Abstract

The company Backflip has built a system that retrieves tweets and subtitles about UK TV series and classifies the sentiment of these Twitter messages. They designated the problem of visualising this data in a compact way, in order to identify TV fashions. This project aims to deliver and evaluate visualisation methods that accommodate the inspection of trends on TV related Twitter content.

Moreover, this project examines the study of *Information Visualisation* on large-scale data. Throughout the exploration of the different methods of visualising the two datasets, of Twitter activity and subtitles, conclusions are made for the visualisation methods and technologies for this data. On the other hand, a visualisation toolbox is created for composing the different visualisations into one query system. This toolbox aims to propose an interface for revealing insights into people's behaviour via visualisations for any Twitter data.

Finally, three visualisations that allow the display of sentiment and volume of tweets for TV series over time are evaluated by an assessment experiment. These visualisations compose some of the main deliverables of this project, since they can correlate the datasets of Twitter activity and subtitles time-wise and improve trends discovery. This argument is justified by Backflip by the fact that time-wise visualisations of the two datasets allow data analysts to overview how people react (sentiment analysis, volume) on what is played on TV (subtitles). Therefore, the assessment experiment evaluates the most adequate of these three visualisations regarding information retrieval, by the level people can perform visual tasks on them.

The contributions of this project are related to the study of *Information Visualisation* and specifically are:

- Examination of the visualisation approaches that enhance information retrieval
- Review of commonly used visualisation methods
- Examination of the Twitter content in order to find the data that can be visualised, providing valuable insights about people's behaviour
- Testing of different visualisation libraries on various data and analysis of their strengths and weaknesses
- Composition of a toolbox gathering the implemented visualisations, revealing fashions via visualisations for any Twitter data
- Correlation of TV Twitter data with subtitles in time-series visualisations, enhancing trends analysis
- Creation of an assessment experiment to evaluate interactive visualisations
- Evaluation of three time-series visualisations about Twitter data in order to find the most capable for knowledge discovery

Table of Contents

List of Figures

List of Tables

1 Introduction

"I had (and still have) a dream that the Web could be less of a television channel and more of an interactive sea of shared knowledge."

– Tim Berners Lee, 1995

Since their establishment, Social Network websites have been a keen interest for millions of users, who share plenty of information about their daily activities (Boyd et al., 2007). Companies acknowledge the fact that they can use the great amounts of data from such websites and are interested in utilising it to retrieve insights about products (Leskovec et al., 2007).

This project focuses to the exploration of trends from Twitter TV related activity. The key to the transformation of Twitter data to fashions about TV series is the *Information Visualisation*. This section describes this project with its aims and objectives and gives the outline of this report.

1.1 Aims and objectives

Backflip introduced the problem of finding trends about TV series from relevant Twitter activity. Hence, they designed a system that retrieves Twitter content related to UK TV series and classifies these Twitter messages, also known as tweets, according to their sentiment. Furthermore, they developed another system that fetches subtitles from each UK TV series and refines keywords about them, also known as concepts. Regarding to their setup, my goal was to find a way to use and combine all the provided data via visualisation methods, in a way that time-wise relations between Twitter content and concepts enhance trends discovery.

As for the use of visualisation methods, the motivation came from the company's realisation that information displayed as plain text is poor for knowledge retrieval. In the same manner, according to Pocock (1981), sight is the sense that helps us perceive the nine-tenths of our external world knowledge. Therefore, this project aims to find the visualisation methods that catch the data analyst's attention and help him understand large amounts of different data.

More specifically, the objectives of this project are:

- Explore which visualisation methods are more efficient for large-scale data
- Provide interactive visualisations that augment the review of different data
- Correlate in graphs the datasets of Twitter content and subtitles in order to assist the observation of their relations
- Correlate the demographics, provided by the Twitter content, with the sentiment and volume of tweets and create visualisations that improve the exploration of people's tendencies
- Compose a query system, also referred as toolbox, that gathers all different visualisations in one interface
- Evaluate the effectiveness of the time-wise visualisations regarding information retrieval
- Propose the most adequate graph for the time-wise analysis of Twitter content, for the purposes of trends discovery

Overall, this project explores the contribution of interactive visualisations in knowledge-base decision making. Web 2.0 technologies, as social networks, provide massive data, but only efficient visual techniques can transform such large-scale data to wisdom (Shedroff, 1994). As a result, this research examines visualisation methods that improve knowledge retrieval on

large datasets. On the other hand, another outcome of this project is the evaluation of visualisations that illustrate social network activity over time. This evaluation aims to conclude about time-series visualisations that enhance the discovery of trends from such data. Finally, this project develops a visualisation toolbox for the composition of the most effective visualisations about Twitter content in one interface. This toolbox is meant to be adjustable on Twitter data related to a domain, for the review of people's behaviours regarding this domain.

Consequently, the distinct deliverables of this project are:

- A toolbox, which gathers all the visualisations and statistics about TV Twitter content. This toolbox provides query methods in order to refine results about UK TV channels, series and episodes.
- Time-series visualisations, which enhance the information retrieval regarding the two datasets. Both Computer scientists and people outside this area, as TV producers, are meant to be the end users of this deliverable.
- Evaluation of the time-series visualisations.

1.2 Outline

This report consists of six main sections. It begins, in Section 2, with an overview of the definitions that play a key role to the clarification of the problem. In the same section, it continues with an in-depth description of Information Visualisation and specific approaches for visualising data. Lastly in this section, work related to this project is noted with an analysis of the inspirations that it brought to this project.

In Section 3, the problem is defined in relation with the data, use cases and challenges it concerns. This section helps the reader understand the state of the problem and what can be done to resolve it. As follows, Section 4 specifies the implementation that aims to achieve the initial goals and overcome the given challenges. In particular, the deliverables of this project are being described as for the features they introduce.

Moreover, in Section 5, the evaluation methodology is explained and the results of its procedures are stated. In the same section, the results are determined for their significance, with the use of statistical tests. Consequently, in Section 6 the outcomes of this project are estimated, clarifying the achievements regarding the stated objectives of this project. Finally, in the *Conclusion* section, the project is overviewed and further possible improvements are declared.

2 Background

This section briefly defines the key domains related to this project. The following paragraphs describe the terminology that will be used throughout this project. The word or phrase in italics is the keyword, which is followed by the corresponding definition. The described definitions aim to help the reader understand the areas of study that this project involves. Also, related work attempts to provide a basis for this project's appraisal.

2.1 Overview and definitions

Social Network is a group of nodes or members, who are linked with one or more kinds of relations (Wasserman et al., 1994). Social networking websites are online services providing utilities as text, picture and video sharing among members. In such websites members use these utilities to share information about their everyday life. This project examines text-sharing activity extracted from the social network Twitter, of which the definition follows.

Twitter is a social networking website, which allows its members to post messages of up to 140 characters. Because of its upper limit about characters, Twitter is also referred to as a microblogging service. According to the research by Makice (2009), microblogging services try to provide a mixture of the three established technologies: Instant Relay Chat (IRC), IM chat status messages, and mobile phones. The same research states that Twitter's restriction about post characters adopts the lowest limitation of short message services (SMS), a text messaging service that is mostly a component of mobile phones.

Because of its directness and simplicity, Twitter quickly became a popular social networking service, reaching over 140 million active users and more than 340 tweets per day (Twitter Team, 2012). Hence, Backflip chose it as the source for social network activity related to TV series. In order to better understand the later described mechanism that Backflip uses to retrieve tweets and some preprocessing steps I applied on the retrieved data, Table 1 provides definitions for Twitter's special characters and terms by Singh et al. (2012).

Table 1: Special characters and terms as described by Singh et al. (2012)

Special Character/Term	Definition
@	Designates a reference on another user. This reference is clickable and leads to the related user account. For example, in "*I love the picture @Alice uploaded"* the word "@Alice" is clickable and leads to the account related to the username "Alice".
#	This character with a following keyword is known as "Hash tag" and is used to mention a common subject of discussion. For example, if a user writes "#superbowl" in his tweet, he is implying that his tweet is relevant to this topic. Hash tags are clickable and link to a list of tweets containing them.
RT @	This term comes from the word "retweet" and is used for forwarding a tweet. In other words, a user can resend ("retweet") another user's message to his own profile using this term as follows:

Special Character/Term	Definition
	"RT @[Username] [*Repeated Tweet]*".
DM @	This term comes from the phrase "Direct Message" and is used to send private messages directly to user accounts. For instance, the tweet *"DM @Peter I need to talk to you ASAP"*, is displayed on the account related to the username "Peter" and only the user with this username has the privilege to see it.

Television (TV) Series is also known as TV programme or television show and describes a group of TV productions that are broadcasted one after the other for a time period in TV. A TV production follows another specific TV production and each of them is called *TV Episode*. TV Series are showed in *TV channels*, which are physical or virtual channels where TV networks are allocated.

Named Entity Recognition (NER), also known as entity identification and entity extraction, is the procedure of processing text to discover and categorise names in it (Northman, 2008). Backflip uses NER to create nametags about TV series in order to use them in queries for Twitter content. Similarly, Backflip, after retrieving subtitles for TV series with an appropriate mechanism, uses NER to create nametags for each subtitle, which nametags are then called *Concepts*.

Datasift is an online service that Backflip uses to extract Twitter content. As the company argues, the *Twitter API*, which is a similar service provided by Twitter, is heavily rate-limited, meaning calls to it range between 150 and 350 queries per hour, and therefore not appropriate for running a production service. On the other hand, Datasift provides a rate limit of 500,000 Tweets per 24 hours. Backflip queries Datasift for Twitter content using hashtags and phrases generated by NER from TV series descriptions. Before querying Datasift, the hashtags are reviewed by a human who may add further tags.

Time-series charts or *visualisations* are two-dimensional charts, in which one of the two axes defines the time. In other words, in *time-series charts* each point of this chart corresponds to a specific time that is determined by one of the two axes of the chart. The time can involve intervals of milliseconds, seconds, minutes, hours, days, months and years. The libraries that were used in this project use this term to designate that one of the two measures in the two-dimensional chart denotes the sequence of time.

Word-cloud, also known as *Tag-cloud*, is collection of words represented closely together in a way that the overall shape looks like a cloud. These words come from a dataset or a text file, where each of it can occur more than once. For the shape to be similar to a cloud, each word has a different size in this representation, according the number of times it occurred in the dataset/ text. Its purpose is to visualise an overview of the text, highlighting the most common words in it (Grudin et al., 2005).

Sentiment Analysis, according to Bifet et al. (2010), is a classification task that attempts to categorise text into two classes, positive and negative, depending on the feeling that is attached to it. Backflip applies a sentiment analysis algorithm on every tweet, the *Multinomial Naïve Bayes classifier*, in order to resolve whether the tweet has a positive or negative sentiment related to it. Despite the short size of tweets, their sentiment analysis is a hard process, since a tweet may contain rich information in a compact form, which may include both positive and negative feelings (Bifet et al., 2010). Therefore, the sentiment analysis

provided by Backflip, may introduce misclassifications, but this analysis is not part of this study.

The *Multinomial Naïve Bayes classifier* is the most popular algorithm for tweets sentiment analysis, since it provides the best accuracy for this domain compared to other text classification algorithms (Go et al., 2009). This classifier is based on Bayes' theorem, which denotes the following probability:

$$P(w|t) = \frac{P(w|s) \times P(s)}{P(w)}$$

where:

- P(s) – probability of a sentiment s to occur (e.g. P(positive))
- P(w) – probability of a word w to occur (e.g. P("Dexter"))
- P(s|w) – probability of a sentiment s to occur given it is described by vector of attributes w (e.g. P(positive|"Dexter"))
- P(w|s) – probability of an instance with attributes described by w given it comes from the sentiment s (e.g. P("Dexter"|positive)) (Pak et al., 2010).

For unseen words, where the zero-frequency problem arises, the *Laplace correction*[1] is used. The above probability is then used in the following formula to classify a tweet, based on its composition of words.

$$sentiment = argmax P(s_i) \prod_j P(w_j|s_i)$$

where:

- sentiment – final classification of a tweet
- $P(s_i)$ – probability of a sentiment i to occur
- $\prod_j P(w_i|s_i) = \prod_j P(w_1, \dots, w_n|s_i)$ - the sum of probabilities described by attribute w_i given they come from the sentiment s_i.

The approach by Backflip is to have two classes for a tweet, positive and negative. They use 2 million tweets hand-classified into positive and negative sentiment to train the classifier. Eventually, the probabilities that define positive sentiment have no sign, whereas the probabilities for negative sentiment are assigned the minus sign. This sign attachment is used so that the final probability, which will be a sum of all the probabilities, will either be a positive value or a negative value, determining positive and negative sentiment respectively.

2.2 Information Visualisation

As Keim (2002) states, large amount of data without any visual representation of it grows to data "dumps". The same research argues that visual data exploration is useful for retrieving insights about the data, even when the aims for the insights are vague. Accordingly, this project intends to enhance the analysis of TV insights from large-scale social network data using visual data exploration or in other words *Information Visualisation*.

To begin with, visualisation is the process of making data easier to perceive using intentional designs that trigger the senses (Dix, 2011). As computer graphics evolved, visualisation methods became easier to use and widely accessible, establishing visualisation as a concrete and considerable case of study. Consequently, visualisation of data is a distinct procedure

[1] A way of handling zero probability values (P(w|s)=0), adding 1 to the numerator and the size of the Vocabulary to the dominator.

with specific steps for producing a meaningful effect. These steps are defined by Ware (2004) as:

- Step 1: Gathering and storage of data
- Step 2: Preprocessing techniques to transform data into meaningful definitions
- Step 3: Handling of graphic hardware and software that enable the representation of the visualization.
- Step 4: Analyzing information by the final observer

The above steps are represented in Figure1, inspired by Ware's equivalent illustration. In the final analysis, Ware acknowledges the gathering of data (Step 1) as the most demanding and time-consuming process. Note that this step, as well as the analysis by the final observer, is related with the according physical and social environment.

Information Visualisation, also known as *InfoVis*, is a visualisation method that was firstly introduced at the 1980s by the researchers of Xerox Palo Alto Research Center to determine the visual representation of large-scale connections, which associates with knowledge retrieval (Mazza, 2009). The term *InfoVis* is used to separate this method from scientific visualisation methods. This separation is required to determine the different data that these two methods involve. Specifically, scientific visualisation methods mostly involve data in the form of fields of numbers of vectors, whereas *InfoVis* methods include data more composite regarding their format. For instance, the data of *InfoVis* methods may combine categorical data, as female/male, with continuous data, as time (Dix, 2011).

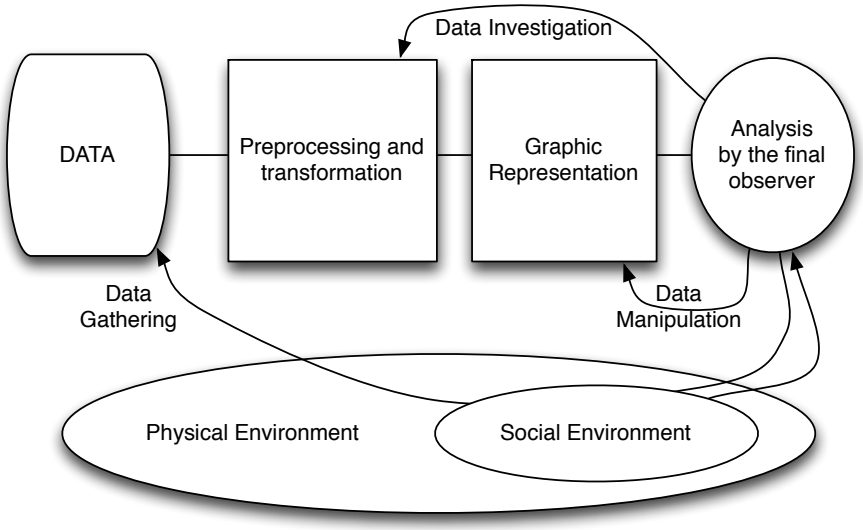

Figure 1: A schematic diagram of the visualization process.

In the same sense, Keim (2002) proposed a designation of the visualisation methods that are in general approved, which reflects the progress of *InfoVis*. These methods are defined as:

- *Standard 2D/3D displays*, which represent relationships of two or three entities respectively, e.g. bar charts,
- *Geometrically transformed displays*, which illustrates the correlation between a pair of variables, e.g. a scatter-plot matrix[2]
- *Icon based displays*, which map data variables onto graphic features, whether they are geometric (e.g. shape, size, orientation) or non-geometric (e.g. color and texture),

[2] A type of mathematical diagram using Cartesian coordinates to display values for two variables for a set of data.

- *Dense pixel displays*, which use pixels to represent dimension values by coloring and gathering them in alongside regions,
- *Stacked displays*, which order data by some scale.

During this research, *InfoVis* follows the procedure proposed by Ware (2004), as earlier described. Therefore, the Twitter data relevant to TV series is

- Stored,
- Preprocessed and transformed to meaningful terms, and
- Represented to visual graphs.

This process is explained in more detail in Section 4 and was applied so that the final observer can understand insights about the data. In addition, in this project all the different visualisation methods defined by Keim (2002) are tested to the given data, with every visualisation developing one or more of them.

