

The Words of the Lord of the Rings

NADIEH

For our very first *Data Sketches* project we chose the topic “Movies,” and we wanted to have some sort of personal connection to whatever dataset we’d end up visualizing. Starting with an open mind I went ahead and did a general search of movies to get a feeling of what might be out there.

I quickly came across budget information per movie and found my way to the OMDb API¹ and IMDb Datasets² where you can download huge files with lots of information on movies and series. Having access to such large databases seemed like a very good starting point, but I wanted to make the process more personal and relatable. So I decided to search for data on my favorite movie trilogy: *the Lord of the Rings* (LotR).

With the popularity of the movies, I was quite surprised that I couldn’t find any structured datasets about them. Thankfully, after digging through more search results and using variations of the search query “Lord of the Rings dataset” on Google, I found a fascinating dataset in a GitHub repo³ with the number of words spoken by each character in each scene, in all three extended(!) editions (see an excerpt in Figure 1.1). How amazing is that?! (/•ヮ•)/*:◦◦♦ I did a few manual checks, comparing the word count in the dataset to scripts available online, and they coincided pretty well. In this case I didn’t need a perfect match, as I was more interested in the aggregated results.

¹ OMDb API, the Open Movie Database: <http://www.omdbapi.com>

² IMDb Datasets: <https://www.imdb.com/interfaces/>

³ LotR by @jennybc: <https://github.com/jennybc/lotr>

Data

Fig.1.1

Film	Chapter	Character	Race	Words
2 The Fellowship Of The Ring	01: Prologue	Bilbo	Hobbit	4
3 The Fellowship Of The Ring	01: Prologue	Elrond	Elf	5
4 The Fellowship Of The Ring	01: Prologue	Galadriel	Elf	460
5 The Fellowship Of The Ring	02: Concerning Hobbits	Bilbo	Hobbit	214
6 The Fellowship Of The Ring	03: The Shire	Bilbo	Hobbit	70
7 The Fellowship Of The Ring	03: The Shire	Frodo	Hobbit	128
8 The Fellowship Of The Ring	03: The Shire	Gandalf	Wizard	197
9 The Fellowship Of The Ring	03: The Shire	Hobbit Kids	Hobbit	10
10 The Fellowship Of The Ring	03: The Shire	Hobbits	Hobbit	12
11 The Fellowship Of The Ring	04: Very Old Friends	Bilbo	Hobbit	339

After finding the dataset, my first thought was: "How many words did each member of the Fellowship speak at each place/scene/location?" Focusing on the nine members of the Fellowship seemed like a nice and structured way to filter the total dataset.

Although the original dataset was structured around scenes, it didn't contain information about the scenes' locations. I found the scenes to be a bit arbitrary; scenes, unlike locations, are more connected to the making of the movie versus the actual universe where the movie takes place. Therefore, I decided to manually add the location to each of the ±700 rows of data. ☺(ಠ_ಠ)º

To do this, I looked at the scripts of the extended and non-extended editions and a map of Middle-earth, all found online. These scripts sometimes mention the location where the scene takes place, but for other scenes, I relied on memory from having watched these films way too many times.

I added two columns: one with a broad location (like a kingdom) such as "Gondor," and the other consisting of a more detailed (or more precise) location, such as "Minas Tirith" (a city within Gondor). While I wasn't always able to find location information for every scene, I still managed to add detailed location information to approximately 90% of the rows in this column.

Eventually this was a labor of love and I tried my very best to add the right data to each scene.

I'm still a bit sad that meant that I had to ignore a few other wonderful characters such as Galadriel and Saruman.

All of this took place during about three hours on a lovely summer Sunday. A little dedication can go a long way!

Sketch

For my sketches, I used an iPad Pro 9.7" with an Apple Pencil that I had recently bought and was eager to try out. After having tried out a handful of apps that were recommended for drawing, I used the one that charmed me most: *Tayasui Sketches*. I found it to have the right level of options—not too little, as I felt with the app *Paper*, and not too many, such as the multitude of features in *Procreate*.

About a month before I started on this project I got an email from a potential client with a rough sketch of a chart that is known as a "chord diagram," but with extra circles in the center. A chord diagram is a type of chart that reveals flows or connections between a group of entities and can be used to show many different types of datasets, such as import/export flows between countries or how people switch between phone brands. It seemed very intriguing, and I have a fond memory of "hacking" the chord diagram for other purposes (such as turning it into a circular version of a more general flow chart to reveal how students went from their educational degree to the type of job they end up doing a year and a half after graduating). During a dataviz design brainstorm for this project, I remembered that sketch and thought something along those lines, with data also present in the center, would be a perfect way to visualize my LotR data.

I decided to place the Fellowship characters in the center with the locations spread around them in a circle. Each character would be connected to the location where they spoke and the thickness of the chord/string represented the number of words spoken by that character at that location.

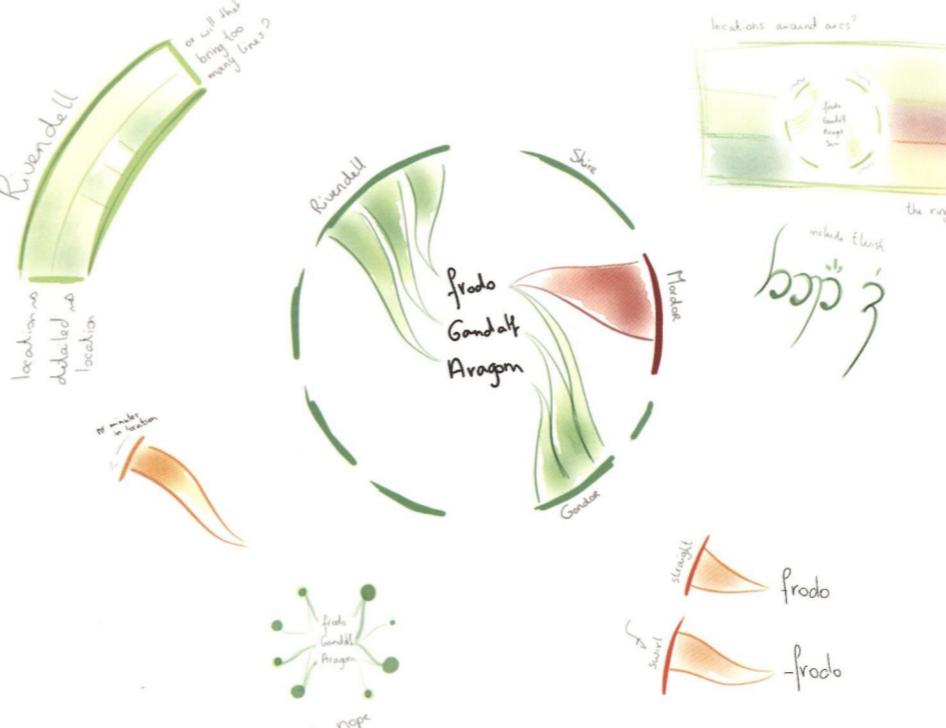


Fig.1.2

The LotR sketch with the Fellowship members in the center and locations around them in a circle.

