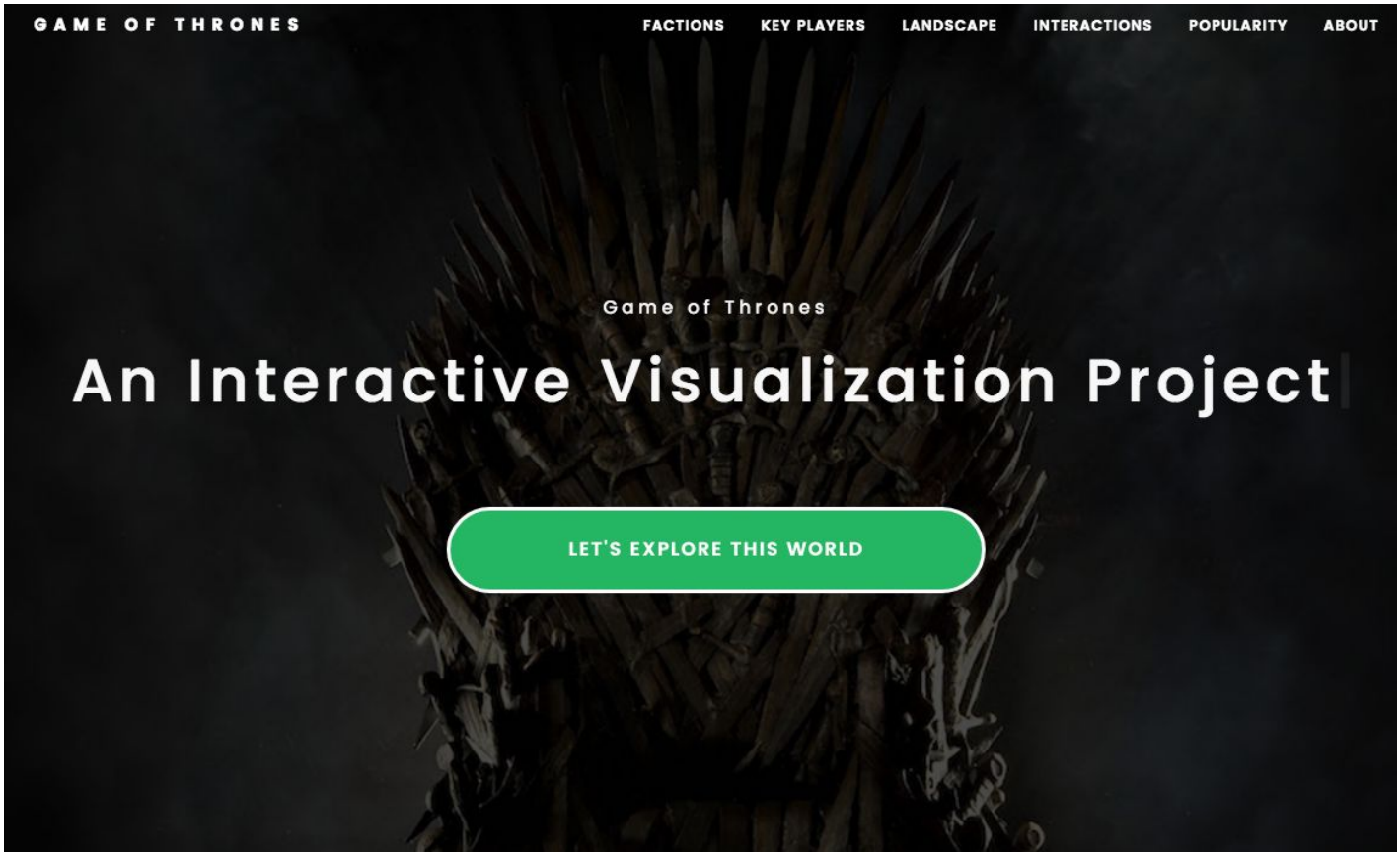
INFOVIZ FINAL REPORT

Game of Thrones - Interactive Visualization



Nihar Dalmia, Lynell Amanna, Shirish Dhar, Meghana Murthy

06.05.2017

Info 247

PROJECT GOALS

The goal of this project is to leverage powerful visualization tools and concepts from the class to understand the world of Game Of Thrones while analysing its effects on its viewers. Our motivation was also to be able to uncover key insights related to the show using information visualization concepts. We used feedback we received from Game of Thrones enthusiasts to narrow our focus to the selected areas of the show that are the most intriguing to most viewers.

As the world of Game of Thrones is massive, it was a challenge for us to narrow down our scope to a specific part of the show. The complex and intricate dynamics between individual characters and their families was relatively unexplored, and thus we decided to focus on these network dynamics in the Game of Thrones world.

Our end goal was to develop a set of interactive visualisations with a gripping narrative dedicated to Game of Thrones viewers, with the focus being on network dynamics. Our completed project involves the following components -

- The overall plot with central characters.
- Individual Character Interactions and storylines
- Family-level allegiance interactions
- Overall response of viewers to key events and characters in the show

RELATED WORK

SHIRISH:

Related Work 1:

An extensive paper on the network dynamics of Game of Thrones - http://www.maa.org/sites/default/files/pdf/Mathhorizons/NetworkofThrones%20%281%29.pdf

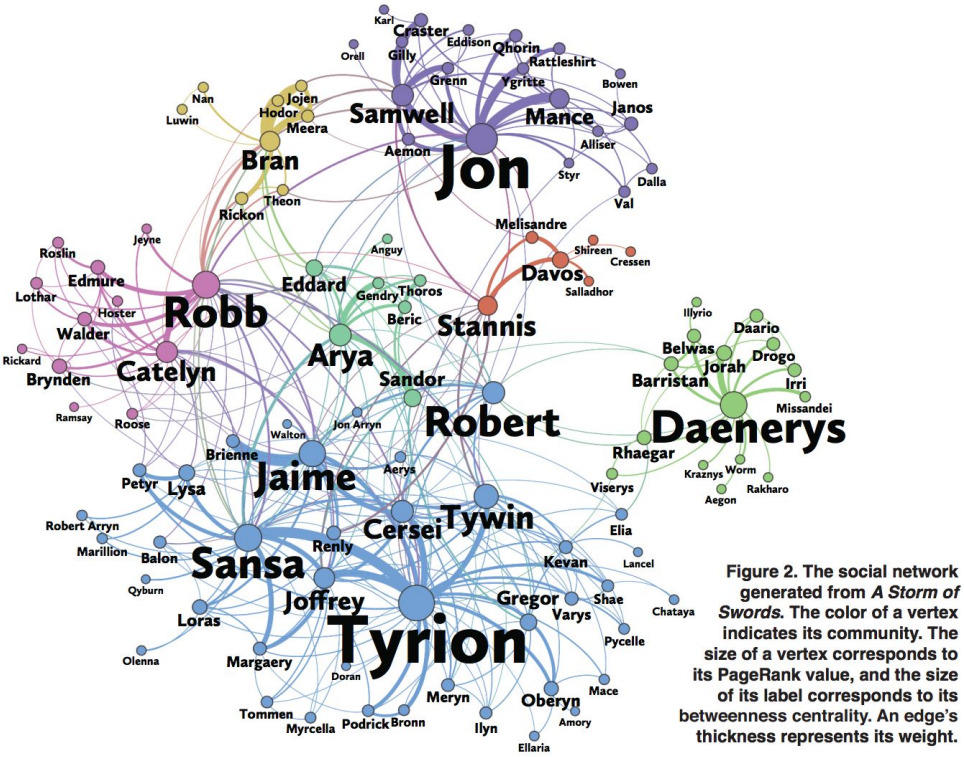


Figure 2. The social network generated from *A Storm of Swords*. The color of a vertex indicates its community. The size of a vertex corresponds to its PageRank value, and the size of its label corresponds to its betweenness centrality. An edge's thickness represents its weight.

This paper was one of the primary sources of inspirations for the focus of this project - Exploring network dynamics in the Game of Thrones world.

Most of my work during the project was focussed on the 'Current Landscape' allegiance visualisation as well as a minor portion on the 'Individual Level Interactions' force-directed visualisation. Both of these have a lot to do with communities, allegiances and the centrality of characters. This paper went a long way in inspiring many design considerations for both the visualisations that I was involved with.

For the 'Current Landscape' allegiance visualisation, this paper showed me that while there are hundreds of characters in the show, a select few stand out from the rest in terms of importance within the show as well as interactions. This paper was the inspiration behind creating the 'Current Landscape' visualisation - an intuitive introduction for viewers to understand the main contenders in the show as well as their supporters.

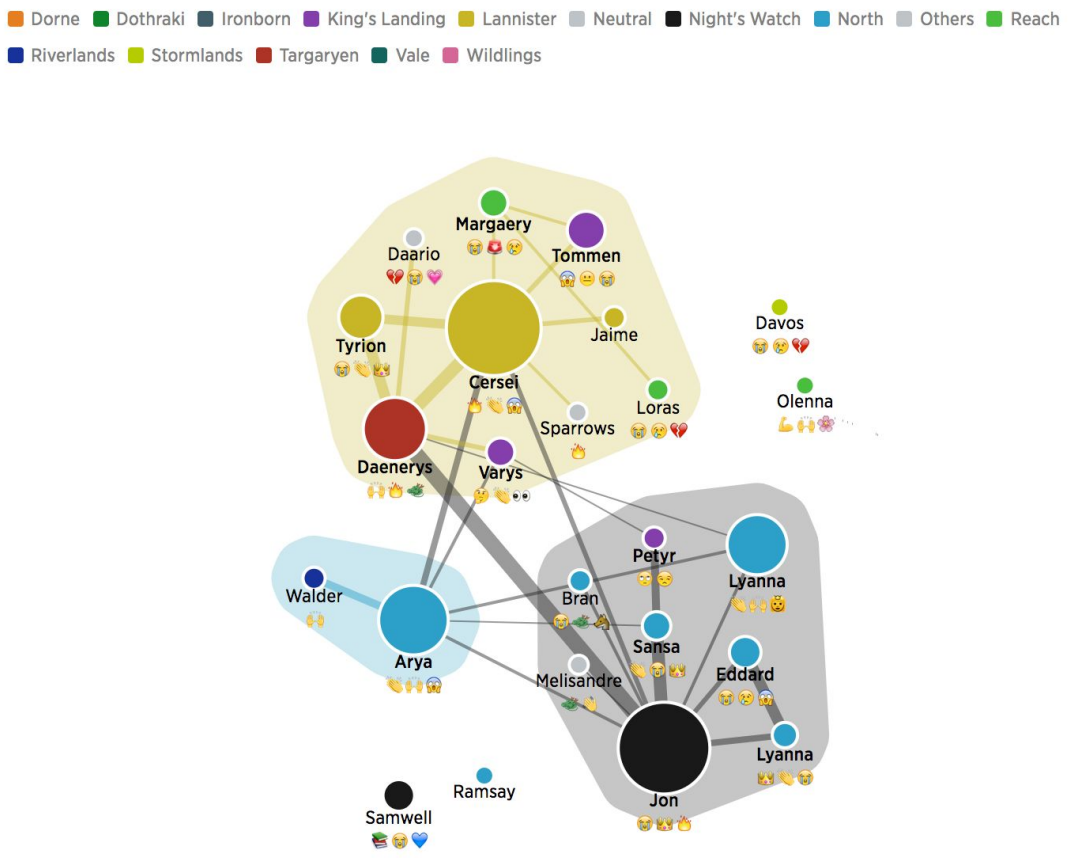
For the individual-level force-directed network graph, this paper helped us realise that while there are a huge number of characters interacting in the show, a well-designed

network graph is bound to show patterns and insights, instead of being a ‘hairball’ network. This inspired us to think of design considerations while creating the basic skeleton for the individual-level force directed network graph. Some of the key initial design considerations we chose were:-

1. To color the nodes of the network by the House each character was born into.
2. To have the length of a link (between two characters) to be inversely proportional to the number of interactions between them. This helped add modularity/clustering in the diagram.
3. Once the planned community detection is implemented, the color of each community would be different from each other as well as different from any of the House colors. This was done to ensure complete clarity and to prevent any ‘color confusion’ for the viewer.

Related Work 2:

In-Class Guest Lecture Twitter Visualisation - https://interactive.twitter.com/game-of-thrones/#?episode=60



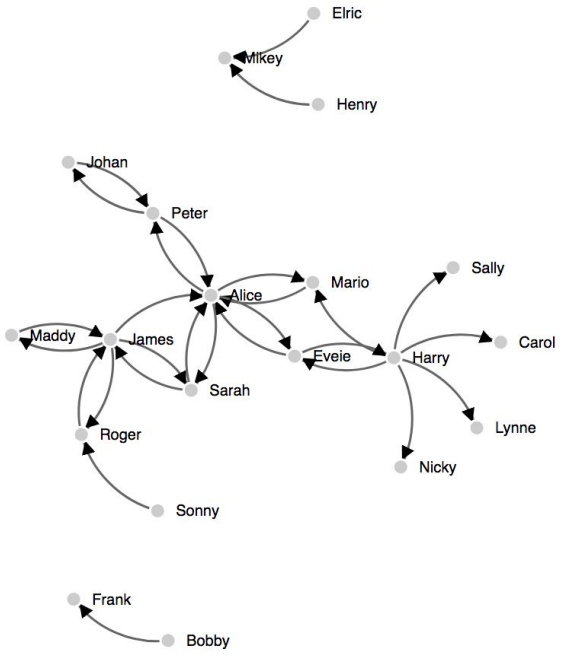
Witnessing this visualisation in one of our guest lectures for INFO 247 was a great source of inspiration and learning for me to apply into the two visualisations I was involved with - the ‘Current Landscape’ visualisation and the Individual-Level Force Directed Graph.

