Chapter 1 Introduction

The focus of this thesis is to explore visual themes of different scripts. The word script is used to refer to the character set used by different writing systems to encode spoken language; e.g. the Latin alphabet, is a script used by many different writing systems to express languages such as English, German, French and Malay. In the field of linguistics some research exists in the development of specific scripts (see related work section xx). However the goal of this project is not to explore the evolution from past to present, but to compare different current scripts in light of their historical connections.

Background

The visual aspect of written language played an important role in the history of writing. Before written language as we know it today was *proto-writing*, a picture writing system. Proto-writing uses ideograms or pictograms - graphic symbols that represent ideas or objects respectively - and does not directly translate to specific words from spoken language. The shape of the symbols conveyed the information and a reader would not necessarily need to know the spoken language of the writer in order to gather the information contained in the symbols. When most *true-writing* evolved (true-writing being a system in which the entire content of spoken language can be encoded), symbols that represent whole words were used to encode information (logograms). Again the shape of the symbol was related to the information, but unlike in picture-writing systems, they represented specific words of the writer's language. From there, the phonetic system stemmed, in which symbols represent sounds that are combined to phonetically construct words from spoken language. The shape of the symbol no longer has a specific meaning; rather it can be combined to create many different

words. Some scripts, such as the chinese characters or egyptian hieroglyphs, preserved both systems such that symbols can function both as logograms and as phonemes. In general, most scripts lost the connection that used to exist between the visual shape and the meaning. However the visual aspect of the scripts we use today stemmed from a long history of evolution.

Problem and Task Definition

Most of us do not encounter many foreign scripts in our day to day. Research estimates that 80% of the people in the world speak 1.69 languages on average (reference http://www.nytimes.com/2012/01/15/opinion/sunday/are-we-really-monolingual.html?_r=0). Considering many spoken languages share the same script, for example the romance languages, Germanic languages and many others use a Latin derived alphabet, the numbers for different scripts are probably higher. While there are several visualizations for spoken languages analysis on the one hand, and some visualizations for specific scripts and alphabets evolutions, I have yet to find a visualization that analyzes and allows comparison between different scripts.

The goal of this project is to expose different scripts and allow a visual exploration of them as a first step. Second to enable an exploration of historical and geographical data in light of the visual connection.

Result

The end result of this project is a web based interactive visualization that compares 11 scripts that are in use today, all of which stemmed from the Egyptian Hieroglyph origin. The visualization displays the evolutionary tree of the scripts on the left (as scraped from Wikipedia in the previously described project), followed by a heatmap mapping the similarity between all scripts (see figure 9).

Each character of each script was analyzed to compiled and an average was created to achieve a numerical representation of the scripts based on its lines and curves. The similarity score between scripts was calculated as a distance between the scripts with axes of lines and curves.

The heatmap diagonal is the intersection of scripts with themselves (identity) therefore we chose to use them in order to show a small scatter plot of the letters distribution. For each character set we used a bi kmeans algorithm to find 3 clusters based on the same distance function described above. The central letter of each cluster was chosen as a representative character for the script. All scripts are similarly scaled so this small multiple can also be used to compare scripts by letters distribution.

Selecting one of the rectangles representing the similarity between two scripts highlights the two scripts, displays an extra data section and shows a larger version of the scatter plots (see figure 10) (TBD). The extra data section shows the full character set, the similarity score on a scale and a world map with current script distribution (*add reference*).