2.3 Visualisation approaches

2.3.1 Interaction in visualisation

Although the described visualisation methods are sufficient to represent information in a comprehensive way, interaction in visualisations can further improve the analysis of large networks (Chuah et al., 1996), like Twitter. Interaction designates the human activity on a visualisation display that causes alterations to the initial design. These alterations allow the user to identify various aspects of the shown data and accommodate the examination of the given information.

Likewise, interaction and distortion techniques can lead to an effective analysis of data. Interaction techniques refer to the direct manipulations of the visualisation by the data analyst in order to augment the examination of data. In the same sense, interaction may correspond to techniques that allow the combination of different visualisations and data. On the other hand, distortion techniques concern methods that allow more detailed exploration of the data, by supporting actions like focus or detailed data on demand (Keim, 2002). The same research reports five main interaction and distortion techniques, which are:

- *Dynamic Projections*, which describe the action of changing the viewing angle of a visualisation, in order to further examine multidimensional data.
- *Interactive Filtering*, which refers to the extraction of a specific subset of the visualised dataset. This process can be done either by direct selection of the subset (*browsing*) or by specification of the options one wants the subset to have (*querying*).
- *Interactive Zooming*, which means the transformation of a compressed overview of the data to a more detailed display of it. This interaction is widely known and applied, giving the opportunity to take a closer look to the data and retrieve more detailed information. Although, this interaction usually refers to the action of getting a closer display of a subset it also means getting more detailed information about it, as you get closer to it.
- *Interactive Distortion*, which stands for the utilities that offer an overview of the data, even when the data is zoomed in.
- *Interactive Linking and Brushing*, which has to do with the connection of different visualisations together. In other words, this interaction combines visualisations in a way that a change to one visualisation can lead to changes on the rest linked visualisations.

All these techniques, except *Dynamic Projections*, were tested to the given data of this project. The *Dynamic Projections* found no application, because in my point of view such interactions are difficult to understand by non-scientists. Specifically, this technique is mostly

used to augment the exploration of multidimensional data, for which I chose less complicated visualisations, as analysed in Section 4, which do not require different viewing angles for information retrieval.

In Section 4 more details are given for the use of the rest techniques. Some of them were implicit in the used visualisation libraries and in most cases I added the rest. *Interactive Linking and Brushing* was applied until a point, since, according to my implementation, it does not follow the rule that changes in a visualisation cause changes to all the rest linked ones. Only alterations in one main visualisation cause alterations to the rest.

2.3.2 Visualisation technologies

In order to apply the described visualisation methods and interaction techniques, research has been done for the available freeware visualisation technologies. Firstly, the most popular technologies offering a user interface to create visualisations are reviewed for their utilities. Consequently, the technology that embraces most of the characteristics of the freeware interfaces is stated, as also the reasons for using it in this project.

To begin with, the review by Machlis (2011) and the technical report by Boer et al. (2011) analyse some free designing tools for interactive visualisations. These tools provide a user interface for creating visualisations, which make them easier to use, since they do not require programming skills. Briefly, the most commonly used such freeware tools are:

- *Google Fusion Tables*: an online service provided by Google, whose primitive operation is to transform data tables into charts or geographical maps. Its main advantages are that it requires little time to transform text data into charts and it provides automatic transformation of addresses to geocodes in order to visualise data on maps. Its main disadvantage is the restriction about the size of data a user can upload (100Mb maximum), which corresponds to a limit on the data that can be visualised.
- *Tableau Public*: desktop software for data visualisation. Its basic feature is to convert relational databases and spreadsheets to interactive visualisations written in JavaScript. As *Google Fusion Tables*, it is easy to use and provides many different visualisations. However, it obligates the user to upload his data in a publicly accessible database, if he is using the free edition.
- *Many Eyes*: is similar with the previous online tools for designing interactive visualisations and was introduced by IBM. It basically shifts data to interactive visualisations, which are written in JavaScript, Java or Adobe Flash[3]. Its main advantage is the rich library of different charts available to use, but its drawback is the limit on the size of data a user can upload, which is 5Mb.
- *VIDI*: a visualisation tool that is designed for the content management system Drupal and creates visualisations in webpage formats. Similarly to *Many Eyes* it offers a wide range of different visualisations, but has the restriction for the data not to exceed the size of 30Mb.

These technologies as argued by Machlis (2011) and Boer et al. (2011) are widely known, freeware and easy to use, as they do not require programming skills. This project aims to find the technology that provides most of the visualisations and features of these freeware visualisation applications, while overcoming the constraints about the size and the secrecy of data.

In order to achieve that, I used freeware JavaScript libraries that are independent of data size, provide a wide range of visualisations, can take data from any source and are cross-browser

[3] Multimedia platform providing animation, video and interactivity between elements

compatible. The used libraries embrace the innovations introduced by HTML5[4], providing scalable visualisations that are lightweight and only require JavaScript to run.

Specifically, all libraries use the canvas element of HTML5, which allows creation of graphic animations with JavaScript. This element makes the visualisation independent of plug-in based applications like Adobe Flash, Microsoft Silverlight and Oracle-Sun JavaFX, which are commonly used in other visualisation methods (Boulos et al., 2010). Furthermore, some of the libraries provide export of the visualisations to Scalable Vector Graphics (SVG), which is also an innovation introduced by HTML5 and allows a visualisation to be scalable for its big display or print. The latter feature is important for this project, as Backflip stated the need of visualisations that allow scalable printing.

2.4 Related work

Trends discovery from Twitter content via visualisation methods has been a research subject before this project. Specifically, Cheong et al. (2009) in their research used a methodology that takes the most discussed topics, also known as *trends* provided by Twitter, with their related tweets, and used artificial intelligence-based data mining methods to classify them in order to retrieve fashions regarding what is going on in the real world. These data mining methods explore the device used, the gender and the country that are related to each tweet via visualisation methods, meaning bar charts and self-organising maps[5], to promote knowledge retrieval.

Furthermore, the designed toolbox for this project is an inspiration of two different online services, the *Tweet Archivist*[6] and the *Sentiment 140*[7]. The first one, as its name defines, is an archive of tweets in an online database available for statistics analysis. This application allows the user to retrieve visualisations about statistics of tweets that include a specific word or phrase, for instance “Coca Cola”. These visualisations are about the volume of tweets, the top Twitter users, the top used words, the top URLs where the tweets came from, the device used for sending the tweet and the comparison of tweets with retweets.

Moreover, the second application offers a query system that retrieves sentiment analysis of tweets. Particularly, this application provides a search engine where a user can query about a word or phrase and the system retrieves tweets that include this word or phrase and the overall sentiment analysis about these tweets. The results of a query are two graphs that display the comparison of Negative with Positive tweets, one by percentage and one by volume of tweets, and the list of the tweets from which the sentiment analysis was retrieved. Each tweet is classified to be either Positive, Negative or Neutral with green, red and white colors in the visualisations respectively. This is a useful tool for product analysis, as a user can retrieve sentiment analysis about a product and make conclusions about its effect on Twitter users.

Consequently, this project delivers an online toolbox that manages to combine the aforementioned ideas for the exploration of TV trends. This toolbox is unique to the features it introduces, as it associates sentiment analysis, volume and statistics about tweets, while providing high-quality interactive visualisations for data exploration. As in the Cheong et al. (2009) research tweets are mapped to what is going on in real world, this project maps tweets to what is going on in TV, exploiting TV subtitles.

In addition, another originality of this project is the more complex analytics that are provided, as retrieval of statistics for UK cities and other unique analyses that are further explained in

[4] Hypertext Markup Language Version 5
[5] Also known as SOM or self-organising feature map (SOFM). It visualises an artificial neural network trained by unsupervised learning on a map.
[6] http://www.tweetarchivist.com
[7] http://www.sentiment140.com

Section 4. The domain of this project is TV Twitter content, but the methodology and the implemented toolbox is meant to be adjustable on any other Twitter content for product analysis. Finally, another contribution of this project is the evaluation methodology for retrieving whether the used time-series visualisations achieve a better understanding of the data by the end users.

2.5 Summary

This section described in brief the domains of this project, clarifying the aims that were aforementioned in Section 1. Most of the terms that will be noted in this document were explained in this section, so the reader has a basis of understanding. Furthermore, the *InfoVis* study was in depth analysed, as it is the main study field that this project is related to.

Likewise, the forms of interaction in visualisations were examined, as they are part of the final deliverables that are evaluated for the conclusions of this project. In order to inform the reader about what is mostly used for *InfoVis* and is freeware, some visualisation technologies were also concisely described. Finally, this section ended with the related work that became an inspiration for this project's deliverables.

3 Problem Analysis

In this section, the problem is analysed in more depth regarding its initial state and the opportunities and limitations it introduces. Particularly, the datasets, the use cases and the challenges of this project are explained, so the reader understands the potentials of this domain and my approach to resolve this problem.

3.1 Datasets

Backflip provided the data for this project. As for Twitter content, the data I was given corresponds to real tweets posted during a specific time interval on Twitter, related to UK TV series. The time interval, during which the tweets were posted, was between 23.30 Sunday 24th June and 01.49 Tuesday 26th June. For Backflip, this time interval matches the day Monday 25th June on TV. This is because they fetch tweets half an hour before a day begins and half an hour after the last played TV series for that day finishes. Their approach to define a day on TV like this has to do with the fact that people tend to discuss about a TV series before it is on air and for some time after it is over.

As mentioned in Section 2, in order to retrieve Twitter data Backflip uses the *Datasift* service. To query this service Backflip uses keywords that are taken from TV series descriptions using NER. These keywords may include the name of the TV series, channel and episode or the names of the people involved in this TV series. In order to match the tweets to a specific episode of a TV series, Backflip retrieves tweets using the mentioned keywords for half an hour before the episode is on air, during the on-air time of this episode, and for half an hour after it is finished. This approach attempts to reassure that the tweets including the keywords match the specific episode, because of the time relation.

As a result, the Twitter dataset I was given includes the outcomes of *Datasift* for the TV series played on Monday 25th June on each UK TV channel. Also, this dataset contains the sentiment analysis score for each tweet as estimated by the Naïve Bayes classifier. In more detail, the dataset includes the names of the TV series, the episode, the channel, the times that the retrieval of tweets began (*Transmission Start Time*) and ended (*Transmission End Time*), the creation time of the tweet, the user (*From User*) that sent it, its text (*Twitter text*), the sentiment analysis score that was assigned to it, and the gender corresponding to it. This dataset is named *Tweets* and is illustrated in Figure 2.

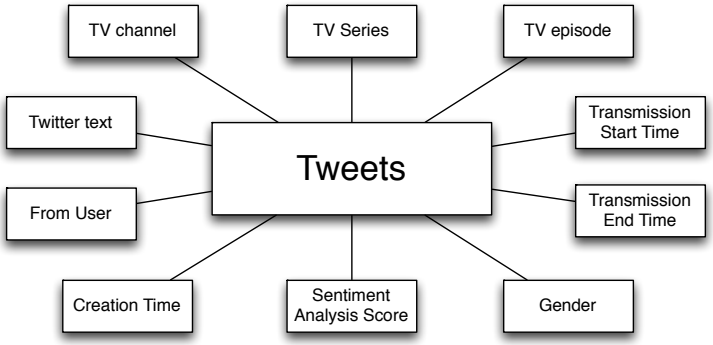

Figure 2: The initial dataset of *Tweets* given by Backflip.

In advance, Backflip gave me for each tweet its JavaScript Object Notation (JSON) file, which is the Twitter format of all the information contained in a tweet. As I explain in more depth in Section 4, I preprocessed the JSON file in order to extract the device that was used

for tweeting[8] and the location of the Twitter user. The dataset I used eventually for this project is illustrated in Figure 3. Part of this dataset is given in APPENDIX A as an example of the actual data I used.

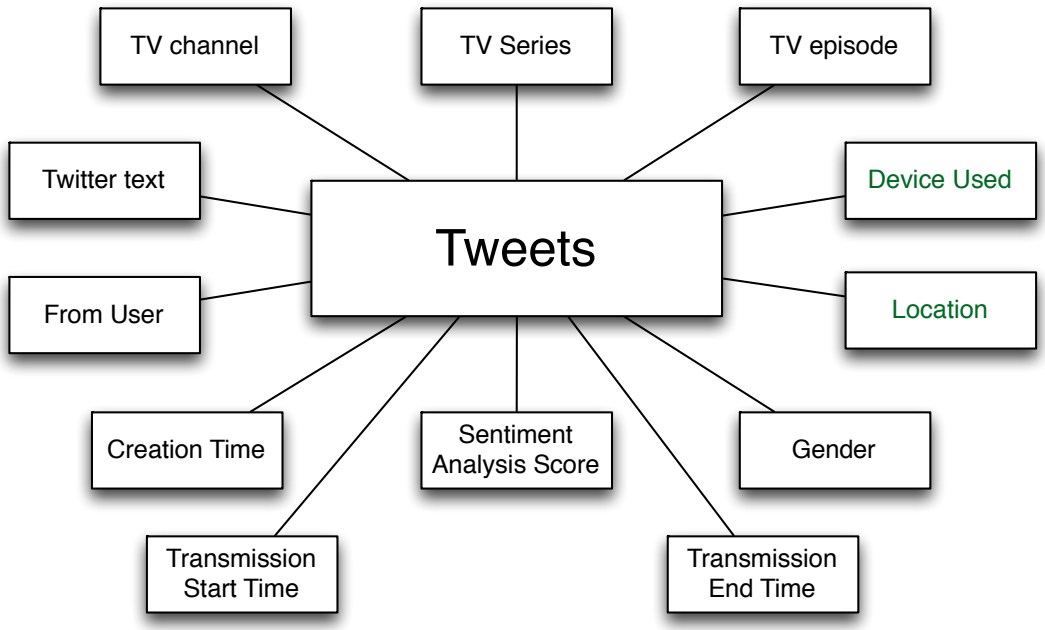

Figure 3: The final dataset that was used for this project. Extensions to the initial data are given in green color.

Additionally, Backflip gave me a dataset of subtitles, also referred to as the *Concepts* dataset. This dataset includes, in the same sense as the *Tweets* dataset, data for Monday 25th June. This data contains each subtitle as appeared in TV for each TV series of each UK TV channel extracted by a system the company implemented. Also, it contains the *Concept* that corresponds to each subtitle, which as aforementioned reflects a keyword that appeared in each subtitle.

In particular, the *Concepts* dataset I was given consists of the subtitle text (*Sentence text*), the name of the episode that it refers to (*TV Episode*), the name of the TV channel, the times that the retrieval of subtitles began (*Transmission Start Time*) and ended (*Transmission End Time*) for this TV series and the subtitle's creation time. This dataset is illustrated in Figure 4. Part of this dataset is given in APPENDIX A as an example of the actual used data.

My goal, as earlier explained, was to enhance fashions discovery from the time-wise correlation of these two datasets. In order to correlate the two datasets I matched the fields *Creation Time*, *TV episode* and *TV channel* that were the same in both datasets. In other words, for each tweet I matched the subtitles that were created at the same time about the relevant TV series and channel. This correlation is demonstrated in Figure 5.

The *Tweets* dataset has 149,098 records and the *Concepts* dataset has 194,248 records. Since one day can fetch this many records, for the purposes of this project I required and used data for one day, Monday 25th June. However, the implementations are adjustable for any amount of data.

[8] The action of sending a tweet.

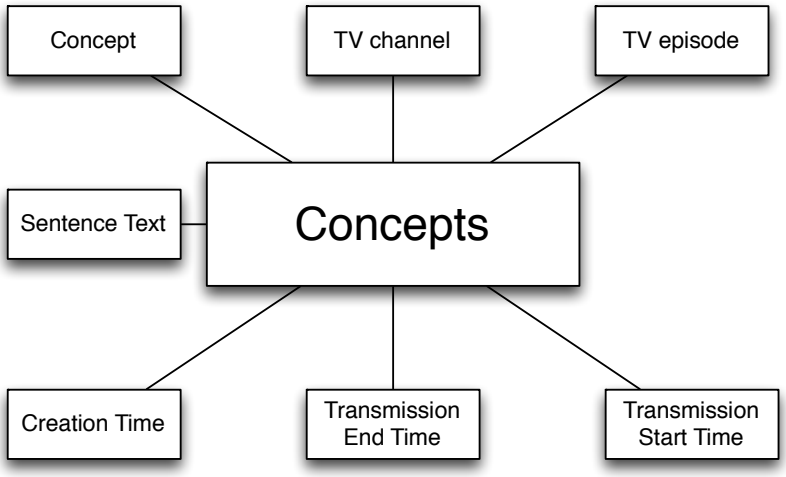

Figure 4: The dataset of *Concepts* given by Backflip

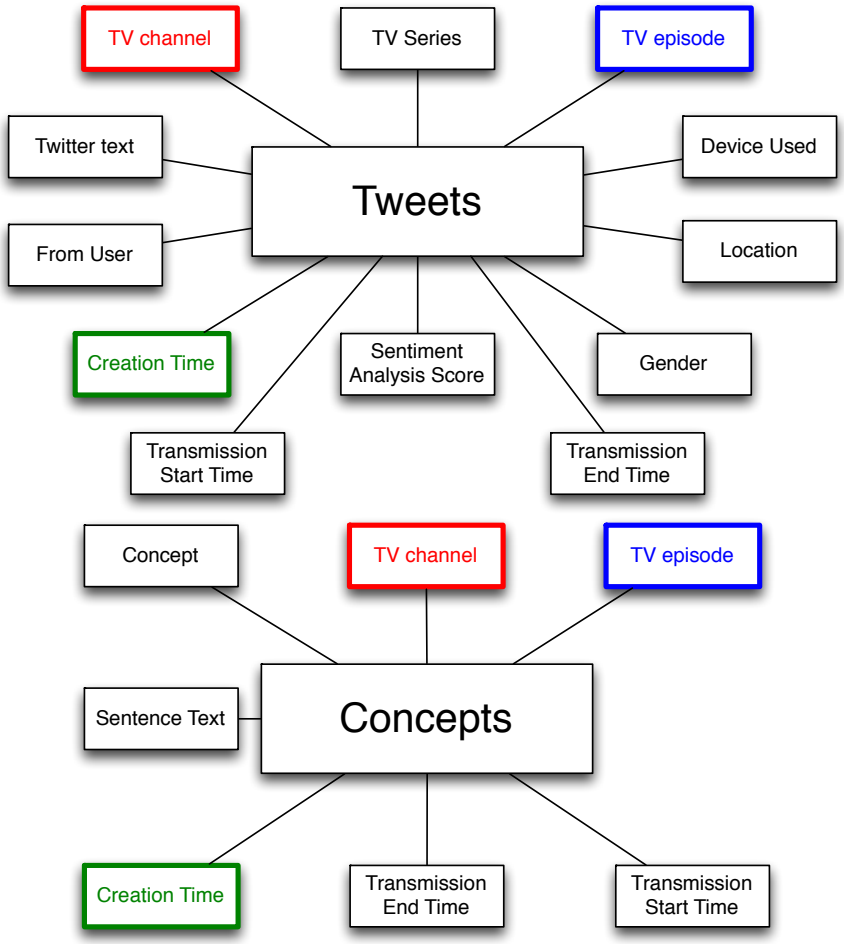

Figure 5: Illustration of the correlation between the two datasets. The coloured boxes define the fields that were matched in both datasets to retrieve the relevant subtitles for each tweet.