For the ‘Current Landscape’ visualisation, this related work went a long way in helping me conceptualise the design process. In this related work, the same set of characters can always be seen with bigger circles than the other, indicating that they are the pivotal characters in the show. This helped me realise that a hierarchy of characters can be formed such that the main contenders to the Iron Throne (the most talked about characters on the show) can be the top level characters, and the remaining characters can be clustered as supporters of these three main contenders. This related work also throws light on how different characters from the same House can be part of different communities and allegiances. This was an inspiration for the ‘Current Landscape’ visualisation, where clicking on each character will highlight all the other characters from the same House, thereby throwing light on the allegiance trends for that House.

For the Individual-Level Force Directed Graph, this related work was structurally identical to the vision for our graph. Nodes would be colored differently and the background communities would be colored differently. The inspiration to cluster communities also came from this related work - all the smaller distant clusters would be separated from the bigger clusters.

Related Work 3:

Basic Directional Force Layout Diagram - http://bl.ocks.org/d3noob/5141278



This graph was an important part of the initial conceptualisation and building of the ‘Current Landscape’ allegiance visualisation for me. It displays nodes and edges in a way that is unique and different from a traditional Force-Directed graph.

This piece of related work helped me build the first skeleton for the ‘Current Landscape’ visualisation. Most force-directed network graphs have parameters called ‘charge’ and ‘gravity’. These parameters decide how different nodes will interact with each other. In the conventional force-directed network graphs, the charge values are very high so that the nodes can be extremely flexible and reactive to other nodes. On the other hand, my vision for the ‘Current Landscape’ visualisation needed a much more stable, barely-moving-on-its-own set of nodes and edges. This was because my visualisation was more about giving the users an intuitive understanding of the current landscape of the show before delving deep into further visualisations. Thus, while I required flexibility in the nodes when users deliberately move each node, I did not want the nodes to behave in a volatile and reactive manner when they are not clicked. This graph has exactly the kind of interactions between nodes that I was looking for. I went on to learn a lot about the inner workings of a force-directed graph - the importance of the ‘charge’ parameter, the ‘gravity’ parameter, the length of the edges, and more, using the aforementioned piece of related work. It allowed me to cater my ‘Current Landscape’ visualisation nodes to my requirements.

That said, I was motivated to depict different characters in my visualisation using their images instead of names, as this would help make the experience much clearer for the users. To understand how to associate nodes with images, and also provide added functionality like hovering and clicking events, I had to refer to tutorials and illustrations separate from this aforementioned piece of related work.

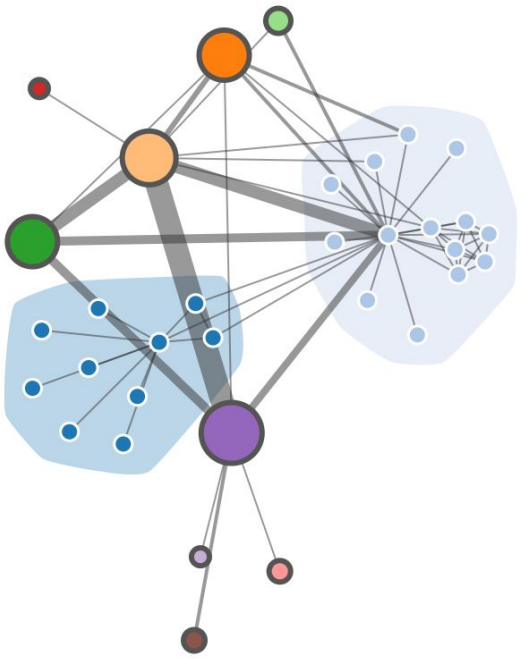
MEGHANA:

Apart from the paper mentioned by Shirish, http://www.maa.org/sites/default/files/pdf/Mathhorizons/NetworkofThrones.pdf, which served to be the starting point for the individual character interaction force directed graph, the following related works helped in tweaking various aspects of the visualization to make it more informative by providing various insights about factions and allegiances.

Related work 1:

Force layout with grouping based on the color: http://bl.ocks.org/GerHobbelt/3071239

The below graph shows an interesting and unique visualization of the force directed graph. This visualization is different from the thousands of other force directed graphs out there in the wild. The nodes are clustered together and hence instead of just showing a hairball which is not informative, it provides clustered nodes which was also the main theme of our visualizations. Through this visualization, I also learned the importance of some of the properties of the force directed graph such as gravity, charge, linkDistance and linkStrength which helped in shaping the structure of the individual character force directed graph by clustering communities and separating them from one another.

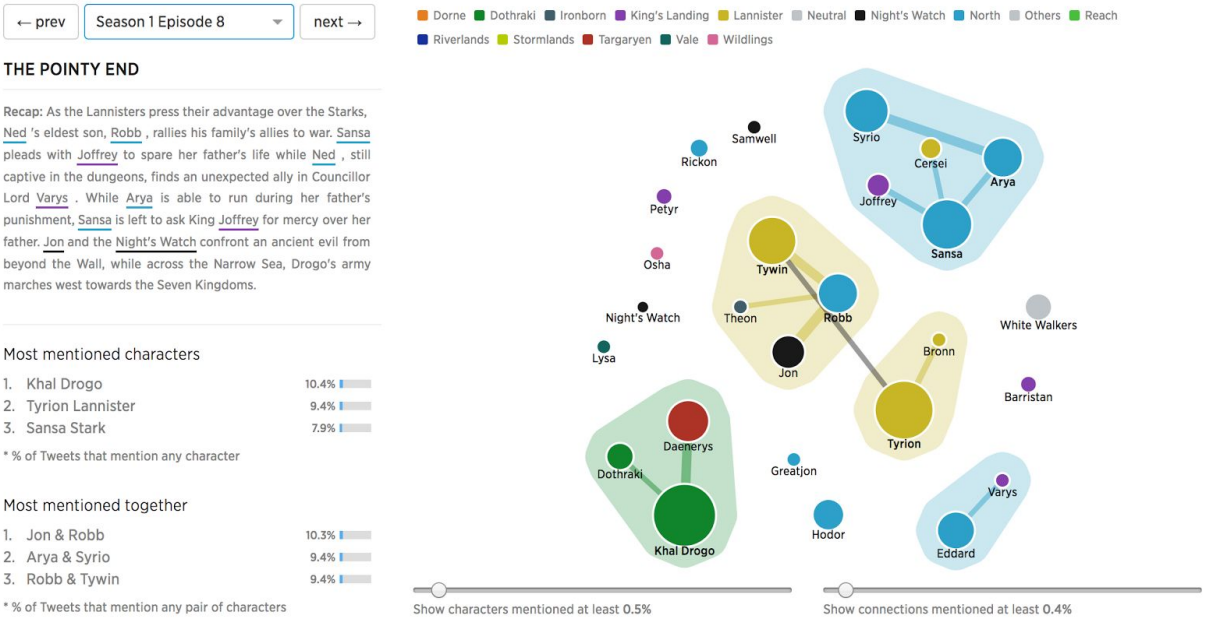


However, in this visualization the clustered nodes could be expanded or collapsed since the node and the community colors were essentially the same. In our case, we had to show that the community population is diverse and hence I decided not to incorporate that functionality and have all the communities expanded to clearly depict the diversity in the communities.

Related work 2:

GoT Twitter Interactive Visualization: https://interactive.twitter.com/game-of-thrones

The Game of Thrones Twitter visualization demonstrated in one of the guest lectures of the course was the major inspiration for the visualization I was involved, the individual character interaction force directed graph. This graph clearly showed the implementation of Gestalt's principles of proximity and grouping and it's importance.



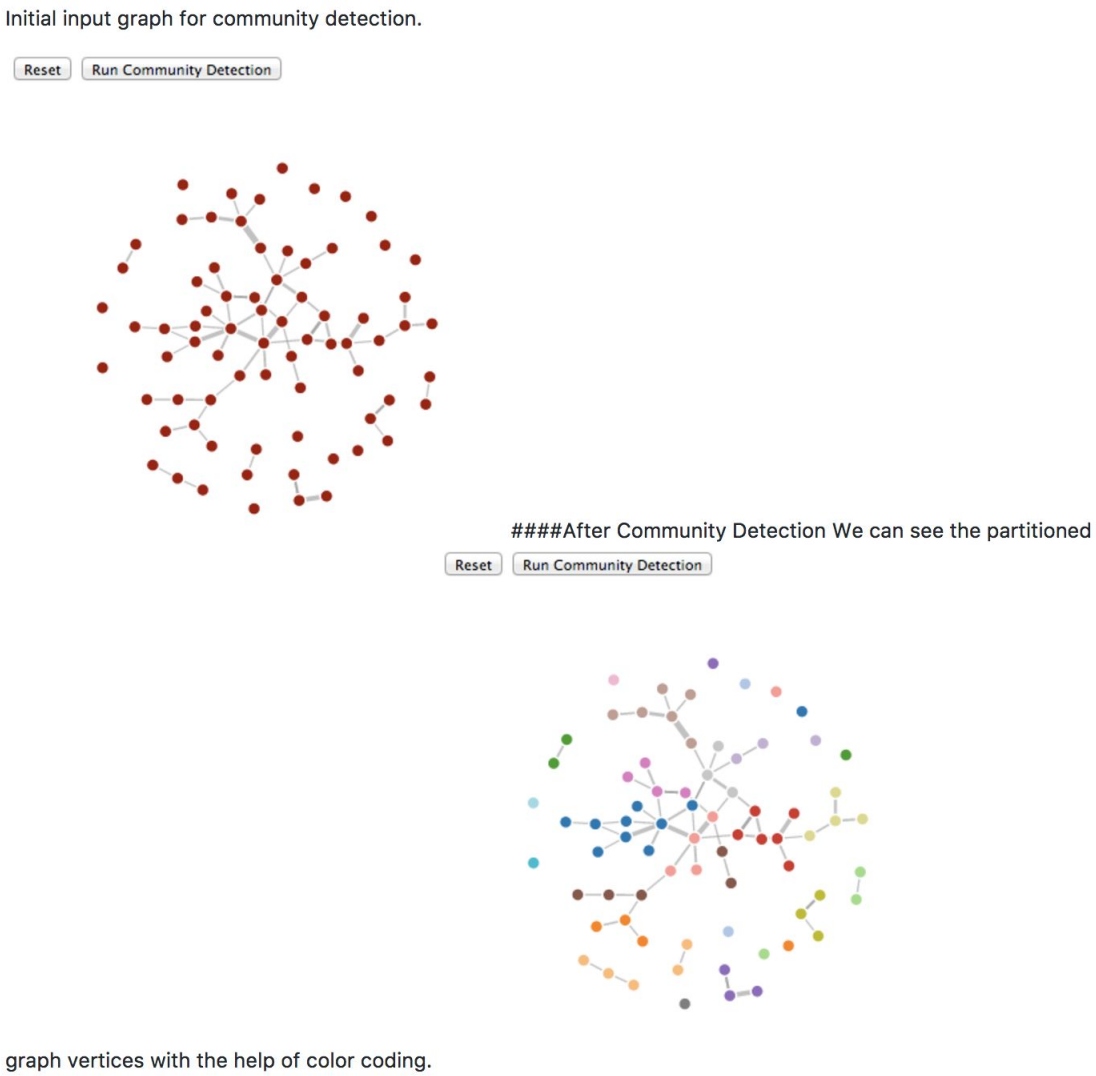
In this graph, each faction was colored differently and the convex hull representing a community containing characters from different factions had a background color. This was very similar to our vision of the individual character interaction force directed graph where we wanted to show how individuals can be born in a different faction but belong to a different community. Hence, instead of Twitter popularity, we utilized the data containing the actual interactions between the characters in the show.

Related work 3:

Community detection: https://github.com/upphiminn/jLouvain

This visualization is an implementation of one of the popular community detection algorithms, Louvian modularity for community detection. This algorithm was key in implementing the community detection for the individual character interaction force directed graph. This algorithm separates the network in communities by optimizing greedily a modularity score after trying various grouping operations on the network. By using this simple greedy approach the algorithm is computationally very efficient. In this visualization the coloring of the nodes belonging to a community is uniform. However, I decided to implement this algorithm to detect the communities and add the community data to the json for each node which already contained the faction data. I then used the

faction data to color individual nodes and community data to color the convex hull representing the communities.



