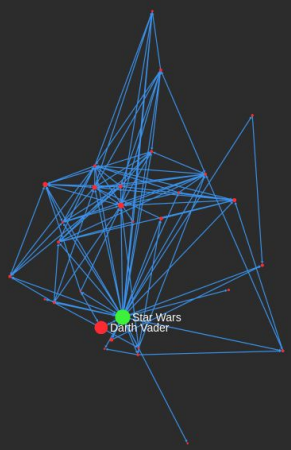
# Major events, News and Scandals: projection of everyday life to Wikipedia




Process Book

Alexandru Mocanu, Manana Lortkipanidze

Data Visualization, EPFL 2018

Contents

# Introduction

This process book serves as a description of our project for the Data Visualization (COM-480) class at EPFL, Fall 2018. The aim of the class was to introduce us to various informative/powerful/state of the art data visualization techniques and methods, while emphasizing not just on technical details or information they convey, but also on esthetic, artsy side of visualization. Data analysis heavily relies on data visualization and material learnt is crucial for anyone interested in the field.

This process book explains the reasoning behind every analysis or design step of our project starting from the initial abstract idea, drafts and sketches, all the way through the final product. This process book guides its readers through our thought process from day one, providing along the way insights into the dataset used, the design choices made, explaining the functionality of visualization, providing details of its implementation and presenting usage scenarios. Nevertheless, we also discuss possible impacts and future improvements as well as concrete use-case scenarios of our visualizations. Finally, technical details will be provided for people interested in further development/re-usage of our work.

## Overview

Wikipedia, supported and owned by Wikimedia foundation (non-profit organization), is not only the most widely used online encyclopedia, but it is also considered one of the most visited/popular websites in the world. Wikimedia foundation operates through donations received by people willing to support them, meaning that Wikipedia is free of charge for everyone.

January 15$^{th}$ of 2001 was the day Wikipedia was launched for the first time by Jimmy Wales and Larry Sanger. It was and is based on a model of openly editable and viewable content. The website contains around 5,769,478 articles in total and is multilingual.

There exists a controversy regarding accuracy of Wikipedia articles, since some accused it for exhibiting bias and presenting a mixture of "truths, half-truths, and some falsehoods". However, review published in 2005 concluded that the accuracy of Wikipedia articles is almost equal to the accuracy of Britannica. Therefore, Wikipedia can be considered as the biggest and possibly the best encyclopedia in the world.

# Motivation

Our initial motivation was to study human behaviour based on the Wikipedia data set. As already mentioned above, Wikipedia is a free and openly editable Encyclopaedia. It is created by volunteers and is one of the most popular websites. Therefore, political news, climate changes, new trends, technologies, movies and basically all-important life events impact its content, edit requests, visitor counts and frequencies. Therefore, we thought it would provide good insight into the human mindset and behaviour patterns.

The questions we asked were the following:

- How can we define, more specifically quantify, the popularity of articles?
- How can we measure spikes in visitor counts?
- Can we identify major events based just on sudden peaks in number of visitors for a specific page?
- Which pages get affected for a specific category of event?
- How does the increase in visitors on the pages affect the visitor count of its hyperlinked pages? More specifically, how far does the interest spread in terms of related pages?
- Finally, and most importantly (purpose of our course project), how can we best visualize the answers to the questions provided above?

Emphasized will be visualizations depicting the popularity and its spread along the network, as well as spike linkage with (major) life events and effect on its neighbours. These two visualizations are the result of our project, that we will discuss in fine detail in the following sections.

# Target Audience

Our target audience is anyone interested in the human mindset and the reflection of real-life events and trends on human behaviour regarding Wikipedia. Our target audience is people curious to see how powerful/widely used Wikipedia articles are and consequently how one can discover patterns of everyday life events through Wikipedia visitor counts or alternatively observe the impact of specific events captured by the number of visitors, hence people who were affected, reached or interested by a specific event/news.

# Dataset

We used a Wikipedia dataset originally retrieved from the SNAP repository. This is a website containing human navigation paths on Wikipedia that were collected through human-computer interaction within a game called Wikispeedia. In the game, players received pairs of Wikipedia articles and had to reach one article from the other through Wikipedia links between them. The dataset provided consists of 4604 different articles. For the purpose of our project, we cleaned and extracted relevant information from the above-mentioned website, consisting of names of articles, their categories and linkage information in the form of an adjacency matrix.

Since the goal of our project was to identify popularity, we looked for spikes in the number of visits and linked those occurrences with the corresponding events in the real world for visualization, we could have used an arbitrary sample from the millions of Wikipedia articles as long as we could identify unusual activities caused by external factors. Therefore, we deemed the obtained articles that we were kindly provided by Mr. Ricaud as sufficient.

Additionally, we queried the number of visits for all days in 2017 for all articles in our dataset from Wikipedia API. This information served as the basis for popularity and spike detection.