↳ Manually Add New Variables to Your Data

As will become apparent in the chapters to come, we often had to do a lot of manual digging to get the dataset that we needed. For this LotR data one of the most important variables—location—wasn't in the base data, or in any other dataset for that matter. The location had to be gathered from unstructured sources, such as the movie scripts and from memory. Although this can take some time, don't be afraid to invest a little effort to build up exactly the dataset that you need. If you only stick with data that is already in a comma-separated values (CSV) format or other straightforward structured file type, you'll be missing out on a lot of interesting topics and variables to visualize!

Data

NADIEH

Fig.1.1

Film	Chapter	Character	Race	Words
The Fellowship Of The Ring	01: Prologue	Bilbo	Hobbit	4
The Fellowship Of The Ring	01: Prologue	Elrond	Elf	5
The Fellowship Of The Ring	01: Prologue	Galadriel	Elf	460
The Fellowship Of The Ring	02: Concerning Hobbits	Bilbo	Hobbit	214
The Fellowship Of The Ring	03: The Shire	Bilbo	Hobbit	70
The Fellowship Of The Ring	03: The Shire	Frodo	Hobbit	128
The Fellowship Of The Ring	03: The Shire	Gandalf	Wizard	197
The Fellowship Of The Ring	03: The Shire	Hobbit Kids	Hobbit	10
The Fellowship Of The Ring	03: The Shire	Hobbits	Hobbit	12
The Fellowship Of The Ring	04: Very Old Friends	Bilbo	Hobbit	339

After finding the dataset, my first thought was: "How many words did each member of the Fellowship speak at each place/scene/location?" Focusing on the nine members of the Fellowship seemed like a nice and structured way to filter the total dataset.

Although the original dataset was structured around scenes, it didn't contain information about the scenes' locations. I found the scenes to be a bit arbitrary; scenes, unlike locations, are more connected to the making of the movie versus the actual universe where the movie takes place. Therefore, I decided to manually add the location to each of the ±700 rows of data. ☺(ð_ð^)☺

To do this, I looked at the scripts of the extended and non-extended editions and a map of Middle-earth, all found online. These scripts sometimes mention the location where the scene takes place, but for other scenes, I relied on memory from having watched these films way too many times.

I added two columns: one with a broad location (like a kingdom) such as "Gondor," and the other consisting of a more detailed (or more precise) location, such as "Minas Tirith" (a city within Gondor). While I wasn't always able to find location information for every scene, I still managed to add detailed location information to approximately 90% of the rows in this column.

Eventually this was a labor of love and I tried my very best to add the right data to each scene.

↳ Manually Add New Variables to Your Data

As will become apparent in the chapters to come, we often had to do a lot of manual digging to get the dataset that we needed. For this LotR data one of the most important variables—location—wasn't in the base data, or in any other dataset for that matter. The location had to be gathered from unstructured sources, such as the movie scripts and from memory. Although this can take some time, don't be afraid to invest a little effort to build up exactly the dataset that you need. If you only stick with data that is already in a comma-separated values (CSV) format or other straightforward structured file type, you'll be missing out on a lot of interesting topics and variables to visualize!

Sketch

For my sketches, I used an iPad Pro 9.7" with an Apple Pencil that I had recently bought and was eager to try out. After having tried out a handful of apps that were recommended for drawing, I used the one that charmed me most: *Tayasui Sketches*. I found it to have the right level of options—not too little, as I felt with the app *Paper*, and not too many, such as the multitude of features in *Procreate*.

About a month before I started on this project I got an email from a potential client with a rough sketch of a chart that is known as a "chord diagram," but with extra circles in the center. A chord diagram is a type of chart that reveals flows or connections between a group of entities and can be used to show many different types of datasets, such as import/export flows between countries or how people switch between phone brands. It seemed very intriguing, and I have a fond memory of "hacking" the chord diagram for other purposes (such as turning it into a circular version of a more general flow chart to reveal how students went from their educational degree to the type of job they end up doing a year and a half after graduating). During a dataviz design brainstorm for this project, I remembered that sketch and thought something along those lines, with data also present in the center, would be a perfect way to visualize my LotR data.

I decided to place the Fellowship characters in the center with the locations spread around them in a circle. Each character would be connected to the location where they spoke and the thickness of the chord/string represented the number of words spoken by that character at that location.

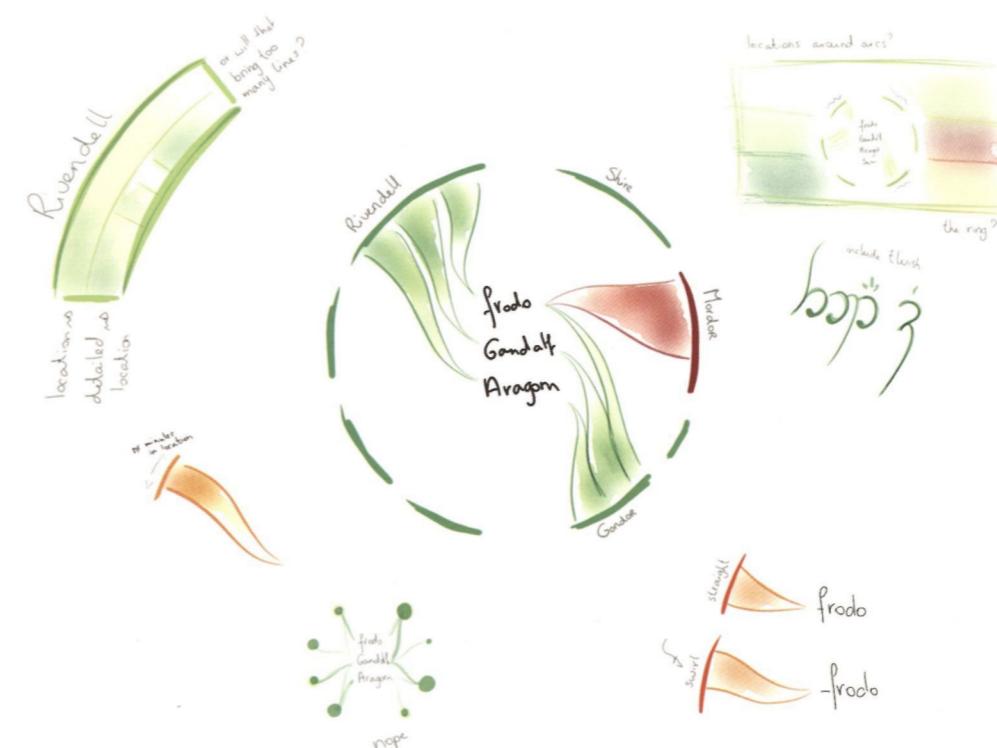


Fig.1.2

The LotR sketch with the Fellowship members in the center and locations around them in a circle.