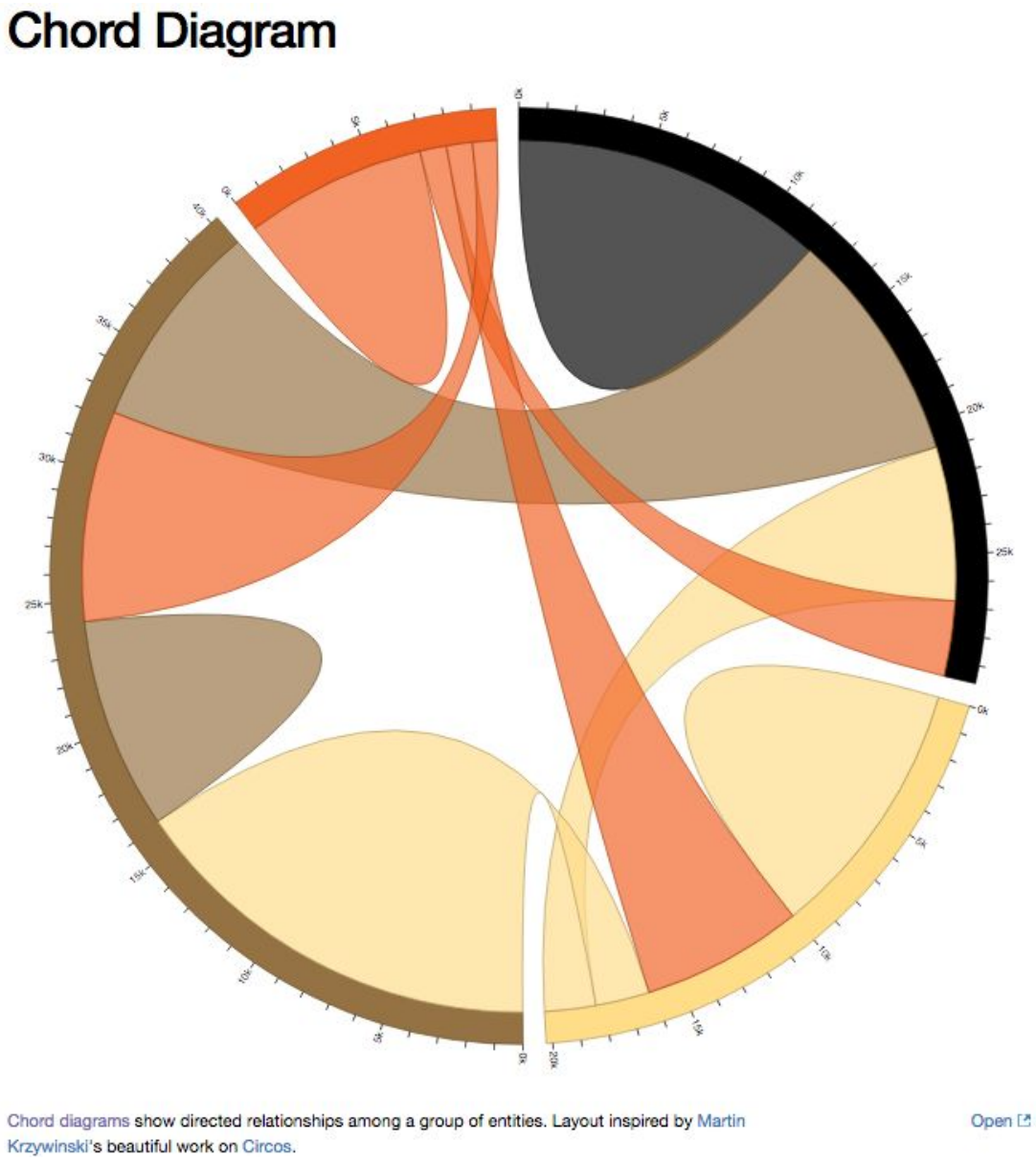
Apart from the above mentioned related work, I referred to a number of tutorials for implementing fisheye, constraining the x and y axis of the nodes, tooltips and onclick for the nodes to highlight its nearest neighbors. These added features not only made the visualization more appealing but also helped in engaging the user with those interactions.

LYNELL:

Related work 1:

Chord Diagram: https://bl.ocks.org/mbostock/4062006

My first introduction to chord diagrams came with the above graph by Mike Bostock. The chord diagram as shown in the link above and many others available on the internet use a number scale or a quantity on the arc making the reading of the graph itself easier.



However, our vision for this assignment was to have families depicted by an arc with a

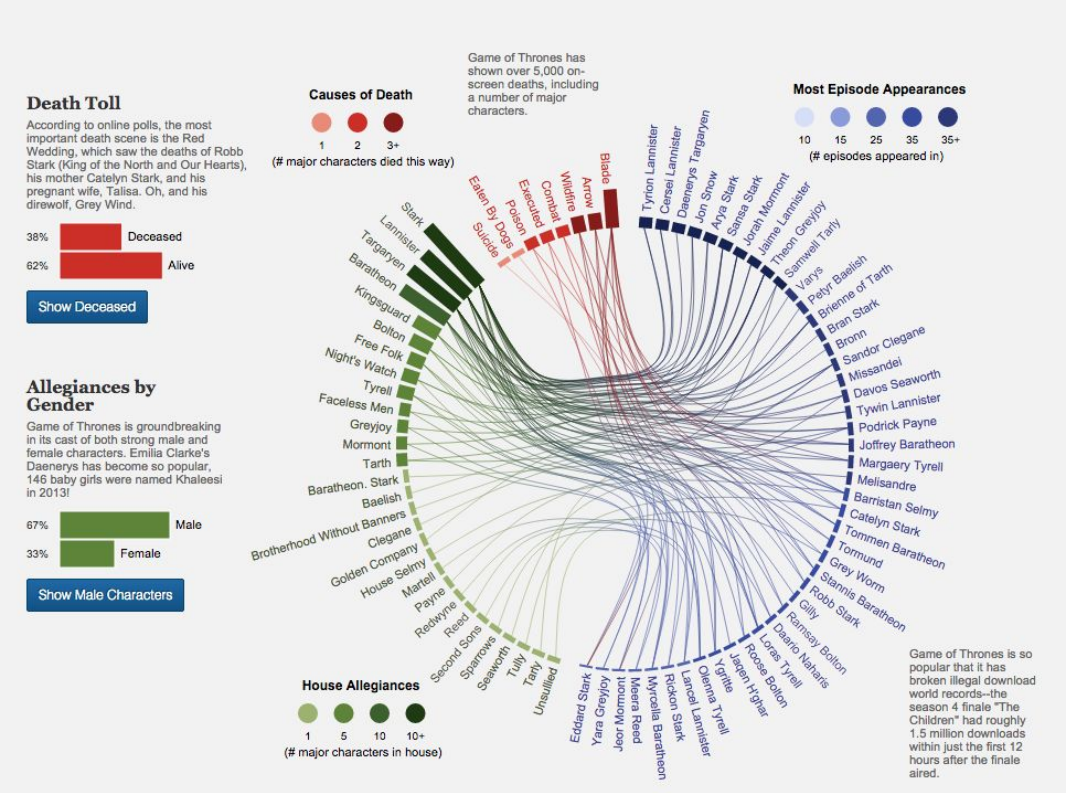
specific color. Having a specific color for a given arc allows for the user to easily identify the direction or flow. The graph was designed to show families interacting with one another, where the colors for the families needed to be consistent throughout the webpage, since it allows for an easier and quicker understanding if this information was retained in working memory.

A major challenge in working with chord diagrams is the amount of information contained within the chord. In order to overcome this and allow the user to focus on certain aspects of the chord information at a time, the fade on hover feature was implemented. The inspiration for this was taken from , https://strongriley.github.io/d3/ex/chord.html Although the two pieces look the same on first glance the second allows for a hover over an arc, which results in all other ribbons in the chord to fade out, allowing the user a clean view of the data within.

Related work 2:

Game of Thrones visualization: http://thronesviz.github.io/

On first glance this display is visually appealing and interesting, as it takes a completely different approach to a chord diagram. The datapoints are disjointed but arranged in position and color to give the impression of an arc like formation. A smart use of Gestalt's principle of closure and similarity. However, on closer inspection the graph contains so much information almost confusing the user with an overwhelming amount of data. Each arc itself is different kind of data making it difficult to understand unless the user spends a good amount of time analysing what the graph has to say.



One way the graph overcomes this overload of information is by providing filters on the page that help filter the data, allowing the user to focus on a specific aspect of the graph as opposed to the whole graph at the same time. This helped us understand how filtering to show relevant data at a time can simplify a visualization making it more appealing while retaining the attention of the user. With this in mind we implemented a filter for this graph to display data one season at a time. This not only solved the problem of 'overload of information', but also helps the user understand a flow of how the show progressed over seasons.

Related work 3:

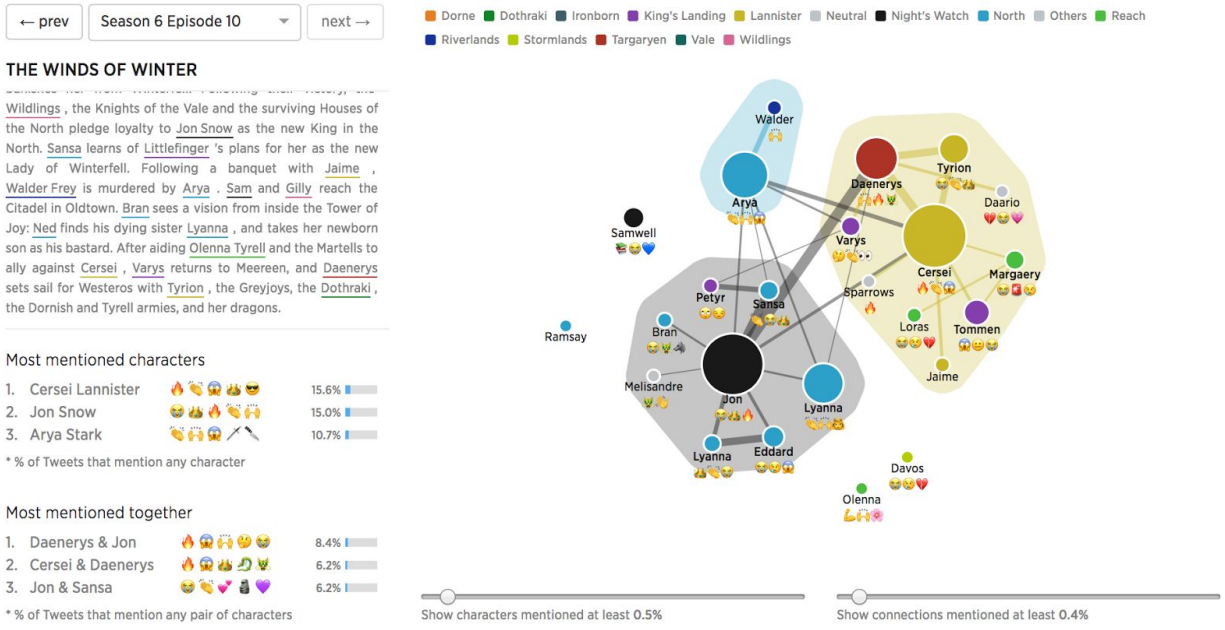
Imports chord diagram : http://mbostock.github.io/d3/talk/20111116/bundle.html

The graph shown below has a single ribbon for each value of the data , for example each company is represented by its own ribbon in the chord. However, this diagram has no means of displaying how a group of companies can be quantified. In order to display the quantities for each ribbon on the chord, a tooltip had to be implemented not only at the chord level but also one level deeper, i.e at the ribbon level.

NIHAR:

Related work 1:

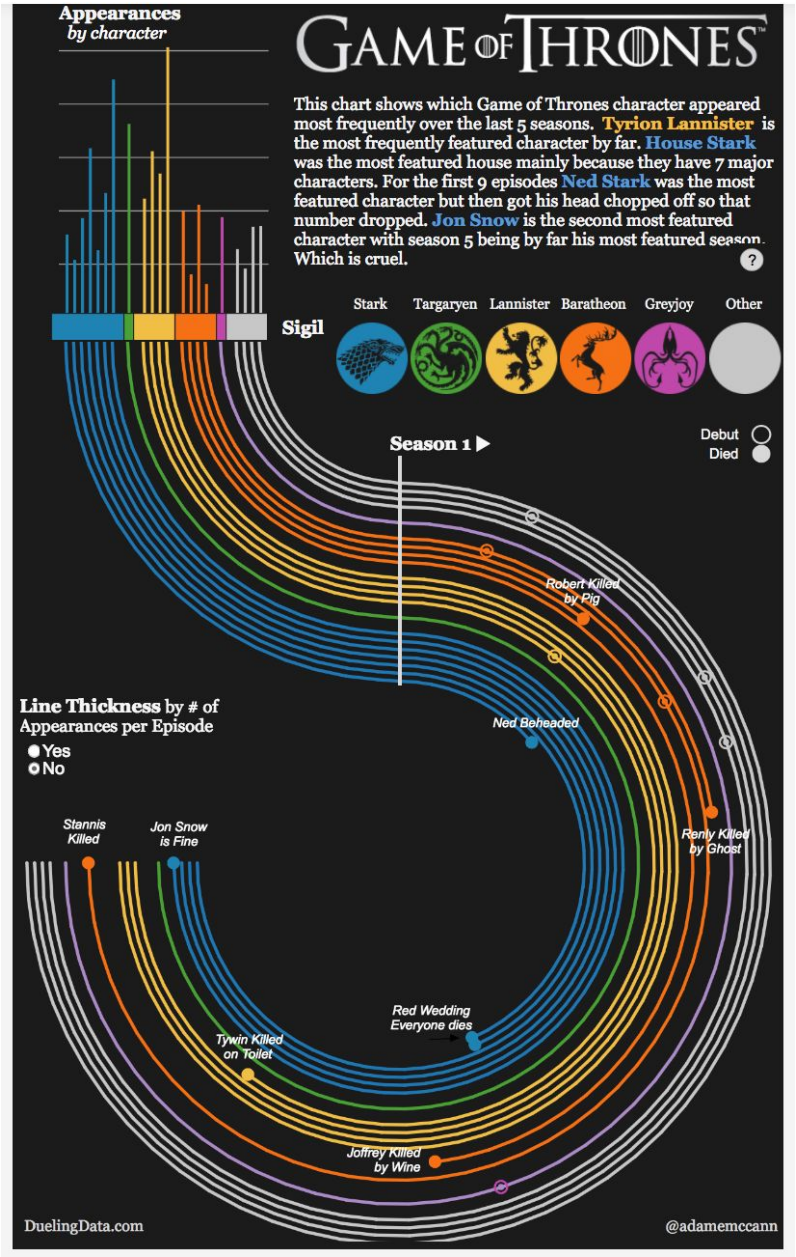
Like my teammates Shirish and Meghana, the inspiration for my work also took off from the interactive Twitter graph on Game of Thrones by Krist Wongsuphasawat (https://interactive.twitter.com/game-of-thrones). However, I will be focussing on a different part of the visualization than my colleagues. Looking at the data Krist had and the visualization he created, I particularly liked some parts. The drop down menu option to select the episode and the brief description to each episode provide the perfect background that one needs to further assess the main graph. Lastly, the forced network graph was a very interesting way of showing relationships between characters based on their Twitter mentions.