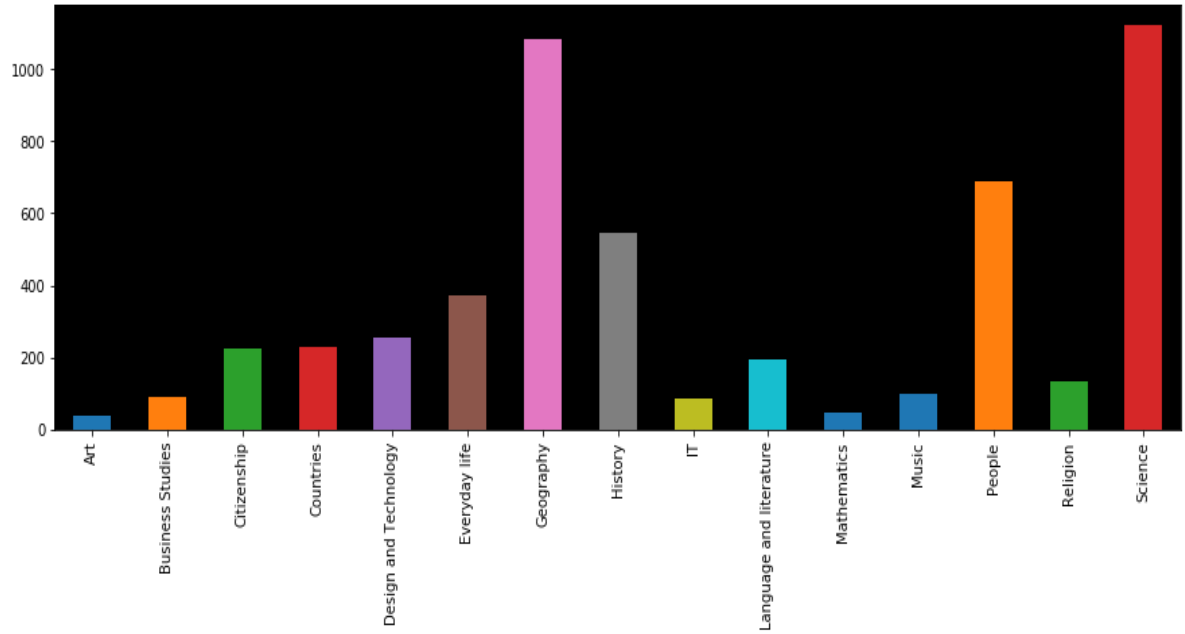
Furthermore, the dataset linking spikes of specific articles with news was created manually by us for the project. We built a dataset of around 200 entries specifying article name, event type, date and source (if applicable) of the event.

Prior to discussing design choices, we think we should briefly introduce the main characteristics and descriptive statistics of our data for the purpose of making further sections more comprehensible and insightful. As well as explain how we measure popularity and identify articles with unusually high visitor counts compared to their baseline.

## Exploratory Data Analysis

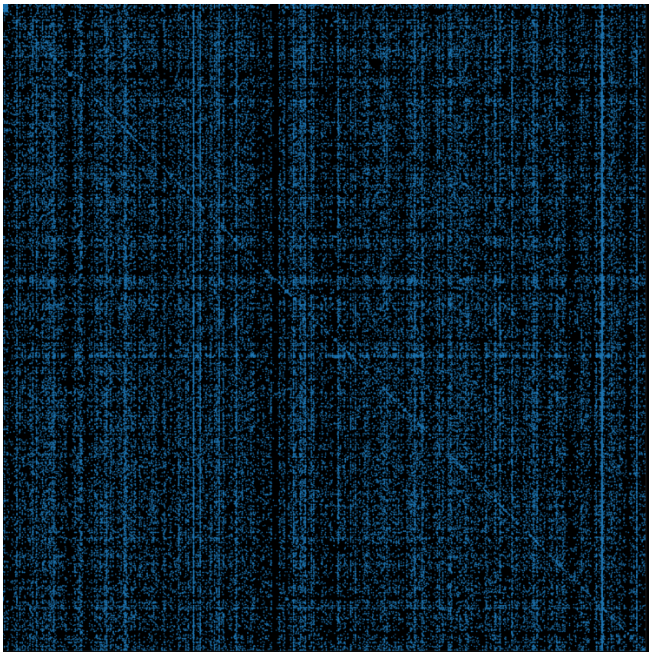
As mentioned above, SNAP repository dataset contains 4604 articles. However, some of them (less than 0.5%) were removed because they were duplicated when representing different categories. As for the categories, they are hierarchical, and we have 15 top level categories in total. It must be noted, that one article can belong to several categories, including several top-level categories. Distribution of articles over categories is as follows:

Distribution of Articles over Categories

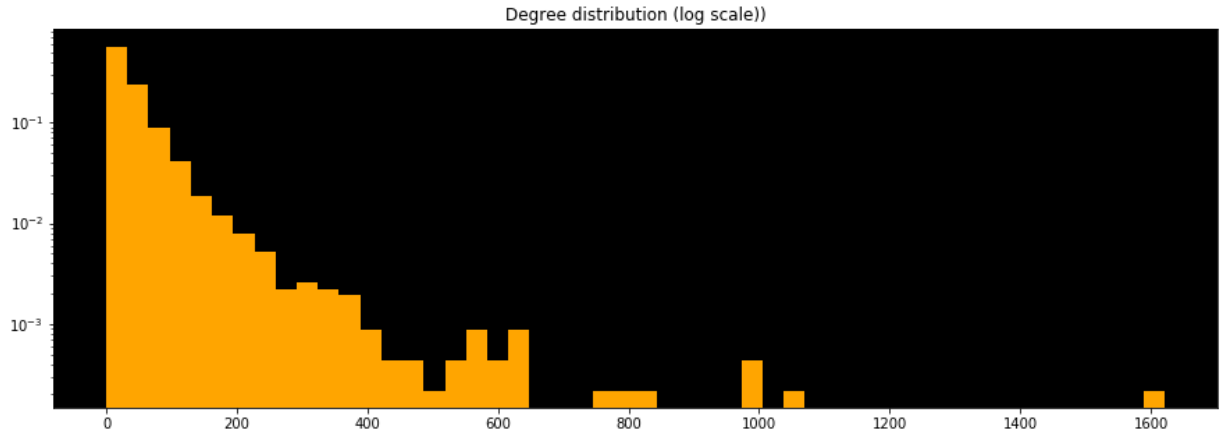



As we see most of the articles in our dataset belong to either science or geography category. Art and mathematics are represented the least, each of them having less than 100 articles in our dataset. Furthermore, as expected adjacency matrix (created by articles as nodes and hyperlinks as edges) is sparse:

Adjacency Matrix



The adjacency matrix above forms a graph with one big connected component and 3 additional isolated articles/nodes (that were then removed from our dataset). Furthermore, observing degree distribution below, we conclude that our network resembles scale free network:




Scale free network means, that we have hubs (articles with very high degree compared to other articles) and the rest of them are sparse. Additionally, we observed that graph has small world property, since diameter of our network is 8. [Meaning information/influence spreads over all network rapidly. [Therefore, it would make sense for popularity or spike to be spread to neighboring nodes. ToDo]

## Popularity of Articles

Popularity of articles is determined using daily visit counts. We plan to visualize n most popular articles/nodes for a selected time frame, providing relevant statistics alongside and possibility to explore its neighboring nodes. We won't go into details of popularity in this section, since one big part of our visualization functionality description will be devoted to the topic.

## Spikes in Visitor Counts

In order to identify spikes in visitor counts we employ different strategies. Firstly, we estimate average visit count for a day over all articles over a year. And then identify days with unusually high average visitor counts, under the assumption that this high average number of visitors was caused by significant spikes in one or several articles. After selection of days throughout the year 2017 employing the described method, we try to identify articles responsible for the unusually high daily average number. For that, we first create baseline for each article, by

averaging their daily visitor counts over a year. Afterwards, we select articles where we observe visitor count higher than their average by at least six time their standard deviation.

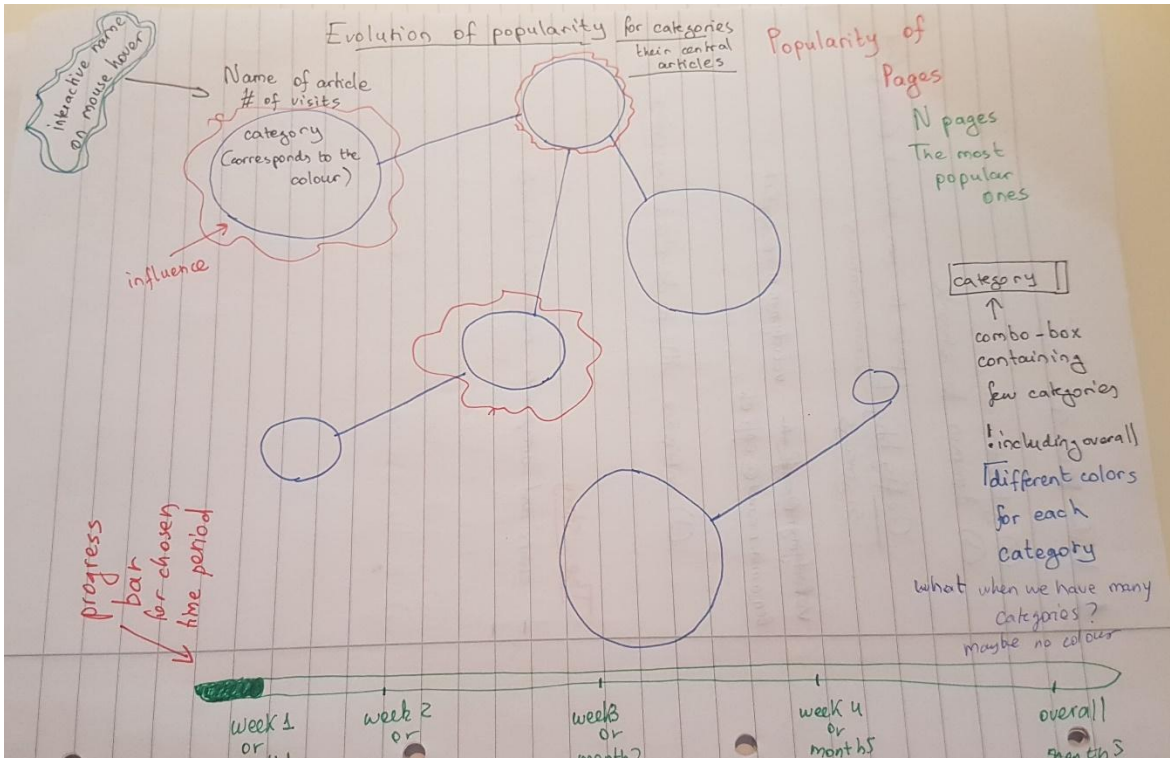
Using the above described methods, we extracted around 300 one of the most significantly spiked articles and then selected 120 so that they are evenly distributed over months. Therefore, our final dataset contains 20 articles for each month where each article provides an information about a day in that month where we observed unusually high visitor count for that page.

# Design Concepts

In this section we will discuss concept of our visualization. Firstly, we will overview design ideas, starting from the initial concepts to the final version. We will as well discuss deviations from initial plan and reasons why we deviated from it.