Fig. 1.3

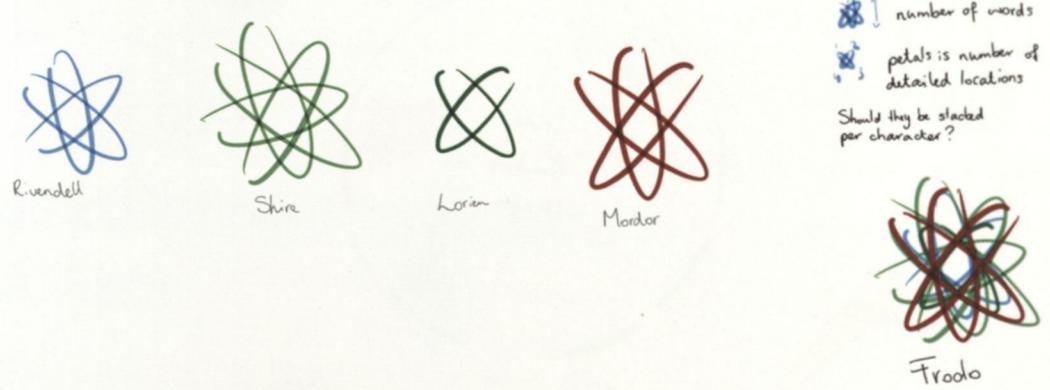
Another sketch, which places the number of words spoken on a timeline per character.



I also tried something more abstract where each location would be a spirograph, with the number of “petals” being the number of detailed locations, each sized to the total number of words spoken. But I definitely liked the chord-diagram-like concept the most.

Fig. 1.4

A final idea that turned the words spoken into spirograph-like figures.



Sketching on the iPad was quite fun. It was very easy to combine techniques, move parts around, and undo things. And even though I don't have the same amount of control that I have with a regular pen on paper, it has been part of my “dataviz repertoire” since this project.

Code

As the design that I sketched wasn't a standard “chart,” I had to figure out how to make it a reality. When this happens (which is often) I always try and find something that lies close to the idea that I've sketched and work from there. In this case, the starting point was already obvious: the chord diagram. I therefore started with the D3.js based code of a basic chord diagram.

For a short explanation of D3.js, please see “Technology & Tools” at the beginning of the book.

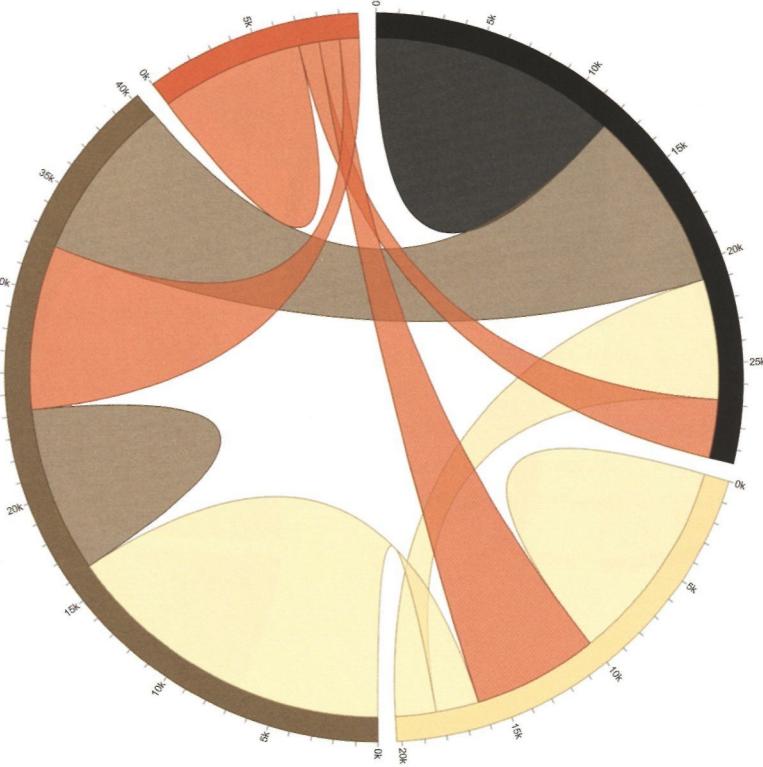


Fig. 1.5

A basic chord diagram is a chart that's often tricky to understand, but it can display a wealth of information.

This approach helps me stay flexible and possibly change a design mid-way when some difficult step isn't working out or looking as intended. For this chord diagram, the most important step was to ensure all the central chords had one end in the center of the circle. If I could pull that off, then I figured I could manage the rest of the design as well. (If I couldn't, I'd have to think of a plan B!)

I never plan out all of my steps; instead I try and focus on the most fundamental change/addition to be made at that point, see if I can make it work, and then think about the next step.

Fig. 1.6

All the chords now flowed towards the center, which created some gaps around the edges.



Fig. 1.7

The LotR data applied to the “tweaked” chord diagram.

After updating some of the formulas to calculate the sizes of the outer arcs to get rid of the gaps, the visual was ready to handle the actual LotR data (Figure 1.7).

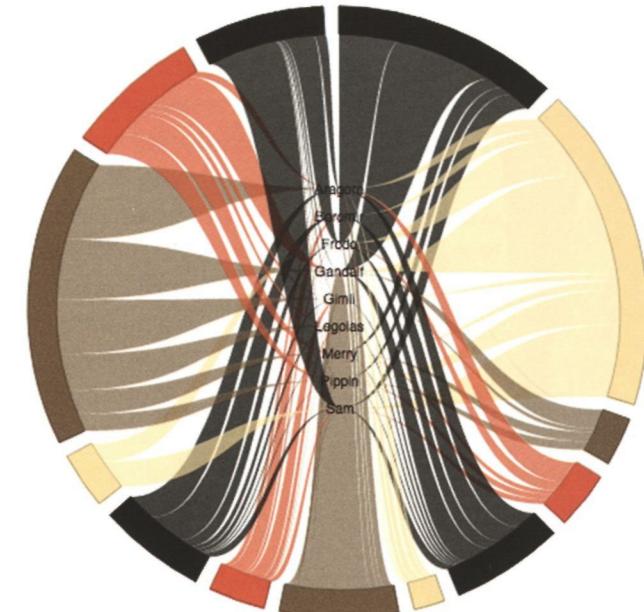
I quite liked the look of all strings flowing to the center, but I had nine members of the Fellowship to put there. I therefore placed all of the character names in the center, arranged them alphabetically (at first), and had each string flow towards the correct character in the center.