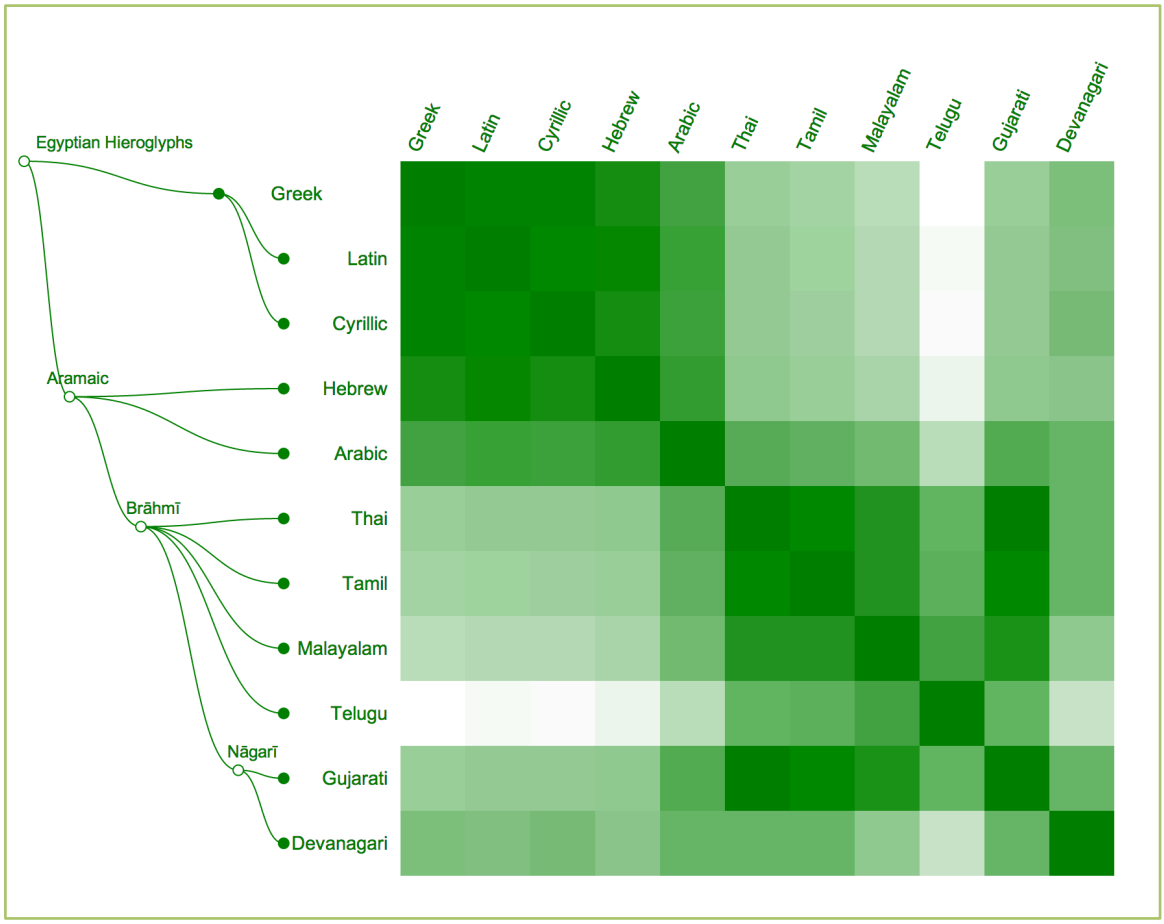

Figure 1: script evolution tree on the left leading to the comparison heatmap

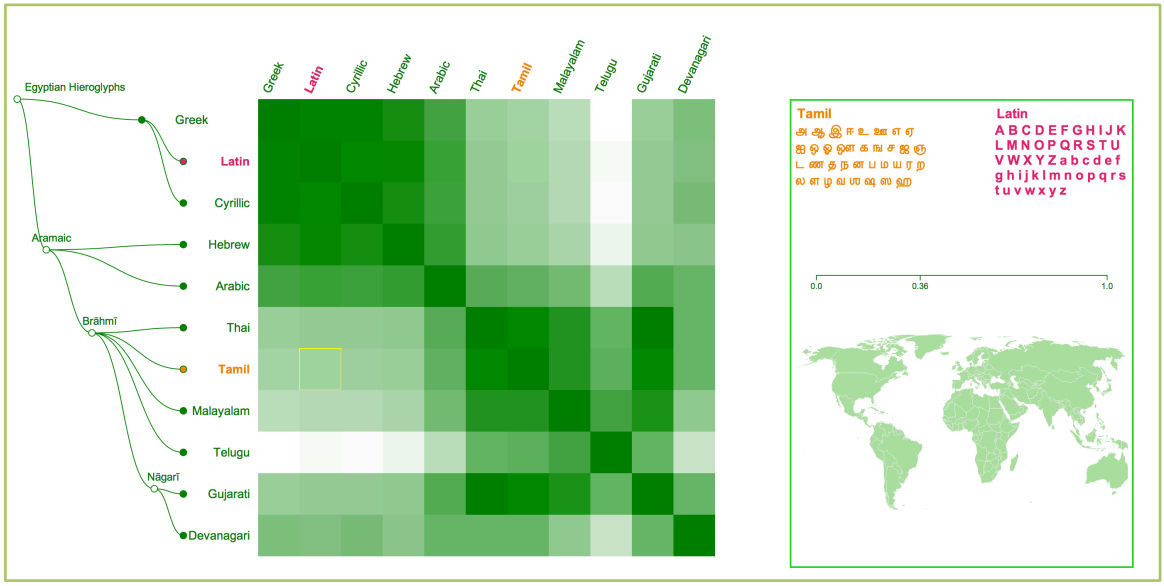

Figure 2: Tamil and Latin comparison selected with extra data on the right

Organization of this document

The related work in chapter 2 covers work done both in the field of linguistics and in the field of data visualization. Chapter 3 contains the implementation details relating technical details of the project. Chapter 4 describes the methods used to implement this project and the reasoning behind them. Chapter 5 is conclusions and summery.

Chapter 2 Related Work

In this section we review related work, from the domains of linguistics, visualizations and art. While most of these projects analyse text visually they do not provide insight into visual themes of the writing systems themselves.

