Vincenzo Bazzucchi, Christian Sciuto, Lorenzo Tarantino

Data Visualization Project

Process book

This is the process book that describes our progresses during the project. When there is a name before a paragraph, that means that that person of the team worked on that.

Goal of the project:

Donald Trump is the first US president to have adopted Twitter as his main communication channel. Increasingly more americans realize that his tweets are destructive, inopportune and unfounded, that they represent an obstacle to a real political dialogue, often resulting in a chaos politics. Moreover, a lot of incoherences can be easily found between his claims on Twitter and reality, which leads us to analyze his tweets and his behavior on the social network in order to detect fake news.

Trump is one of the most unpredictable president ever elected as he is impulsive, he tends to improvise in difficult situations and he manipulates the crowds. Therefore we would like to characterize his behavior on Twitter by studying his tweets during different chronological periods and while he faces different topics. We would also like to understand which emotions lay under these tweets and show them.

We would like to show Trump's Twitter usage in different periods and topics. In order to do this we would like to create an interactive visualization, so that the user can choose what information to focus on. We will also try to build visualizations to describe the emotions we detected in his tweets.

Datasets:

We used two datasets:

1) Trump's tweet dataset: you can find the dataset here.
   The dataset contains all the tweets of Trump from 2009 until now.

There are “condensed” and “master” tweets. We are going to use the “condensed” ones, because it contains all the practical informations we need.

2) Word-emotions dataset: you can find the dataset here.
   The dataset contains a lot of different informations, we are going to use 'NRC-Emotion-Lexicon-Wordlevel-v0.92' dataset, that contains 14.000 words. Every word is associated with 8 emotions and a positive or negative value.

Analysis of the datasets:

For the pre-processing part of the project, we worked on python. We used plots and bar charts using the ‘matplotlib’ library in order to gain insights on the data. The visualizations were very simple and not interactive, but they showed us the right directions to take in order to produce better visualizations using Javascript and D3.

Users:

We thought that the users of these visualizations will be every one that is interested in Trump’s personality. For this reason, we thought to use intuitive but effective visualizations to explain our data: one user can come from any background, so it has to be not difficult to interact with the visualizations. Furthermore, the visualizations have to be well explained because we don’t know how much a user can know about Trump a priori.

Process book

12/11/2017:

We decided that we wanted to show visualizations regarding Trump’s tweets.

1) Topic Detection: we focused on dividing Trump’s tweets in different topics and study the differences between the tweets that are in different topics. This topics will be detected using LDA. The main idea is to show which words he uses in which topic, if a topic is generally liked by his public or not...

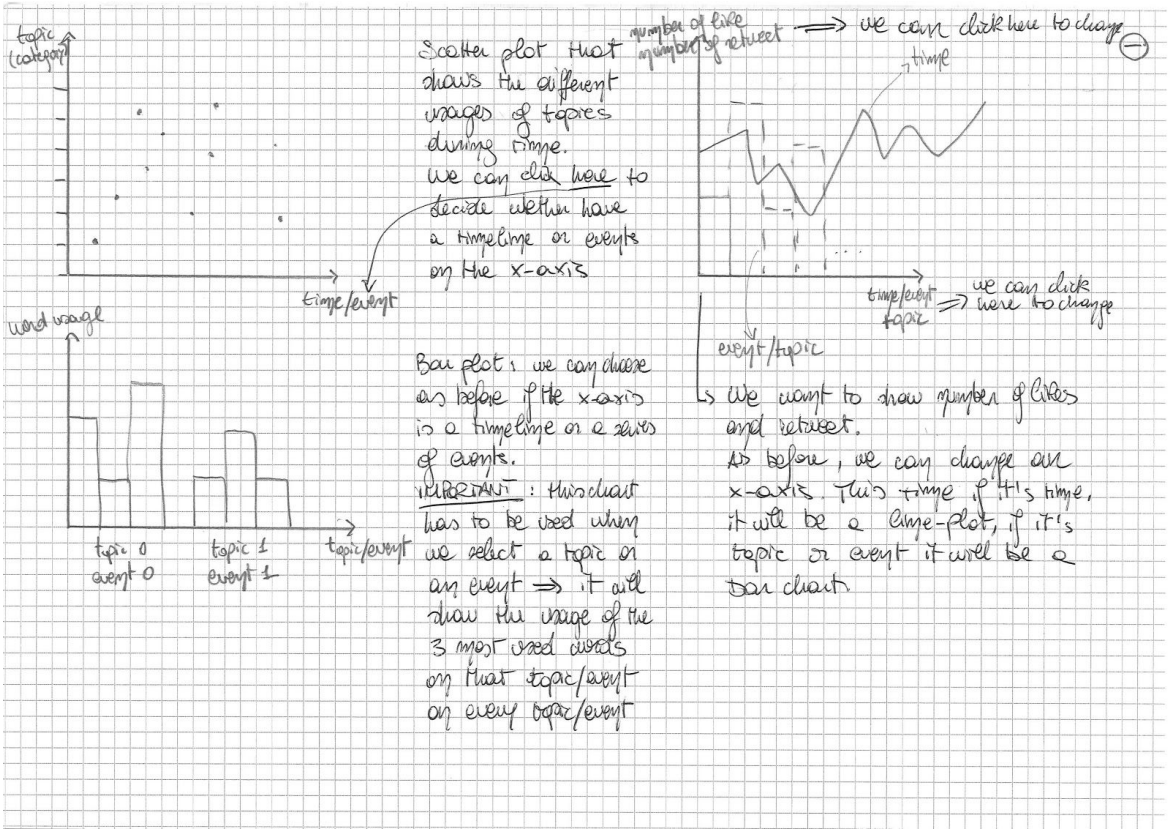
2) Sentiment Analysis: find a way to express the emotions that are beyond the text of Trump's tweets and show them, comparing different topics and events.