## Initial Design

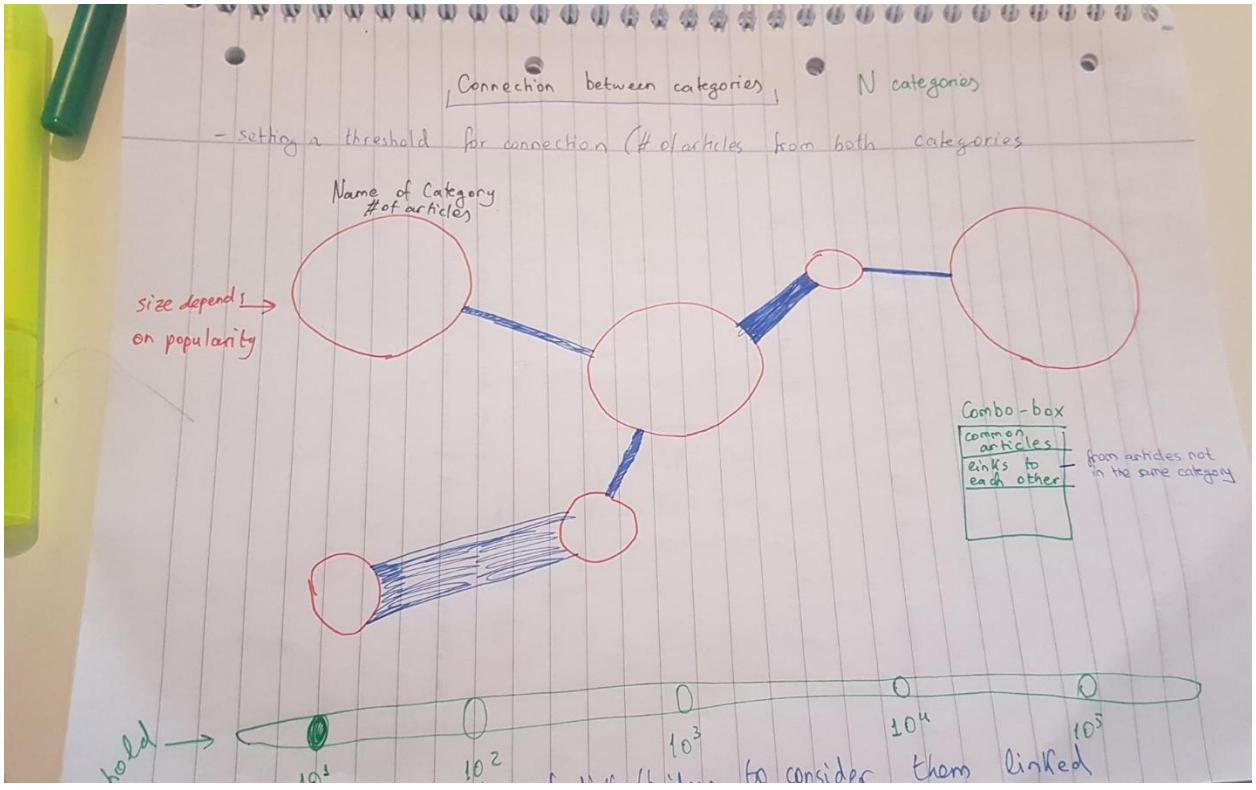
Firstly, our initial idea was to visualize popularity of articles given their visitor count through network where nodes would represent articles and edges hyperlinks between them.




Article Popularity Visualization

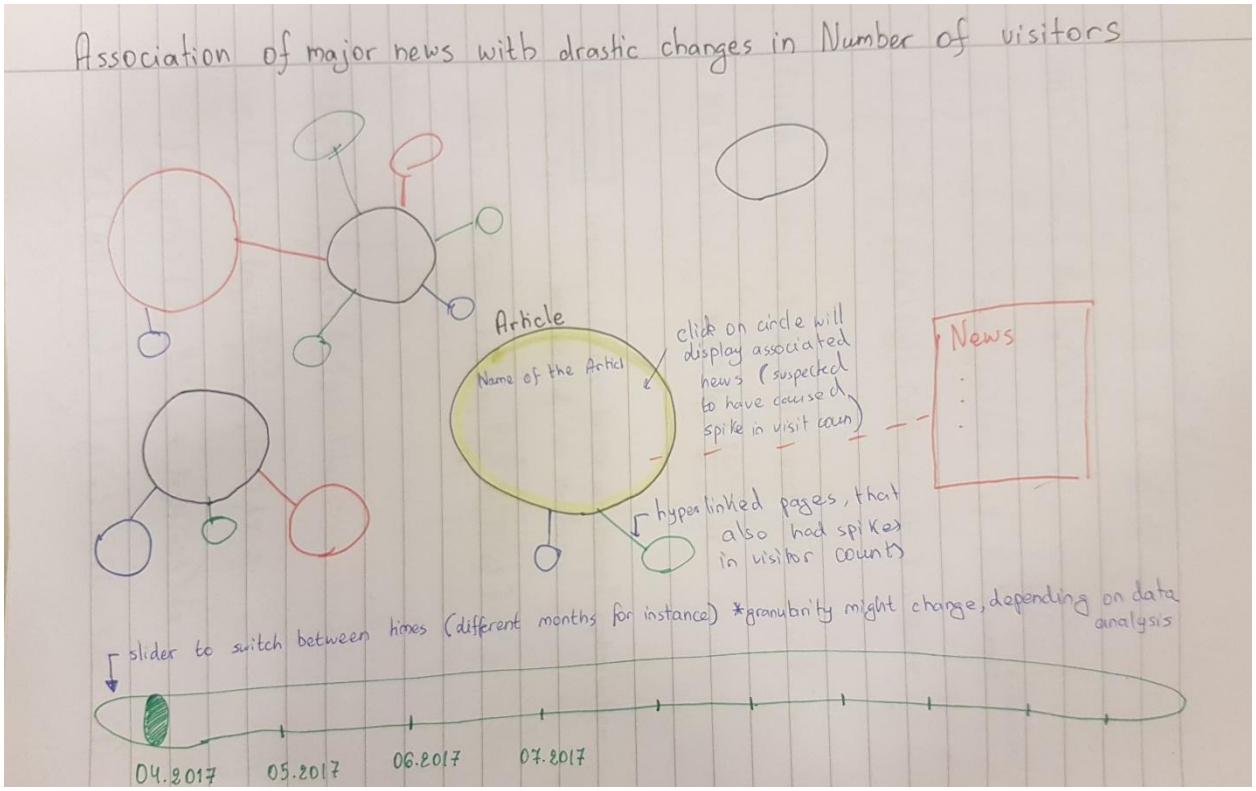
Furthermore, we wanted to make node size proportional to the visitor count as well as color the nodes depending on their category. Additionally, we wanted to have a slider at the bottom to visualize development of popularity of nodes throughout time (either weeks or a day of a month). Moreover, we thought of providing a text box where user would select number of nodes to be displayed. We decided to have a text box, because we have around 5000 articles/nodes in total and it would not be feasible and would make website lag if we displayed them all at once. Therefore, we decided to introduce an upper bound to the number of nodes allowed to be displayed at once and let user choose number of nodes. Additionally, we wanted to add a filtering option depending on categories, so that user could see only articles belonging to the selected category.