Visualization

Rule-based Visual Mappings – with a Case Study on Poetry Visualization (Abdul-Rahman et al. 2013): the writers created a tool the analyses text in general and poetry in specific in a visual way. The text is analysed both phonetically and semantically including the connections and features of the different blocks. The approach also uses a visual tool in order to analyse text, and is an example of collaboration between two remote domains in order to create a new tool for analysing poetry. However this tool is used to investigate the inner structure of the text rather than it's visual form and does not currently promote the comparison between different text segments.

TBD text visualizations

Linguistics and language visualizations

The language tree on sentence structure done at Stanford university (http://humanexperience.stanford.edu/languagetree) uses visualization methods somewhat similar to the ones used in this thesis, in order to describe sentence structure relations between different spoken languages. The said article uses a tree to express the evolution of languages with color coding to indicate the sentence structure used in each. A map is displayed on the side to allow for the comparison to geographical data. However the color coding used in the map

does not correlate to the color encoding of the tree, making the visual comparison challenging.

The lexical distance between European languages, done by Teresa Elms based on linguistics research by Kostiantyn Tyshchenko (http://elms.wordpress.com/2008/03/04/lexical-distance-among-languages-of-europe/), shows another interesting comparison of spoken languages. The color and size is used in the visualization to relate historical information about those languages, while the distance and lines are used to relate the lexical distance between them. This allows the user to visually compare the relation, historically and lexically between the languages.

TBD
http://profs.etsmtl.ca/mmcguffin/research/2010-drawer/lopez-pacificvis2010-drawer.pdf

Clustering and heatmap code and technical reference:
http://isites.harvard.edu/fs/docs/icb.topic1362126.files/hac.pdf
http://blog.nextgenetics.net/?e=44

Wordcollider is an artistic visualization done by Moritz Heller inspired by particles collision that accelerates two phrases into each other, giving a different visual theme to each letter's phonetic characteristics (Figure 11). The end result is a visual representation of the two phrases phonetically. Though the approach is interesting and creates an intriguing visual "footprint" of the text, it does not allow for exploration or comparison and is in essence an attempt to visualize sounds (phonetic characteristics). Moreover this approach "visualizes" phonetics, pr in other words, attempts to give a visual form to sound, while this thesis focuses on analyzing the character sets themselves.

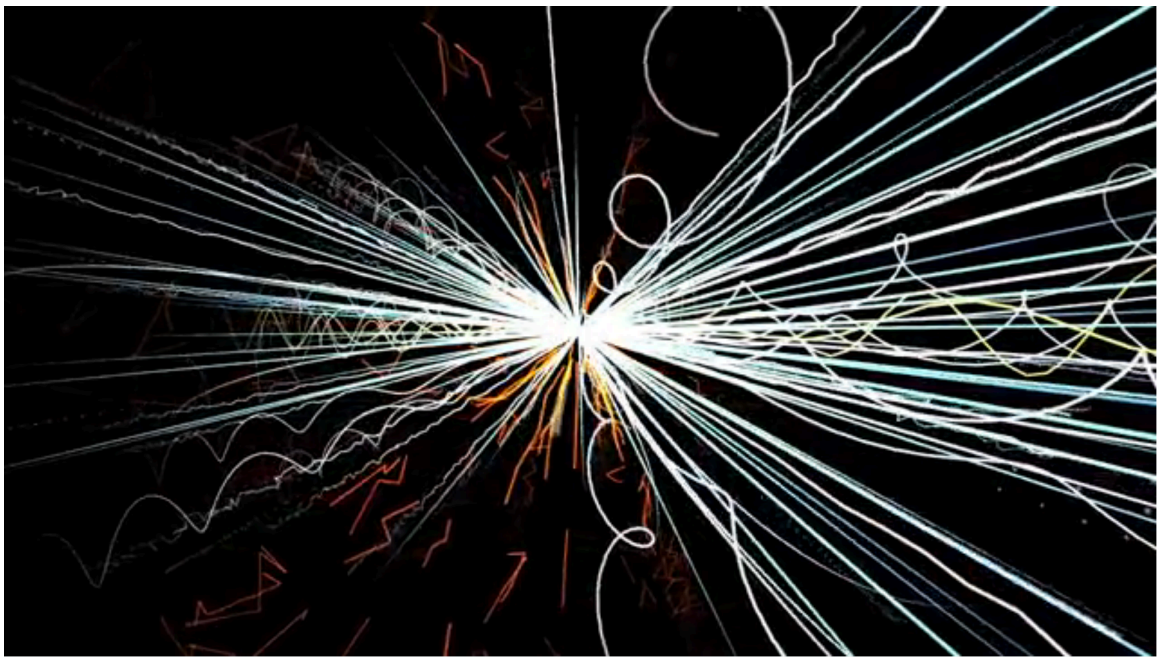

Figure 3: an image out of the wordcollider video

Null sets is an artwork that visualizes text files as images, representing their size and structure. The result is an abstract image of varying color that creates a visual rhythm that is based on the text and therefore represents the structure. Although this project takes a similar approach in the concept of visualizing text and despite the fact is allows for a comparison between different text segments, the end result is not derived by and does not indicate of the visual themes of the original text.