3.2 Use cases

According to my goal to improve trends analysis via visualisation methods, I had to create use cases for the visualisation of the data I was given. In order to do this, discussions were made with the Backflip company, determining what could be useful to learn from Twitter content in general and in combination with TV subtitles. As a result, in this paragraph I explain the concluded use cases that attempt to improve information retrieval for the domain of TV series.

3.2.1 Discovery of trends

As Cheong (2009) argues, Twitter, as any popular social network, can bring out opinions from the masses, which can later be used for decision support and economic analysis. For the same purposes, this project attempted to create visualisation methods that illustrate opinions and tendencies related to TV series from Twitter content.

In order to discover tendencies, the contents of tweets were analysed and the most informative data for this project were used in visualisations. Particularly, the used information from each tweet was:

- the gender (female, male, unknown)
- the location (UK city)
- the device used, and
- whether a tweet is a Retweet or not

This data composes the demographics that can be resolved from Twitter content. They are informative for trends discovery, because demographics allow the discovery of patterns in people's behaviour (Bhatnagar, 2004).

In advance, the volume of tweets and the sentiment analysis score for each of them were also used for trends discovery. The volume of tweets shows the amount of people discussing over Twitter for a TV series. It was used in relation with other characteristics, such as time and sentiment analysis. The sentiment analysis was the key to distinguish the opinions of the Twitter users about a TV series. In other words, sentiment analysis defined positive sentiment analysis score for TV series that were liked and negative for TV series that were not liked.

The analysis of Twitter content brought out use cases for the visualisation of data for three research fields: TV channel/series, TV episode and all channels together. As noted, the use case for TV channels and series are the same, as they both can resolve information about one subject of study (TV channel or TV series) for different time intervals. On the other hand, TV episodes are usually once on-air and on a specific time period; therefore they concern a different use case. Also, use cases about all channels include more than one subject of study (all channels) and therefore they are separate from the rest. All of the use cases are explained in Table 2, with the field of study they concern. The field of study lists the subject being analysed and the use case provides details of the areas to investigate.

Table 2: Use cases for the visualisation of demographics, sentiment analysis and volume of tweets for trends discovery.

Field of study	Use case
TV Channel/Series	• Overview the average sentiment analysis score of tweets per minute and by gender for a time interval. • Relate the volume of tweets and their sentiment analysis with the parts of day: Dawn, Morning, Afternoon and Night

Field of study	Use case
	for a time interval. • Illustrate the sentiment analysis for each TV series/TV episode on air for a time interval, from the most liked one (highest sentiment analysis score) to the less liked one (lowest sentiment analysis score). • Visualise the volume and the sentiment analysis of tweets on a UK map for a time interval. • Compare the amount of tweets with the amount of retweets for a time interval. • Overview the devices that were used for tweeting, from the most used one to the less used one, for a time interval.
TV Episode	• Overview each tweet per creation time and gender regarding its sentiment analysis score for the time an episode was on air.
All TV Channels	• Overview and compare the sentiment analysis and the volume of tweets for all channels for a time interval. • Illustrate all channels from the most discussed one (highest volume) to the least discussed one (lowest volume) and compare the sentiment analysis for them for a time interval.

In the above use cases the *Concepts* dataset are not taken into account. In the next paragraph the use of this dataset is explained regarding the purposes of trends discovery.

3.2.2 People's reaction to what is played on TV

This project, besides revealing insights about people's behaviour from demographics, attempts to map this behaviour to what is played on TV. This is achieved by using the *Concepts* dataset. This dataset, as earlier explained, consists of subtitles about each UK TV series and can reveal what was discussed in TV from the content of the subtitles.

Specifically, the *Concepts*, as Backflip defines, can be thought of as topics of discussion, because they usually include mentioned names. For instance, the *Concept* of the subtitle "Did Angela ask for money?" is "Angela", meaning in this subtitle the name "Angela" was a topic of discussion. Having the *Concepts* and not only the subtitles makes it easier to highlight what exactly is discussed on TV and use it for information retrieval.

In order to map *Concepts* to the rest analysed data, the *Concepts* dataset was correlated with the *Tweets* dataset time-wise and by matching the name of TV channel and episode. This mapping allowed the creation of another two use cases for the visualisation of the provided data. These two use cases aim to improve the understanding of people's reactions to what is played on TV. Their descriptions are:

- Overview for a TV channel/series the average sentiment analysis and the volume of tweets by gender for a time interval. After the selection of one minute in the visualisation, get the tweets that were retrieved at that minute with their sentiment

analysis (positive/negative). Also, get the Top 20 *Concepts* that were retrieved 10 minutes before and up until the selected minute.

- Overview for a TV channel/series the average sentiment analysis and the volume of tweets by gender over a time interval. After a selection of another time interval in the visualisation get the Top 20 *Concepts* and the tweets with their sentiment analysis (positive/negative).

As noted, the above use cases have two specific constraints: the Top 20 *Concepts* and the specific time intervals. The constraint about the Top 20 *Concepts* attempts to limit the number of *Concepts* that were created during the mentioned time interval. That is because, many subtitles can be retrieved for a time interval and the most common ones are meant to capture what was going on TV during that time period. In the first use case, the time interval of 10 minutes has to do with the fact that people tend to react to an incident some time after it occurred. This project defines that people respond to something that happened on TV in the following 10 minutes after it occurred. In the second use case, the time interval is not limited and therefore the *Concepts* correspond to the selected time period.

The above two use cases can embed the visualisations for the rest use cases. That is because these two use cases are based on time intervals, a measure that can bind the different data together and combine different visualisations in one. As a result, these two use cases for the *Concepts* dataset were designed to be time-series charts with enclosed interactions that can resolve the rest use cases.

3.3 Challenges

So far, the problem has been examined in depth regarding its state and my approach to resolve it. However, the process of analysing the goals and the data of this project and creating a plan to unfold the problem involved some challenging tasks. In this paragraph three main challenges are described: the manipulation of data, the transformation of data to knowledge and the evaluation of visualisations as the mean of trends discovery from Twitter data.

As in most researches, the data of this project introduced extra effort in the problem's resolution. For this project the Twitter content was the most complex dataset. That is because the JSON object of each tweet, from which the most information can be retrieved, involves many different data that can misguide the information extraction. For instance, in a JSON object there may be demographics about the user who sent the tweet and demographics about the user who's tweet was resend by the first user. In order to make correct extraction of information for this sort of cases, the structure of the JSON object was in depth analysed. The preprocessing method is further analysed in Section 4.

In addition, the location, a feature that was part of the demographics was not well defined in the JSON objects and required further preprocessing. This feature was not well defined, because users of Twitter have no constraints about what to enter as their location. In most cases they enter a phrase not related to location or they leave the location blank. As a result, the mapping of tweets' volume and sentiment analysis to a geographical map needed preprocessed data that refer correctly the name of the city that the user is located. Again, the method for this preprocessing is described in Section 4.

As well as manipulating the data, transforming it into knowledge was also a difficult process. That is because there are a lot of ways to demonstrate data to the observer, but only some of these ways are effective for information retrieval. For the creation of meaningful representations of data I had to conceive many use cases for data visualisations and introduce them to Backflip to get their feedback. Since the goals of information retrieval were vague, meaning the company were interested in finding any insights about TV series, many ideas for visualisations were dismissed in the process of finding the essential ones.

Even when the goals of information retrieval were more distinct, the quest for the appropriate visualisations was still a difficult task. For example, after the decision that time-series charts would combine the two datasets and augment the understanding of their relations, the representation of their embedded parts needed to be effective as well. Specifically, the representations of *Concepts* and tweets eventually had a specific format that attempts to accommodate their understanding by the end user. More details about the final implementation are given in Section 4.

Finally, the most challenging task for this project was the evaluation of the three main time-series visualisations. These visualisations needed to be evaluated, as they are the main deliverables of this project and attempt to transform many different data into knowledge. The evaluation of visualisations has been a difficult task in general for *InfoVis* and some researches tend to skip this process (Ellis et al., 2006). However, the results of this project could not be valuable unless an evaluation method defined the efficiency of the main deliverables.

In order to conclude to the most appropriate evaluation method extensive research has been done in the area of *InfoVis* tools' evaluation. Eventually, the appropriate evaluation methodology was found, which assessed how people actually manage to retrieve information from a visualisation and the time they need to do so. However, although the methodology was well defined, the process of inviting people to participate and making this evaluation experience as user friendly as possible was also a demanding task.

3.4 Summary

This section allowed the reader to take a closer look to the problem that this project aims to resolve. The datasets that play a key role to the implementation of this project were thoroughly explained. The reader should now understand the available information from Twitter content and the ways it can be related to other datasets, in this case the *Concepts* dataset.

What is more is that the problem has been broken down to visualisation use cases, which explain the implementations described in Section 4. These use cases were separated to those who correlate demographics data with sentiment analysis and volume of tweets and those who correlate the two datasets, the *Tweets* and the *Concepts* dataset. This separation attempts to clarify that the first use cases can be used for any Twitter content in the same manner and the latter use cases are designed specifically for the domain of TV series in combination with TV subtitles.

Finally, this section ended with the challenges of this project. These challenges are noted so that researchers on the same domain will have a better understanding of the state of this problem. Also, with the given challenges the resolution of this problem can be critically evaluated.

4 Implementation

This section attempts to describe the implementation process and the final deliverables of this project. It begins with an introduction of the technologies that have been used and continues with the implementations, giving examples of their use. Images are provided for each implementation, to augment the understanding of each deliverable.

4.1 Libraries and software

As reported in Section 2, this project needed freeware visualisation tools that have the less possible limitations and many abilities. Therefore, I decided to use JavaScript libraries that offer already implemented visualisation graphs and can be adjusted to any amount and type of data. Moreover, I attempted to use these libraries that are richer in the features they provide and have a friendly user interface. Specifically, I used the JavaScript libraries: Highcharts and Highstock JS, Google Chart Tools and amCharts JavaScript charts. All the tested libraries and the advantages of the selected ones are noted in Section 6.

4.1.1 Highcharts and Highstock JS

The Highcharts and Highstock JS are free open source libraries, which include charts written in pure JavaScript and are provided by Highsoft Solutions AS. The Highcharts library offers interactive charts, like line, area and column charts. On the other hand, the Highstock library provides time-series charts and its name comes for its design for stock market visualisations.

These libraries are developed in JavaScript so that the end user can add or adjust functions in the visualisations. Both of them are cross-browser compatible, including smartphones[9]. They can be used for free for non-commercial purposes, requiring a relevant trademark on the bottom of each visualisation. All the features and options for the modification of the visualisations are well explained in the provided documentation of each library.

Besides being open-source and cross-browser compatible, both of them provide a rich gallery of charts. Some examples of Highcharts visualisations are given in Figure 6 and examples for Highstock visualisations are presented in Figure 7. In advance, all JavaScript events are supported from both libraries, which allow the design of interactions in the visualisations. For example, the selection event is supported, which allows the creation of functions that are triggered when a user selects an element in the visualisation.

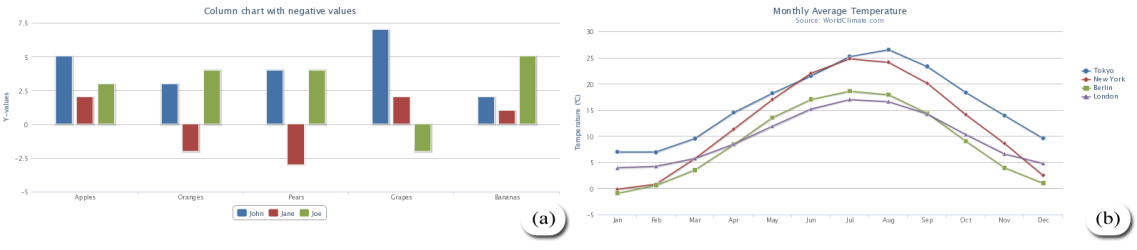

Figure 6: Highcharts library examples. (a) Column chart with negative values (b) Basic Line. Taken from http://www.highcharts.com/demo/

Moreover, both of them can load external data and can be updated real-time with the use of Ajax[10]. What is more is that they allow the export of each visualisation in four graphic

[9] A mobile phone with an embedded operating system and more advanced utilities than regular mobile phones.
[10] Asynchronous JavaScript and XML

formats: PNG, JPEG, PDF and SVG. The PDF and the SVG formats are scalable for printing, an essential characteristic for the Backflip company's need to print the visualisations for meeting presentations.

Figure 7: Highstock library examples. (a) Point markers only (b) Single line series. Taken from http://www.highcharts.com/stock/demo/

I used these two libraries mostly to represent the Twitter data time-wise. In some cases, I used the JavaScript *Date* object to define the format of the time, whereas in other cases the representation of the time did not follow a particular format. The representation of time without the use of the *Date* object was an examination of other ways to display time, in case the *Date* format is not supported by a visualisation. From the three evaluated time-series charts, one is implemented using the Highcharts library and one using the Highstocks library.

4.1.2 Google Chart Tools

The Google Chart Tools application is one of the free products provided by Google Developers[11] and offers a rich gallery of interactive visualisations. Each chart in this gallery is implemented as a JavaScript class, which a user can adjust and embed to his website. All charts are rendered inside a website using the HTML5 and SVG technology, without requiring additional plugins.

The Google Chart Tools offer a wide range of different charts, from simple line charts to more complicated in structure hierarchical treemaps. These two examples are given in Figure 8. As aforementioned, they are implemented in JavaScript and therefore are fully customisable and cross-browser compatible. Also they can load external data by using the JavaScript *DataTable* class, which is a JavaScript representation of data created by Google.

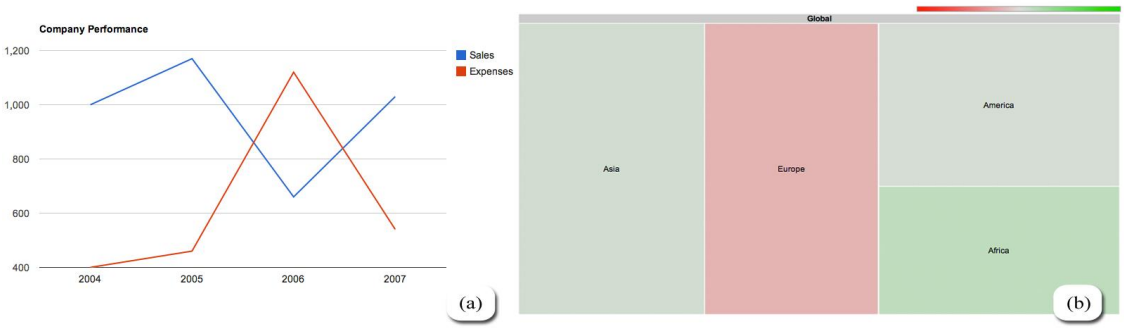

Figure 8: Examples of visualisations from Google Chart Tools. (a) Line chart (b) Treemap chart. Taken from https://developers.google.com/chart/interactive/docs/gallery

In addition, these tools provide backward compatibility for three years. In other words, if Google decides to discontinue this technology, all visualisations will still be supported for at least 3 years. This feature is essential; because it reassures that no more work needs to be

[11] Google website for developer tools, APIs and open-source code.

done if this technology is deprecated for the visualisations using it. Moreover, this technology allows the display of live data. In particular, they allow continuous data queries supported by the *Chart Tools Datasource protocol*. This protocol is designed particularly for data sources that are recognized and handled by Google, like *Google Fusion Tables*. Although Google forces the use of its supported data sources, server-scripting languages can overcome such limitations.