However, the question that we were trying to answer was how does the popularity of characters vary across episodes and how is the response of the people to different incidents on the show. Thus, we decided to have a similar filter and description like we observed in the visualization above. However, the main graph needed to be replaced by something more adept at showing changes in crowd reaction over seasons and episodes.

Related work 2:

The second work that inspired my final product was Adam McCann's visualization of game of thrones using Tableau (https://public.tableau.com/en-us/s/gallery/game-thrones-0).



There are multiple design choices that look great in the above visualization and have inspired our design. Firstly, the House Banners on the right of the screen which function as highlight on hover is an intuitive and aesthetically pleasing element to have. Secondly, The color scheme corresponds to those of the actual houses, and that makes following graphs intuitive for Game of Thrones fans in particular. Lastly, and more importantly, I particularly like how abnormalities in graphs are accompanied by annotations which provide context to the data point. In our graph, we plan to incorporate similar annotations to depict anomalies in the the graph behavior.

Related work 3:

A key inspiration towards the implementation of this graph came from within the School of Information itself. The Soccer Guru project (https://kpotluri.github.io/SoccerGuru/ by Keshav Potluri, Sameer Bajaj and Safei Gu) had a player information dashboard which had a feature where we could filter the dashboard by clicking on the club's crest. This seemed extremely intuitive and adds life to the graph with the presence of so many pictures.

In our visualization, we plan to have a similar filter with faces of the popular characters of the show. The user can click on one of the faces, and the entire dashboard will be updated.

DATA SOURCES

1. http://beta.wind-and-words.com/
2. Twitter Game of Thrones Character Popularity data
3. http://gameofthrones.wikia.com/wiki/Game_of_Thrones_Wiki

TOOLS

EXPLORATORY DATA ANALYSIS:

We used ipython notebooks to clean the data along with, tableau and excel to perform preliminary data analysis and cleaning on the data we gathered. The pandas library in python was used to join, filter and aggregate the data to bring it into the formats we needed for our individual visualizations.

VISUALIZATIONS: Three of our visualizations were developed using D3 and one using Tableau.

WEBSITE: The website includes HTML, CSS, JS along with Bootstrap libraries. Certain aspects of the webpage act as bridges to tie together the four visualizations and establish a clear path/journey for the user to experience.

PROCEDURE

The target users for this visualization are viewers of Game of Thrones or Game of Thrones enthusiasts. In order to achieve the above goals our web application takes the users through the following journey.

INTRODUCTION:

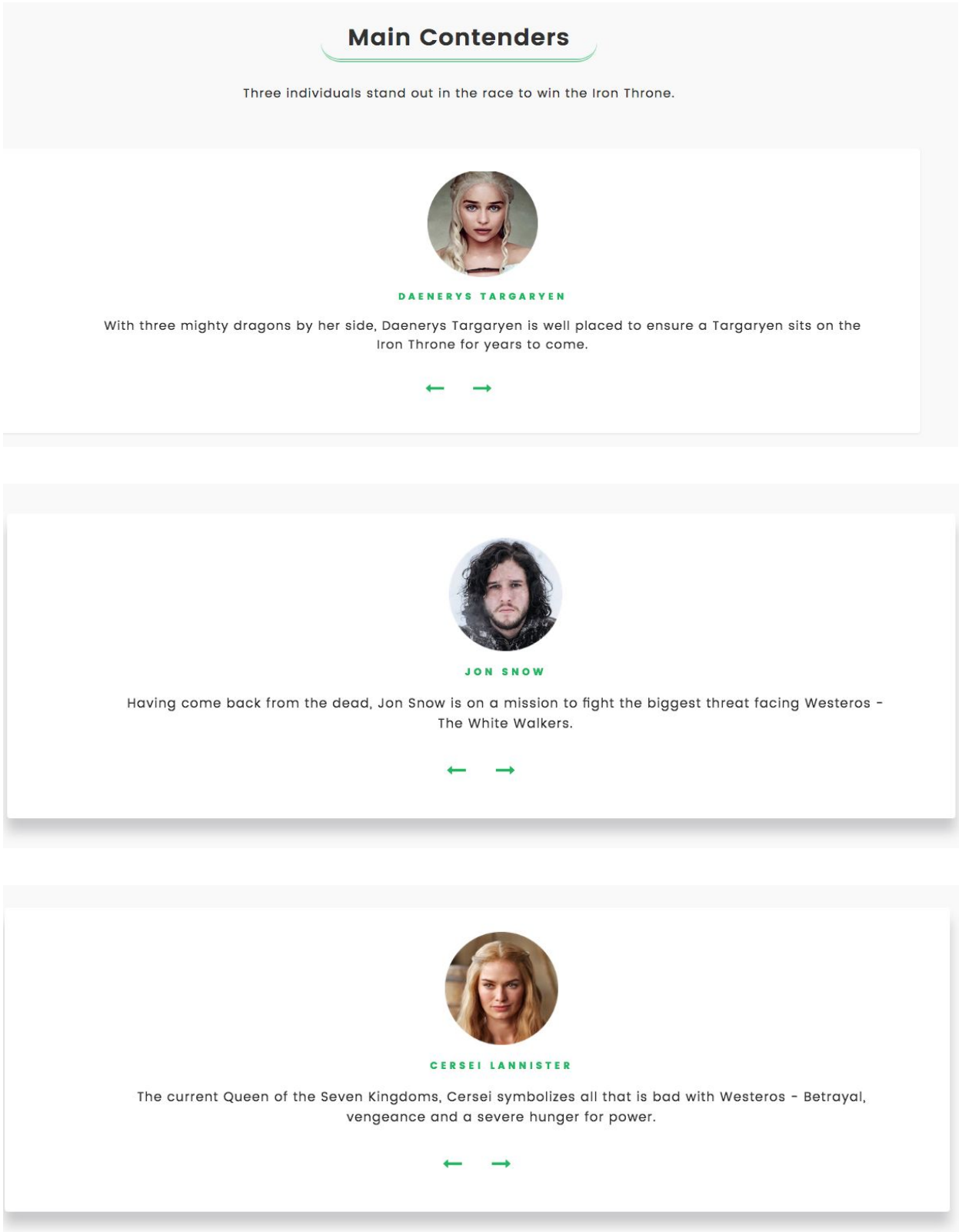
We first introduce the users to the key groups in the Game of Thrones. The first step is an introduction into the 6 most powerful factions in the world of Westeros. Each house is represented by a card that can be clicked on to highlight certain information about the house along with the key member of the house. Since the show and the plot itself is extremely complex with a very large number of characters interacting with one another, this method allows the user to take get an overall idea of the focus areas of this interactive page.

The above cards can be clicked on to view key information about each house as shown below. In order to click through each image, a navigation using arrows on the right or left was implemented. To dismiss the image and return to the page, the green cross on the top left of the image can be clicked.

KEY MEMBERS:

This section of the web page goes a step further and displays the three main contenders to the throne. Each of these characters are shown using a side scrolling animation with some key information displayed about each character. This sets the tone for the rest of the web page and also paves the way for the first visualization.

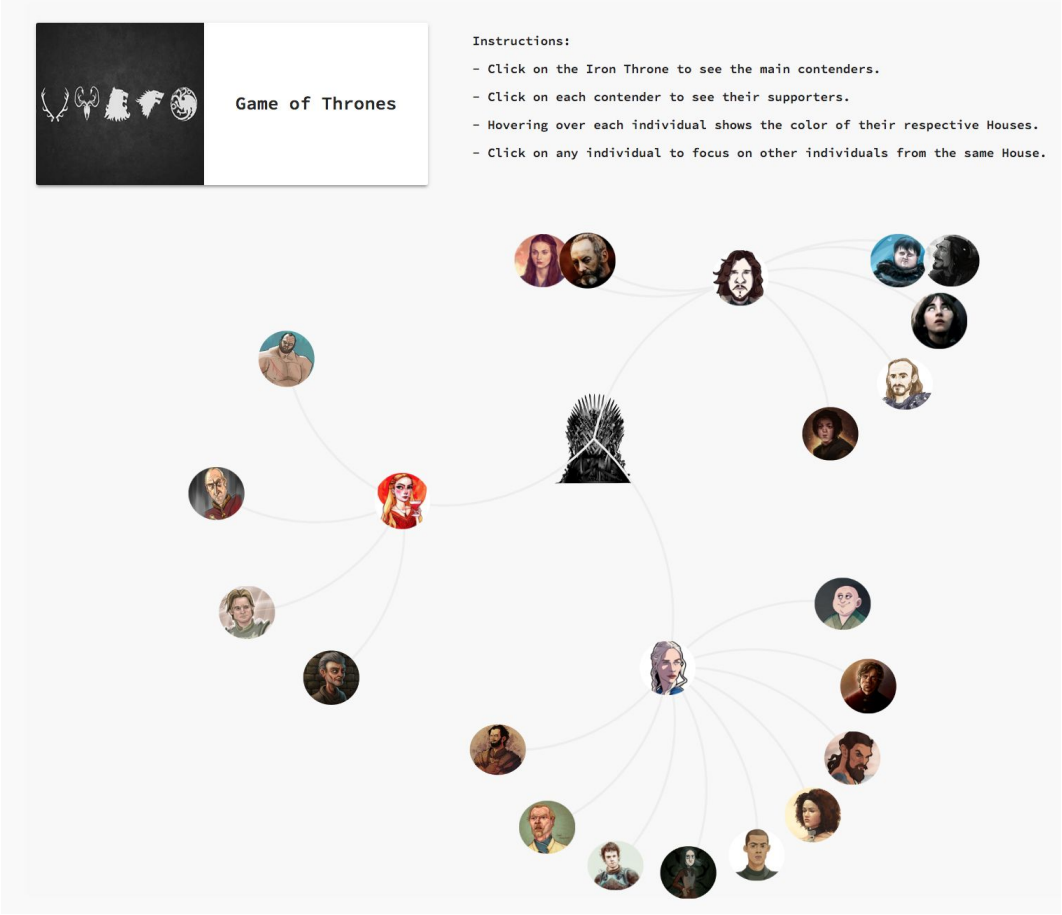
The following images display how this was achieved on the web page:



Each of these cards can be navigated to by using the green arrows depicted on the

screens above.

The ‘Current Landscape’ Visualisation



The main motivations behind this visualisation were:-

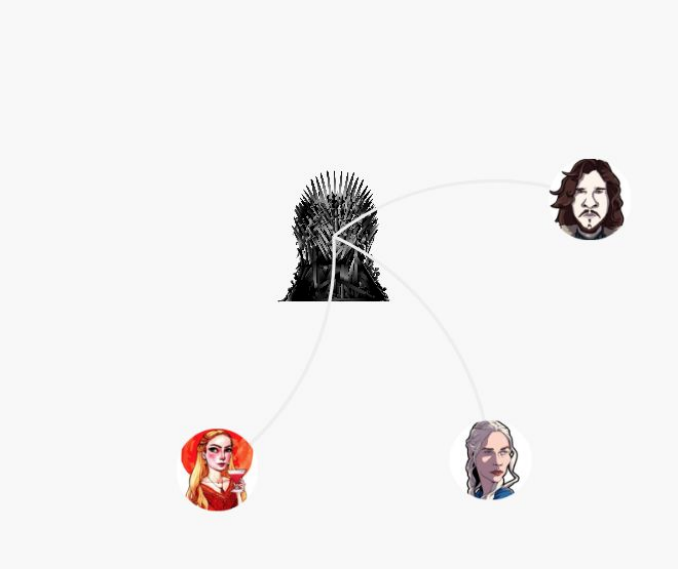
- The world of Game of Thrones is a very complex and often difficult-to-comprehend, with tens of Houses, allegiances and hundreds of characters. We decided to start off our project portal with a simplified hierarchy of the Game of Thrones world. Our research identified that while there are hundreds of characters, some select ones stand out from the rest. Jon Snow, Daenerys Targaryen and Cersei Lannister are currently the three main contenders to the Iron Throne (the main objective of the show). Most of the other main characters are allied to one of these three main characters. Thus, we decided to create an allegiance visualisation that can allow a user to explore the allegiances of different characters on the show and understand the current landscape of this

complex world.