Scripts visualizations

Prof. Robert Fradkin at the University of Maryland has made several animations (executed by Charlie Seljos) that demonstrate the evolution different alphabets, for example the evolution of the latin alphabet from Phoenician. It visually shows how letters gain their visual form and displays the change of style over the course of history. It provides an interesting insight into how the characters transformed into what we know and use today. While it is a fascinating exploration of the history and evolution it offers no analysis of main visual

themes, and although the evolution of several alphabets is available, it is very difficult to compare the visual themes of different writing systems.

Adna Something, in her Hebrew book HarpatkaOT, lays down the transformation for each character in the Hebrew alphabet from the Canaitic origin to their modern form. The book also shows the “sibling” characters in Latin and Arabic scripts, which stemmed from the same origin. Thus this book allows an in depth, per character, analysis of these scripts.

Finally, Jack Kilmon offers font files for ancient scripts, which opens the possibility of a visual exploration of scripts that are no longer in use as a possible continuation of this project.

Previous projects

The next section describes two projects, *letters space distribution* and *the origin of languages and their scripts*. These projects were done as part of the visualization class with professor Hanspeter Pfister and together they form the groundwork that led to this thesis project.

Letters space distribution is a project that aims to explore the distribution in space of glyphs in text segments of varying languages. It allows for the exploration of the space distribution for a writing system used by a specific language and a comparison between the different languages. The space distribution is obtained based on text segments which provide sets of characters from each language. The bitmap, the image of the character as a 2D array of pixels, is retrieved for each character. Then the bitmaps of all the characters are overlayed, counting the number of times each pixel is visited. This generates a grey scale “heat map” showing how often a pixel is used by the characters in the given text segment and provides a visual insight into how the characters of that writing system are distributed in space. Note that all characters in given text

segments were used, therefore if a character repeated multiple times it had a stronger impact on the resulting space distribution. For example Hindi's space distribution (Figure 1) shows almost all characters have a line across the top, since these pixels are most strongly colored.

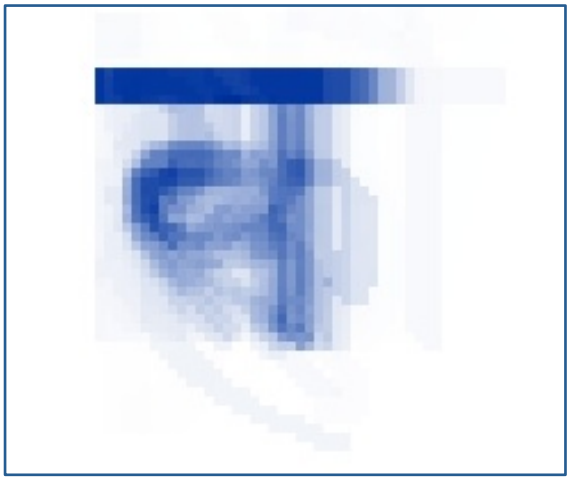

Figure 4: Hindi's space distribution

The first part of this visualization (Figure 2) contains a small multiple showing the space distribution derived from 10 different languages, on the first row - English, Portuguese, German, French and Malay which all use a latin derived alphabet; on the second row - Hebrew, Arabic, Hindi, Thai and Chinese which each use their own distinct writing system.