First ideas:

- From all tweets we are going to show different topics, word usage, topic proportion, number of likes and retweet per topic and sentiment analysis
- Following the same procedure as above, but doing the visualizations only for some handpicked events
- Using the Trump's tweet debunked dataset of the Washington Post, shows stats of Trump's fake news (only if we have time to do so)

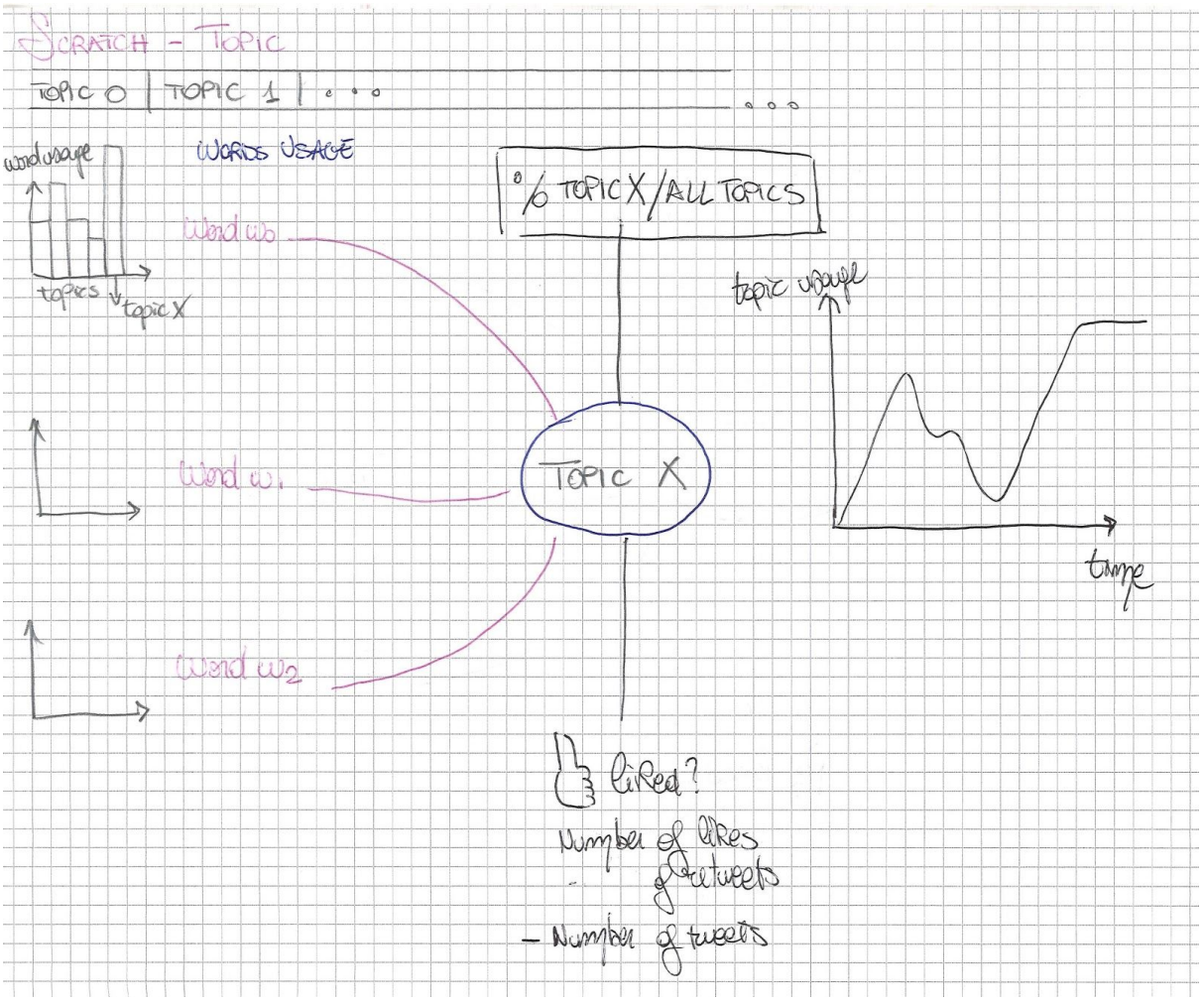
14/11/2017:

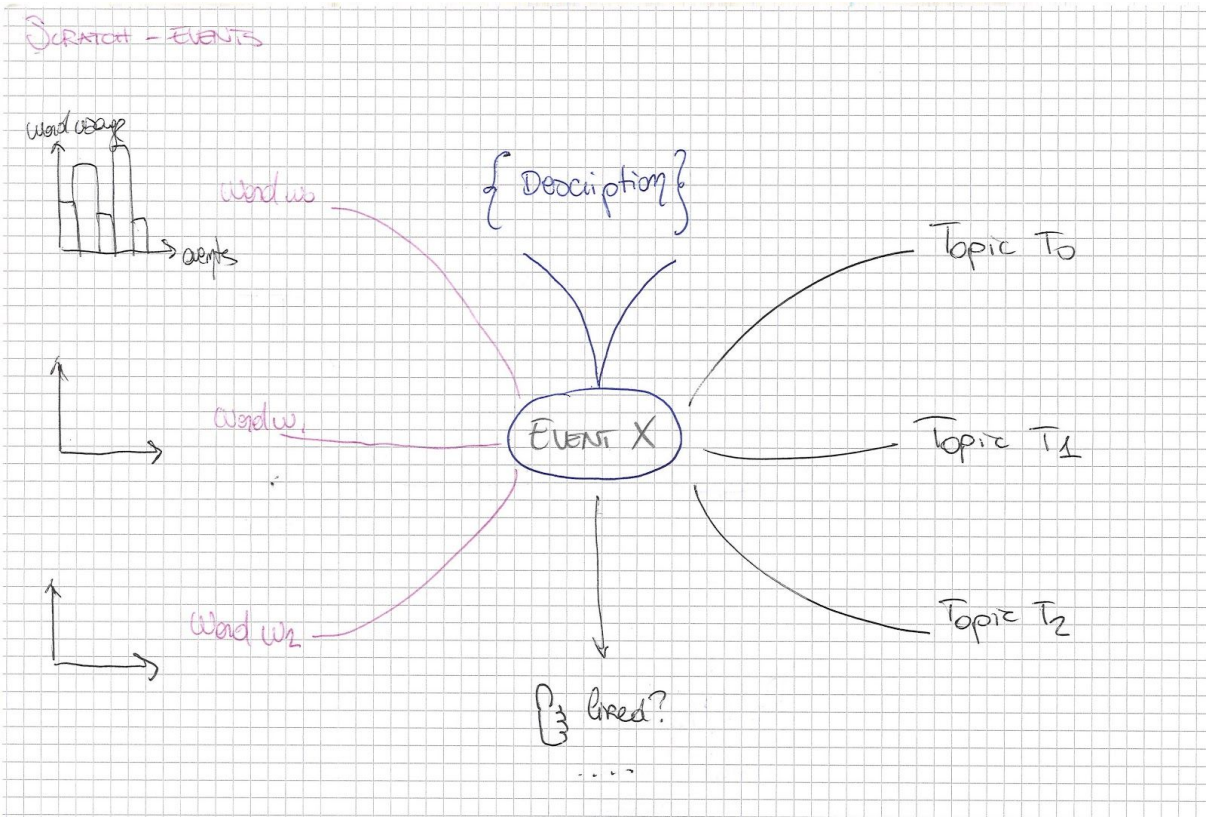
We started to do some sketches about our ideas on the project.