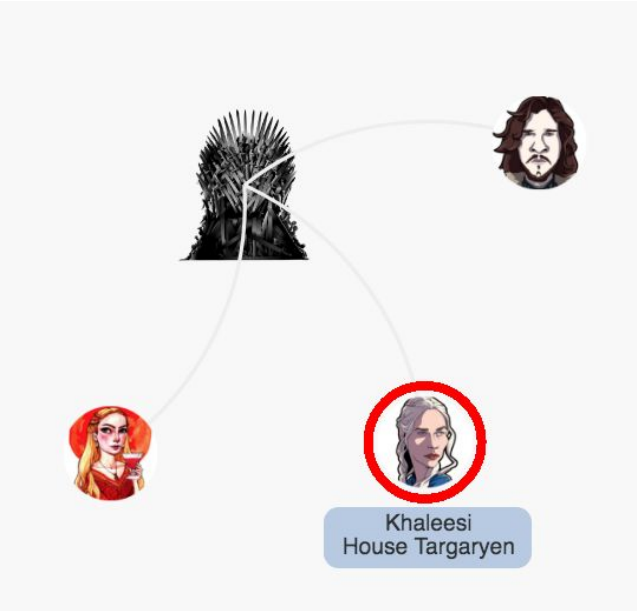
- One of the most intriguing parts of the show is the dynamics between the House someone is born in and their current allegiance. For example, Tyrion Lannister was born into House Lannister, but goes on to support Daenerys Targaryen, a direct competitor to Lannisters. We wanted an intuitive visualisation that could allow users to observe this relation between houses and allegiances. A user can click on any character and observe where the allegiance of other characters from the same House lies.

Features of the Visualisation:-

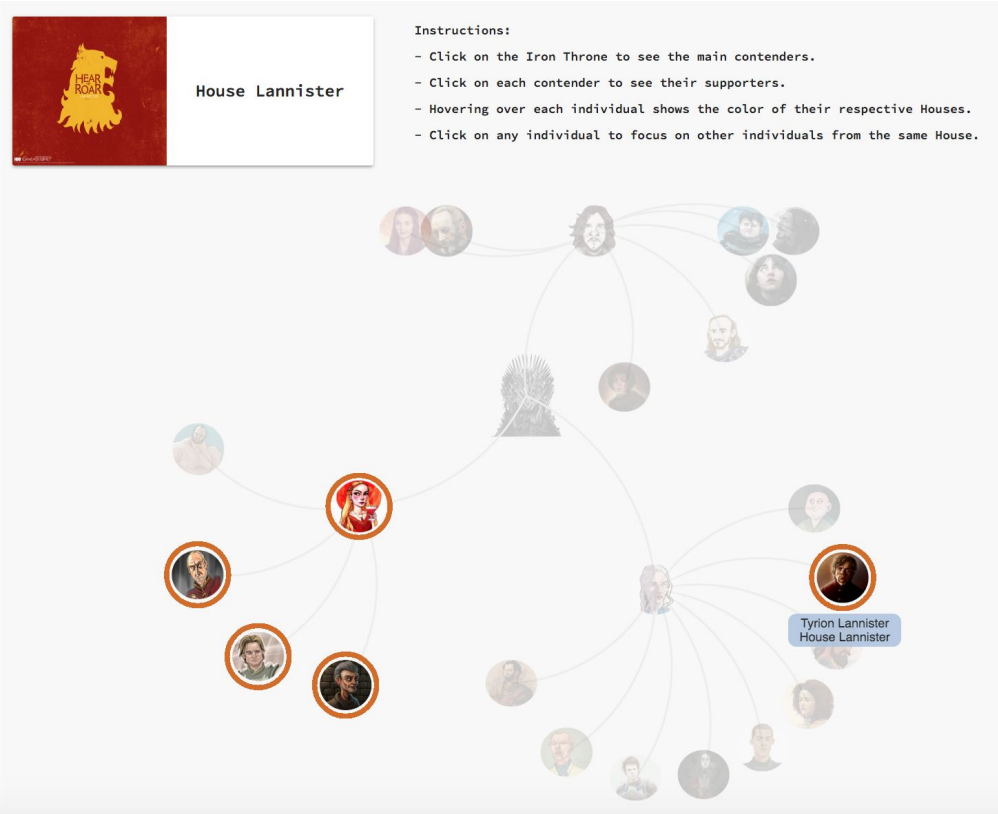
- Each node in the graph is depicted by an image of the character. This helps viewers relate to and identify the characters much faster.
- Clicking on the Iron Throne will display the three main contenders to the Throne. Clicking on each of these three main contenders will display their respective supporters in the show.



- Hovering over each character will display more information regarding that character - the name of the character, their House, as well as a colored stroke around the character displaying the color of their House.



- Clicking on any root node (supporters of the main characters) will highlight all the characters in the visualisation who hail from the same House as the clicked character. It will also ensure that the information card on the top left displays the House in consideration. This helps users to understand the patterns in allegiances for all characters in the same House. For example, clicking on a character from House Lannister highlights all the Lannisters in the visualisation. This particular instance tells us that while most Lannisters support Cersei Lannister, there is one particular Lannister - Tyrion Lannister - who supports Khaleesi.

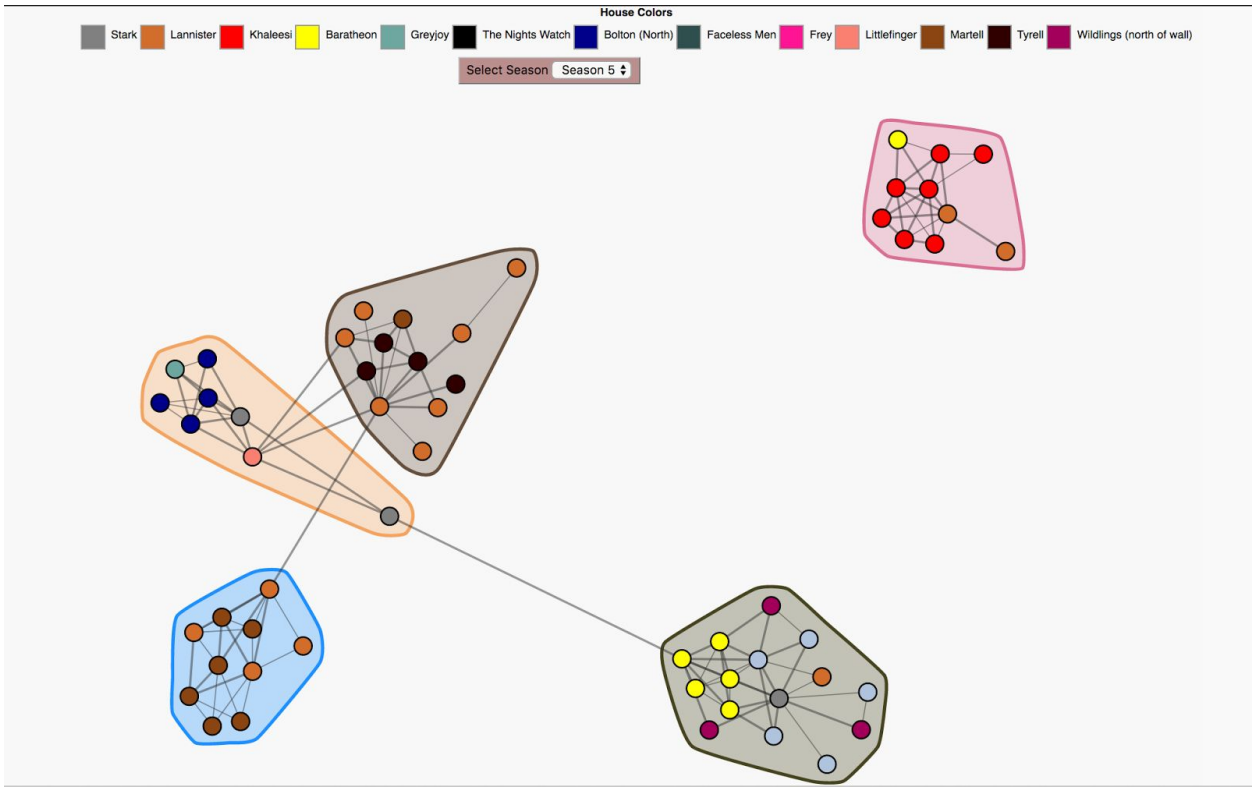


- Clicking on any root node also changes the information displayed by the card on the top left of the visualisation. Each card displays the crest of the clicked character's House as well as the name of the House.
- Instead of being a static visualisation, this visualisation is very flexible. Any node can be moved by the user to adjust the orientation based on their requirements.

INTERACTIONS

After the user has been introduced to the high level plot of the show and its contenders and supporters, we move on to talk about how characters within the show interact with one another. Our motivation for displaying interactions came through feedback from GoT enthusiasts as well as the theme of the show itself which is characters that were born into a family that interact across families forming allegiances directly impacting the plot of the show.

1. INDIVIDUAL CHARACTER INTERACTIONS



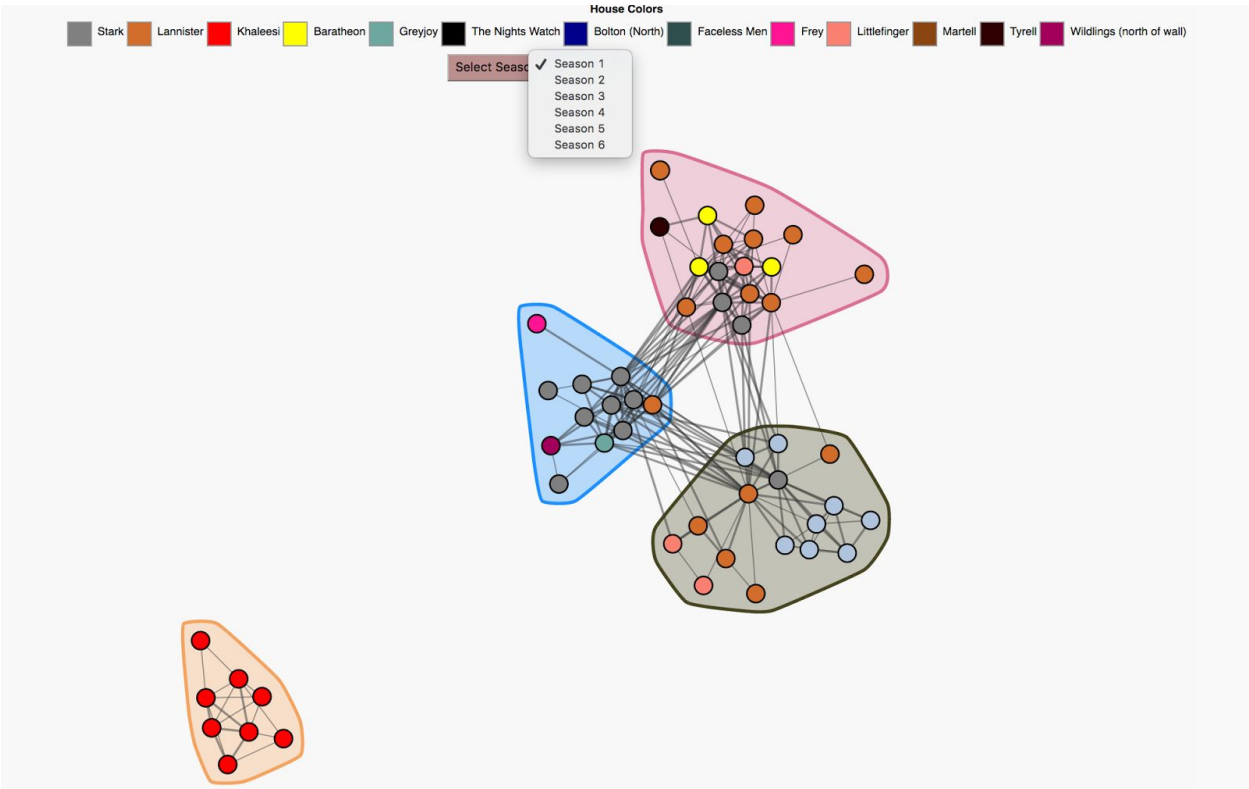
Motivation:

After visualizing a high level depiction of the main contenders of the iron throne and their supporters, the next important part of the show was to provide enriching information about individual character interactions and the depiction of diverse communities clustering characters belonging to different houses. This visualization helped in gaining a deep understanding of the story line of GoT even for a user who is not familiar with the show.