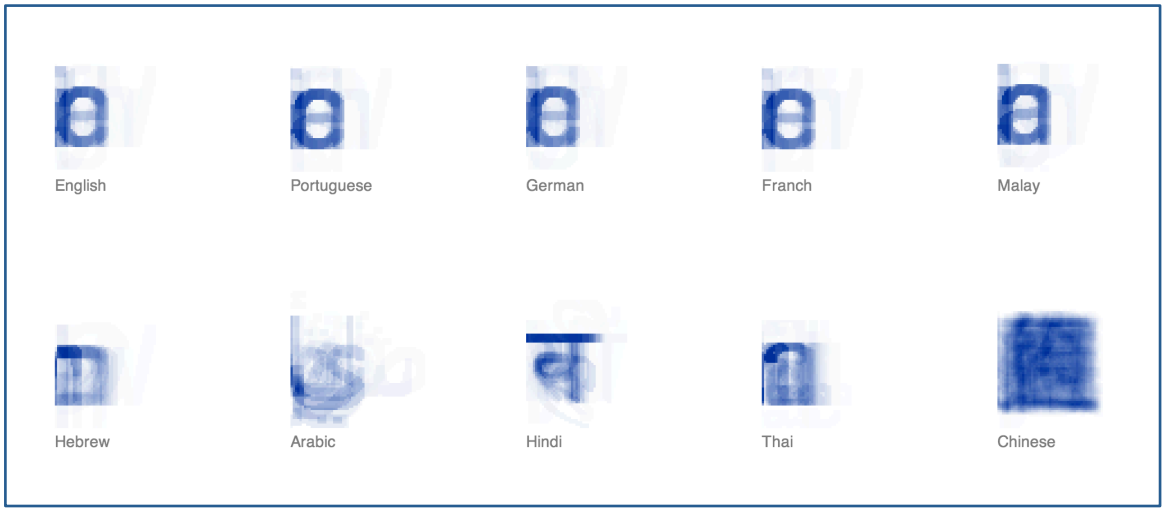

Figure 5: space distributions of 10 selected languages

Selecting a specific language brings up a page that displays a larger interactive version of the space distribution for that language on the left, all the characters that appear in the text as a bubble chart on the right, and the original text at the bottom of the page (Figure 3). The size of the bubble corresponds to the number of time that character appears in the text which is displayed at the bottom. Hovering over a pixel in the space distribution section on the left highlights that pixel in orange as well as the corresponding letters (i.e. letters that occupy the highlighted pixel) in the bubble chart on the right (Figure 4).

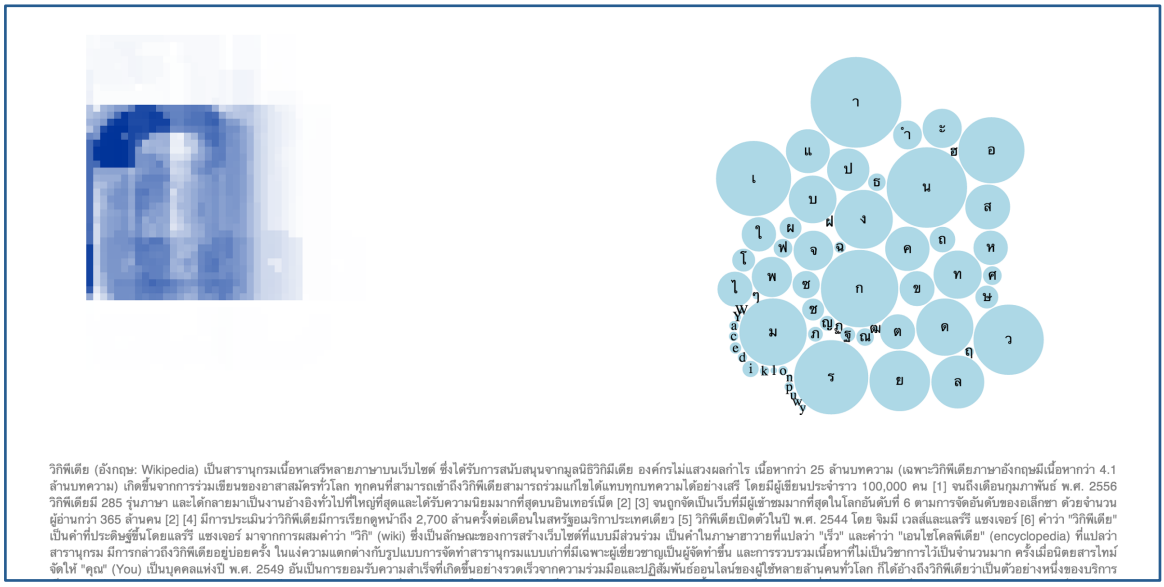

Figure 6: the page for Thai showing the space distribution on the left and the bubble chart on the right