Furthermore, we wanted to create an influence visualization. We wanted to show how popularity of one articles could affect popularity of neighbouring nodes. For that, we wanted to give weights to links. Meaning, that edge width from one article/node to the other would correspond to the influence it had on its popularity. The rest of the visualization (including slider and colour selection of nodes) would have been the same. However, since it was dubious how to define influence and since we came up with a better scenario of visualization, namely popularity and link with events, we decided not to develop the above mentioned visualization further.




Popularity Influence Visualization

Main visualization plan during the initial phase was to display nodes having unexpectedly high popularity (as described in spikes in visitor counts section) and link them with events that potentially caused this interest. We wanted to have a time slider, as in previous visualizations as well as colors depending on category of nodes. We wanted to make nodes clickable to display pin with event description, date and link to the published news. Furthermore, we planned to display several neighboring images on click (two clicks as compared to the news pins), selecting the ones that also experienced relatively high spike in visitor counts.




Linking Spikes in Visitor counts with Events/News

## Deviation from Initial Ideas

First deviation from the initial plan, as mentioned above was to omit influence visualization. The second deviation was to add several statistics that we deemed important for the better insight. The third one was to remove coloring depending on categories, since we have 15 different ones. This section will concentrate on concepts of small/extended visualization we decided to include during the implementation phase before providing concrete implementation details.

### World Map

One of the most important concepts developed after the initial phase was a way to visualize origin of news in connection with articles/graph. We decided to include this information to see where the source of the events was and to provide better insight into how news from various parts of the world affect Wikipedia visit count. For that purpose we decided to add the world map to the last sketch. At first we wanted to position graph and map as rows one below the other. However, upon closer inspection we decided to place them side by side to evade appearance of scroll bar.

### Popularity Statistics

We also decided to include popularity statistics, that would appear upon selection of a specific day/month on the slider. We wanted to display bar chart of the several most popular articles displaying exact number of visitor counts.

### Category Statistics

Category Statistics was added to the popularity visualization to display visitor count distribution for a given date over categories. We decided to place corresponding bar chart to the right of the graph and make it dynamically change with the slider and present updated statistics.

### Visit Count Statistics

Additionally, we decided to include visit count statistic for visualization linking spikes in article with news around the globe. We thought it would be interesting to see how big the spike was and what was the baseline visit count otherwise. We decided to change visit count plot dynamically as well, to update with regard to date and article.

### News Description

Furthermore, we added news description to the visualization linking spikes in article with news to give more detailed information. More specifically, we give summarized content regarding news with affected article, date and link to the source of the news.

## Implementation Model

The project consists of two visualization: popularity and news, which will be presented below. The main page consists only of a presentation of the dataset used and brief descriptions of the two visualizations along with links to them.

### Popularity

The first visualization concentrates on presenting how the popularity of the Wikipedia articles, considered as the number of visits, evolves over all the days of 2017.

The visualization contains three main panels:

- The **graph** presents the articles in the form of a network, where the nodes represent articles and the edges represent outgoing hyperlinks.
- The **article popularity bar chart** presents the 10 most popular articles for the given day.
- The **categories popularity bar chart** presents the 10 most popular categories for the given day.

There are also two controllers:

- The **nodes slider** used for selecting the number of nodes to display. This lets you to select between 0 and 200 nodes to display in steps of 5. The limit is set at only 200 as for a higher number of nodes the visualization would become pretty hard to follow due to the large number of edges.
- The **time slider** which selects the day for which we want to display statistics.

The visualization permits the following types of interactions:

- To change the number of nodes to display, you just have to drag the nodes slider. The number of nodes displayed will be shown to the left of the slider.
- To change the day for which you display data, you will have to drag the time slider. The date will be displayed to left of the slider in the format *DAY/MONTH*.
- Hovering over the nodes will highlight their outgoing edges and display the article's name.
- To better explore parts of the graph, you can zoom on the region of interest.
- To view an article's top neighbours, you just have to click the article's node. This will change the node color to green and display the top neighbours. The number of neighbours to display can again be changed using the top slider. Also, after selecting a node, you can then select one of its neighbours and you will change the focus on that neighbour.
- To drop the focus on a green node, just click it and you will be taken to the previous context.

The graph's nodes are red, the edges are blue, the selected node is green and the highlighted edges are white. We chose this range of colors to keep everything simple and clean. We chose to change the color of the selected node to keep track of which node to click in order to get back to the previous context, as this would have been an almost impossible task if we were to keep all nodes red, especially considering that we can do nested node selections.

The article bar chart was added in order to better see the difference between the numbers of visits for the different articles. This respects the principle which states that lengths are easier to compare than areas.

Below, you can see representative screenshots of the popularity visualization.