Furthermore, the provided charts allow interactions that can be triggered by the following JavaScript events:

- *Select*
- *Ready*
- *Error*

The *select* event corresponds to a click on a specific element of the chart. The *ready* event occurs when the chart is rendered, meaning when the chart has been drawn in a website. The *error* event is triggered when a syntax error has occurred during the creation of the chart. All charts support the *select* event, although it may require a different handling (Google Chart Tools, 2012). On the other hand, the *ready* and *error* events are not supported from all of the charts.

I used this software to implement visualisations about location and demographics and to create the last evaluated time-series chart. These visualisations, as will be further explained in Section 5, allow the quick understand of the data that is visualised. However, the fact that these charts do not allow many interactions or modifications makes them less appealing for the visualisation of complex data.

4.1.3 amCharts JavaScript charts

The amCharts JavaScript charts is another freeware library, provided by amCharts, a small company in Lithuania. They offer already implemented interactive visualisations written in pure JavaScript. The graphic design of these visualisations is primitive compared to Highcharts and Google Chart Tools, but this library offers a very rich gallery of charts easily adjustable on any data.

As the other libraries, it is cross-browser compatible, using JavaScript and HTML5 elements to overcome any incompatibilities. It allows the load of any data and supports the JavaScript *Date* format for the creation of time-series charts. It is free even for commercial use, but the free version has the trademark of the product on the top left corner of the charts.

They offer a wide range of additional tools and interactions embedded in the visualisations. Besides the common zoom-in utility, they offer interactions like transforming one visualisation into another with a corresponding option. For instance, a bar chart can be transformed to a column chart by using the *rotate* option. Furthermore, they can have multiple axes, allowing the visualisation of many data in two dimensions. Such an example is shown in Figure 9.

I used the amCharts JavaScript charts to visualise demographics about Twitter content. It could also be used for the implementation of time-series charts, since amCharts offers the library JavaScript Stock chart, which is similar to the Highstock library. However, it does not yet support export of the visualisations in graphic formats as the Highstock library does and so it was not used for implementation of the time-series charts.

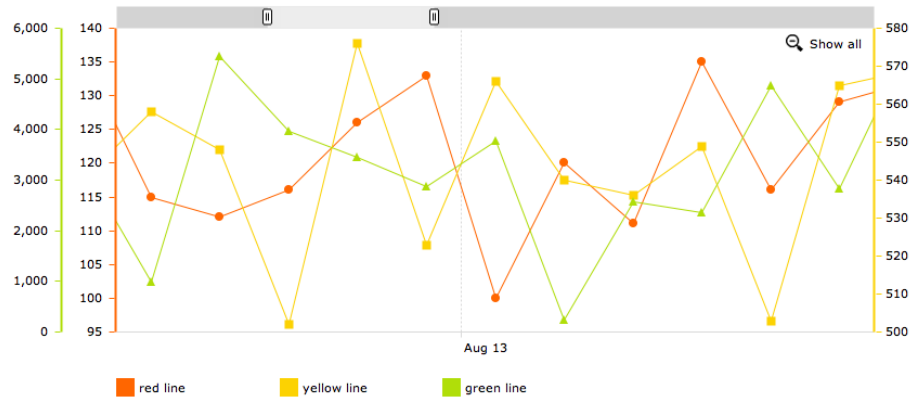

Figure 9: Line chart with multiple value axes, an example from the amChart library. Taken from http://www.amcharts.com/javascript/line-chart-with-multiple-value-axes/

4.2 Data preprocessing

As aforementioned in Section 3, one of the challenges of this project was the manipulation the data it involves. In order to get the data in the correct format for visualisation, the JSON object of each tweet was preprocessed. The data that was eventually extracted and preprocessed from the JSON object was the location of the user who sent the tweet and the device he used for tweeting.

For a better understanding of the required preprocessing, the structure of the Twitter JSON object needs to be analysed. In general, JSON is a text-based representation of data that is lightweight, can easily be generated and parsed from machines and is used as a data-interchange language (JSON.org, 2012). The JSON object of each tweet as noted earlier is extracted from the *Datasift* service. Therefore, it may have a different structure from JSON objects retrieved from the *Twitter API*.

So, the JSON objects for this project included data about the *interaction*, meaning the information about the tweet that was sent, like the text (*content*), the creation time (*created_at*), and the *username* corresponding to this tweet. Also, they included information about the *user*; meaning his profile settings and any other information the user allows to be displayed, like for example *location*.

Furthermore, if a tweet is a retweet then the format of the JSON object is more complicated. That is because it includes data about the *retweet*, meaning the creation time of the retweet etc. Also, it includes data about the user whose tweet was *retweeted*. An example of such a JSON object is given in APPENDIX B for a further examination of this data structure.

So, the structure of each JSON object in this project was not necessarily identical to the structures of the rest. That is because, some of them had extra fields, when they were related to retweets and the rest did not. Therefore, the use of already implemented parsing libraries was not possible, in order to extract for each JSON object the information needed, as the structure of the fields was not definable.

To overcome this inconvenience, text preprocessing of these JSON objects was required. In order to get the needed content in each case, a preprocessing procedure would match the labels of the fields that included the information and would extract the corresponding data. However, there was a possibility that the extracted data could refer to the *retweeted* user and not the user corresponding to the tweet. Therefore, the different creation times would verify whether the extracted information matches the actual user or the *retweeted* one. As a result, the extracted information about the device and the location of the user was verified that it

corresponds to the user who is related to a particular tweet. However, the recovered data need also further preprocessing for their representation in the visualisations.

Particularly, in order to retrieve the UK city[12] from the *location* field of the JSON object another preprocessing procedure was needed. This procedure checked whether a word inside the *location* field corresponds to a UK city and then transformed it to the actual name of the city. The cities of UK were taken from Wikipedia[13]. For example, the *location* field could include the data "Great city of london" and after the preprocessing procedure the location "London" would be saved in the corresponding tweet of the *Tweets* dataset. This preprocessing was necessary for the representation of the locations to the map visualisation provided by Google Chart Tools.

In the same sense, the device used for tweeting was also preprocessed. This information was held under the field *source* and could include the device and the application used for tweeting. For example it could contain the text "Twitter for iPhone". The preprocessing procedure worked as the previous one for locations. Specifically, it searched for a word in this field corresponding to a device for tweeting, and then inserted the name of the device to the *Tweets* dataset for the specific tweet. The proper names of the devices were inserted in an array after listing and editing the distinct sources extracted from the JSON objects. This step was necessary in order to count the use of devices, even when they have different applications for tweeting.

Finally, a small preprocessing step of the data took also place. In order to visualise the comparison of tweets with retweets, each tweet was checked whether it included the special Twitter term *RT @*, which defines whether a tweet includes a retweet or not. This preprocessing step was easily done using some SQL filtering, while extracting the data from the dataset.

4.3 Visualisation toolbox

One of the main deliverables of this project is the visualisation toolbox. This toolbox is an online service that attempts to compose a query system for visualisations for the purposes of product review. In other words, this toolbox is meant to be a search engine for products discussed on Twitter, retrieving visualisations for this Twitter content in order to accommodate trends discovery. This toolbox suggests a different approach for Twitter data exploration, as analysed in Section 2.

For the domain of TV series the implemented toolbox offers a query system for TV channels, series and episodes. However, the visualisation toolbox is designed to be adjustable on any Twitter content. In the next paragraph the structure of this interface is analysed.

4.3.1 Overview

To begin with, the implemented toolbox is called "Visualise It". It is an online service, which gives a prominent place to its query system for visualisations, as can be seen in Figure 10. It involves two query methods, a search engine and a browse catalog. Via either one of these query methods, the user can retrieve visualisations corresponding to his query, if there are matching results.

Specifically, the search engine gives three options: Channel, TV Series, and Episode, which refer to the type of product the user is querying for. The user selects an option, enters his query in the search engine and then clicks on "Search" to retrieve the equivalent results. The query that has no matches returns a relevant message. On the other hand, the browse catalog

[12] Including cities from the countries: England, Scotland, Wales and Northern Ireland.
[13] http://en.wikipedia.org/wiki/List_of_cities_in_the_United_Kingdom

displays all the Channels, TV series and Episodes for which there are corresponding results in the database. In this implementation all the results are for one day, Monday 25th June. As a result, the user is not asked whether he would like to pick results for another day. That is an additional feature to be added in the toolbox, for data archives involving more days.

Figure 10: The index page of the visualisation toolbox "Visualise It".

As analysed in the use cases, in Section 3, the Channels and the TV Series resolve the same visualisations, but for different subjects of information retrieval. Particularly, queries for Channels derive visualisations that analyse the TV series that were played in each Channel. In the same manner, queries for TV series result visualisations for the analysis of their Episodes. These visualisations are the same in structure and logic, but include different data. They are further analysed and showed in paragraph 4.5.

Moreover, queries for Episodes derive a visualisation that shows the sentiment analysis of all tweets for one Episode per creation time. This visualisation is implemented with the Highcharts library and is later evaluated as a time-series chart. This visualisation is a time-series chart, because it shows the tweets as they are created during the time that an Episode is on air. It is again explained and showed in paragraph 4.5.

Above all, there are analyses available, which do not require the user to enter a query. These analyses are displayed on top of the search engine, as can be seen in Figure 10, and are named: "Overall Statistics", "Most Frequent Concepts", and "Volume VS Sentiment for all Channels". These sections of the toolbox include visualisations that concern the use cases for *all channels* plus a visualisation for the 20 most frequent *Concepts* on a specific day for all channels. The last use case attempts to capture what has been discussed mostly in all channels and is represented by a *Word-cloud*.

In all visualisations additional utilities are embedded. The most common ones are the *Interactive Filtering* and the *Interactive Zooming*, as analysed in Section 2. In advance to *Interactive Zooming* I added options to enlarge the visualisation, which redesign the visualisation in a new window and in a bigger size. As I mentioned in Section 2, the *Dynamic Projections* were avoided for the representation of multidimensional data, in order to simplify the visualisations. To display such data visualisations that allow multiple axes in two-dimensions were used. Such an example is given in paragraph 4.5.

4.4 Time-series charts

This paragraph describes some of the most essential deliverables of this project; the time-series charts. These visualisations are important for this project as they attempt to enhance the information retrieval about social network activity over time, one of the main interests of Backflip. Many time-series charts were tested, but in my opinion only three of them were easy to understand and appropriate for trends discovery. Therefore, these three time-series charts are analysed here as for their features and functions. Each of them is described after the notation of the corresponding use case. The code for the first time-series visualisation is given in APPENDIX C as an example of the way I implemented the evaluated visualisations.

Use Case: *Overview each tweet per creation time and gender regarding its sentiment analysis score for the time an episode was on air*.

For this use case the *scatter plot* from the Highcharts library was used. In this visualisation the data is represented as a collection of points. Each point is displayed at the coordinates that correspond to each pair of values of the data. These two values in this case are the sentiment analysis score and the creation time of the tweet. The time in this chart is represented with the JavaScript *Date* object. The Highcharts library allows the display of time like this, which automatically fixes the axis of time to represent fractions of minutes. This visualisation includes *Interactive Filtering*, allowing the display of the results by gender. Also, it allows *Interactive Zooming* with controls to return to the initial graph. An example of this visualisation retrieved from a query about the episode "Be Careful Tombliboos!" of the TV series "In the Night Garden" of the channel "BBC 2" can be seen below in Figure 11.

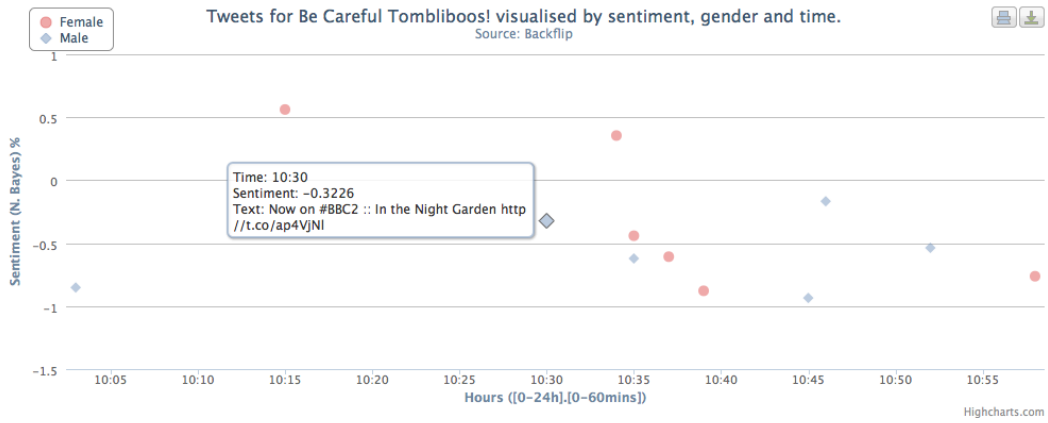

Figure 11: Example of the visualisation for the use case: *Overview each tweet per creation time and gender regarding its sentiment analysis score for the time an episode was on air*. In this example the mouse is over a point for a tweet sent by a male at 10:30 with a negative sentiment analysis score.

Use Case: *Overview the average sentiment analysis score of tweets per minute and by gender for a time interval*.

For this use case the *multiple series* chart from the Highstock library was used. In this visualisation the data involves the average sentiment analysis score, the gender category and each broadcast minute of a channel. This data is represented as continuous lines. Each line represents a gender category, meaning "Females", "Males", "Unknown" and "All". For each minute, which is displayed in the horizontal axis, a gender category either corresponds to an average sentiment analysis score or not. This means that for each minute either tweets were retrieved for some gender categories or not. The vertical axis illustrates the average sentiment analysis, from -1 (negative) to 1 (positive). Again, this visualisation offers the *Interactive Filtering* technique, providing the option to filter out the displayed lines. Also, it involves the technique *Interactive Zooming* and *Interactive Distortion*, meaning it gives the option to zoom in and out, while providing a navigator for the whole visualised data on the bottom of

the visualisation. In Figure 12 an example of this visualisation is given, zoomed into the time interval 12:20 to 13:30.

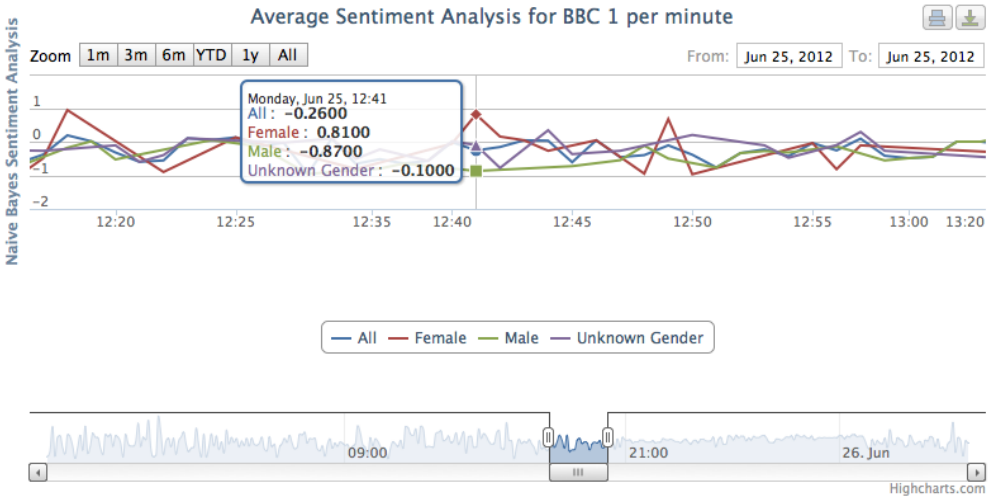

Figure 12: Example of the visualisation for the use case: *Overview the average sentiment analysis score of tweets per minute and by gender for a time interval*. In this example the time interval is from 12:20 to 13:20 and the mouse is over the minute 12:41, showing the average sentiment analysis score per gender.

Use Case: *Relate the volume of tweets and their sentiment analysis with the parts of day: Dawn, Morning, Afternoon and Night for a time interval*.

For this use case the *Bubble Chart* by the Google Chart Tools was used. This chart allows the display of five different variables in a two-dimensional graph. It is defined by two axes and from points rendered according to the two axes. These points are circles designated by their colour, area and label. Their colour defines the part of the day (Dawn, Morning, Afternoon, Night). Their area denotes the volume of tweets for this TV series. Their label shows the name of the TV series. The vertical axis is the average sentiment analysis score for a TV series and the horizontal axis shows the hours that correspond to the parts of days. The advantage of this visualisation is that it gives a quick overview of the sentiment and volume analyses for each TV series and each part of day. The drawback is that it does not allow *Interactive Zooming*. An illustration of this visualisation is given in the Figure 13 with results for the channel "BBC 1".

