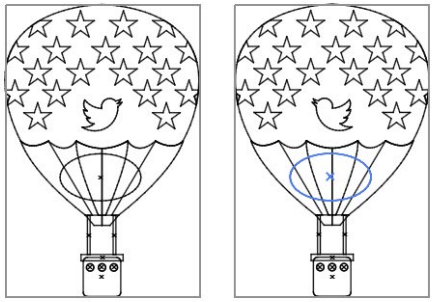
Twitter 2016, 'Twitter Developer Documentation - Object: Tweets', viewed 1/9/2016, <https://dev.twitter.com/overview/api/tweets>

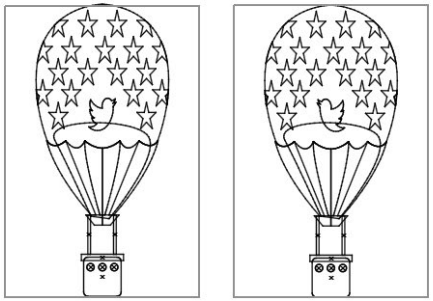
The Young Turks, Hillary Clinton DNC Speech: BEST Lines ,video recording, YouTube, viewed 30 October 2016, <https://www.youtube.com/watch?v=pYj5R_AdTqc>

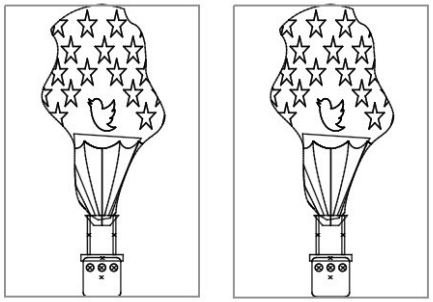
Illustration development

Animation stated designed in Adobe Illustrator









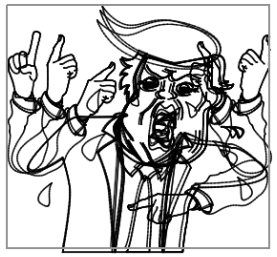
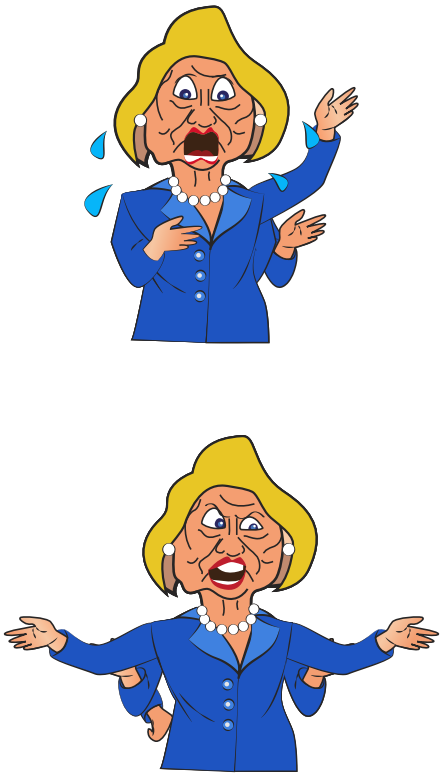