15/11/2017:

After confronting with the Professor, our ideas had to be developed more, so we changed the visualizations we had in mind.


SCRATCH - TOPIC
TOPIC 0 | TOPIC 1 | ...
word usage
WORDS USAGE
topics
topic X
Word w1
Word w2
% TOPIC X / ALL TOPICS
topic usage
time
TOPIC X
Liked?
Number of likes of tweets
- Number of tweets


SCRATCH - EVENTS
word usage
events
Word w1
Word w2
{ Description }
EVENT X
Topic T0
Topic T1
Topic T2
Liked?
...

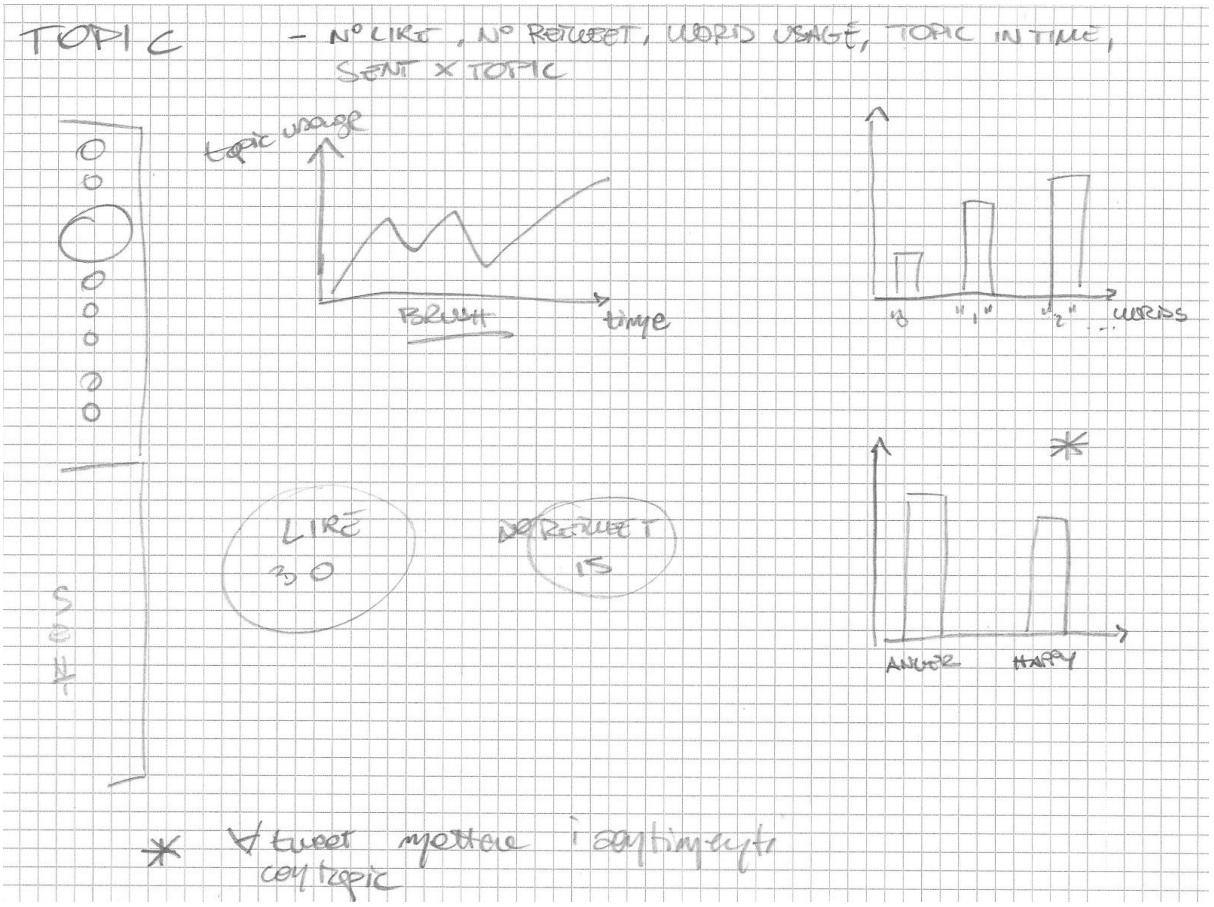
17/11/2017:

Lorenzo: Topic Detection completed. I used LDA to cluster the tweets in 8 different topics. Doing some analysis on the documents of the topics I found out that I could put together some of them
The topics are business, shows, interviews and debates, various, foreign politics, that contains the China topic too, internal politics, that contains Hillary Clinton and Obama topics as well.