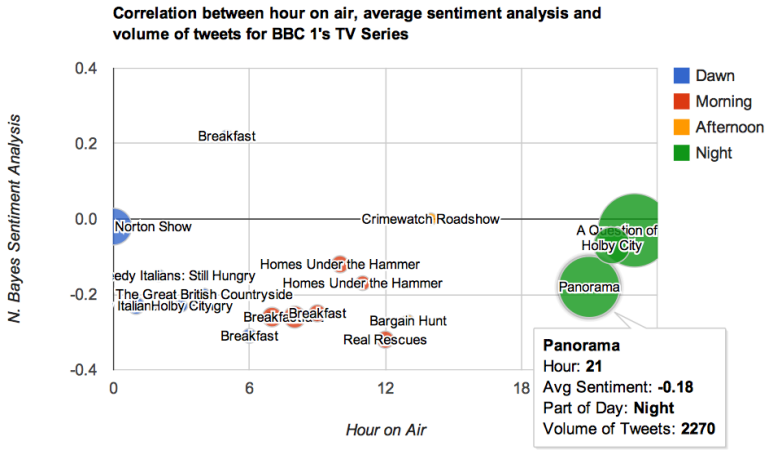

Figure 13: Example of the visualisation for the use case: *Relate the volume of tweets and their sentiment analysis with the parts of day: Dawn, Morning, Afternoon and Night for a time interval*. In this example the mouse is over the TV Series "Panorama" played during the Night.

4.5 Other visualisations

This paragraph will give an extensive outline of the visualisations embedded in the visualisation toolbox and not assessed by the evaluation procedure described in Section 5. More visualisations than these noted were tested, but after the decided use cases only the most contributing to information retrieval visualisations were kept. Discussion about the aborted scenarios and visualisations is remarked in Section 6. The visualisations are explained in the order of the use cases in Section 3.

Use case: *Illustrate the sentiment analysis for each TV series/TV episode on air for a time interval, from the most liked one (highest sentiment analysis score) to the less liked one (lowest sentiment analysis score).*

This use case was illustrated using the *column chart* of the amCharts library. This chart is pretty simple, as it involves two variables, the average sentiment analysis score and the name of the TV series. The vertical axis represents the average sentiment analysis score. The horizontal axis shows the name of each TV series from the one with the highest average sentiment analysis score to the one with the lowest one. Each TV series corresponds to a different column colour, so that the end user understands the data visualised are each time for a different TV series. In Figure 14 an example of this visualisation is given, illustrating all the TV series for "BBC 1" for Monday 25th June according to their average sentiment analysis score.

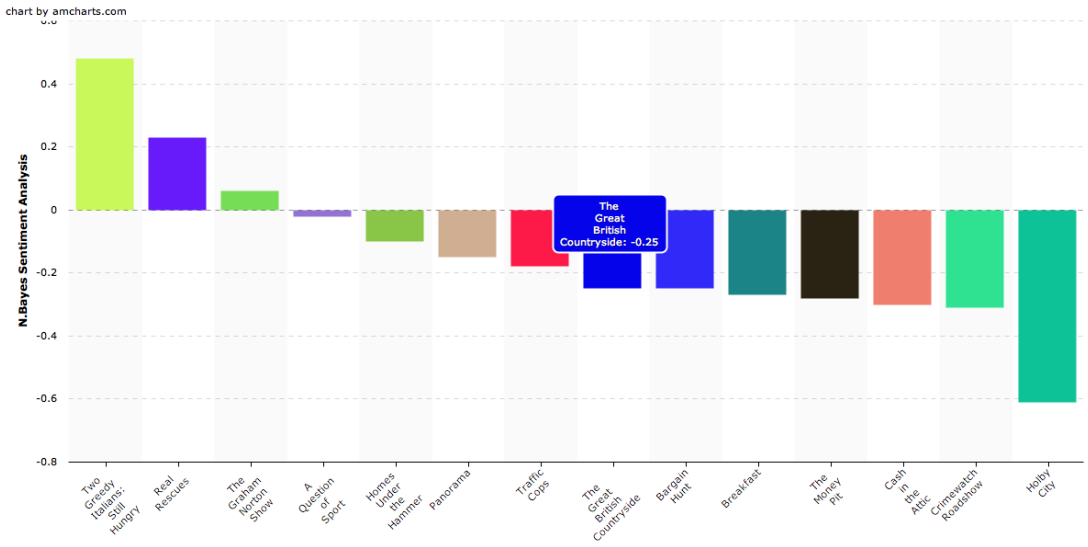

Figure 14: Example of the visualisation for the use case: *Illustrate the sentiment analysis for each TV series/TV episode on air for a time interval, from the most liked one (highest sentiment analysis score) to the less liked one (lowest sentiment analysis score).* In this example the mouse is over the TV Series "The Great British Countryside" with an average sentiment analysis score equal to -0.25.

Use case: *Visualise the volume and the sentiment analysis of tweets on a UK map for a time interval.*

This visualisation was created with *Geo Chart* provided by the Google Chart Tools. The implementation of this visualisation required the preprocessing described in paragraph 4.2, in order to get the strings of the locations in the format this visualisation recognises. The main body of this visualisation is a UK map on which circles define the volume of tweets and their average sentiment analysis score. The area of the circle denotes the volume, meaning the bigger the volume the bigger the circle in the visualisation. The colour of each circle defines the average sentiment analysis score; positive scores are closer to green, negative scores are closer to red. The location is a contradictable demographic from Twitter content, since people

have no constraints about what to enter about it. Also, many times the location is left blank from Twitter users. For these reasons, the data available for this visualisation is not sufficient for information retrieval. In Figure 15 an example of this visualisation is given from the results about the channel "BBC 1".

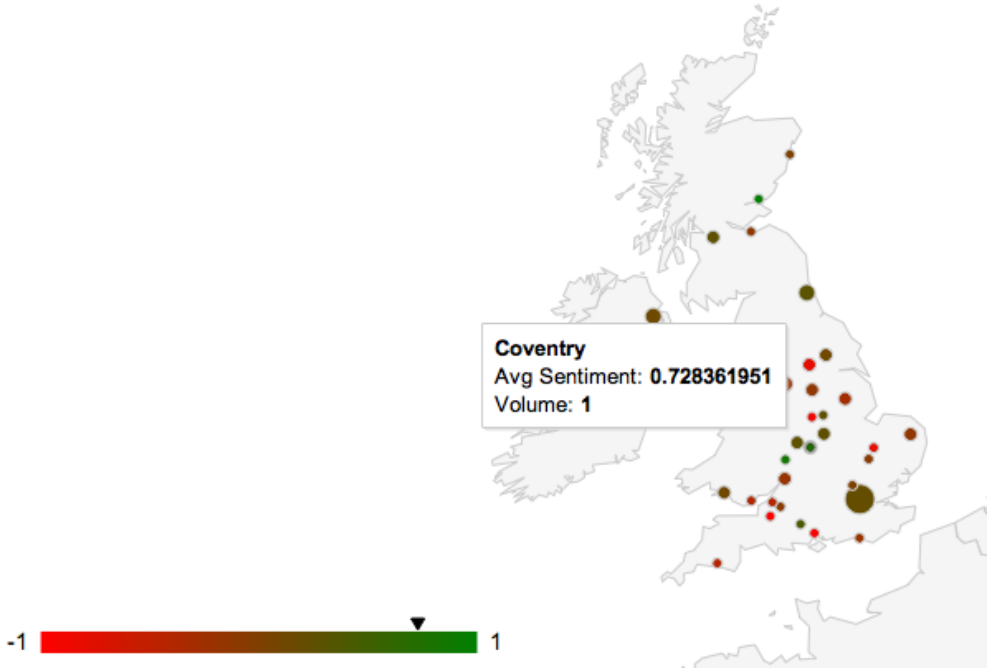

Figure 15: Example of the visualisation for the use case: *Visualise the volume and the sentiment analysis of tweets on a UK map for a time interval*. In this example the mouse is over the UK city "Coventry" with only one tweet and a positive sentiment analysis score.

Use case: *Compare the amount of tweets with the amount of retweets for a time interval*.

This visualisation was created with the *pie chart* provided by the Google Chart Tools. This visualisation is very simple as the data it represents. The values for this visualisation are the labels "Tweets" and "Retweets" and their count represented as a percentage. The *pie chart* was picked because of its simplicity to represent the comparison between two values. The goal is that the end user instantly understands the analogy between the tweets and retweets. Figure 16 illustrates this visualisation from the results for the channel "BBC 1".

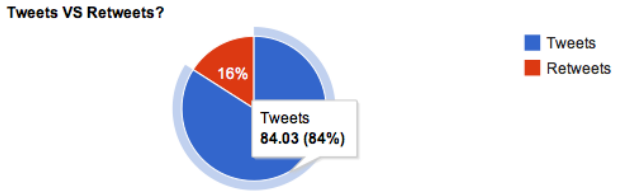

Figure 16: Example of the visualisation for the use case: *Compare the amount of tweets with the amount of retweets for a time interval*. In this example the mouse is over the tweets' piece of the pie, representing the 84.03% of the whole volume of tweets and retweets.

Use case: *Overview the devices that were used for tweeting, from the most used one to the less used one, for a time interval*.

This use case was implemented with the *column chart with images on top* by the amCharts library. This visualisation is a simple column chart, but with the ability to add images on the top of each column. In this case the illustrated variables are the volume of tweets in

percentage format and the names and images of the used devices. For this visualisation the preprocessing step for the extraction of the devices' names was necessary, as descripted in the paragraph 4.2. This visualisation attempts to visualise this data in a way that the end user instantly recognizes the used devices and compares their corresponding volumes. An example of this visualisation is given in Figure 17 from the results for the channel "BBC 1".

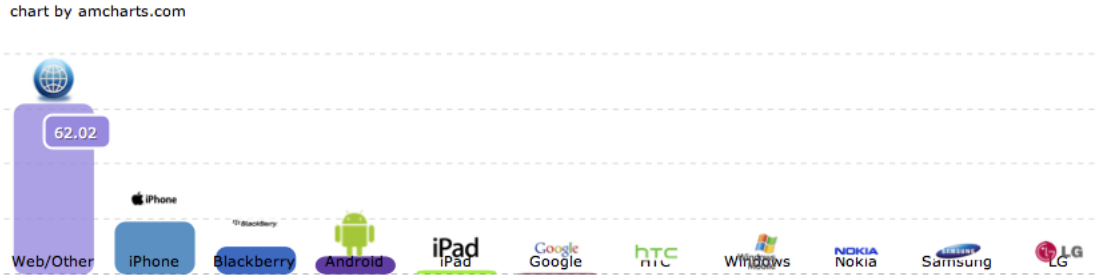

Figure 17: Example of the visualisation for the use case: *Overview the devices that were used for tweeting, from the most used one to the less used one, for a time interval*. In this example the mouse is over the device corresponding to "Web/Other", which was the device used for the 62.02% of tweets.

Use case: *Illustrate all channels from the most discussed one (highest volume) to the least discussed one (lowest volume) and compare the sentiment analysis for them for a time interval*.

This use case corresponds to the analysis provided as an option above the search engine with the name "Volume VS Sentiment". This use case was implemented with the *dual axes, line and column* chart by the Highcharts library. This is one of the cases where the data includes more than two variables and the *Dynamic Projections* technique is avoided. In this visualisation the variables that need to be displayed are: the average sentiment analysis score, the volume of tweets and the name of each channel. In order to simplify their display a visualisation that allows multiple axes was chosen. In this chart there are two vertical axes; the left one represents the average sentiment analysis score and the right one the volume of tweets. In the horizontal axis the name of each channel is displayed. Again, this visualisation offers the *Interactive Filtering* technique, providing the option to display the volume or the average sentiment analysis score or both of them in the chart. The results for Monday 25th June for this use case are shown in the visualisation in Figure 18.

Use case: *Overview and compare the sentiment analysis and the volume of tweets for all channels for a time interval*.

This use case corresponds to the analysis "Overall Statistics" and is illustrated using the *Treemap* provided by Google Chart Tools. This visualisation demonstrates the structure of a *data tree*, which is a hierarchical format of data linked with each other by some relation. In this graph some elements embed the rest, if they are higher in hierarchy. Specifically, in this use case each channel includes its TV series. Each element in the graph is represented by a rectangle. The rectangle's area defines the volume of tweets. The colour of each rectangle defines the average sentiment analysis score; positive scores are closer to green, negative scores are closer to red. This visualisation attempts to accommodate the understanding of the relation between channels regarding the volume of tweets and average sentiment analysis score, by the use of sizes and colours for its elements.

Also, in this visualisation the *Interactive Linking and Brushing* is applied. This technique is used when a user clicks on a channel, besides revealing the TV series on the *Treemap* visualisation, the *Bubble Chart* of Figure 13 appears. This graph again visualises volume and average sentiment analysis score for each TV series of a channel, but also correlates it with the parts of day. The *Interactive Linking and Brushing* technique is not entirely applied, since

only alterations to the *Treemap* result alterations to the *Bubble Chart* and not vice versa. The results of this use case for Monday 25th June are displayed in Figure 19.

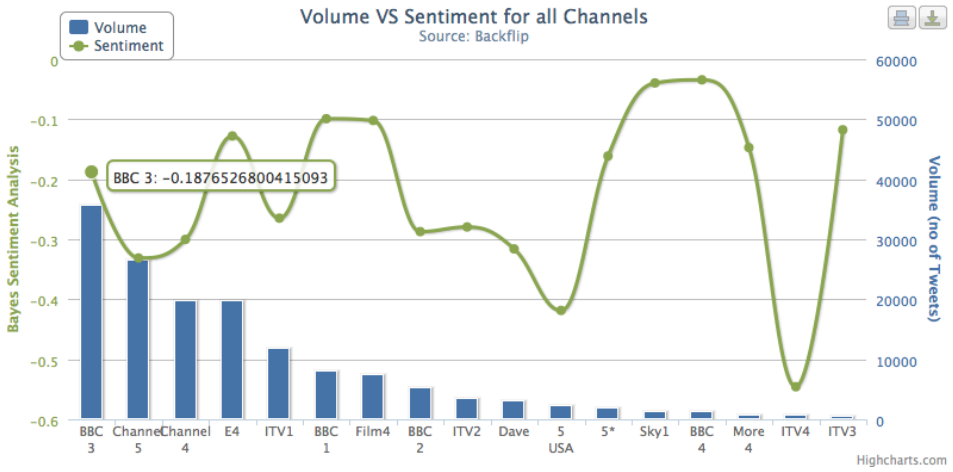

Figure 18: Example of the visualisation for the use case: *Illustrate all channels from the most discussed one (highest volume) to the least discussed one (lowest volume) and compare the sentiment analysis for them for a time interval*. In this example the mouse is over the most discussed channel "BBC 3" with average negative sentiment analysis score.

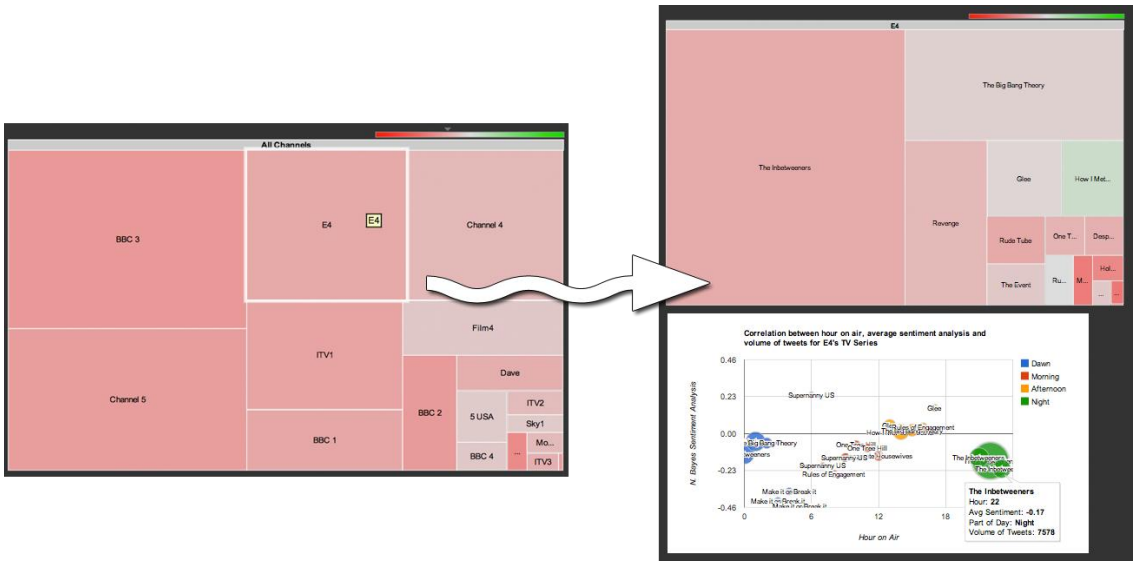

Figure 19: Example of the visualisation for the use case: *Overview and compare the sentiment analysis and the volume of tweets for all channels for a time interval*. In this example a click on the channel "E4" reveal its TV Series and the corresponding *Bubble Chart*.

Above all, additional visualisations were implemented for the improvement of trends discovery. These visualisations are simpler to their development compared to the ones described earlier. One such visualisation is the analysis "Most Frequent Concepts", which is illustrated as a *Word-cloud*. This *Word-cloud* demonstrates the top 20 most discussed *Concepts* over all channels and is created with the open-source jQuery[14] plugin by Luna Ongaro, the jQCloud[15]. This visualisation can be seen at Figure 20, as an example of "Most Frequent Concepts" of Monday 25th June. As can be seen in this example, the word "yeah" is retrieved mostly from all subtitles of all channels as a *Concept* on this day, since it has the

[14] JavaScript library meant to simplify web client-side scripting.
[15] http://www.lucaongaro.eu/demos/jqcloud/

biggest size compared to rest words in the visualisation. The colours denoted are the default of this library and designate the volume of each word as well. Less frequent words have a colour close to purple and the most common ones have a colour close to turquoise.