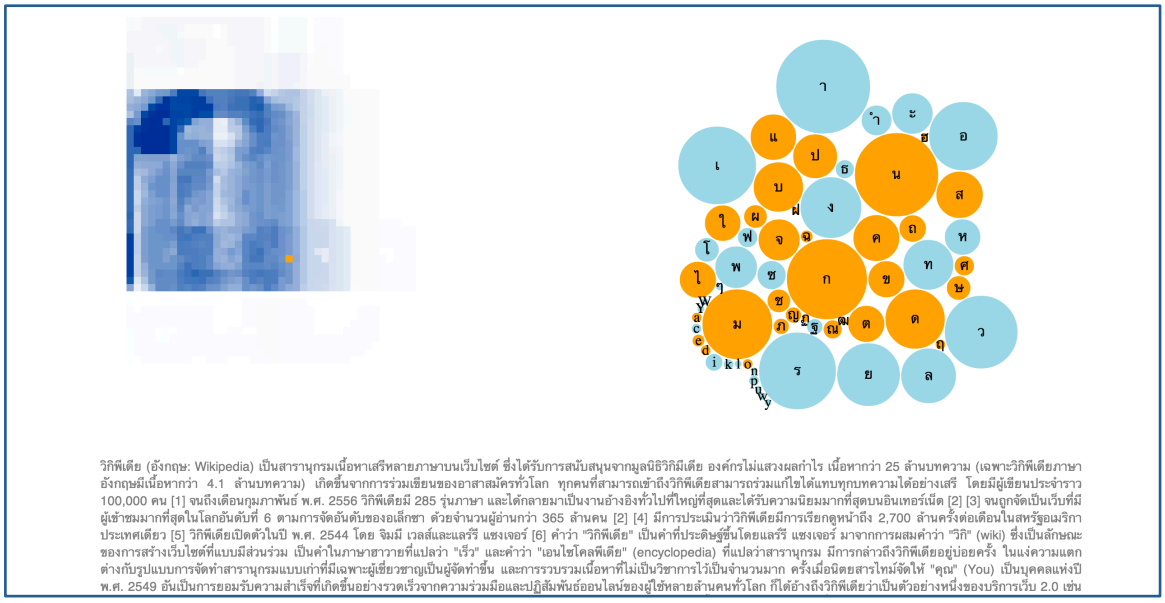

Figure 7: space distribution and characters of Thai with a specific pixel and corresponding letters highlighted

Since the space distribution is based on the frequency and use of specific characters, we expected to find a greater difference between the languages that use a latin derived alphabet, however this visualization shows that the space

distribution of these languages is remarkably similar. The Malay text is slightly different as it seems to use the letter *a* more heavily than *e*, unlike the other latin derived languages. This slight difference as well as the high similarity of the other latin derived scripts may be explained by the fact that the spoken languages English, Portuguese, German and French all have a common Indo-European origin and share a closer history than Malay. Chinese (Figure 5) was probably the most interesting to note as it was the most distinctly different - it showed very dense characters with even distribution over a square, and a significantly higher amount of characters used with a smaller occurrences rate.

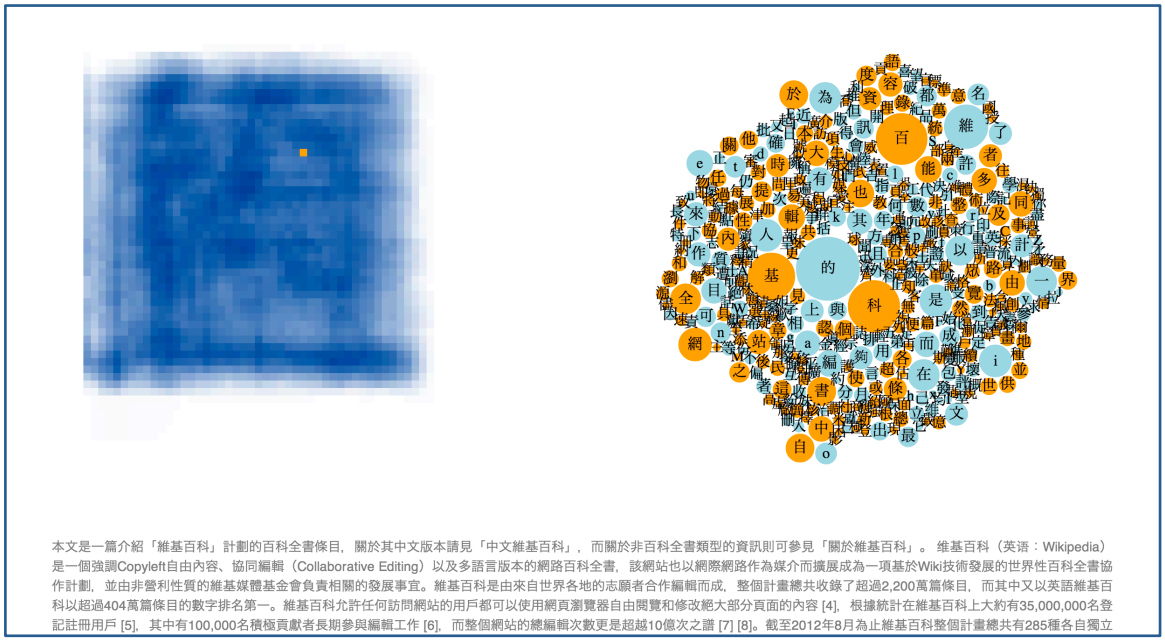

Figure 8: space distribution and characters of Chinese with a specific pixel and corresponding letters highlighted

The origin of languages and their scripts is a project that aims to collect and visualize the evolution tree of different languages and their writing systems. Origins of spoken languages are displayed on the left and origins of writing systems are displayed on the right. Clicking on a language family brings up the

tree for that family (Figure 6). Languages that have an alphabet connected to it are larger and in full color. Selecting one of those nodes displays the alphabet node at the same level, connected to it's origin. The selected node is highlighted while the rest are greyed out (Figure 7). At the bottom there are details on demand - a frame displaying the wikipedia article about the language or the alphabet (Figure 8). Selecting the alphabet will open the writing system tree, with that node already selected. When a script tree is open, selecting one of the selectable scripts will present a list of all languages that use that alphabet.