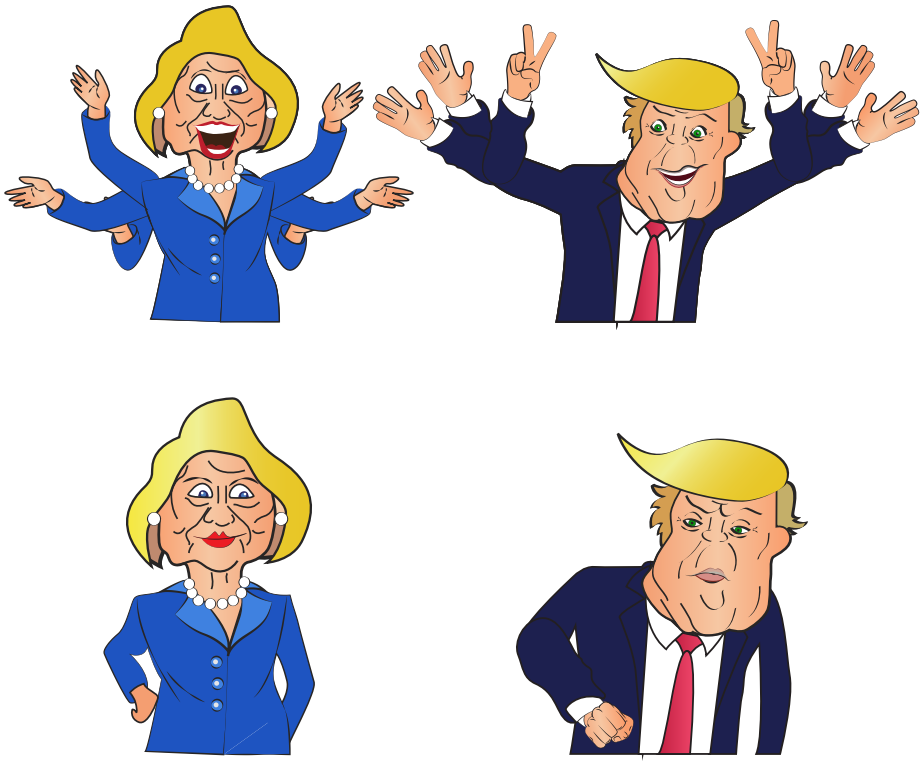
Illustration development

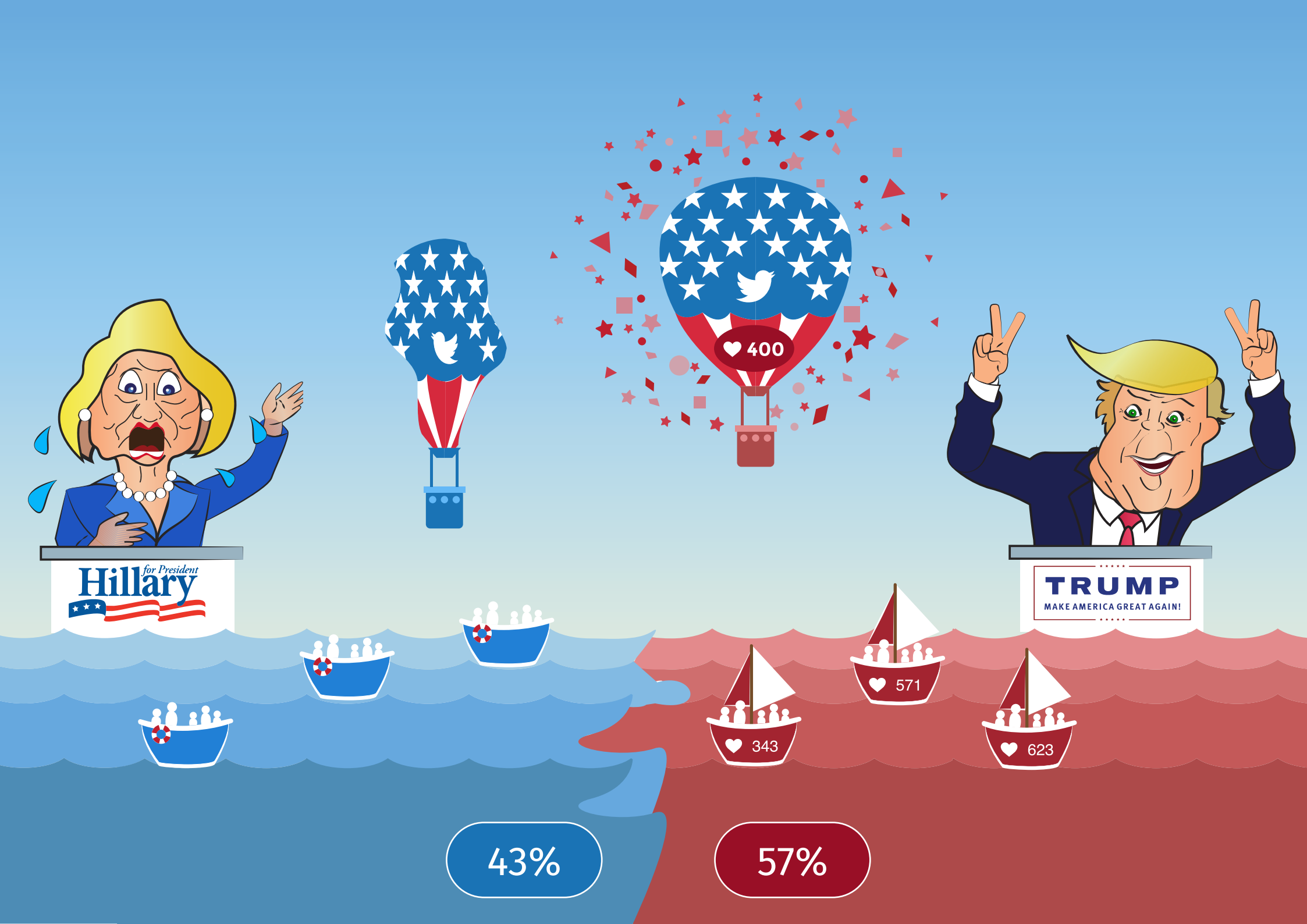
I designed three states for each emotion to make the animation

Link to the video: https://vimeo.com/mikhanosha/us-election

Link to GitHub: https://github.com/Daymeo/sea-of-tweets-js






400
for President
Hillary
TRUMP
MAKE AMERICA GREAT AGAIN!
571
343
623
43%
57%