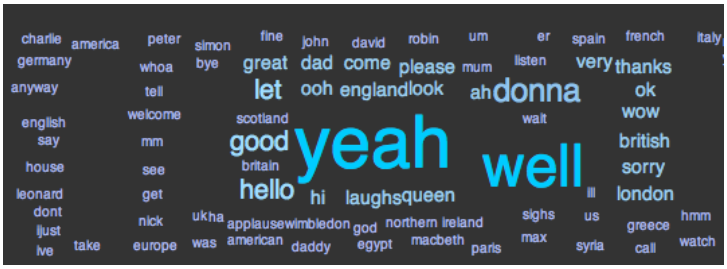

Figure 20: Example visualisation of the analysis "Most Frequent Concepts" implemented as a *Word-Cloud*.

Besides visualisations, images were used as well in the visualisation toolbox to improve information retrieval. Examples of such images are the representation of genders (female, male) and the representation of sentiment of a tweet (positive, negative). These image displays of data are usually part of other visualisations and aim to make the recognition of gender and sentiment faster to the end user. These images are given in Figure 21.

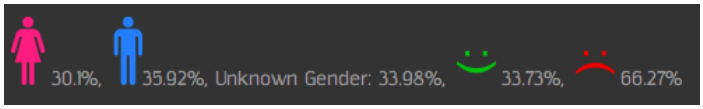

Figure 21: Example of the additional images for the enhancement of information retrieval. It shows the results for the channel "Channel 4", which define that mostly men discussed about it sending mostly negative tweets.

4.6 Time-series charts with other visualisations

As described in Section 1, some of the crucial deliverables of this project are the time-series visualisations that manage to improve information retrieval. In order to improve knowledge discovery, the three time-series charts described in Section 4.4 were linked to other visualisations. This link establishes the technique *Interactive Linking and Brushing* until a certain point. That is because as aforementioned in this implementation one visualisation is considered the main one and the rest linked ones change according to alterations in the first. This technique augments information retrieval because it provides the user with more data than this embedded in the main visualisation.

The first implemented time-series chart (Figure 11) displays the sentiment analysis score for each tweet for one TV episode by creation time. This visualisation was linked with the *Concepts* dataset, meaning for each point in the graph the *Concepts* that were retrieved at that time or up to 10 minutes earlier are displayed. The *Concepts* are illustrated as a *Word-cloud*. Also, for a selection of a point the user is informed about what has been selected and which tweet and gender is correlated to this point. An example of this visualisation is given in Figure 22. The additional images as mentioned in the previous paragraph are used for the quick recognition of sentiment and gender by the end user.

The second time-series chart (Figure 12) illustrates the average sentiment analysis score of tweets per minute and by gender for a time interval. This time-series chart was also correlated with the *Concepts* dataset. Either with a selection of a time interval using the *Interactive Zooming* technique or with a selection of a specific point in a line in this chart, the corresponding *Concepts* and tweets are displayed. The visualisation of *Concepts* is again with the use of the *Word-cloud*. Again, the image additions explained in the previous paragraph attempt to decrease the time the user needs to understand gender and sentiment. This visualisation with its links is illustrated in Figure 23.

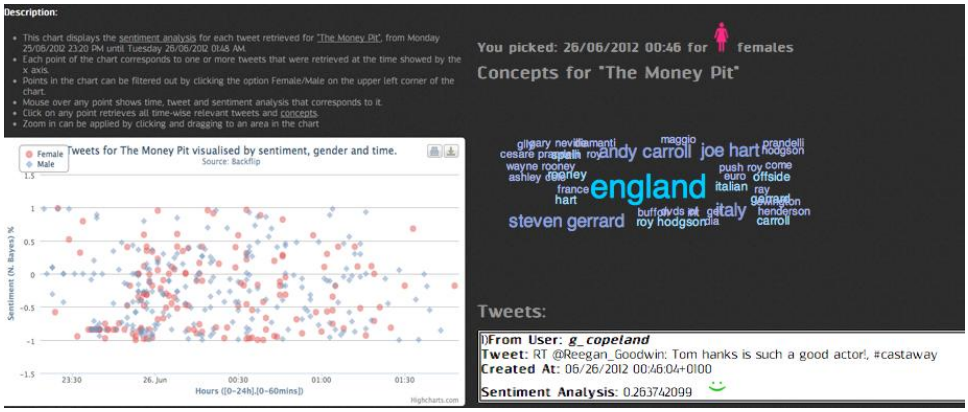

Figure 22: Example of the visualisation for the use case: *Overview each tweet per creation time and gender regarding its sentiment analysis score for the time an episode was on air* in combination with a *Word-cloud* about related *Concepts*. Tweets are also displayed for the selected time 00:46 for females.

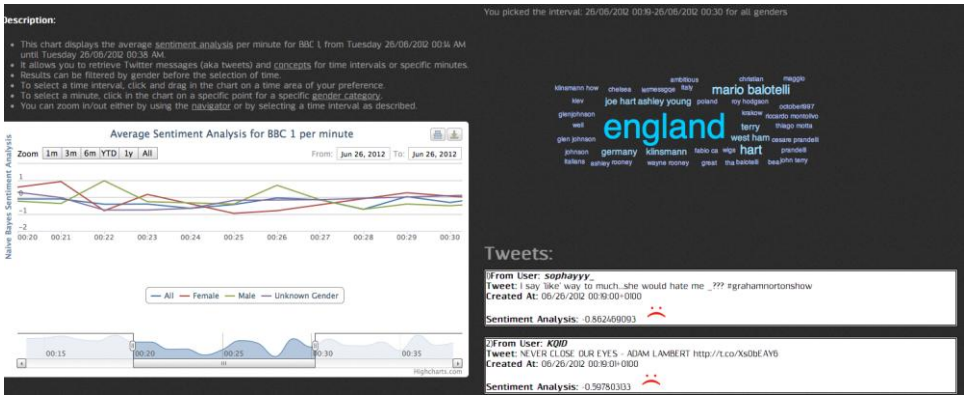

Figure 23: Example of the visualisation for the use case: *Overview the average sentiment analysis score of tweets per minute and by gender for a time interval* in combination with a *Word-cloud* about related *Concepts*. Tweets are also displayed for the selected time interval 00:20-00:30 for all genders.

Lastly, the third time-series chart (Figure 13) was used in correlation with other visualisations about Twitter content for the use case "*Relate the volume of tweets and their sentiment analysis with the parts of day: Dawn, Morning, Afternoon and Night for a time interval"*. This illustration has been described in the previous paragraph and can be seen in Figure 19. The correlation of the two visualisations attempts to provide the user with an overview of the sentiment analysis and the volume of tweets for each channel on a specific day. From this overview the user can go deeper into the visualisation and retrieve information about the sentiment and volume of each related TV series, which are displayed per part of day.

All these three time-series charts are assessed via the evaluation procedure analysed in Section 5. The first two time-series charts are evaluated in correlation with other visualisations, as described in this paragraph. However, the third time-series chart could not be embedded in the evaluation procedure in combination with the visualisation described here. That is because it involves a lot of data that could not be hosted in the online database that is used for the evaluation procedure. However, they are all assessed on the same basis, which is their efficiency regarding information retrieval on data displayed over time.

4.7 Summary

This section described the implementation process required for this project. This description intended to give the reader an overview of the research and work done regarding the visualisation of the given data. It began with an analysis of the used libraries in order to give an insight of the selected tools for the implementation of this project. These libraries are

appraised as for their strengths in comparison with other freeware libraries in Section 6, in order to argue my choice to use them.

Consequently, the data preprocessing was analysed for the reader to understand the transformation of Twitter activity to valuable information for *InfoVis*. Furthermore, the visualisation toolbox was inspected as for its contents and utilities. This toolbox is one of the main deliverables of this project and aims to bind all the information in a query system that a user can easily handle to retrieve insights about a product, in this case a TV product.

Lastly, all the visualisations embedded in the visualisation toolbox were described as for the information they hold and their functions. They were also examined as for the interaction and distortion techniques they involve, which were defined in Section 2. In the end, the time-series charts bound with other visualisations are noted. These time-series charts will be part of the evaluation procedure explained in the following section.

5 Evaluation methodology

"*One of the great mistakes is to judge policies and programs by their intentions rather than their results*"

– Milton Friedman, 1975

It is essential for a research to evaluate its outcomes. The aspect of this research is the knowledge retrieval via visualisation methods. In their research Ellis et al. (2006) argue that most researches implementing visualisation methods for knowledge retrieval tend to avoid evaluating their final outcomes. In the same research it is argued that some researches assess their results by giving examples of their use. Moreover, Faisal et al. (2007) argue that usually Human-computer interaction assessment procedures evaluate the performance of an interface isolated from the data it involves and the knowledge retrieval.

In this project the goal is to represent data in such a way that the final observer will retrieve insights about it. Therefore, the visualisation experience is evaluated for two purposes: efficient representation of data and enhancement of knowledge retrieval. In order to assess both of these factors, the visual tasks defined by Wehrend and Lewis (1990) are used to compose a research experiment about three visualisation deliverables of this project. This evaluation method will be analysed in the following two paragraphs. The three assessed visualisation deliverables correspond to the three time-series charts that attempt to improve information gain about Twitter content displayed over time.

5.1 Overview

This evaluation methodology is based on the visual tasks by Wehrend and Lewis (1990) and is inspired by a similar approach by Morse et al. (2000). This methodology attempts to assess the three time-series charts implemented for this project isolated from the rest implementations. The reason these implementations are evaluated comes from Backflip's need to find a time-wise representation of TV related Twitter data in order to retrieve insights about TV series. Although the three time-series charts involve different data, their goal is the same, which is the information gain. The used methodology evaluates these visualisations for the level a user can handle them and retrieve knowledge about the represented data.

This methodology involves three questionnaires, one for each of the assessed visualisations. These questionnaires include 10 questions about the 10 visual tasks, which are described in the following paragraph. These questions require from the user to interact with the visualisation, get information from it and select one of the multiple-choice answers. While a user is handling the visualisation to answer the questionnaire, the time he needs to do it is measured. Therefore in the end, the effectiveness of each visualisation is assessed by the analogy of the correctness of the answers and the time needed to fill the questionnaire. Additionally, the experiment involves a feedback section, where each participant can argue about his preference between the three time-series charts.

Consequently, the results of this experiment are inspected about their content, meaning the subjects (demographics of participants), the time needed to fill each questionnaire and the correctness of the answers. Finally the results are explained as for the visualisation they concern and in comparison with the rest, using significance tests. Also, from these results and the provided feedback some initial conclusions for the whole process are noted. This is a user-centered evaluation methodology, since the user defines with his actions (answers, time) the effectiveness of each visualisation regarding knowledge gain. The process of this methodology is illustrated in Figure 24.

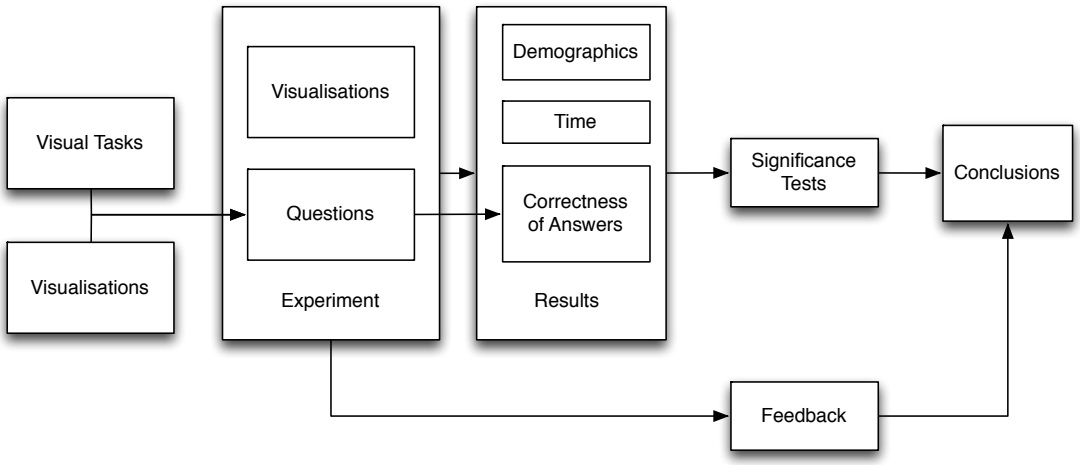

Figure 24: The evaluation methodology illustrated.

5.2 Visual tasks

As aforementioned, the assessment experiment of this evaluation methodology is based on the visual tasks denoted by Wehrend and Lewis (1990). As argued by this research, these tasks denote operations on visual elements and can evaluate these elements in isolation with their application domains. These operations are: *compare within entities*, *associate*, *distinguish*, *rank*, *cluster*, *correlate*, *locate*, *categorise*, *identify* and *compare between relations* and their definitions follow, according to a similar analysis by Morse et al. (2000).

Compare within entities describes the decision about visual elements based on the relations between their attributes. In this project, this action requires the user to observe elements embedded in the visualisation and answer a question related to their characteristics comparison. For instance, if there were two circles in a graph a question for this task could be "Which circle of the two has the biggest area?".

Associate asks the user to make conclusions about the connections between visual elements in a display. For the purposes of this experiment, this task asks the user to decide which element in the visualisation is correlated with the greatest amount of external to the visualisation data. For example, in a visualisation where clicking on an element retrieves external data a probable question for *Associate* would be "Which element is associated with more external data than the rest?".

Distinguish refers to the action of determining the correct value of an element's variable given a collection of values. In this project, this task gives the user for an element in the visualisation a value corresponding to one variable and requires him to find the value of another corresponding variable from a collection of multiple-choice answers. Such an example of this task in a visualisation with elements corresponding to two variables would be "What is the value of X for the element having the value Y for the variable Z?"

Rank is the action of placing in order visual elements according to their values in a visualisation. For this project, this action requires the user to review the values of specific elements and order them as one of the ways described in the multiple-choice answers. For instance, for a visualisation with illustrated circles a task about *Rank* would be "Rank circles A, B, C regarding their area, from the bigger to the smallest".

Cluster is the action of gathering the elements that have some attributes in common. For the purposes of this project, this task asks the user to filter the visual elements based on a specific attribute. For example, if a visualisation has circles illustrated in a Cartesian coordinate

system[16] a question for this task could be "Which circles are centered in the origin of the system?".

Correlate describes the action of making a decision regarding shared attributes between elements in a visualisation. In this project, this task asks the user which visual elements follow a given rule based on shared characteristics. For instance, in a visualisation where its elements are represented in an axis with integer values from -10 to 10 a probable question about *Correlate* would be "Which elements have positive values?".

Locate refers to the action of finding the place in a visualisation where an element should be illustrated. Here, this task asks the user where they would place a new element in the visualisation, given specific values for the element's variables. For example, in a *scatter plot* chart where the horizontal axis defines the temperature and the vertical axis defines the mm of rain, a question for this task would be "Where would you place a point with the attributes (Temperature: 32°C, Rain: 100mm)?". An answer to such a question could be something like "Above element A and on the right of element B".

Categorise is the action of finding a description for elements sharing common attributes. In this project, this task asks the user to choose a category that its name describes correctly a collection of elements regarding their attributes. For example, in a visualisation where elements with negative values for one attribute are illustrated with red color, a question for *Categorise* would be "Which category would describe the elements with red color?" and could have the answer "Negatives".

Identify describes the recognition of an element in a visualisation given its characteristics. Here, this task asks the user to find an element in a visualisation, which corresponds to a value of one of his variables. In order to explain the element that is related to this value the answer involves the values of the rest variables. For instance, in a *scatter plot* where the horizontal axis defines the time and the vertical axis defines a score from 1 to 100 the user could be asked "Which element has score close to 50?". The answer should then involve the value of the time to describe the related element.

Compare between relations assesses whether a user can select an area on a visualisation based on the analogy of the relations between elements. Here, the user is given a rule about the relation between elements in a visualisation and he is asked to ignore the part of the visualisation where the elements are not related according to this rule. This is another way of selecting the part of the visualisation that corresponds to a given analogy of relations. In a visualisation were some parts have negative values and other have positive values a relevant question would be "Which parts of the visualisation would you ignore if you wanted only positive values?".

Now that the definitions for the visual tasks are given, the experiment can be analysed as for the questionnaire it involves. These questionnaires include questions testing the described visual tasks and are further analysed in the next paragraph.

5.3 Assessment experiment

One step of this evaluation methodology is the assessment experiment. As aforementioned this experiment attempted to measure the effectiveness of the three implemented time-series visualisations regarding information retrieval. This experiment retrieved results about: the demographics of the participants, their performance on visual tasks for each visualisation and their feedback about the visualisations and the whole procedure.