Christian: Sentiment analysis started: using a dataset offered by the National Research Council Canada and created by Saif Mohammad, I am starting to study how to use the datasets and how to link the emotions to the tweets that Trump posted.

19/11/2017:

Since the analysis of Fake News wasn't going in the right direction, we decided to focus only on topic detection and sentiment analysis. This led to take into account another kind of visualization.



Lorenzo and Vincenzo: As shown above, we changed our idea of visualization. Here we will explain in detail what was the idea behind that sketch.
Layout: on the left we will have two “buttons”, Topic and Sentiment. Once you click on one of them, let’s say Topic, a series of bubbles appears on the right. Every bubble has a title and refer to a specific topic. The size of every bubble is proportional to the number of tweets that every topic contains. If you click on one of this bubble, they are all going to move on the left, on the right of the two buttons, disposed in a vertical manner. The one you clicked on will be bigger than the others, in order to indicate you which stats are showed on the right. When the bubbles move to the left, different plots appear on the space left: word usage, number of retweets and favorites, sentiments of the topic, usage of the topic in time.
It will be implemented by *Vincenzo*.
Topic/Sentiment usage plot: it will have on the x-axis a timeline, from the start of the usage of that topic to the end (most of them go from 2009 to 2017). It has to contain a *brush* in order to make the user able to focus on precise time period if he wants to. It will be implemented by *Christian* using “Billboard” - changed from C3.
Word usage and sentiment/topic plot: these two plots will be categorical bar charts: on the x-axis of the first plot there will be the 20 (or another similar number to decide in the future) most used words in the topic selected (or sentiment if it was selected the sentiment button and the an emotion bubble). The y-axis will show the count of these words.
On the x-axis of the second plot there will be the emotions (or the topics if sentiment was chosen) and on the y-axis the sum of the values of those emotions for the selected topic (or the average value of the emotion for the different topics if sentiment was chosen).
It will be implemented by *Lorenzo* using “Billboard” - changed from C3.

Christian: Sentiment analysis completed: every tweet has 10 different emotions associated with a value: *anger, anticipation, disgust, fear, joy, sadness, surprise, trust and positive and negative*.

21/11/2017:

We distributed the work on the visualizations: Vincenzo will think about the layout and the web page, Christian and Lorenzo will start to work respectively on the statistics plots of the topics and of emotions.

22/11/2017:

Lorenzo and Christian: we found that our plots are very similar, so we started to think of a way to reuse our functions. We discovered that C3 doesn't work with the last version of D3 (v4), so we found out a library called "Billboard" that is a fork of the C3 project but that is compatible with D3 v4.

Vincenzo: I started working on the frontend code. I cloned the skeleton at https://github.com/kikohs/d3-es6-boilerplate but had to apply many customizations to fit our project. I download the necessary dependency and modified the webpack scripts as the production script did not work because of breaking changes on one of the plugin used and because the development script was very slow and contained some features that we did not need.

23/11/2017:

Vincenzo: I decided to organize the code using the following classes

- Dashboard: entrypoint of the application, this class instantiates the sidebar and handles the user's choice
- Sidebar: simple svg component representing a list of buttons. It receives a list of objects, one per button, and executes the callback associated with a button in case of click.
- Bubbles: represents a bubble plot. Starting from a list of objects containing a name, size and callback, this class renders a bubble chart. When of the bubbles is selected the bubbles become smaller and move to the left of the screen. To create the bubble plot D3's pack and hierarchy functions were used.
- Manager (SentimentsManager, TopicsManager): they create and manage the bubble for the corresponding category and instantiate the correct visualizer.
- Visualizer (SentimentVisualizer, TopicVisualizer): the visualizers are responsible for displaying the plots when an emotion or topic is chosen. They compute the series required for plotting and inject them in the super class (Visualizer) so that it can create the necessary div and finally draw the plots using Billboard.js

25/11/2017:

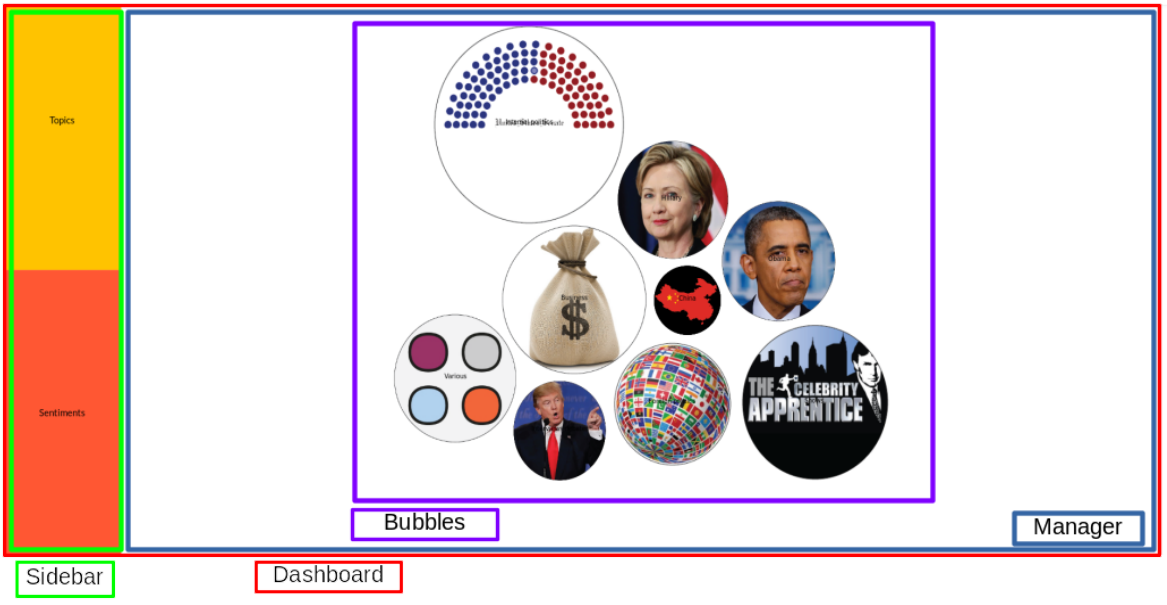
Lorenzo and Christian: the plots are ready. We implemented the Visualizers classes prepared by Vincenzo. In the super class (Visualizer) we implemented the general methods to plot the three different charts we are showing clicking on Topic or Sentiment: a line plot and two bar charts, implemented in Billboard.js. For every plot a new div will be created. In the two children

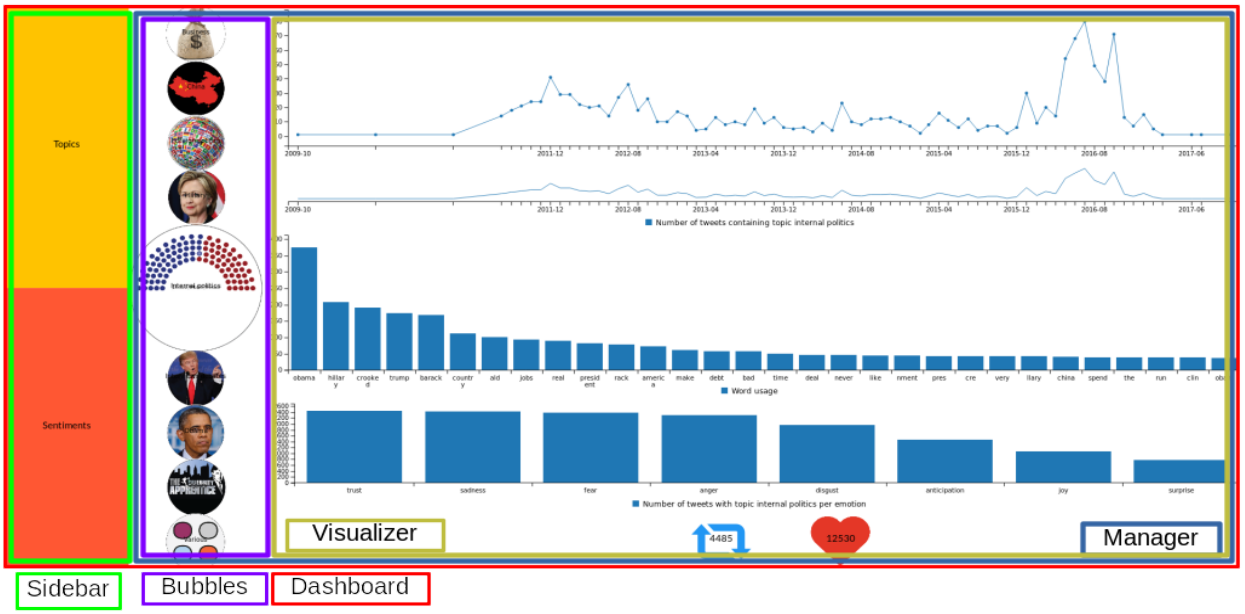
classes (TopicVisualizer and SentimentVisualizer) we prepare and process the two different kinds of data in the constructor in order to be able to call the Visualizer methods on them.