Figure 9: the Turkic origin evolution tree

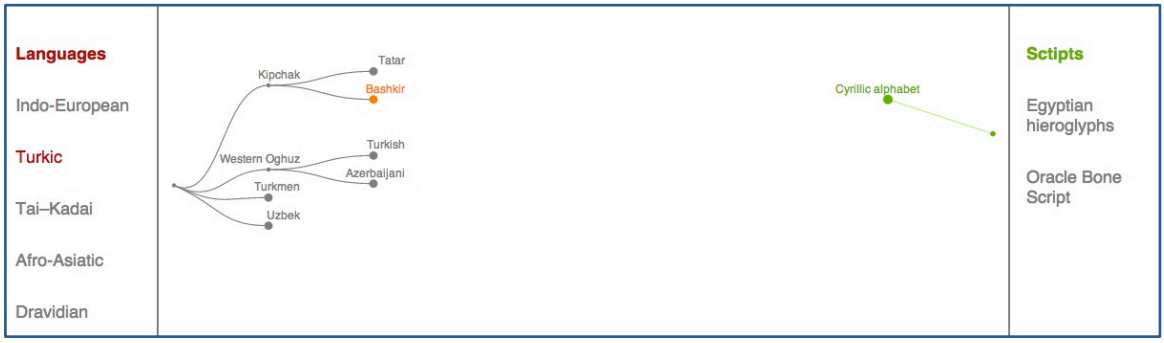

Figure 10: the language Bashkir is selected and the corresponding alphabet is displayed

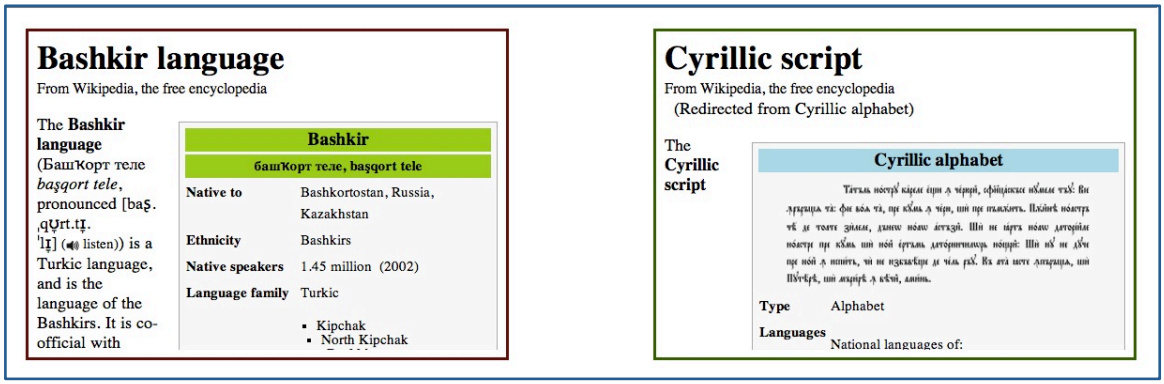

Figure 11: additional information from Wikipedia on the selected language and script

Implementation

The chapter describes the implementation details from a technical perspective. The project has two implementation parts – data gathering and analysis and web-based visualization.

Data Collection and Analysis

The data was collected and analyzed using python with the help of the Numpy library for matrix manipulation and the FreeType_py library for glyph information retrieval. Using a specific ttf file and Unicode index per scripts, the code analyzes the characters and outputs to the file output.json. The two secondary processing files – scripts_heatmap_results.py and scripts_char_relation.py, then use this output. The code in scripts_heatmap_results uses the data in order to generate the required information for the heatmap, and captures this information in the json file scripts_heatmap.json. The scripts_char_relations file uses the data in the output file to generate chars relations information per script, each saved into a separate file under the chars folder. Lastly, char_clustering.py uses the individual json files from the previous step in order to create the clusters of each script. These are saved to individual script files under the chars_outputs folder (see figure 12).