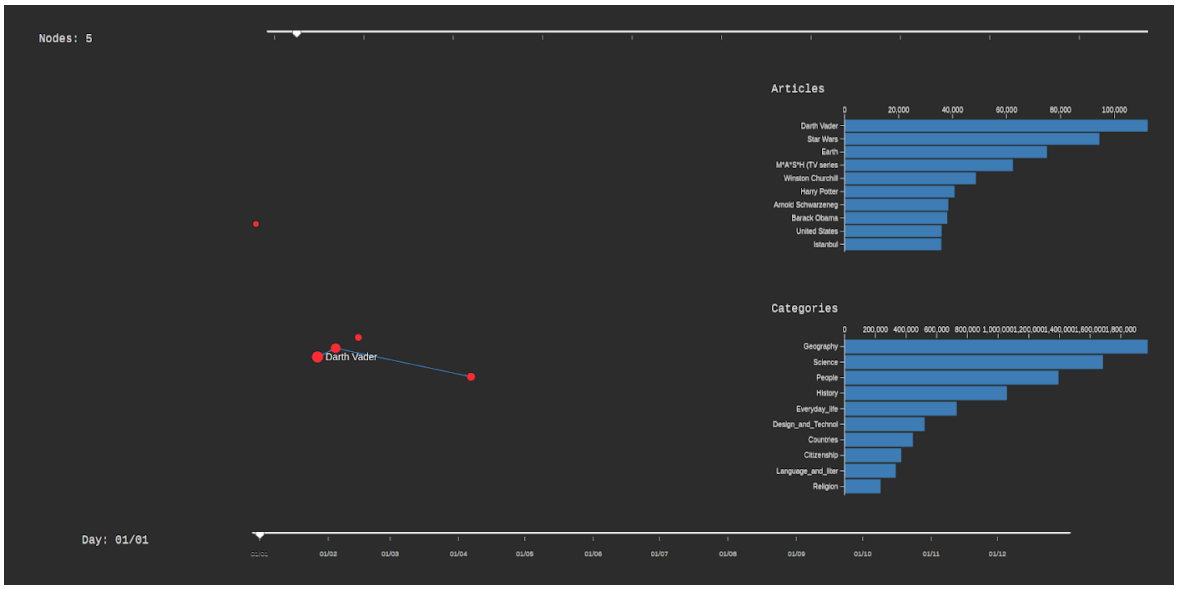



Figure presenting the initial view of the popularity visualization

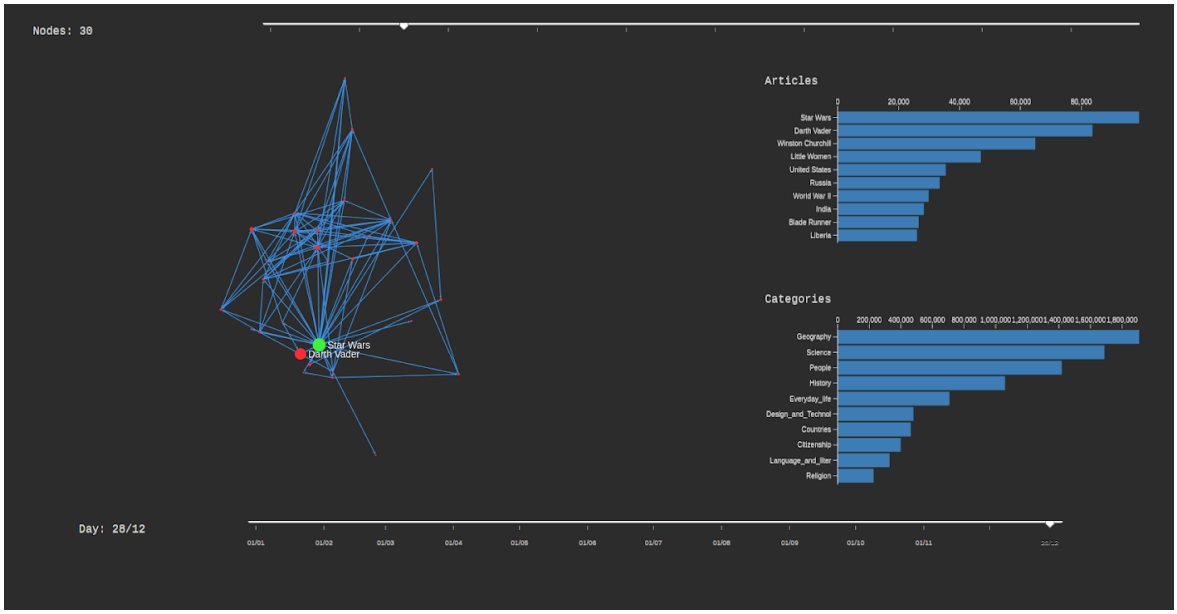



Figure presenting a highlighted node along with its popular neighbours

## News

In the second visualization, we want to show the greatest spikes that occured in the number of visits for the articles in our dataset and to link these unusual events to news that may explain them.

The visualization contains four main panels:

- The **graph** displays for each month of the year 2017 the articles that had the greatest spikes in the number of visits. The greater the spike, the larger the node. The map displays the locations on the globe linked to the top news for that month.
- The **map** shows the locations for the news for the current month. If a node is selected and it has news associated to it, the map will only highlight the country associated to that particular node.
- The **bar chart** pops-up in the bottom-left corner of the screen when a node is selected and it shows the number of visits for that month for each day of the current month.
- The **news window** pops-up in the bottom-right corner of the screen when a node is selected and it displays: the name of the article, the date for which it reaches its peak and, if there are any news associated, the country linked to the news, the type of event that occurred and a link to the news article or Wikipedia page motivating the spike.

As before, there are two sliders, one to control the number of nodes displayed in the graph and another one to change the month for which we want to display the articles with the greatest spikes.

Additionally, there is also a play button used for starting an animation.

