Our product

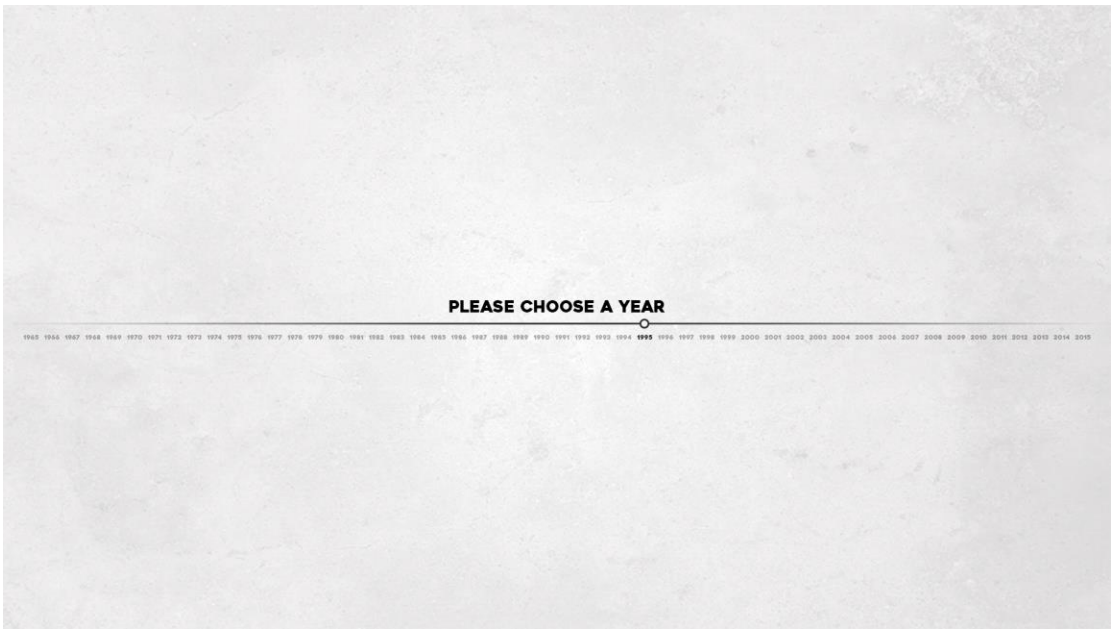
Our goal is the following : create a visualization which shows the link between historical events and songs, through their lyrics. Our website will contain one main animation, illustrating this.

For that purpose we had to gather events data. Since we could not find any relevant dataset for our project, we created our own by scrapping information from https://www.onthisday.com. This site has the advantage of showing only major events which would perfectly fit our needs. Indeed, for each year from 1965 to 2015, we were able to extract the day, month and year of the event, as well as a one-sentence summary. Having collected these data, we used the Wikipedia Python API, to gather the article's url associated with the event as well as the summary of the article. At the end we were able to gather 1115 events over 50 years. The complete processing script can be found [here](https://github.com/com-480-data-visualization/com-480-project-artemis/blob/master/data%20analysis/data_processing_events.ipynb).

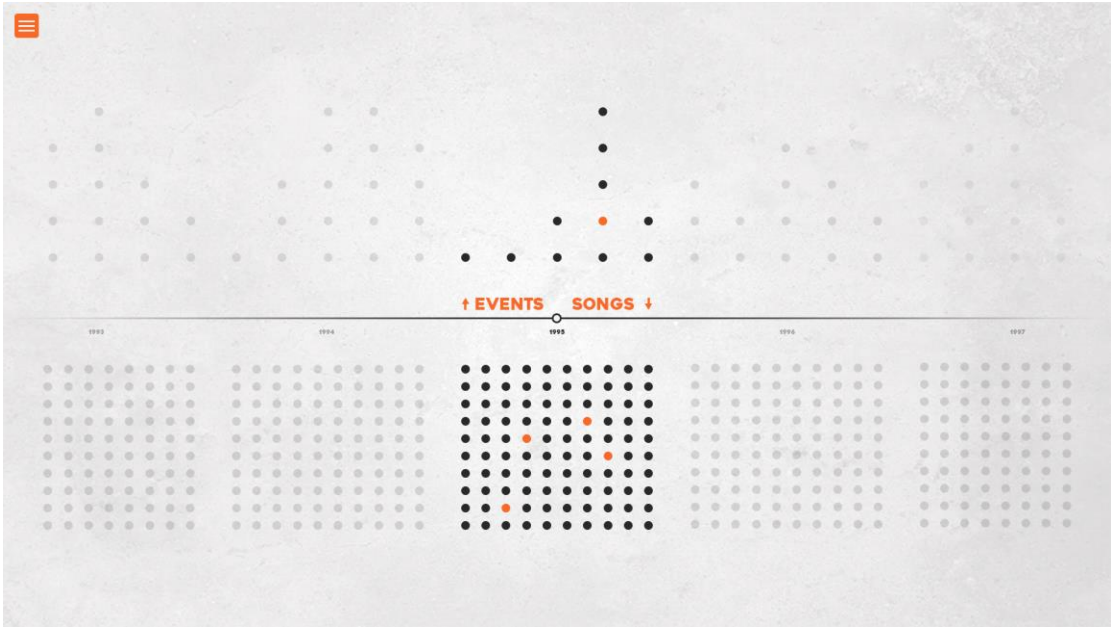
Sketches

The following 5 sketches show our first idea of the rendering. It is meant to evolve throughout the project and may not be exactly the same at the end of the project, but this is the general idea. You can have a look at the initial version of the website [here](https://github.com/com-480-data-visualization/com-480-project-artemis/tree/master/website).