However, Figure 1.8 showed me that the inner strings looked a bit odd, especially at the very top and bottom of the visualization (such as the top-left orange strings). It just didn’t feel *natural* to me. Thankfully, these string shapes were created as scalable vector graphics (SVG) paths. These are quite flexible and allow you to create diverse types of shapes; they can be straight, circular, or curved. And as I wanted something else than the default that the chord diagram supplied, I had to finally resign myself and learn how to construct these SVG paths myself.

The chord diagram code is part of D3.js itself.

Fig. 1.8

Making sure the strings end up at the correct vertical location.



I found a great website⁴ that helped me a lot to get the hang of the more complex SVG paths. It showed one thing: a *Cubic Bézier Curve*, and I could move it around while seeing how its path “formula” changed. Using the website, I tried out different shapes and slowly built up an understanding of how to create more elegant S-shaped curves. I went through many, many tweaks of shapes, each one getting a bit closer to a natural looking curve flowing between the outer arcs and inner characters.

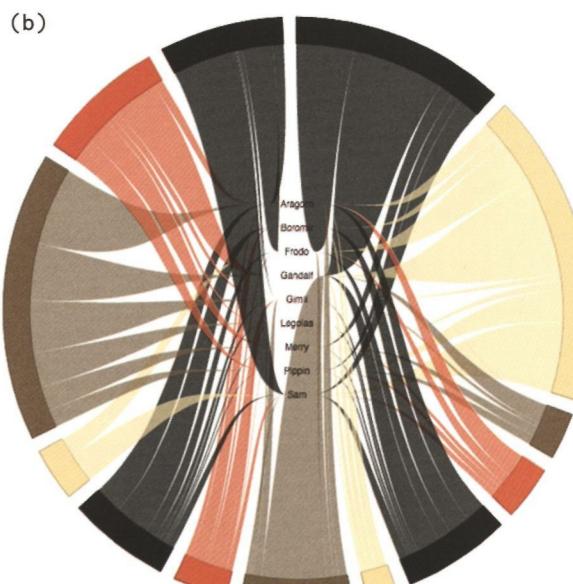
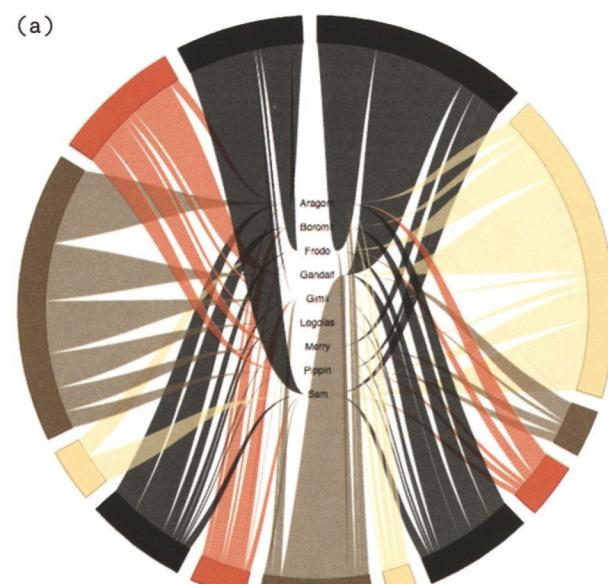
A *Cubic Bézier Curve* is a line that is defined by an end and starting point, and two “handle points” that shape the curved look of the line.

I’m switching from “chords” to “strings” to describe the inner lines from here on out.

For a short explanation of SVG, please see “Technology & Tools” at the beginning of the book.

Fig. 1.9
(a & b)

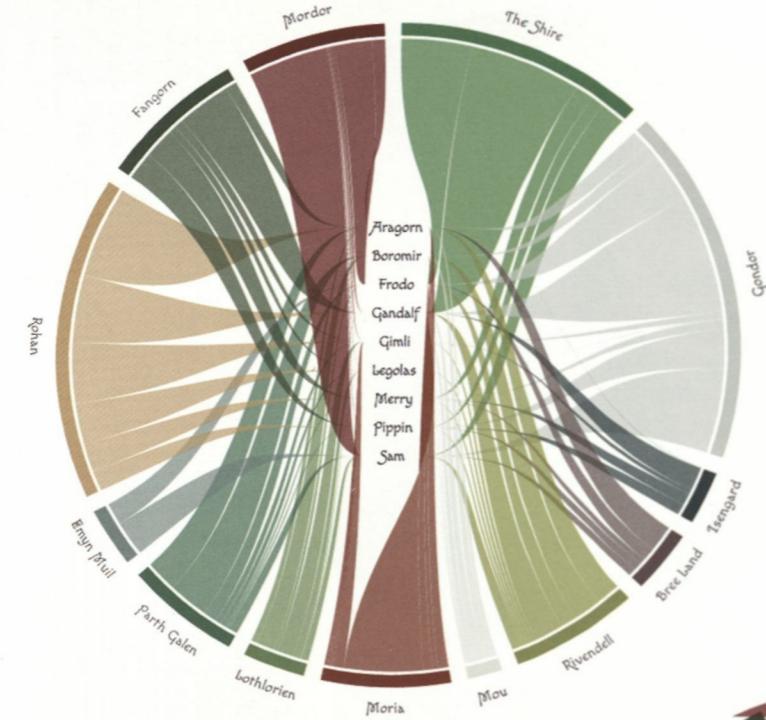
Two in-between states while I was trying to make the inner chords look more natural



⁴ SVG Cubic Bézier Curve Example: <http://blogs.sitepointstatic.com/examples/tech/svg-curves/cubic-curve.html>

Fig. 1.10

Not quite the expected result while I was trying to move the two circle halves outward.



I was now happy with the resulting shapes and moved on to updating the color scheme. I color-picked several stills of the movie locations to get a color that I felt represented each location, without being the same as another location (a challenge, as there were many green locations) (Figure 1.10).

But even with these changes, the next step that I needed to take was practically screaming at me from the screen: fix the center, which looked way too squished. There was definitely a need to create empty space above and below the inner section. This would not only fix the squished feeling, but also to give the strings in that region the room to really flow in a nice shape.

Moving the two halves outward meant I again had to wrangle those SVG paths, however, this time it was easier, because it mostly amounted to adding or subtracting a horizontal offset to the right or left half, respectively (Figure 1.11).