[16] A coordinate system for which the coordinates of a point are its distances from a set perpendicular lines that intersect at the origin of the system (WordNet, 2012)

To begin with, this experiment was implemented as an online system, which can be found here: *http://www.vis-eval.us*. In this system, the *Welcome* page was introducing the participants to the assessment procedure and was asking them to provide information about themselves. This information is referred to as the demographics of each participant. This demographics included the age, gender, level of education and English of the participant. The interface of this page is given in Figure 25. As can be seen in this figure, in this *Welcome* page the expected time per participation is given, which corresponds to 40 minutes. This time was the average time spent, retrieved from the first five participants of this experiment.

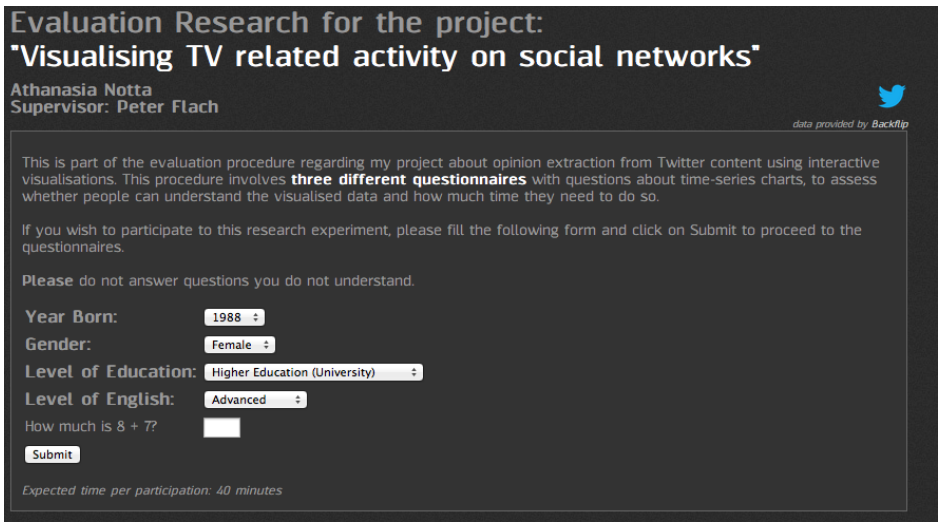

Figure 25: Demonstration of the *Welcome* page of the online system for the assessment experiment of this research.

Continuously, the next step of this assessment experiment was the questionnaires for each time-series visualisation. Each participant who submitted his personal information in the *Welcome* page of this online system was led in the following page, which included the first time-series chart for evaluation and its corresponding questionnaire. In total, three questionnaires were created, each one assessing one visualisation regarding the performance of the described visual tasks on it. In other words, the three questionnaires involved questions that required the participants to perform each of the visual tasks in order to answer a relevant question.

So, after proceeding to the filling of the survey, each participant was led to three pages consequently, each including 10 questions relevant to the 10 earlier described visual tasks and the visualisation under inspection. For each question 2 to 4 multiple-choices were given as the probable answers. Only one of them was correct in every case. The participant could submit an answer or leave the question empty, meaning without a selected answer from the multiple-choices.

For the performance of the visual tasks two measurements were taken: the time to complete each questionnaire and the correctness of the answers. One of these questionnaires is described in Table 3. In this table the "Task" notes the visual task that is assessed and the "Question" gives the question as appeared in the survey. The questionnaire given in this table was created for the visualisation of the use case: *Overview the average sentiment analysis score of tweets per minute and by gender for a time interval* (Figure 12). The illustration of this questionnaire as appeared in the online system is given in Figure 26. The rest questionnaires are given as further reading in APPENDIX D.

Table 3: The questionnaire for the evaluation of the time-series visualisation illustrated in Figure 12. For each question the corresponding order in the questionnaire (No.) and the relevant visual task (Task) is given.

No.	Task	Question
1	Compare within entities	Between the times 00:19 and 00:23 Male and Female reach high picks regarding sentiment analysis. Which of them reaches a higher pick than the other?
2	Associate	For all genders, which of the following times is associated with more tweets?
3	Distinguish	At 00:21 which gender has average sentiment analysis close to 1 (positive)?
4	Rank	Rank times 00:21, 00:22 with respect to the amount of concepts that they contain (gender does not affect the result).
5	Cluster	At 00:22 which gender/s has/ve negative sentiment analysis?
6	Correlate	At 00:25 what is common between all genders regarding sentiment analysis?
7	Locate	If a new tweet were retrieved with characteristics [female, 00:24, 0.85] where would you placed it in the visualisation?
8	Categorize	In which general category would you place points in the visualisation between minutes 00:24 and 00:25 regarding sentiment analysis?
9	Identify	Between 00:28 and 00:30 which time has for all genders and males the same sentiment analysis?
10	Compare between relations	Between times 00:24 and 00:26 which point has values for all gender categories?

As earlier noted, the time was one of the performance measurements for the effectiveness of each visualisation regarding knowledge retrieval. This idea was an influence from a similar approach by Morse et al. (2000). The time is measured from the moment a participant is led to a questionnaire till the moment he submits his answers. This calculation attempts to show the time needed for each participant to: get familiar with the interface, understand the questions, perform the visual tasks, wait for the visualisation's responses and fill his answers. In addition, the other measurement of this experiment is the correctness of the answers. For this calculation, I counted the number of correct answers, while keeping a trace of all the answers. The trace for each questionnaire included the result for each separate question. Questions left empty were also recorded in every questionnaire's trace. All these records will be used for the further investigation of this survey's results.

Moreover, the final part of this online survey was the *Feedback* section. In this section the user was thanked for his participation and he was given a form to provide his feedback. This feedback was in a form of questions, where the participant could submit his opinion about the preferred chart and also the easier to understand and use chart. Furthermore, he was asked to say how he spend most of his time during this survey. The options for the latter were about spending time to: understand the questions, understand the data, understand the chart and

decide the answer. This *Feedback* page attempted to give a better insight about people's opinion for the visualisations and the assessment experiment. The demonstration of this page is given in Figure 27. Further conclusions will be made from the collection of feedback in the *Discussion* section.

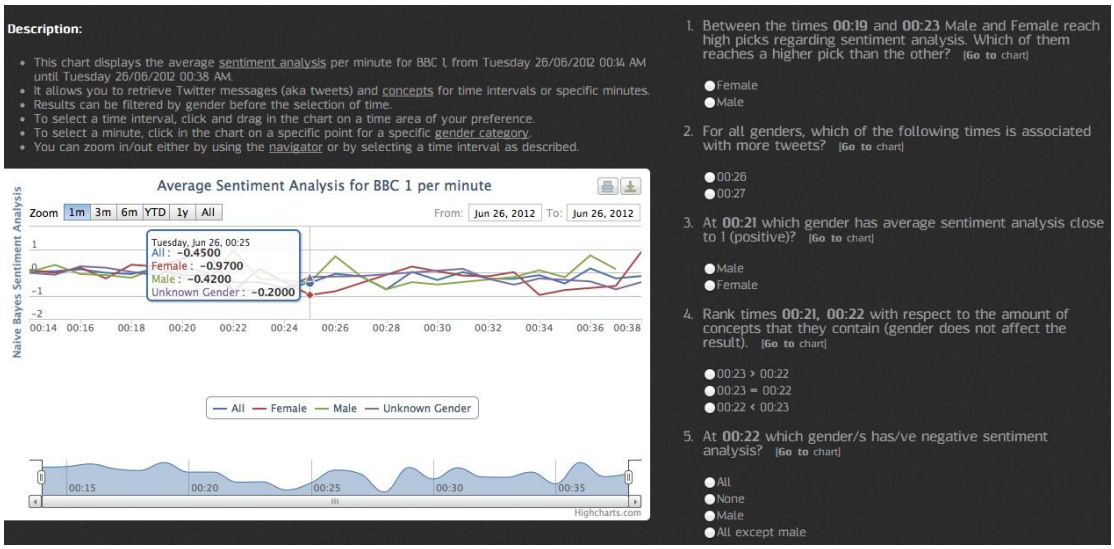

Figure 26: Illustration of the first in row online questionnaire.

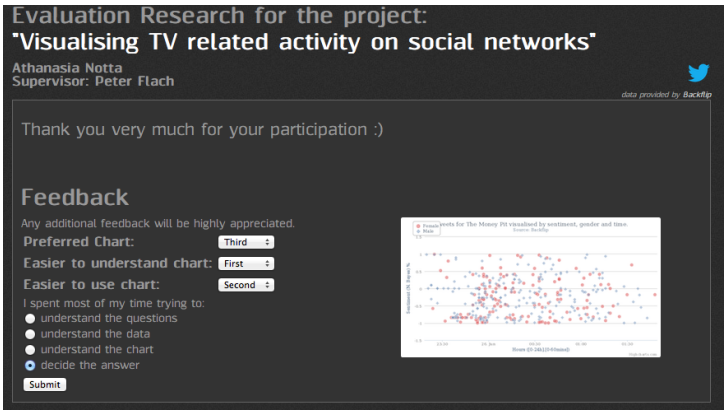

Figure 27: Demonstration of the *Feedback* section of the online system for the assessment experiment of this research.

5.4 Results

For this assessment experiment MSc students from my department and people working in Backflip were invited to participate. The survey was online for two weeks in total and had eventually 32 participants, who fully filled each questionnaire. All the retrieved results are given in APPENDIX E. After the collection of the results, the estimates of time and correctness of answers of each visualisation's questionnaire were calculated. In the following paragraphs, these estimates are analysed and statistical tests are used to measure whether the gathered results resolve significant differences between the evaluated visualisations.

5.4.1 Statistical testing

Statistical testing is necessary in order to discover whether the results of different evaluated methods are statistically significant. In this experiment, the statistical testing is essential to argue about the differences between the results of the three visualisations. In the following paragraphs the results of this survey are examined using the statistical test *Student's t-test*.

Therefore, this paragraph intends to briefly describe this method and its application in the results of this experiment.

To begin with, this statistical test was introduced by William Sealey Gosset, who noticed that in small samples the sampling distribution is not normal. This test assesses in such samples the statistical significance of the difference between two means (Jackson, 2005). This test is applied using the following formula:

$$t = \frac{\bar{d}}{std(\overline{d})}$$

where t is the result of the *t-test*, $\bar{d}$ is the difference of the means and $std(\overline{d})$ is the standard deviation of the differences. The result of this equation is a real number and is usually referred to as *p value*[17]. If the *p value* is less than 0.05, then it can be assumed that the difference between the samples is significant (Vaughan, 2003). For this experiment, I used the *two-tailed paired t-test*, which is a variation of this method and compares results of different models derived from the same units. In this case the different models are the visualisations and the same units are the participants.

The *t-test* and its variants are based on the assumptions that the data follow the normal distribution and that the different compared units are independent (Zhang, 2006). The first assumption stands for this experiment, since time and correctness of the answers follow the normal distribution. Also, the second assumption stands as well, as the compared units, the results of each participant, are independent.

5.4.2 Subjects

This online survey was purposed for people who were over 18 years old, had a basic educational background (at least Primary/Secondary education) and knew English at some level (Beginner, Intermediate, Advanced, Proficient). Therefore, the *Welcome* page provided a form to the participants, where this information was required in order to proceed to the filling of the questionnaires. This data composed the demographics of this experiment's subjects.

The gathered results showed that most participants were men (19/32 ≈ 59.3%) and ages for both genders ranged from 23 to 43. The population for men and women for the age groups 23-27, 28-32, 33-37, 38-43 is illustrated in Figure 28. The average age of the participants was ≈ 27. Also, the level of education was mostly "Higher Education (University)" (28/32 = 87.5%) and the level of English was mostly "Advanced" (23/32 ≈ 72%). There were no statistically significant differences between the results for any of these demographics.

[17] The *p-value* is a measurement to define if the results of an experiment are caused by chance or not. The lower the *p-value* the smaller the probability that the results are due to chance. (IFFGD Glossary, 2012)

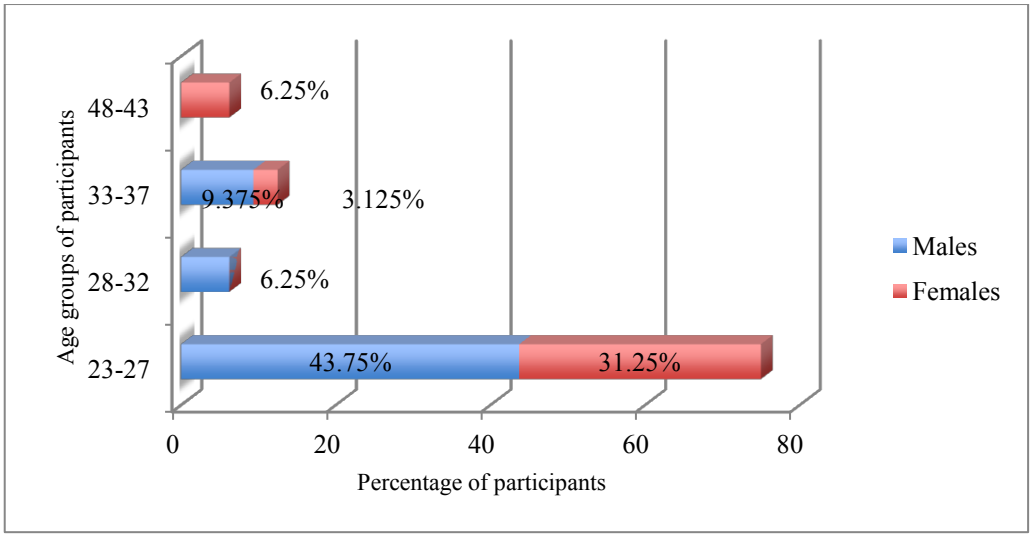

Figure 28: Illustration of the survey's demographics about age and gender.

5.4.3 Time to completion

The time to complete each questionnaire was a measurement for the efficiency of each visualisation. In this paragraph this time is analysed for every visualisation and comparisons with statistical tests define whether the time differences between visualisations are significant. On average, the time spent for the whole experiment was close to 55 minutes.

For the first visualisation (Figure 12), the time spent for interacting with it and filling the questionnaire was on average close to 22 minutes. This time ranged from 7 minutes to 1 hour and 14 minutes. For the least time spent (7 minutes) the submitted answers were mostly wrong (60% wrong answers), whereas for the most time spent (1 hour and 14 minutes) all the answers were correct. One could therefore argue that the more time spent for exploring a visualisation the better the results. However in this case, this argument is only valid for the times that have the highest variance from the mean.

Regarding the second visualisation (Figure 11), the average time to complete the corresponding questionnaire was close to 22 minutes, as for the previous visualisation. In this case, the minimum time spent was close to 5 minutes, whereas the maximum was close to 1 hour and 7 minutes. Again, the correctness of answers was very low for the minimum time spent (5 minutes), where only 2/10 questions were answered correctly. However, for the maximum time spent (1 hour and 7 minutes) the given answers were not completely correct, which is a contradiction to the results for the first visualisation. Even when too much time is spent for the questionnaire of this visualisation, the number of correct answers is not necessarily bigger than the one for the average time spent.

For the interaction with the third visualisation (Figure 13) the average time was close to 12 minutes. This is the minimum average time spent of all the three visualisations. However, the range of time spent is similar with the ones for the other visualisations and is from 5 minutes to 1 hour and 3 minutes. Again, contradicting the argument for the first visualisation, for times far from the mean no significant difference was noted regarding the number of correct answers compared to times closer to the mean.

As earlier explained, in order to examine the significance of time differences between the evaluations of the three visualisations, *two-tailed paired t-tests* were applied. These tests were between the times each participant resulted for each visualisation. The times for each visualisation for the 32 participants are illustrated in Figure 29. The test for the first and the second visualisation showed that there is no significant difference between the two sets of

results ($p \approx 0.6123$). However, the comparison between the first with the third ($p \approx 0.0008$) and the second with the third ($p \approx 0.00005$) showed that the time difference is statistically significant for the third visualisation. In other words, the test resolved that the assessment of the third visualisation requires significant less time than the rest.

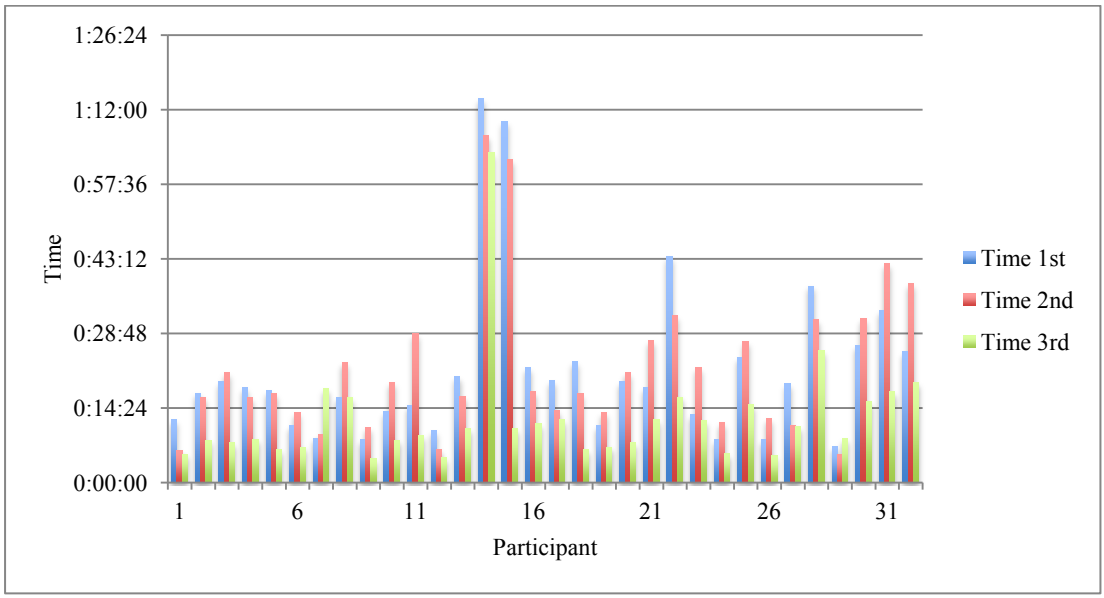

Figure 29: Illustration of the time measurements for each of the 32 participants, per evaluation process of the three visualisations.