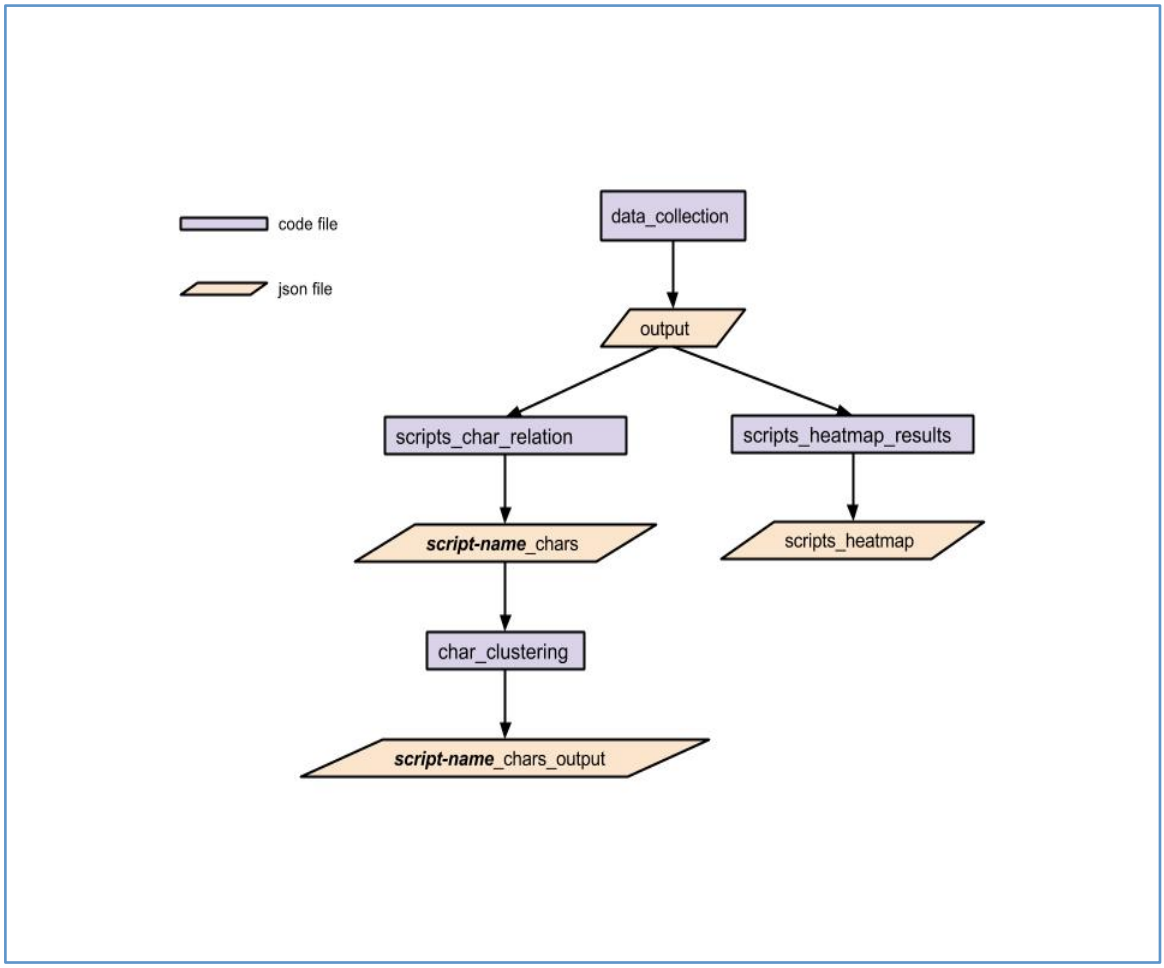

Figure 12: data flow in python code for data collection and analysis

The information for the evolution tree was collected as part of the project *the origin of languages and their scripts* described in the related work chapter. The data was scrapped from Wikipedia using the Pattern library Wikipedia API (*add reference*). The file created and used in that project was adjusted to include only the scripts that are compared in this thesis. More details on scripts selection can be found in the methods section of this document.

Data Visualization

The visualization is a web-based application built with html, css and javascript, using the D3 javascript library (*add reference*). The output data files created in

the python data section is transferred to the visualization folder under the data folder to be used by the application. The index.html file in the root of the folder is a bare-boned html file that does little more than importing the different sections and calling into the different javascript functions in the js folder. The different parts of the visualization are implemented in separate files and are generated separately, which provides encapsulation of each element. The parts are:

- *tree* - which implements the evolution tree (*add reference to technical resource used*)
- *heatmap* – which implements the main heatmap comparing the scripts along with the top labels (*add reference to technical resource used: http://blog.nextgenetics.net/?e=44*)
- *data* – which implements the extra data section on the right and includes the world map based on the topojson library (*add reference*) and uses a static data file, which contains static data on the different scripts such as links to their Wikipedia articles and geographic information.
- *char_relation* – which implements the per-script scatter plots, both the large version which includes the letters and the simple smaller version displayed inside the heatmap. This file uses the clusters of each script generated by the char_cluster python file, in order to display the clusters and the representative character.

All these sections have a javascript file and a css file in the respective js and css folders. General styles are captured in the style.css file, which contains styles that are either application wide or are used by more than one section.

The root folder also contains a README file listing the folder structure and has running instructions. In order to run the application locally the python simple web server was used. For an easy run a simple sh file called run_server was added to the root directory.

Application folder structure:

index.html

→ css

- tree.css
- heatmap.css
- data.css
- char_relation.css
- style.css

→ js

- tree.js
- heatmap.js
- data.js
- char_relation.js

→ data

- tree.json
- scripts_heatmap.json
- static_data.json
- world.json