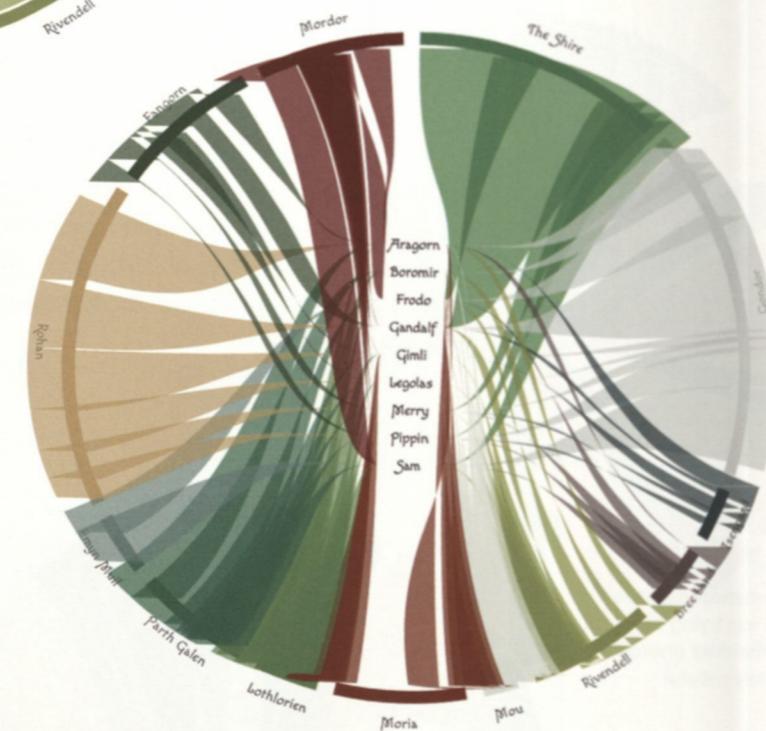


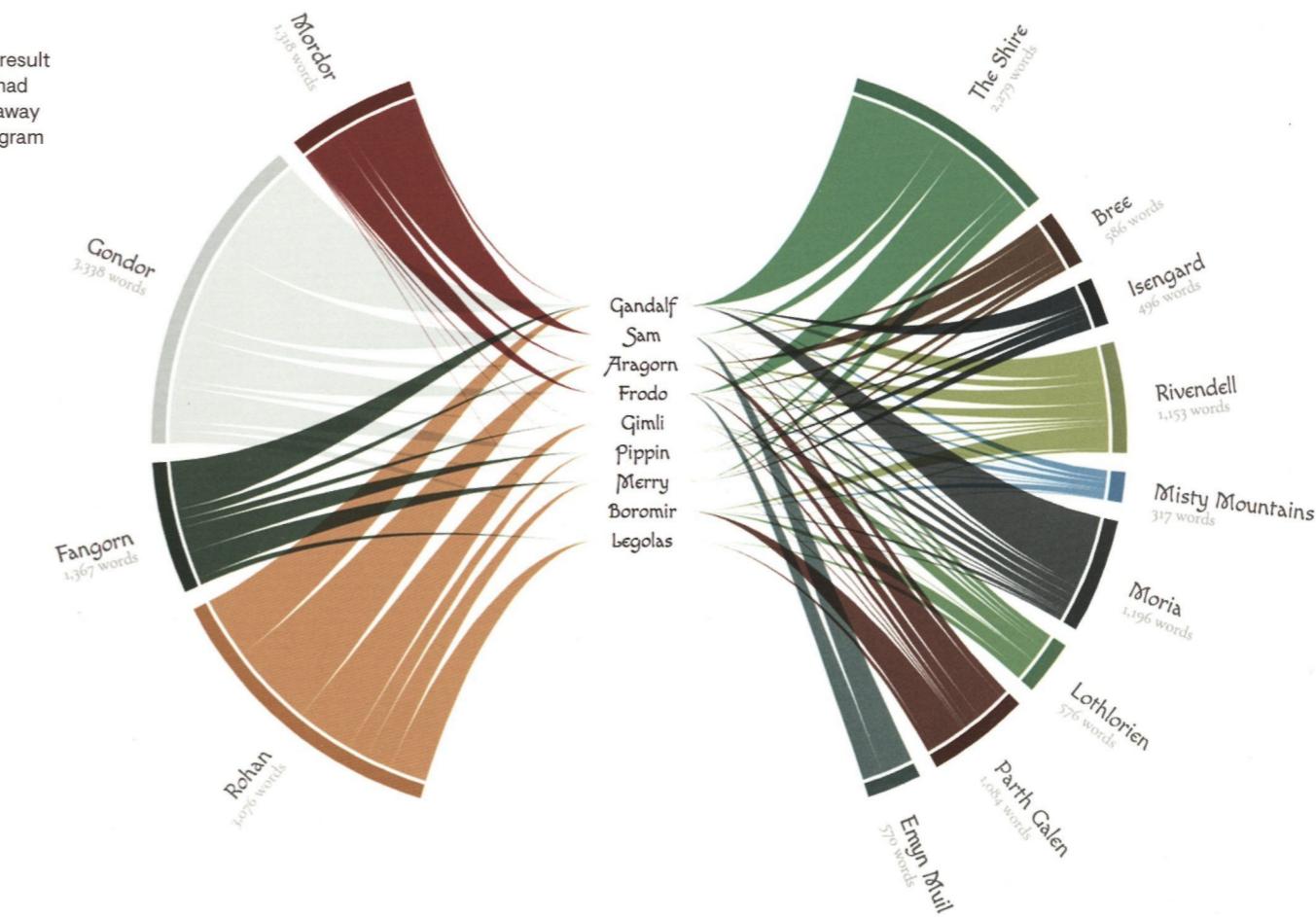
Fig. 1.11

Fig. 1.12

After I visually stretched the two halves to my liking, I ranked the broad locations clockwise from the top in the order that they first appear in the movie or where the bulk of the scenes are taking place. This thankfully divided up the two halves almost symmetrically. A few shape adjustments and color tweaks later, I ended up with something that could be its own new chart form (Figure 1.12). I even asked around on Twitter for advice on what to name it. I got back some really great suggestions, such as Ginkgo + Leafs, Butterfly + Wings, Labrys + Blades, Bowtie, even the “Eye of Sauron.” Eventually I decided to call it Loom + Strings. It seemed the most appropriate, plus it was also suggested by the creator of D3.js. (*≥▽≤)

The one thing I added at the very end of the project was interactivity. I created a simple mouse hover that gave viewers the option to inspect each location or character in more detail. Shirley offered the suggestion to completely fade out locations where a character had no spoken lines. I also implemented it vice versa; when you hover over a location, the characters without spoken lines there are dimmed. And to help the viewer understand the types of insights that you can get from the visual, I added a short paragraph with some fun insights per character that shows up when you hover over their name.

Finally, I wanted fonts that reminded me of Middle-Earth scripts. Bilbo’s handwriting and the Elvish and Dwarfish scripts are so ingrained throughout the movie, and very beautiful, that I had to use it in the visual as well. As for location names, I spent a lot of time searching for the correct translations of the different locations in Elvish (and “Moria” in Dwarfish, of course) as well as the inscription of the One Ring. (*≥▽≤) But I feel that it makes a major impact on the overall feel of the final piece, so it was thankfully worth it!