The visualization offers the following options of interaction:

- To change the number of nodes to display, you just have to drag the top slider. The number of nodes displayed will be shown to the left of the slider.
- To change the month for which you display data, you will have to drag the bottom slider. The date will be displayed to the left of the slider.
- Hovering over the nodes will highlight their outgoing edges and display the article's name.
- To better explore parts of the graph, you can zoom on the region of interest.
- If you click a node, its color will change to green and its top neighbours will be displayed. Apart from this, statistics about the number of visits for that node for the current month will be displayed on the bottom-left corner of the screen. In addition to that, data about the news that caused the spike for that article will be displayed in the bottom-right corner of the screen. Last but not the least, the country corresponding to the selected node will be highlighted. The number of neighbours to display can again be changed using the top slider. Also, after selecting a node, you can then select one of its neighbours and you will change the focus on that neighbour.
- To drop the focus on a green node, just click it and you will be taken to the previous context.
- Hovering over the map, a pop-up window will appear for the selected country displaying, if the country is highlighted, the news for that month.
- To start an animation that takes you through the news evolution in the year 2017, click the play button. The animation will start from the month that you are currently at and it will present one interesting article from each month until the end of the year. You can

pause the animation by again clicking the button. When reaching the end of the animation, by clicking the play button, you will reset the slider to zero. No node should be selected when starting the animation.

For the graph we used the same color selection as before. The map's default color was chosen to make it look earth-like, while the highlighting color for the map was chosen to match the nodes' default color.

Below there are a few representative screenshots for the visualization.