Features:

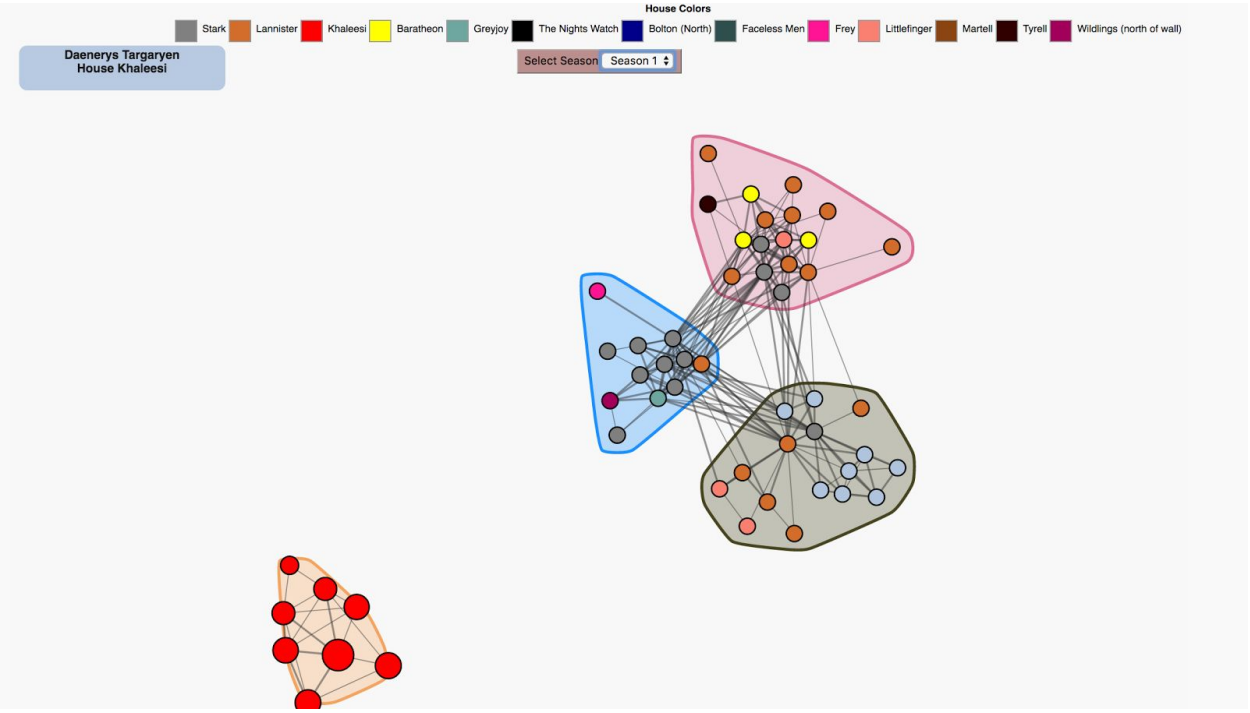
- Season wise filter:

  As shown in the visualization, the season wise filter helps in visualizing how the various communities diverged in the beginning and converged towards the end.



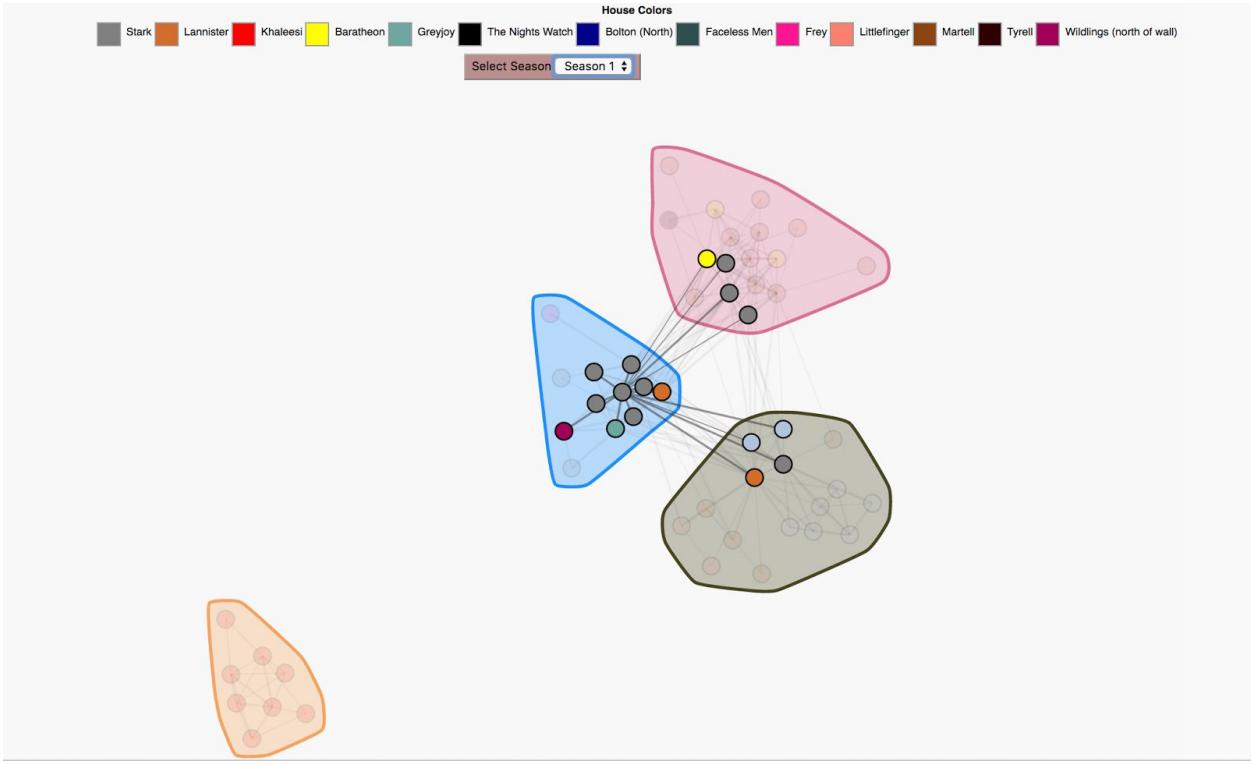
- Hover on character:

  Upon hovering on the individual nodes of the visualization, a tooltip provides the basic details about the node and also fisheye helps in enlarging the node the user is pointing to thus helping it stand out in the crowd. This feature helped in making the graph more readable and also added an animation that made it very appealing.

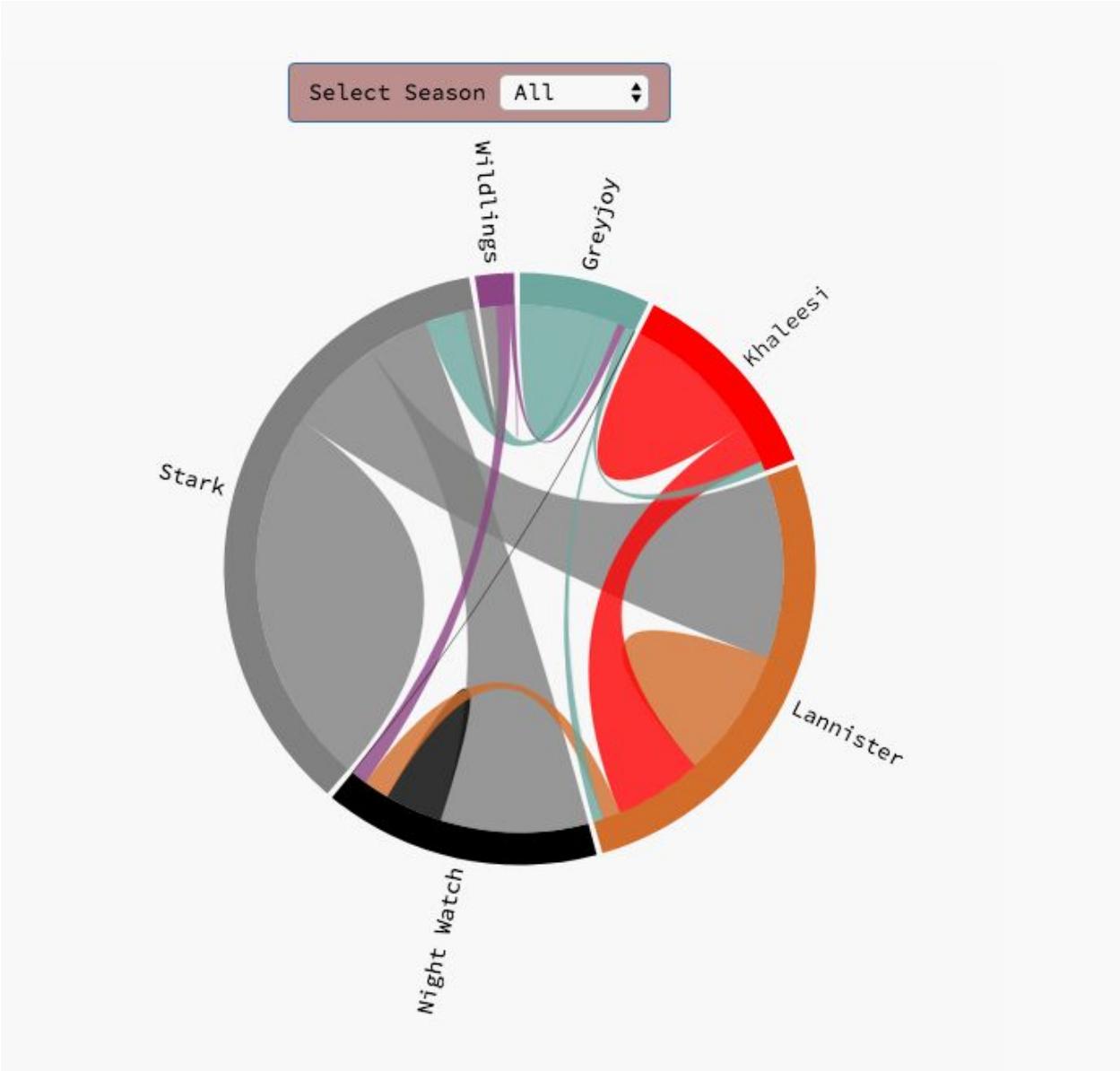


- Click on character:

  On the click of a node, all the other nodes which have direct interaction with the clicked node will be highlighted and the opacity of the rest of the visualization will be reduced. This again helped in making the details of individual interaction available to the user at a more granular level.



2. FAMILY LEVEL INTERACTIONS

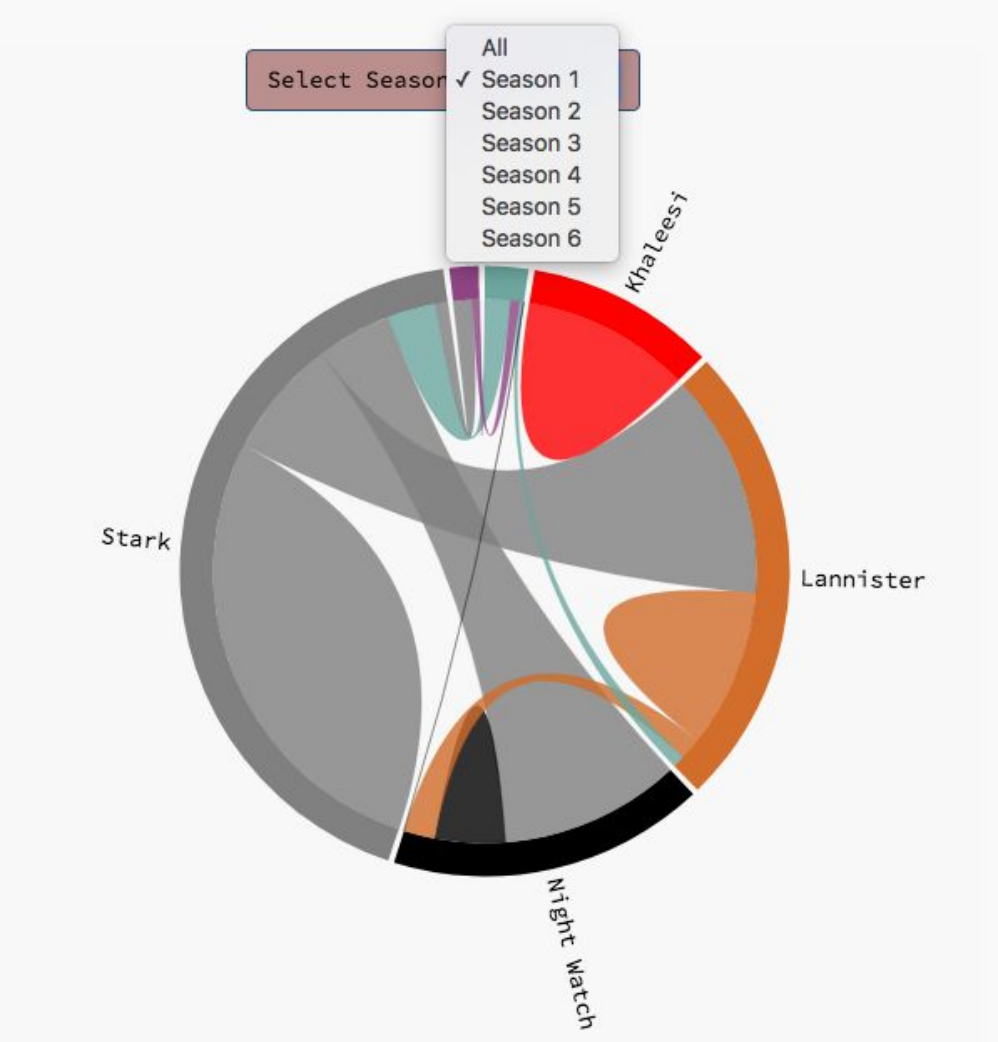


Motivation:

Following the graphs for the initial landscape and the individual character level interactions, the next most important aspect of the show was to display actual houses, the major ones and the interactions that the characters belonging to each house had across other houses. This again ties into the plot of the show where families form allegiances with other families. Our goal with this graph was to depict how that changes from season 1 to 6.