Due to more technical reasons I needed two names, one for the overall shape and one for the inner lines.

I'll never know for sure if I got the Elvish translations correct though...

↳ Remix What's Out There

NADIEH

When I design my visuals, I always use plain pen and paper (or its “digital pen” equivalent). I don’t think about remixing other people’s work. I instead try to figure out what would work well for that dataset and the question the visual has to answer. But after looking at my finished designs, I see similarities with visuals that other people have made. It’s therefore very rare that I start my JavaScript code from scratch. I typically try to find a code snippet or example that most closely resembles my design and start adjusting that. This can be as simple as the code for a bar chart or donut chart, but in this case I started from the well constructed code of a D3.js chord diagram. Having this code as the base made it *much* easier to slowly alter it until it looked like my sketched idea. I literally don’t think that I could’ve created this project without having Mike Bostock’s code, which included so many smart ways to set up the arcs and inner chords, to repurpose.

38

Reflections

It was quite a lot of fun working with a dataset about a topic that I love. I learned several new things about the film from creating the visualization, which is something I always strive for (although that’s probably true of most visuals). Also, having successfully adapted a part of D3.js code into something different gave me a real confidence boost in my understanding of JavaScript going forward! I did have to hold myself back several times throughout this project from not just laying my laptop aside to watch the whole trilogy again.

29

MOVIES

The Words of the Lord of the Rings

↳ LotR.VisualCinnamon.com

Who's speaking in Middle Earth

How many words have the members of the Fellowship spoken across Middle Earth during all 3 extended editions of the Lord of the Rings

In more than 11 hours of the LotR trilogy all characters combined speak approximately 32,000 words. The 9 members of the Fellowship alone take up about 17,000 of these words, a bit more than half. In the visualization below you can find out how many words a member has spoken at each general location throughout the trilogy.

The members have been sorted from Gandalf who had the most lines, to Legolas, who spoke even less than Boromir, even though the latter was only in one movie (and some extended scenes).

Hover over characters or locations to get a more detailed overview

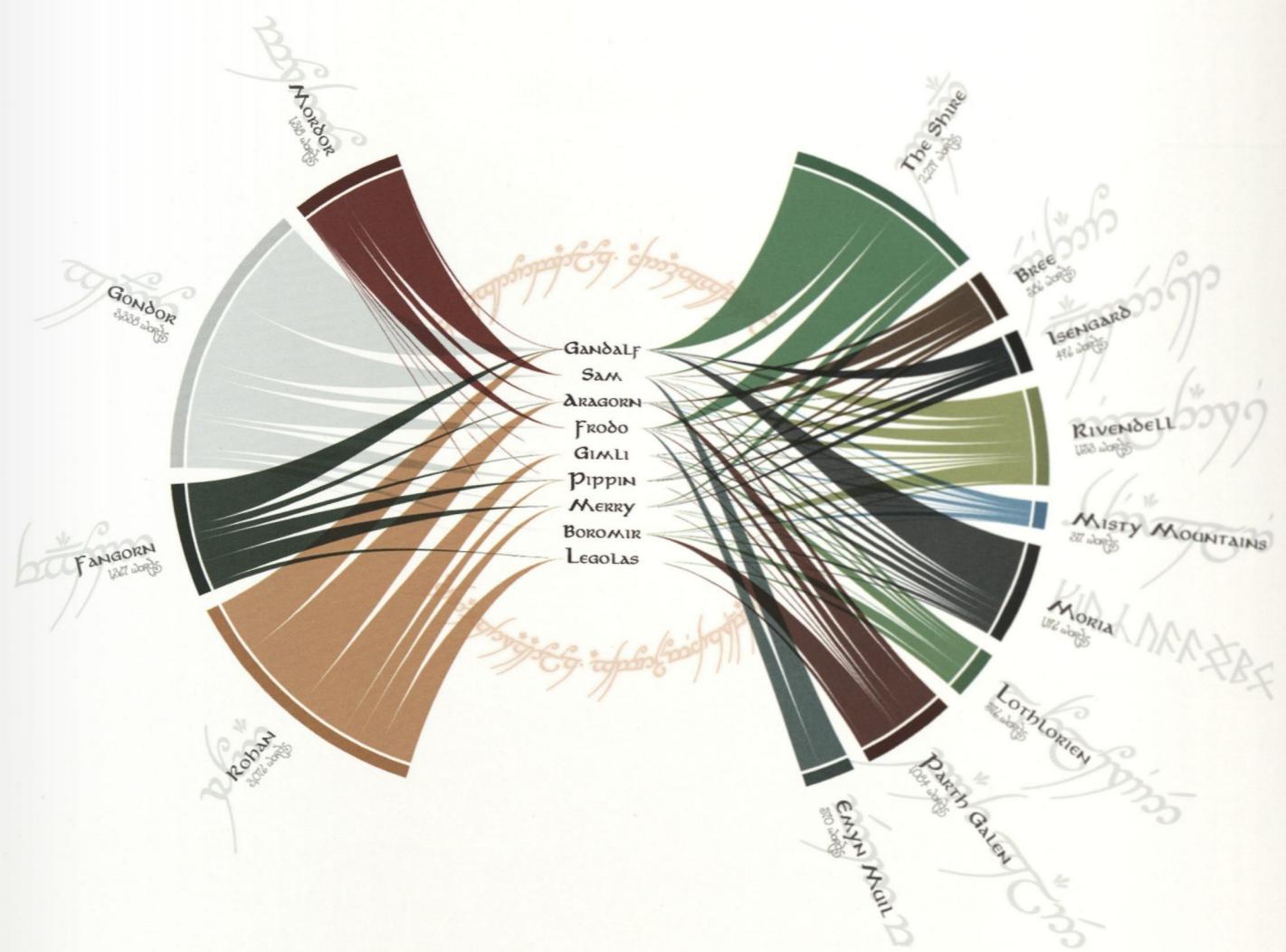


Fig.1.13

The final result of the Lord of the Rings visual with the Elvish or Dwarfish translations behind each location name, and the inscription of the One Ring in the center.