Vincenzo: a first prototype of the layout is ready. The implementation of the layout and user interaction turned out to be much more challenging than planned. This results in a very rough prototype in which we identified many improvements that will be implemented in the following weeks. These can be found on a GitHub issue we opened.

26/11/2017:

Vincenzo: I put together the layout code and the visualizations created by Christian and Lorenzo. I also added a small “welcome screen” with the title of the project, our names and a button to launch the visualizations. Finally the page was published using github pages.
In the following pictures, we indicate which class is responsible for each element of the visualization:





05/12/2017:

Vincenzo: I started to implement the comparison of two or more bubbles (emotion or topic). To do this I want the user to be able to drag and drop bubbles into a specific area to select them and then start the visualization which will show stacked plots.

06/12/2017:

Vincenzo: After implementing drag and drop and bubble collision with d3 and svg, I realized that performances are not great at all. The drag and drop is laggy and not smooth at all. Therefore I refactored the Bubbles class to use the HTML5 canvas.

07/12/2017:

Vincenzo: Move bubbles (collapse et inflate) using d3-force simulations

12/12/2017

Lorenzo: Find pictures to replace colored bubbles and cut/resize them.

15/12/2017

Lorenzo: Write introduction for visualization

18/12/2017

Vincenzo: Adapt interaction to new page. Prepare drag and drop zone to compare topics, implement two bubbles selection and add buttons to enter or reset selections.
Replace colored bubbles with pictures prepared by Lorenzo.
The implementation of the stacked plots for comparison was straightforward thanks to the existing class hierarchy: we created a VisualizerStacker which

extends our previous Visualizer and whose constructor receives one visualizer object for each selected bubble. This allows to extract the data ready to be stacked and visualised. Minor adaptations were required in the Manager class.
Lorenzo: implemented the VisualizerStacker class following the same procedure used for TopicVisualizer and SentimentVisualizer, this time with the goal of comparison of two or more bubbles together. Implemented with Billboard.js.

20/12/2017

Lorenzo: Implemented the webpage, starting from a W3 template. Implemented stacked plots for topics and emotions comparison.
Christian: Description of the visualization and data finished: introduction, structure of the visualization and conclusions.

22/12/2017

Vincenzo: hunt some bugs down.

Peer assessment

Vincenzo

The teamwork went quite smoothly. We did not meet often because we have different course schedules which made finding time for meeting physically a bit challenging. We mainly communicated through chat messages. The only real discussion concerned the design of the visualization: we had slightly different ideas about how it should look. We sat down at table armed with pencil and papers and we agreed to the design we finally implemented. Being the only one in the team with some (little) javascript experience, it was challenging sometimes to explain to the others how to do some things but they were flexible enough to understand what I wrote and to do the necessary for the project.

Lorenzo

Everyone was prepared during the meetings, even if they didn't happen very often because we preferred to communicate through chats. Since I didn't have any Javascript experience it was quite difficult to start the project, but it went better working on it. Everyone contributed as he could do due to past experiences. There has always been respect and open mind from everyone. Luckily we didn't have any real disagreement since Vincenzo had the best ideas and we followed them.

Christian

I had already worked with Lorenzo on previous projects, fortunately Vincenzo turned out to be an excellent team-mate as well as being a point of reference given his previous experience in the field. As pointed out by my two colleagues, most of the time we communicated via chat, preparing meetings especially in the initial part of the project during the lab hours of the course. There were no important conflicts during the decision-making process and the elaboration of ideas for the project, indeed there was a good understanding between us. Given my inexperience in the disciplines of this course (Html, Javascript first experience) I was very lucky to have comprehensive and available colleagues.