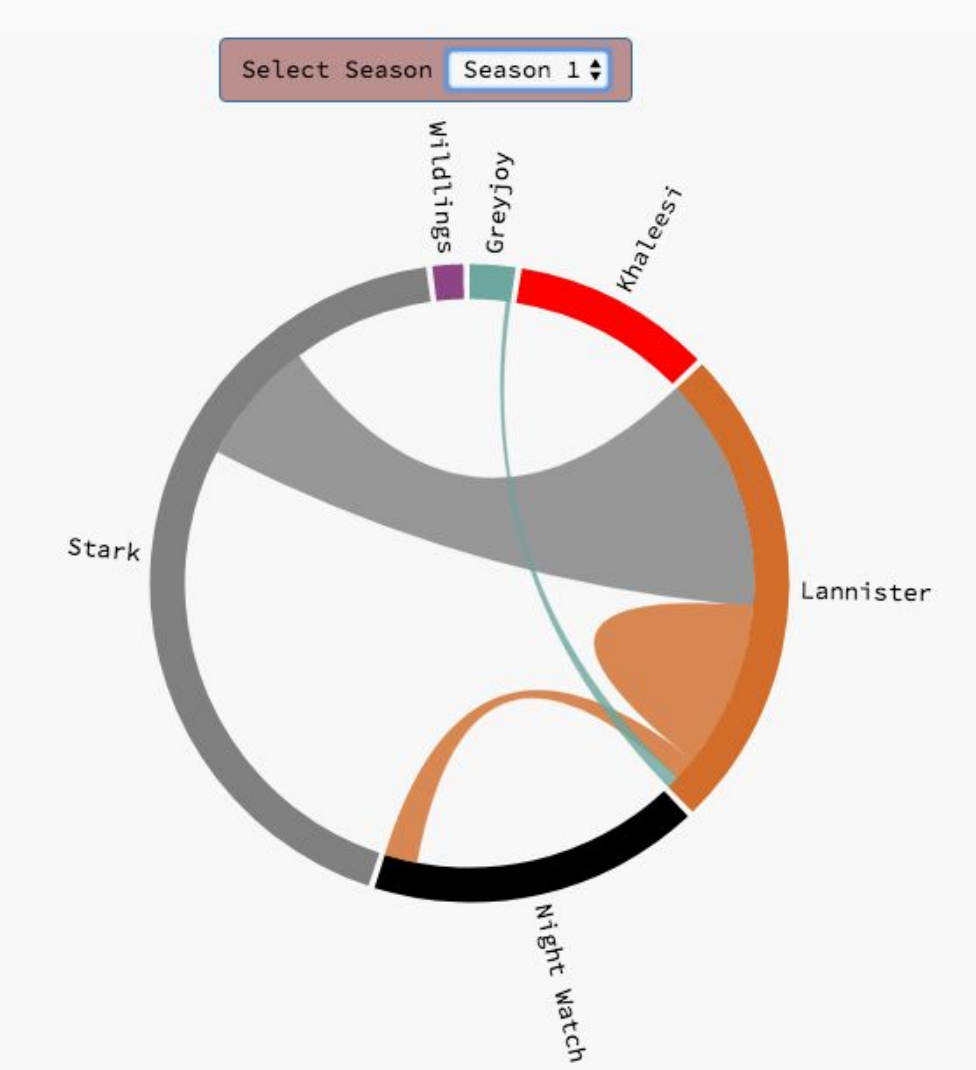
Features of the visualization:

- Season wise filter



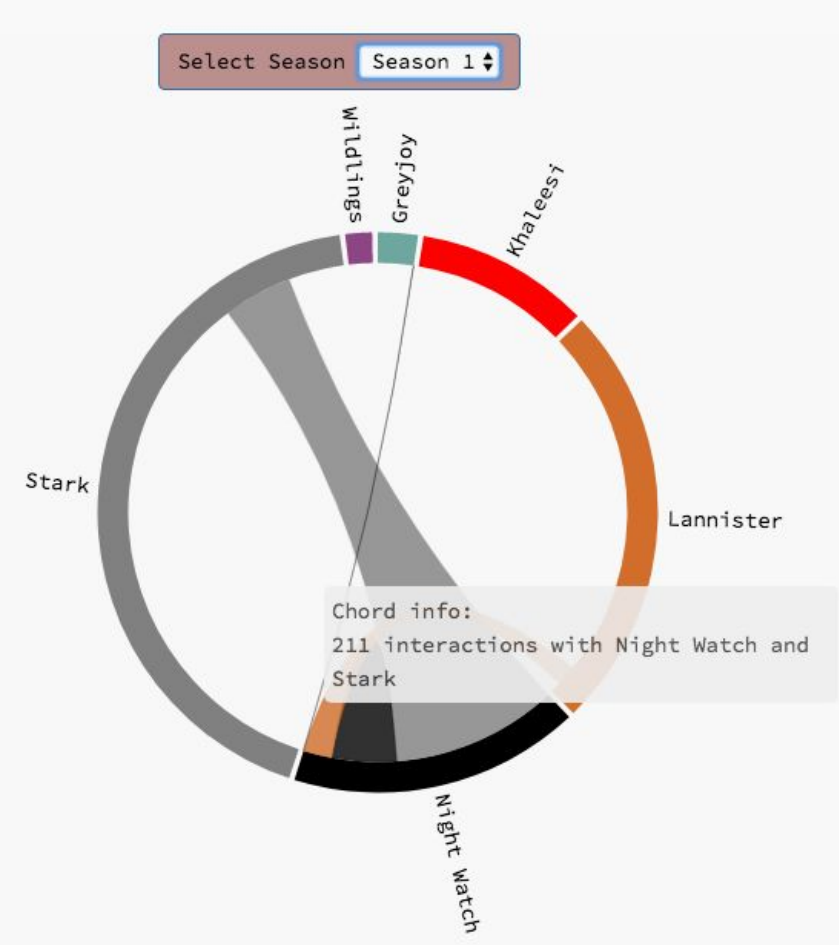
As shown above the filter allows the user to click through each season and interact with the graph one season at a time.

- Chord wise fade on hover



As seen in the graph above the “Lannister“ arc has been highlighted along with the data for each family interaction with the lannister showing while all other interactions not involving lannisters have been faded out. This helps the user get a clear picture of the data without a cluttered view

- Ribbon wise interaction across families

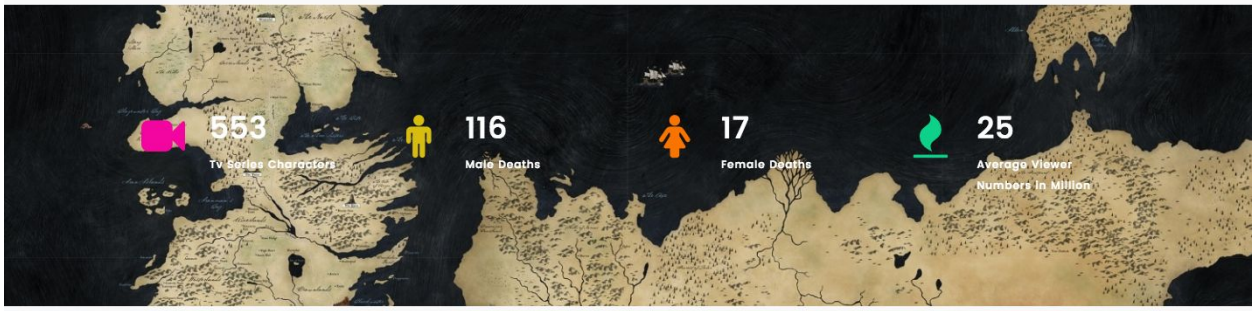


The above image displays the information that the tooltip holds. This can be viewed for each ribbon in the graph making it easier for the user to quantify the different interactions between families. The user no longer has to rely on the size or thickness of the ribbon or make guesses about how many interactions each family has had with another.

STATISTICS:

Once we've discussed the show itself along with character interactions, family and allegiances and plots, we take a step out of the show and look at the statistics from the show such as the total number of characters in the TV show, the number of male deaths and female deaths and the average viewer numbers in millions. This section of the webpage was introduced to mainly set the tone for the upcoming visualization which talks about viewer responses and how viewers reacted to events that took place in the

show.



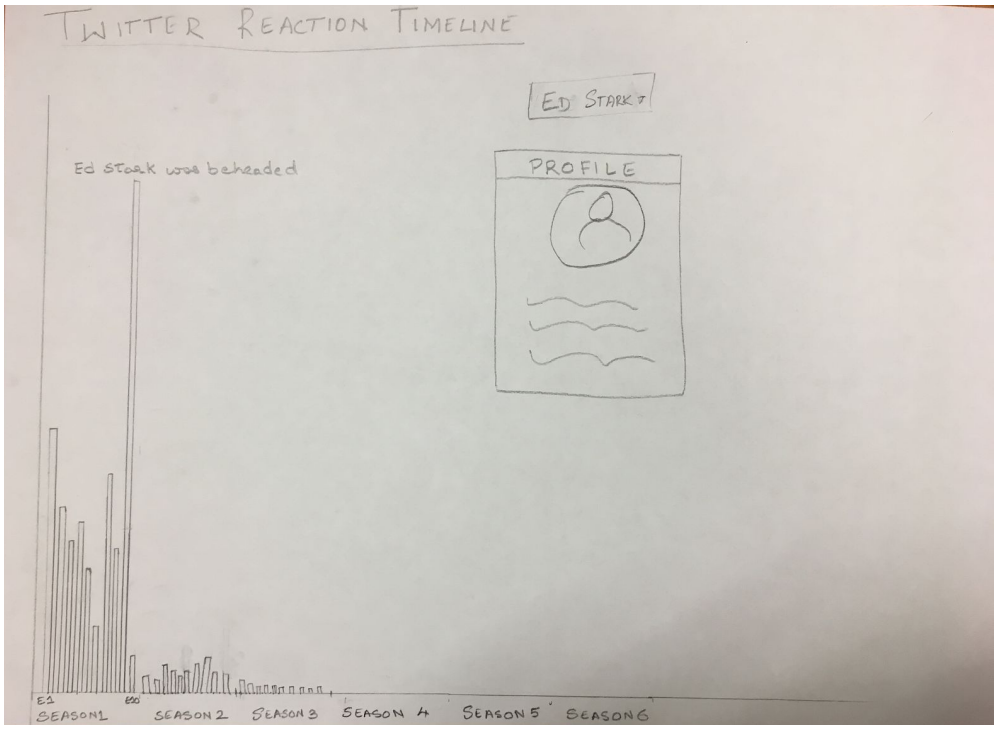
This is not an interactive section on the webpage, however to keep the users engaged, the map of the show itself was placed in the background. The numbers on the screen dynamically increment when the page is loaded.

CHARACTER POPULARITY GRAPH:

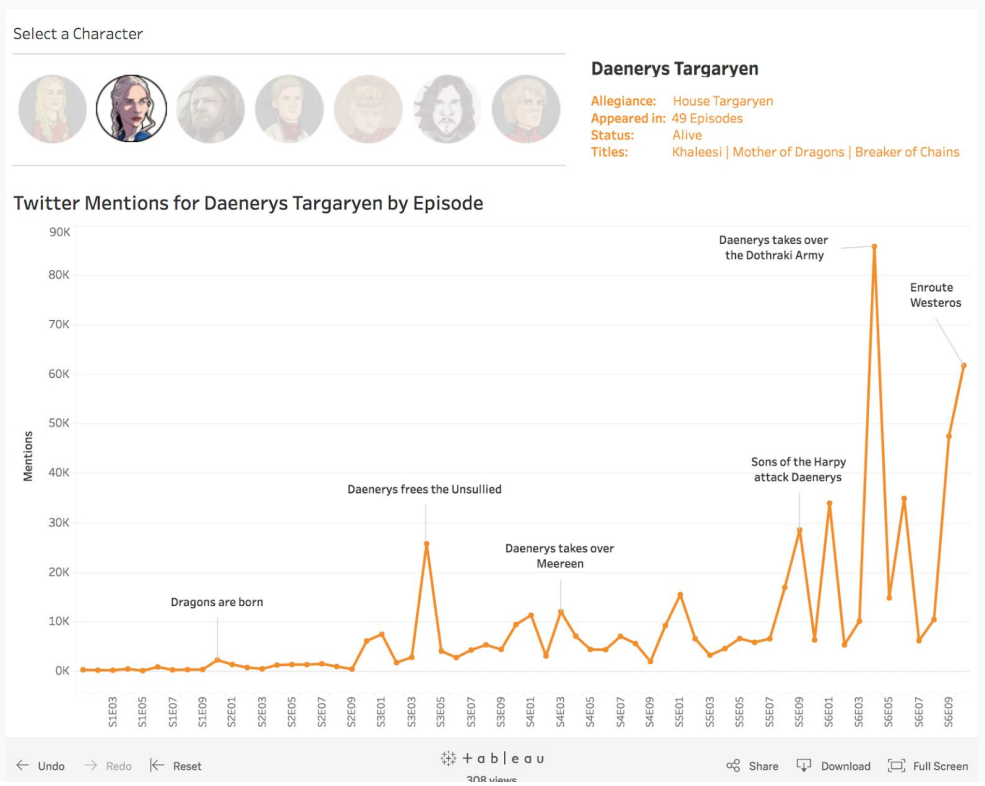
Motivation:

Up until this point, we were visualizing about characters and their relationships in the show in the inward direction. However, we also wanted to display the outward relation that the characters had with the fans of the show. We wanted to answer the question - How did the popularity of characters vary over the episodes and seasons. The popularity was depicted by the number of Tweets (in USA) which mentions the character's name within 24 hours of airing of the show. Furthermore, Game of Thrones has some major twists and turns, and we wanted to visualize how these plot twists in the TV show reflect on the audience hoping to see some spike in the number of mentions for a given character in direct correspondence to their story lines.