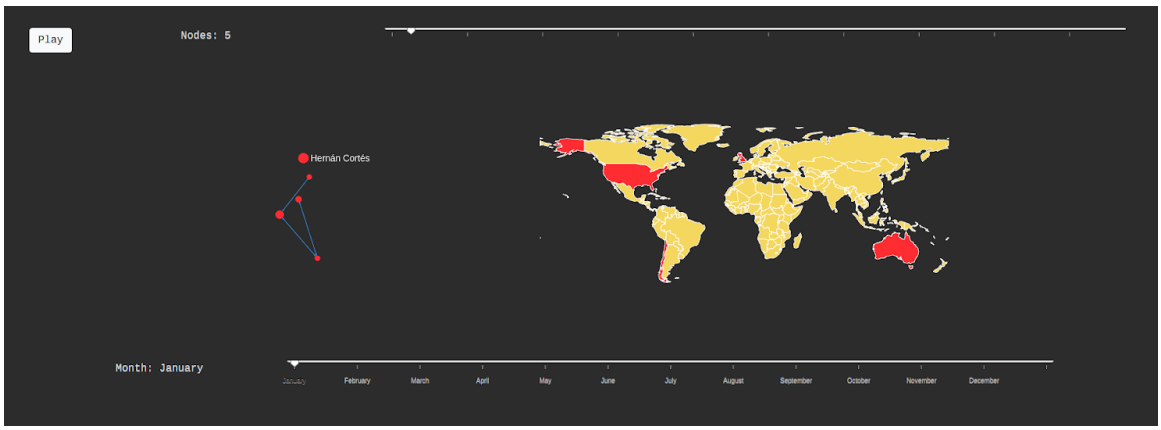



Figure presenting the initial view of the news visualization

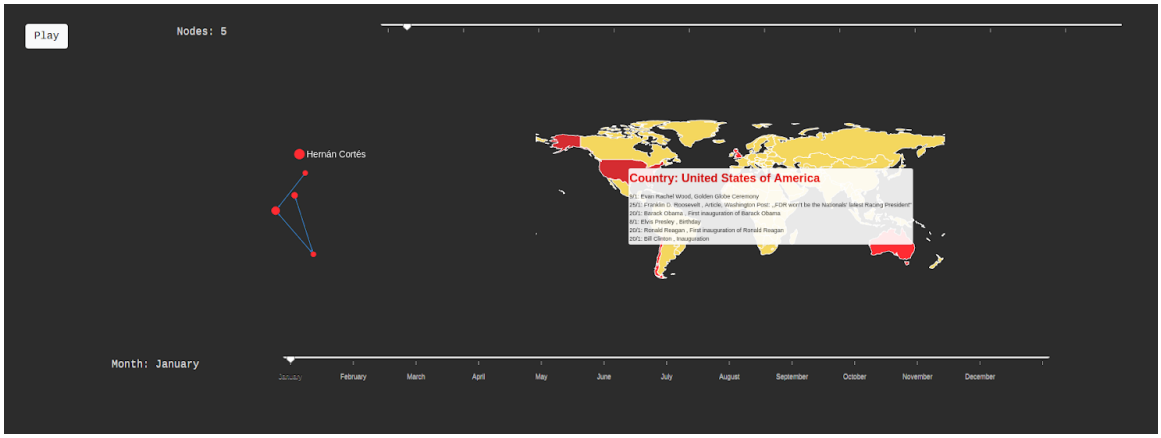



Figure presenting the news associated with a country for a given month

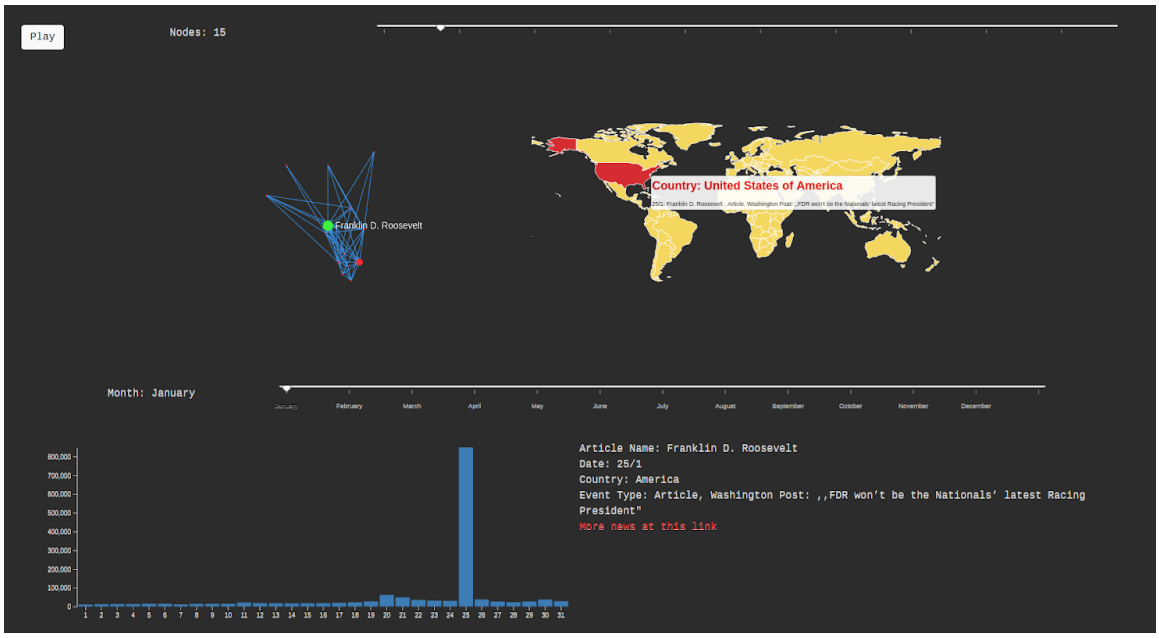



Figure presenting a highlighted node along with its statistics and news

# Technical Details

The pages containing the visualizations were created using Jekyll.

The libraries/packages used in the project are:

- d3 - for general html manipulation
- sigma.js - for drawing the graphs
- d3-simple-slider- for creating sliders

The project's structure is as follows:

- assets - directory containing several types of files
    - css - style files
    - data - data files which contain:
        - article_subject.csv - main categories for each article
        - dataDAY_MONTH.json - data about the visit counts for each article
        - dataMONTH.json - data about spikes in the number of visit counts
        - edges.json - data concerning the edges in the graphs
        - nodes.json - data concerning the nodes in the graphs
        - categories.json - number of visits for each category for each day
        - wiki_news.csv - small dataset of news

        - world.geojson - geojson file used for drawing the map
    - **scripts** - the scripts used for creating the visualizations
        - graph.js - script used for drawing the graphs, the sliders, the charts and for connecting all the elements together
        - world_map.js - script used for dealing with the map
- _includes - directory containing html files for creating the divs filled with the visualizations' elements
- _layouts - directory containing layouts used by Jekyll for creating the web pages
- lib - directory containing libraries/packages used in the project's implementation
- notebooks - ipython notebooks used for collecting, analysing and grouping data into the datafiles presented above
- index.md, news.md, popularity.md - markdown files used by Jekyll for creating the web pages

# Evaluation

We think our visualization and **dataset we created** for this project provided great insights into questions we asked initially. Below, we will provide a brief overview of the initial questions and ways we solved it not just conceptually, but visually as well:

- How can we define, more specifically quantify, the popularity of articles?

  Conceptually: We extracted visit counts for 2017 and linked the popularity with visit counts.

  Visually: We linked the diameter of the node/article with its popularity. Additionally, we provide popularity statistics, as explained in the popularity statistics section.

- How can we measure spikes in visitor counts?

  Conceptually: We decided to measure spikes in terms of mean and std of each article over a year. We selected articles where we observed visitor count higher than their average by at least six time their standard deviation.

  Visually: We linked spikes with the diameter of the nodes/articles as well. The bigger the diameter, the higher the spike for the given article on a given day. For better insight, one can select as many nodes as they want to check articles with the most unusual behaviour. Furthermore, we provide visit count statistics for each article selected over a given month.

- Can we identify major events based just on sudden peaks in number of visitors for a specific page?

  Conceptually: Yes, extracted articles with the methods described above resulted in finding a related news more than 85% of the time and almost all of them represented major/worldwide events even if we worked with randomly selected 5000 articles. Therefore, we conclude that through spikes in articles one can identify big event.

  Visually: We visualize news on map that were linked with articles. One can just slide through all months and check news appearing on the map and summary of the news. Conclusion is the same as in conceptual part.

- Which pages get affected for a specific category of event?

  Conceptually: We created dataset linking specific events with the articles. However, the list is not exhaustive and only contains articles that get affected the most/more drastically than the others.

  Visually: One can hover over countries ("red" ones, as shown above) on a map. The list of the news contains the name of the articles affected, among other things.

- How does the increase in visitors on the pages affect the visitor count of its hyperlinked pages? More specifically, how far does the interest spread in terms of related pages?

  Visually: One can click on a node. Once selected, neighbouring (most affected) nodes will be displayed. Furthermore, one can click on those neighbouring nodes to see their visitor count statistics and see how they got affected. One can continue following the path of neighbours as long as they want, seeing how effect slowly diminishes.

# Further Work

At the moment, there are still some things to deal with in the code. One example is taking care of the delays produced by reading the quite large data files, which due to asynchronicity leads to sometimes trying to add edges that are already present. Another issue is resizing the window while the animation for the news visualization is running. This leads at the moment at completely losing the link between the animation and the new time slider that is drawn.

Apart from the coding part, there are still many improvements to do on the web pages layout. They currently have a very simple and not quite esthetic look and this could be majorly improved, maybe by just using some templates that would be flexible enough to also allow customizing the divs in which to include the visualizations.

# Peer Assessment

We were only two people in our team, since one had to leave his study program. We were both prepared during team meetings and attended all lectures/lab sessions together. We both contributed useful ideas and code for the visualization. We had to work hard in order to compensate for a missing member, therefore we had to work and think as a team. We had no disagreements during the work process and we were both very respectful towards each others ideas and encouraged each other to express them.