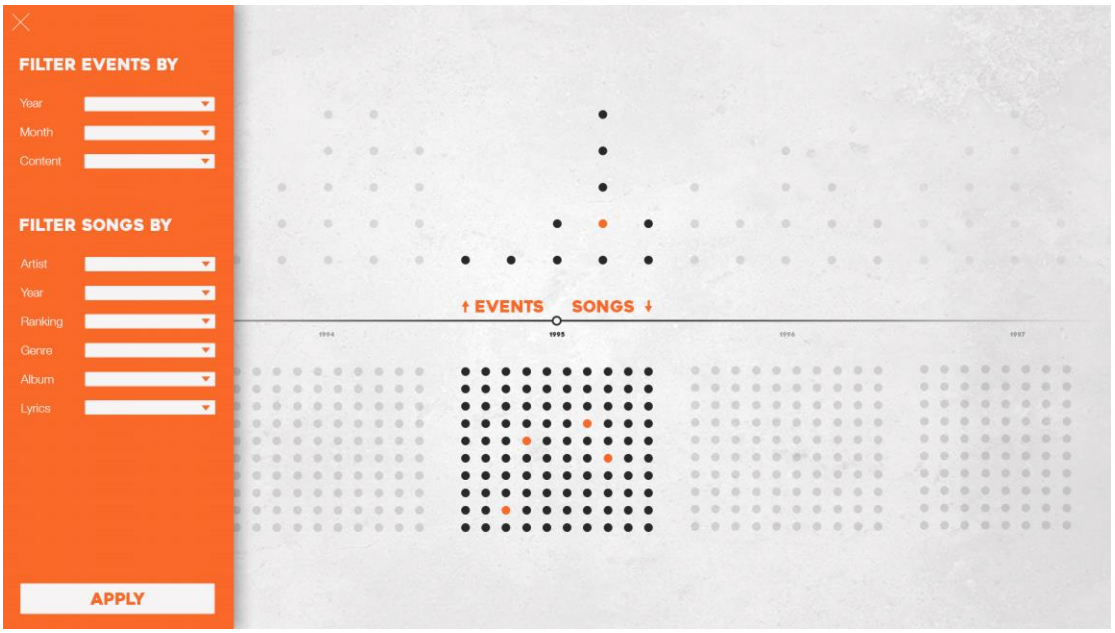
The user lands on the home page, which is quite light. He is asked to choose a year.



Once he has chosen a year, the time line is zoomed in, centered in the year the user chose. The events are displayed above the time line while the songs are displayed below. The data points related to the chosen year are highlighted, and the ones related to the 2 previous and 2 folowwing years have reduced opacity. If the user hover over an event data point, the connected song data points are highlighted in orange as well.



The user is also offered the possibility to filter the resuts, i.e., filter the events or/and the songs.



Once the user choose an event data point, the informations appear in the upper half of the screen, while the related song data points are still highlighted in orange.

Once the user choose one of the connected song, the informations about the song appear in the lower half of the screen. As we will use Name Entity Recognition to link the events and the songs, we will also highlight the words that allowed us to make a connection between the 2 data points. Since we also extracted the wikipedia article for each event, we will add a link pointing to that article, in order for the user to read the entire article.

Tools

We will use several tools for the visualisation Past lectures :

Week 4 - D3

Week 5 - Interactions

Week 6 - Mark and channels

Week 7 - Design for data viz (already used, for designing our interface)

Week 9 - Text visualization : for processing our text, and may be useful if we want to display it (this may come later)

Week 10 - Graphs : may be useful for the links between songs and events

The NLP tools we use / might use for improvements include the following algorithms : LDA, Doc2Vec, Word2Vec, Glove and BERT if time allows. We are using dedicated libraries offered in Python : Scikit-learn, Gensim, Spacy, Nltk and, if time allows transformers (https://github.com/huggingface/transformers).

In our Roadmap below, steps 5 and 6 imply some user interaction, and therefore some live computations. We considered using tfjs (tensorflow.js), but it might not be compatible with all the end task vectorisation tools we are currently computing in Python. We therefore decided to have a server with minimal Python backend to compute vectors for text input by user, using the Flask framework, and leaving the visualization do D3.js. Using flask implied some changes in the code structure, which are

currently in progress on the flask branch. This way, if we were to not use it because step 5 and 6 are not done, we can go back to a fully HTML/CSS/JS server.

Roadmap

1. Work on a "simple" NLP algorithm, to link songs to events

    - Using Name Entity recognition to extract entities from the lyrics, and from the wikipedia excerpt.

    - Matching the lyrics and Wikipedia excerpt which have Entities in common.

2. Work on the animations

    - The apparition of the dots when clicking on a year.

    - The highlighting of the dots when hovering the mouse over a point representing a song or an event.

    - The apparition of a description when clicking on a dot (event or song)

    - The navigation through the timeline.

3. Work on the website (in parallel of 2)

    - Explanation of our algorithm.

    - Navigation between the upper part (main animation) and the lower part (description of the project) of the website.

4. (optional - if time) Improve our NLP algorithm, to make the links between lyrics and events more coherent / accurate / meaningful.

5. (optional - if time) Enable users to submit their own written text, and show them the link between their text and our events.

6. (optional - if time) Develop another page, that would be more semantic / meaning oriented, rather than time. This page would enable the user to visualize semantic links such as embeddings in 2D between Lyrics and events. This way we could for example clusterize texts, and show where the text given by the user positions itself compared to the Events, Lyrics and Clusters.