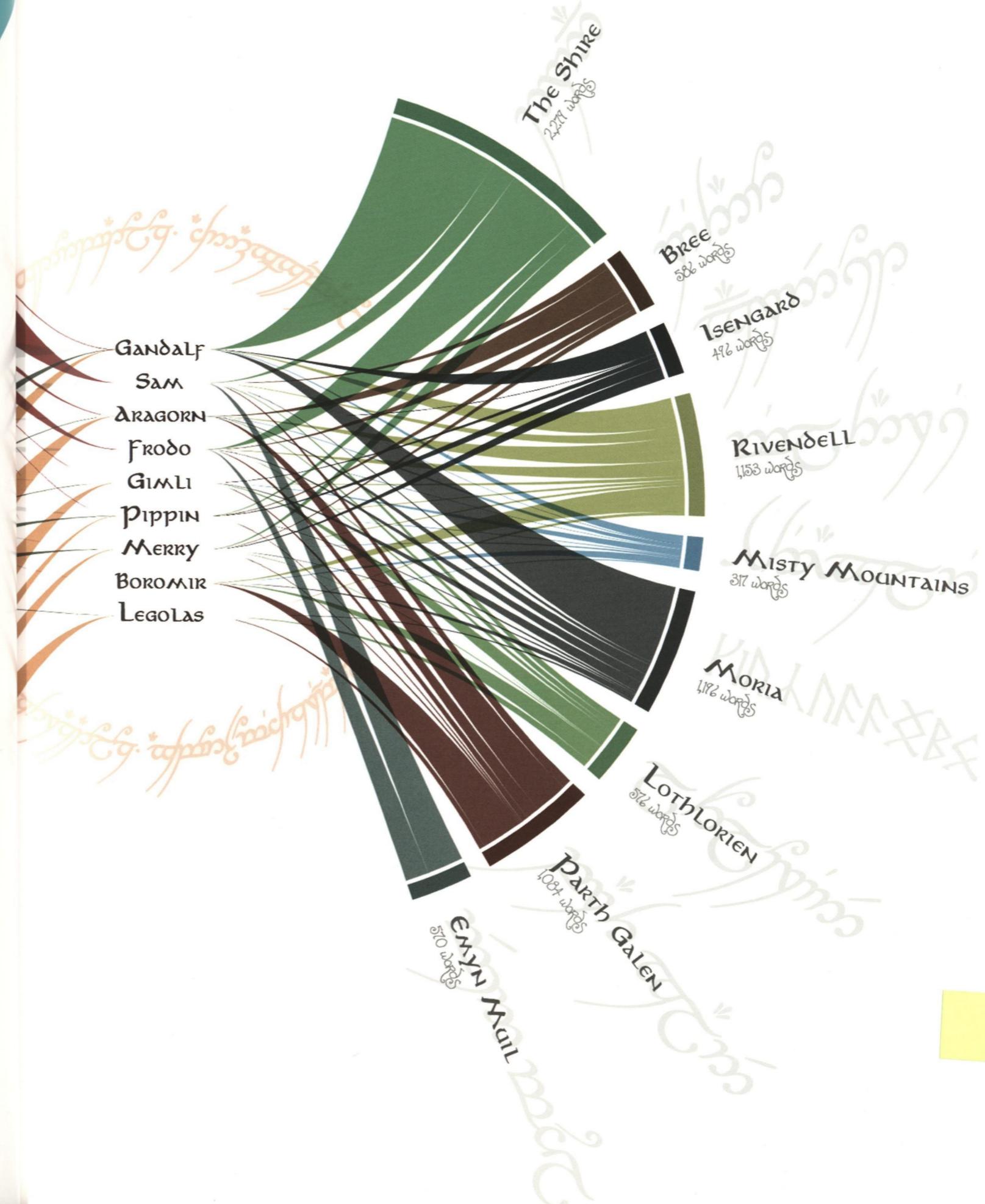
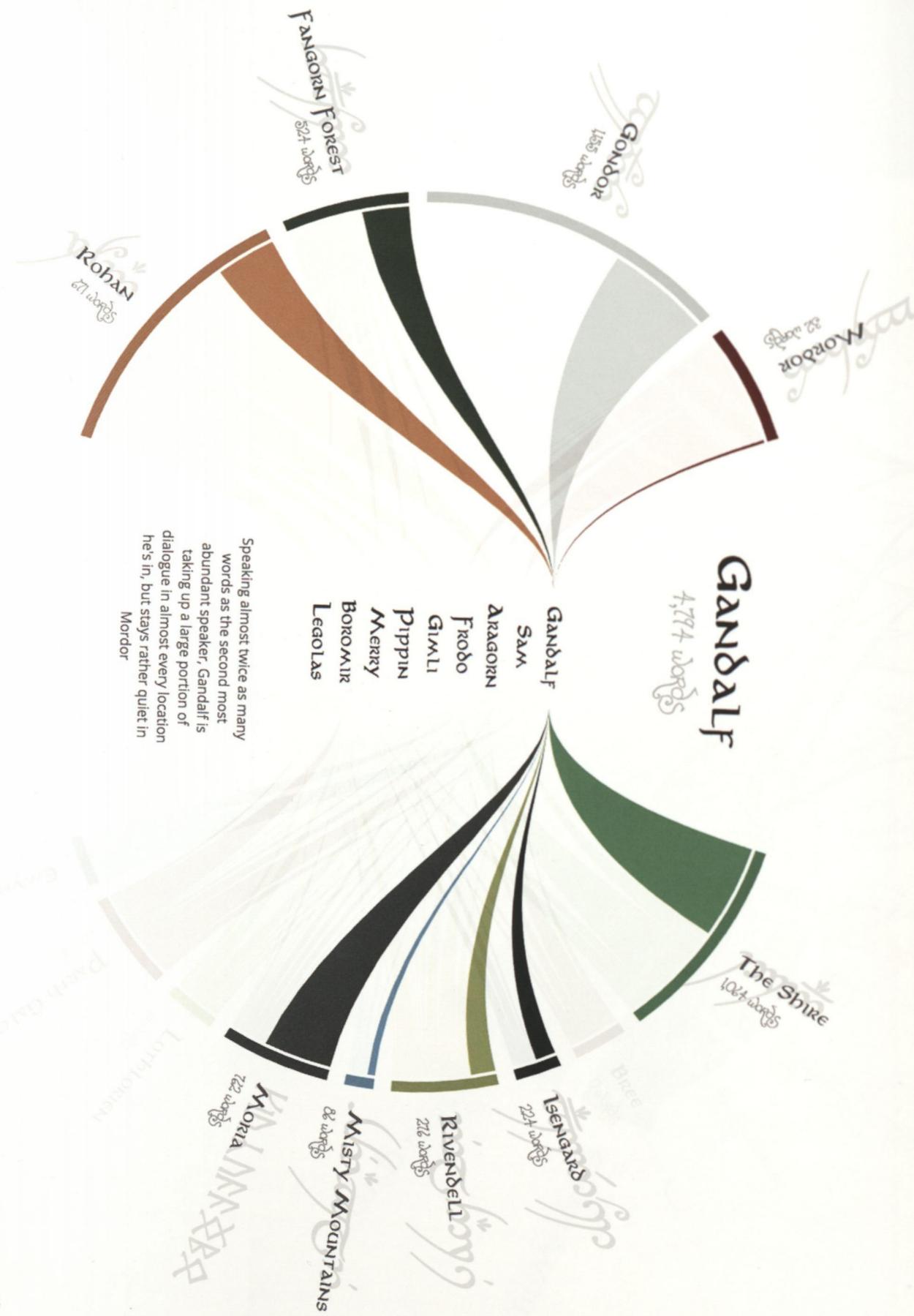


Fig.1.14

Hovering over a character's name shows their strings only, hides the locations where they had no spoken lines, and updates the outer values of words to only theirs.