Based on initial research and feedback we got, the initial sketch we made looked like this:



And the final visualization looked like this:



Design Strategy:

Following were the features that were critical to our visualization:

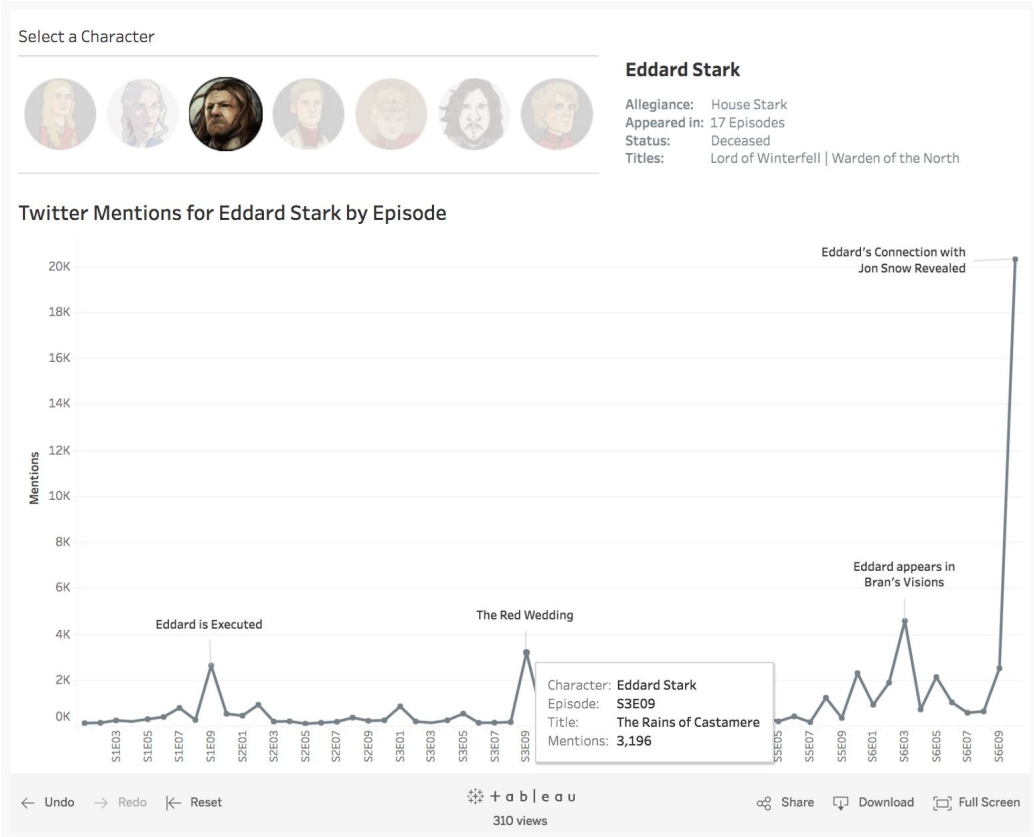
1. A filter for characters
2. A brief description of the character chosen
3. A graph showing number of tweets vs episode for the chosen character
4. Annotations explaining the major event in the show which correspond to peaks in the graph

We started off this design using D3 as we desired to show a transition in the graph where in the line chart would build from left to right episode by episode, revealing the spikes in the graph and the reasons behind those spikes. It was extremely tedious a task to get the annotations at the right spot for each of the highs in the graph, and even when we managed to do that, the graph still had bugs in it. Thus, we decided to skip the transition feature, and make the visualization using Tableau instead.

Tableau as a platform enabled us to do what we desired to. There were a few features of Tableau that we used that are not too conventional. Following are the design choices we made:

1. We used character thumbnails as filters for the graph. Since we had a limited set of characters, we felt that this provided a more intuitive and aesthetic approach to filtering
2. We initially proposed that we would use a bar graph, however, we decided to go with the line graph instead as line graphs capture time series data along with peaks and troughs more intuitively than bar graphs do
3. As proposed, we also chose to annotate the peaks for each of the characters. In order to do so, we had to mark out each data point, as each annotation is character specific and cannot be applied outside of the context of that character
4. There is also a brief description of the chosen character on the top right hand corner of the graph
5. The color chosen for each character is based on the faction they belong to, and follows the same standard as the rest of our visualization
6. We chose to have a hover functionality on the filter (to show the character name) and on the line graph to give more information about the data point

Interactions



The following are the interactions available in the graph

1. A user can click on the character thumbnails to see Twitter reaction timeline for that character
2. The thumbnails have a tooltip with the characters name
3. Once a user chooses a character, all other characters are faded away
4. The peaks in the line graphs are annotated to give a brief description of the major event that happened in that episode in relation to the chosen character
5. Each data point in the graph space has a tooltip which shows Character name, Episode number, the title name of the episode and the exact number of Twitter mentions

VALIDATION

We performed extensive user testing ensuring that our sample user base for testing was representative and hence contained ardent fans of Game of Thrones who had deep understanding of the story plot and also novice users who had not followed the show.

We particularly chose these groups of users as we wanted to ensure that our visualizations were not only for a limited target population of game of thrones enthusiasts but also those who have not followed the show. From the feedback that we received through user testing, it was evident that both the user groups found the visualizations to be highly engaging and interactive.

For user testing, we designed a set of questions to understand if the users were able to answer those with the aid of our visualizations:

1. Who were the main contenders for the iron throne?
2. Which house was Tyrion Lannister born into and whom does he support now?
3. In season 1, some of the Starks move south. Can you identify the Starks, community they belong to and the individuals that they interact with?
4. In the chord diagram, what is the significant change that you notice in the interactions of Starks and Lannisters across all 6 seasons?
5. Can you identify the peak points for the Twitter mentions for Daenerys Targaryen and what are the events associated with it?

Positive feedback:

1. The introduction to the show with a visualization showing only the main contenders of the show helped in gaining an overall picture of the show. The images of characters, instead of the plain nodes, helped users better understand the main characters of the show and their supporters.
2. The individual character interaction graph provided granular details of the show and how the plot changed in each season. Some of the GoT enthusiasts mentioned that this graph helped them recall easily the various changes in plots across different seasons. Some of the novice users mentioned that this visualization helped in understanding the formation of various communities and the details of individuals belonging to those communities.
3. The chord diagram consolidated the interactions and abstracted it to a higher level providing family level interactions. This helped in understanding which communities interacted the most and how the interactions differed across seasons. People particularly noted the dip in communications of the Stark faction over the seasons, its resurrection in the final season.
4. The tableau visualization showing the Twitter popularity of a character was very helpful in understanding the key events associated to the main characters of the show and the number of mentions after the key event. The character selection was intuitive and the annotations helped greatly in understanding the spike in

mentions.

Criticism and the incorporation of the feedback:

1. For the ‘Current Landscape’ Visualisation, based on the first iteration, we received feedback for highlighting all the characters belonging to a house with a stroke around the node instead of adding an additional node with the image of the house.
2. For the individual character interaction graph, the legend was initially not present which made it slightly difficult to identify the house colors. This was incorporated in the later iterations.
3. For the character popularity graph, users wanted to be able to compare the popularity of two characters simultaneously which we are considering to implement as a future enhancement.

CHALLENGES

1. Overall
    - The primary challenge with this project was the overly broad nature of the world of Game of Thrones. Deciding on a narrowly tailored focus for our project was an important challenge we had to accomplish. Most of the current work related to the show is centered around the deaths and wars that are so famous in the show. There was sparse research that had gone into exploring the extremely complex and intricate network dynamics between individuals as well as families in the Game of Thrones world. Character A could be born into one family, but over the course of the show, could find themselves as part of a completely different community. This is what we decided to focus on - The complex network dynamics in the Game of Thrones world - both at an individual level as well as family level.

2. The ‘Current Landscape’ allegiance visualisation:
    - In the initial prototype of the visualisation, clicking on the root nodes (the supporters) would open up another layer of nodes that depict the House of the respective supporter. After some user research and feedback, we realised that this was not an intuitive feature. Clicking on the three main

contenders opens up their supporters, but clicking on the supporters opens up their houses, which was a different type of information altogether. Thus, we pivoted in our design process and decided to display the House feature using colored backgrounds for each character on hover. The feedback from our post-change user research was very positive, and users loved that the House of each character is now encoded using color.

- The biggest challenge in this visualisation was to add the functionality wherein a user can click on a character and all the characters from the same House get highlighted, irrespective of which main character they are allied to. This feature was integral to the visualisation because it would allow users to identify allegiance patterns for each House in the show. The issue was that every time a main character was clicked and their supporters displayed, the dynamic dataset would only include the newly added characters - the supporters who just got displayed on the visualisation. This means that the scope of characters would always be local to one of the three main characters, not the entire visualisation. Thus, upon clicking any root node, only the characters who support the same main character and have the same House as the clicked node would get highlighted. The solution to this was reached by using HTML DOM elements. Instead of parsing the dynamic dataset and highlighting specific nodes from that dataset, the HTML DOM elements were used as the dataset. This means that any HTML DOM elements that are displayed on the screen were considered part of the dataset. This solution helped us highlight any characters that were displayed on screen successfully.

3. The ‘Individual-Level’ Network Visualisation

- The first challenge encountered was the cleansing of the data to obtain season level interaction details from a huge dataset consisting of interactions for all seasons. The data was analyzed in great detail as the force directed graph expects a json input for nodes and links in a particular format. Python was used to modularize the data to season wise from the huge dataset and further the json was created using the CSVs created for each season.
- One of the major challenges with this visualization was to ensure that the network diagram was not just a hairball but was modular and clearly conveyed the story of individual interactions and community formation. Force directed graphs in D3 have various properties which can make or break the visualization. The charge, gravity, linkDistance and linkStrength had to be calculated based on the x and y

coordinates of the node to ensure that nodes belonging to the same community were clustered together and at the same time different communities were separated from each other and there was no overlap. We went over many iterations with different logic for the calculation of these parameters to ensure that the visualization served its purpose.

- Another huge challenge with this visualization was dynamically identifying the communities that were formed each season based on the interactions of the different characters. Most of the visualizations that were available online usually grouped the nodes based on their color but in our visualization the community was different from the node color which represented the house the individual was born into and this was the main theme of the visualizations too. After extensive research about various modularity algorithms for community detection, Louvian modularity for community detection provided the best results wherein we were able to identify the continuous changes in the story line across seasons. To color the communities and color the convex hull, we appended each node's data with the community results that were obtained from the algorithm and assigned the convex hull color based on this result.

4. The 'Family-Level' Chord Visualisation

There were two main challenges for this graph - the data cleansing and formatting, and the second level tooltip

- For the family level interaction graph the data cleaning was implemented in python using pandas to join two separate data frames and then perform a group by on each of the filtered families to derive their interactions for each season. There was a lot of manual cleaning that had to be done for certain families and interactions making the process tedious and extremely time-consuming.
- In order to better understand what each of the ribbons in the chord diagram represented it was important to display some information/ quantity to the user about that interaction in order to make comparisons possible. Most chord diagrams online did not have this function, the fade on hover was what most chord diagrams relied on to convey information. But since this chord diagram had a non numeric scale, it was extremely important to incorporate some method to quantify these interactions.

5. The 'Character Popularity' Visualisation

- Due to the nature of our annotations, we could only display one character's

popularity at a time, else the graph space would get crowded with too many annotations. Thus, we needed to have a filter where in the user could only select one character at a time. This was a challenge while using character thumbnails as filters since it allowed selection of multiple options for filters. We had to dig deep into Tableau filter functionality to modify the settings to let the user only choose one character at a time

- The second challenge was due to a lot of variance in the Twitter mentions data for characters. For example, Jaime Lannister's number of mentions vary from approximately 100 to 15000, whereas Jon Snow's mentions vary from 100 to 136000. Thus, getting all this data on the same scale meant that we would not be able to see the subtle differences for characters who are not as popular in the absolute sense. To tackle this problem, we chose to use dynamic Y Axis which will calibrate itself according to the highest and lowest mentions per character.

REFERENCES (Links to demos, documents, or whatever is needed to show the visualization)

1. https://interactive.twitter.com/game-of-thrones/
2. https://public.tableau.com/en-us/s/gallery/game-thrones-deaths
3. https://public.tableau.com/en-us/s/gallery/game-thrones-0
4. https://kpotluri.github.io/SoccerGuru/
5. http://bl.ocks.org/GerHobbelt/3071239
6. https://github.com/upphiminn/jLouvain
7. ttp://bl.ocks.org/eesur/be2abfb3155a38be4de4
8. http://www.coppelia.io/2014/07/an-a-to-z-of-extra-features-for-the-d3-force-layout/
9. http://mbostock.github.io/d3/talk/20111116/bundle.html
10. http://bl.ocks.org/d3noob/5141278
11. http://www.maa.org/sites/default/files/pdf/Mathhorizons/NetworkofThrones.pdf
12. https://bl.ocks.org/mbostock/4062006
13. https://strongriley.github.io/d3/ex/chord.html
14. http://thronesviz.github.io/
15. http://bl.ocks.org/d3noob/5141278

WORK ASSIGNED

	Proportion of Workload

Task	Meghana Murthy	Lynell Amanna	Shirish Dhar	Nihar Dalmia
Data Collection	-	-	50%	50%
Data Preparation and Cleansing	25%	25%	25%	25%
Website Design and Creation	-	100%	-	-
‘Current Landscape’ allegiance visualisation	-	-	100%	-
Individual-Level Force Directed Graph	90%	-	10%	-
Family-Level Chord Diagram	-	100%	-	-
Character Popularity Visualisation	-	-	-	100%