Speaking almost twice as many words as the second most abundant speaker, Gandalf is taking up a large portion of dialogue in almost every location he's in, but stays rather quiet in Mordor



Fig.1.15

Hovering over a location reveals all the Fellowship members that spoke there. With the section in the Shire happening (almost) exclusively before the Fellowship is formed, only the four hobbits and Gandalf have lines there.

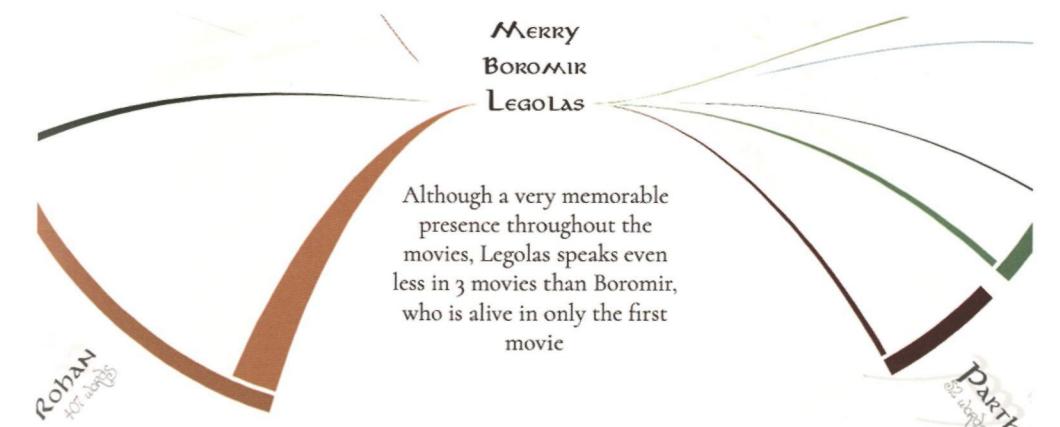
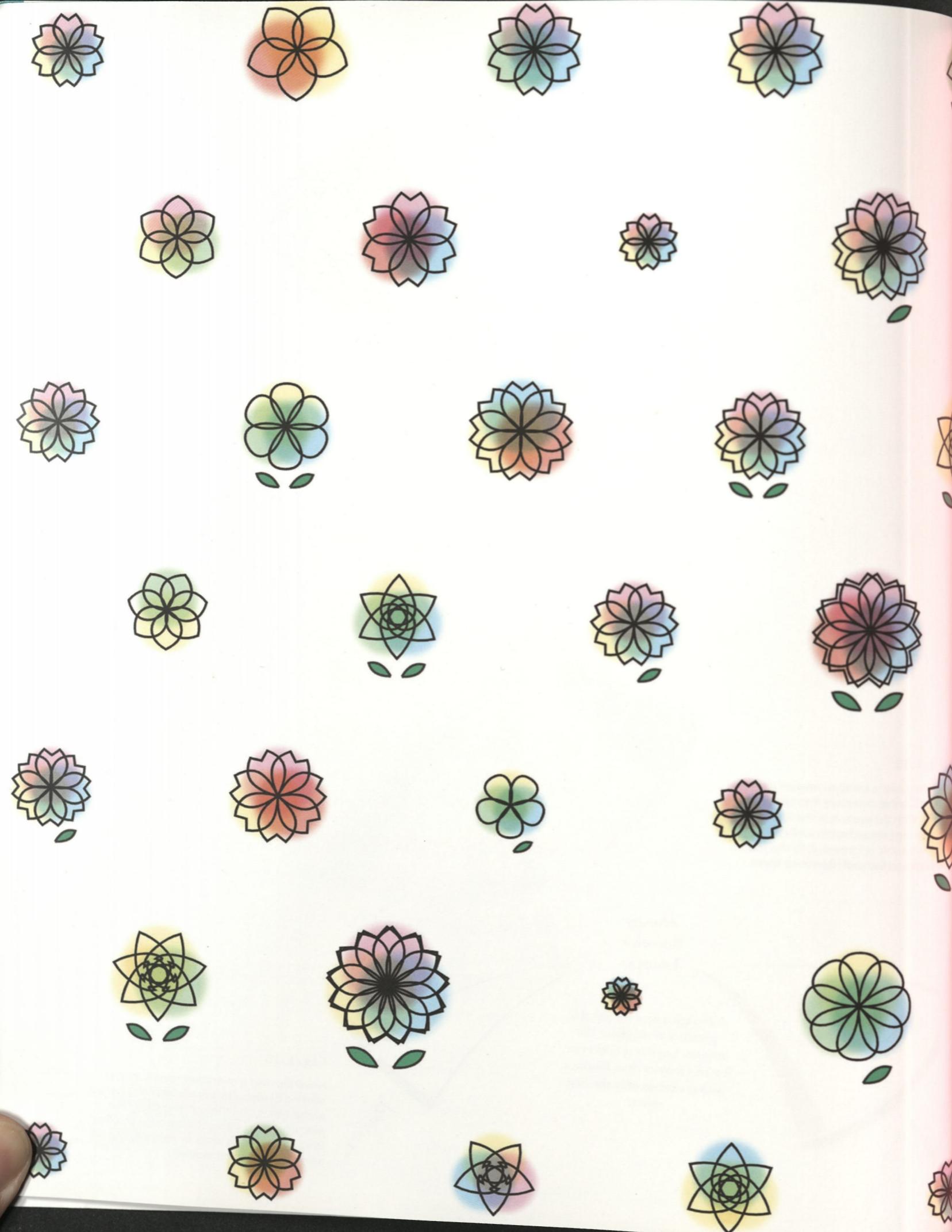


Fig.1.16

Although a very memorable presence throughout the movies, Legolas speaks even less in 3 movies than Boromir, who is alive in only the first movie

Hovering over a character reveals a small section of text with some interesting insight about him, found through this data. (For example, how Legolas talked even less than Boromir, while the latter is really only [truly] alive during the first movie!).



Film Flowers

SHIRLEY

I am absolutely horrible with pop culture references. And while I like to blame the fact that I grew up in various non-English-speaking countries for half of my childhood years (and had my head buried in textbooks for the rest of them), I know it's also because I just didn't watch that many movies growing up.

So I was pretty excited to examine movies for our first topic: how many blockbusters had I seen (or not seen) in my lifetime? And since it was July, I decided to concentrate on summer blockbusters.