5.4.4 Correctness of answers

For each questionnaire and each participant, the correctness of answers was enumerated. As a result, the correctness of answers showed which questions more frequently corresponded to correct, wrong and blank answers. This resolution allows the determination of the question that was the easiest or hardest for the participants to answer correctly. In this paragraph the correctness of answers is analysed per visualisation and statistical tests determine the significance between the different results.

For the questionnaire about the first visualisation (Figure 12) the participants answered correctly on average 7.3 questions out of 10. The most frequently answered correctly question was about the visual task *Compare within entities*, with 93.75% counted correct answers. The most frequently false answer was for the question about the visual task *Associate*, with 43.75% counted wrong answers. From the questions left unanswered, there was not one that had a significant count overall.

In the questionnaire about the second visualisation (Figure 11) the correct answers on average were 6.06. This is one less correctly answered question than the first visualisation. As analysed, the most usually answered correctly question was about the visual task *Compare within entities*, with 87.5% counted correct answers. This task resulted as well the most correct answers in the first visualisation. The question that was mostly answered wrong was about the visual task *Cluster*, with 68.75% counted wrong answers. As noted, this percentage is higher than the one for wrong answers regarding the first visualisation. The usually left blank question corresponded to the *Rank* task, with 15.625% counted blank answers.

In the last questionnaire about the third visualisation (Figure 13) people correctly answered on average ≈ 7.6 questions. This is the highest calculated average for correct answers from the three visualisations, but really close to the equivalent result about the first visualisation. The participants answered most frequently correctly the question about the *Compare within entities* task, with 96.875% counted correct answers. On the other hand, they usually

answered wrong the question about the *Correlate* task, with 40.625% counted false answers. Finally, more than half the participants left the question about the *Identify* task blank, with 56.25% counted blank answers. The last calculation about questions left blank is the highest of all the three visualisations.

Now that each visualisation has been analysed as for the correctness of the answers it derived, it is essential to see the relations between true, false and blank answers between the visualisations. In Figure 30 the answers that were correct are illustrated according to the percentages of their corresponding counts for each visualisation. As can be seen, the first and the second visualisations have similar results about the first three questions (*Compare within entities*, *Associate*, *Distinguish*). However, their results differentiate in later questions. The third and the second visualisation seem to have analogous results for questions 6-8 (*Correlate*, *Locate*, *Categorise*). Also, the third visualisation has a big difference in its results regarding the ninth question, which as aforementioned is about the *Identify* task.

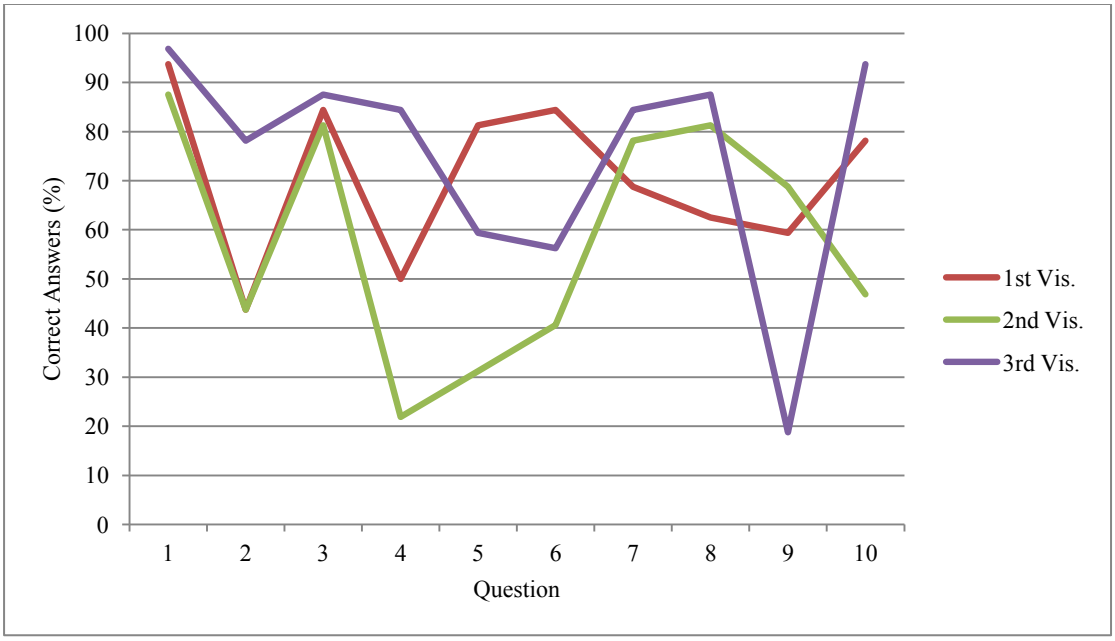

Figure 30: Illustration of the percentage for correct answers per question and visualisation experiment.

In Figure 31, the percentages for wrong answered questions are illustrated regarding the visualisation they concern. One can observe that the percentages overall are lower than these for correct answers. Furthermore, again the first and the second visualisation have similar results about wrong answers for the first three questions (*Compare within entities*, *Associate*, *Distinguish*). Also, all the visualisations have analogous results for the questions 7-9 (*Locate*, *Categorize*, *Identify*). The highest accumulated percentage for wrong answers, as can be seen, corresponds to the second visualisation and with high picks for questions 4-6 (*Rank*, *Cluster*, *Correlate*).

Consequently, Figure 32 shows the percentages for questions left blank for each visualisation. Overall the calculated percentages for these types of answers are below 20%, which means that at most 2 questions per questionnaire were left unanswered. As can be seen, for the first four questions (*Compare within entities*, *Associate*, *Distinguish*, *Rank*) these percentages are similar. The highest pick is reached by the third visualisation for the question about the task *Identify*. This percentage explains the equivalent low percentage for true answers for this visualisation.

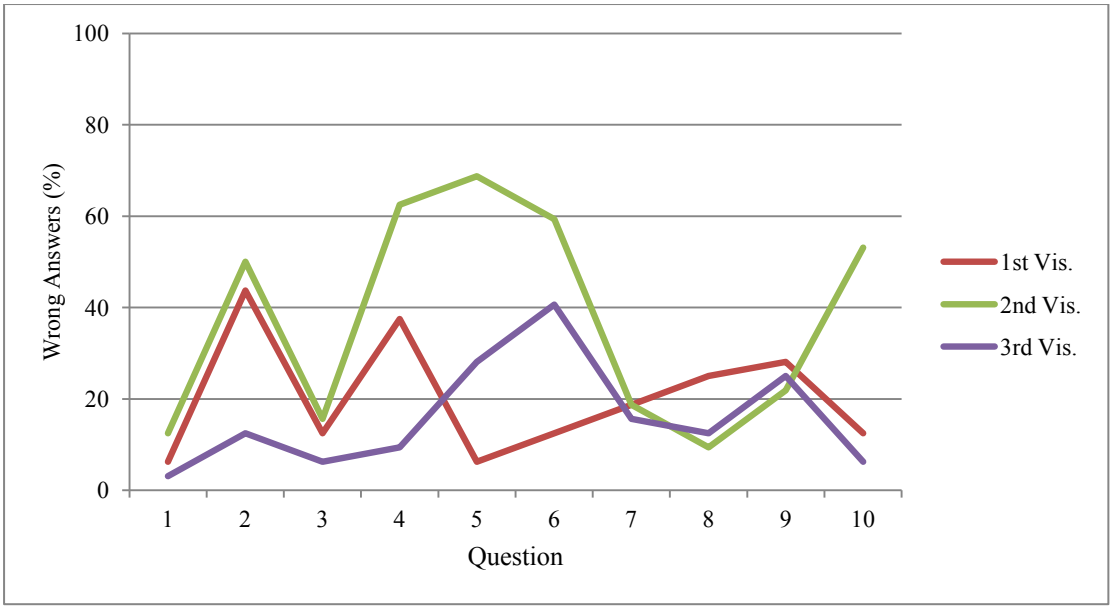

Figure 31: Illustration of the percentage for wrong answers per question and visualisation experiment.

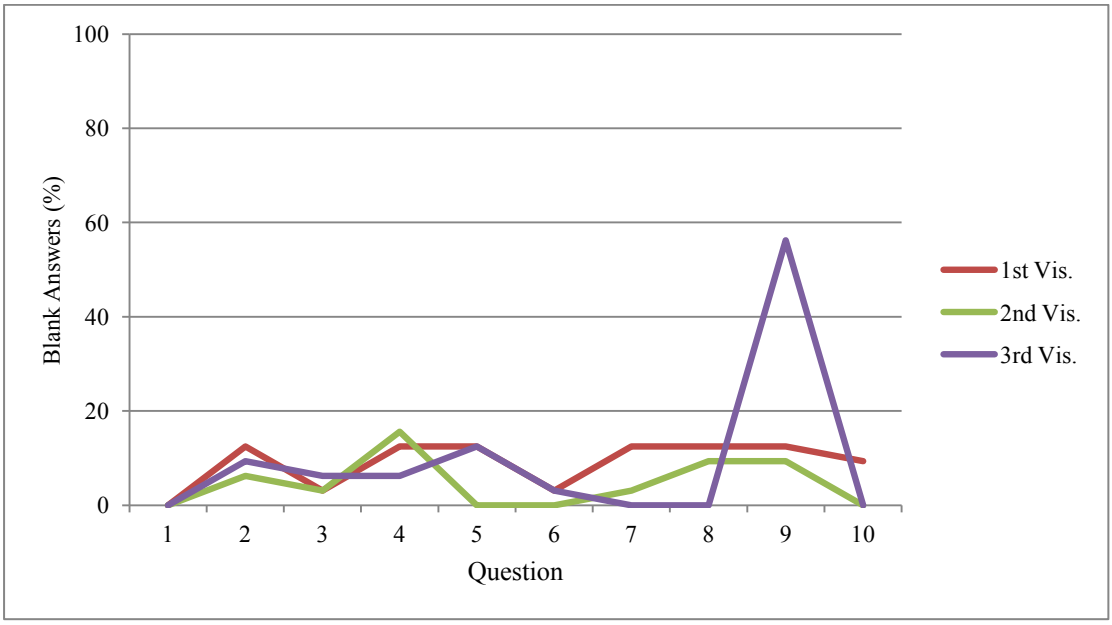

Figure 32: Illustration of the percentage for blank answers per question and visualisation experiment.

Moreover, the *t-test* for the first and the second visualisation regarding the correctness of answers revealed that the first visualisation retrieves significantly more correct answers than the second ($p \approx 0.0008$). However, the differences between the first and the third visualisation regarding this measure were not statistically significant ($p \approx 0.27$). Finally, the comparison of the second with the third visualisation with this measure showed that the first results significant more correct answers than the latter ($p \approx 0.000007$). As a result, both the first and third visualisation have better results than the second, regarding the correctness of answers.

5.4.5 Visualisations performances

As earlier explained, two measures define the performance of each visualisation regarding the visual tasks: the correctness of answers and the time spent of the equivalent experiment. In order to retrieve the visualisation that achieves the best performance, I calculated for each

participant and each visualisation the number of correct answers per minute. In other words the performance was calculated according to the module:

$$Number\ of\ correct\ answers/Minute$$

For the first visualisation the average performance is close to 0.45. This figure corresponds to spending on average 22 minutes for answering correctly all the questions in the first visualisation assessment. The average performance for the second visualisation is close to 0.37, which means that on average at least 27 minutes are required for answering correctly the 10 questions. Lastly, the average performance for the third visualisation is close to 0.8, which is the highest performance of all the visualisations. This result means that only 12.5 minutes are required on average to correctly answer all the questions in the third visualisation's assessment.

The number of correct answers over time is illustrated in Figure 33. Specifically, in this illustration the vertical axis represents the average time spent and the horizontal axis represents the number of correct answers. So, each point in this graph shows for each visualisation how much time is needed on average to have the according number of correct answers. As expected, as the time increases the number of correct answers increases as well, with a few exceptions. One remarkable exception is for the second visualisation, where for people that only answered correctly 4/10 questions spent on average more time than people who gave more correct answers. Another important thing to notice is that for the third visualisation people spent almost the same amount of time, when answering correctly from 5 to 10 questions.

Furthermore, the *t-test* for the performances of the first and the second visualisation revealed that the first achieves significantly better results ($p \approx 0.05$). In the same sense, the comparison of the second with the third visualisation showed the latter results significant better performance than the first ($p \approx 1.21967E\text{-}08$). Finally, the differences between the first and the third visualisation regarding this measure were as well statistically significant ($p \approx 1.37548E\text{-}07$). From all tests about performance one can easily conclude that the third visualisation achieves the better results.

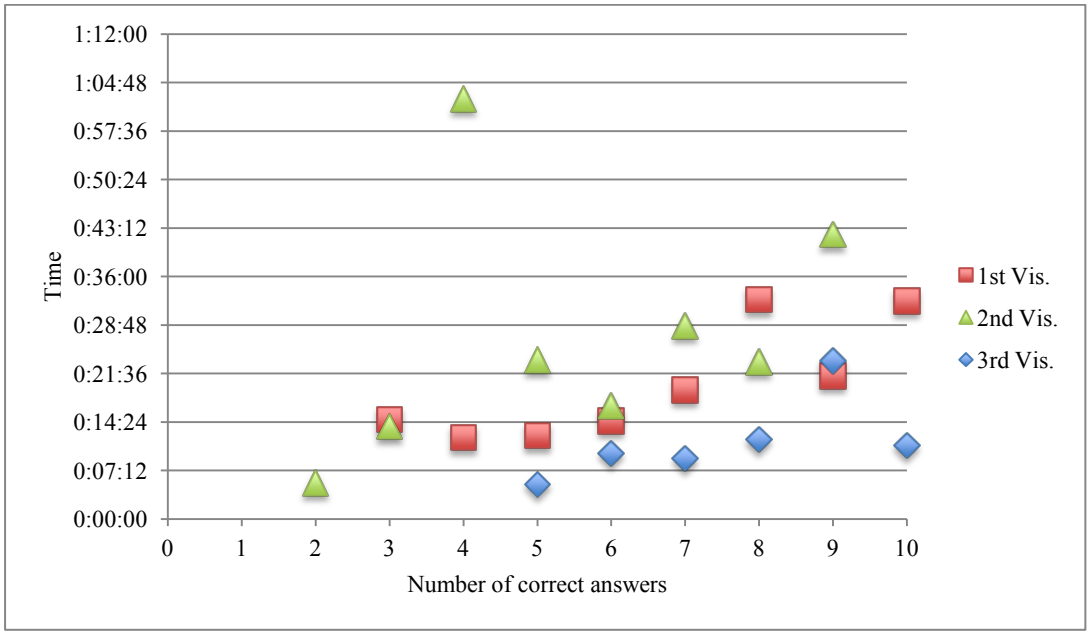

Figure 33: Illustration of the average time per number of correctly answered questions for each visualisation.

5.4.6 Feedback

The final analysis of this experiment concerns the provided feedback. As earlier described, in the end of the experiment, a *Feedback* page was displayed, asking people's opinion about the visualisations and the evaluation procedure. This was not a necessary step of the evaluation methodology, but it offered insights about the difficulties a participant might have dealt with during the assessment procedure, and his preferences regarding the visualisations.

Moreover, half of the participants submitted their feedback. For the selection of the preferred visualisation, most of the people (43.75%) liked the second visualisation (Figure 11). This is a surprising result, since the second visualisation had the biggest number of wrong answers and the worst performance of all the rest. Additionally, for the question about the easier to understand visualisation, most people (56.25%) selected the third visualisation (Figure 13). This was expected, since overall the participants spent less time to complete the corresponding questionnaire and answered more correctly than for the rest visualisations. In the same manner, most people (75%) designated the third visualisation as the easiest to handle of all the three. Again, this was anticipated as the better handling was reflected to the time needed to fill the corresponding questionnaire and to the correctness of answers.

Finally, in the *Feedback* page the participant was asked about the process that required most of his time throughout the whole experiment. These processes were: *understand the questions*, *understand the data*, *understand the chart* and *decide the answer*. Most of the people (43.75%) selected the process regarding the data. However, many people (37.5%) selected the option about understanding the questions, which gives insights about the questionnaires. Less people (18.75%) submitted that they spent time understanding the visualisation, which is a good feedback about the acceptance of the used visualisations. Notably, no participant submitted that he spent time for deciding the answer of each question. So, according to the feedback people spent most of their time understanding the visualised data and the questions. The time consumption according to the provided feedback is illustrated in Figure 34.

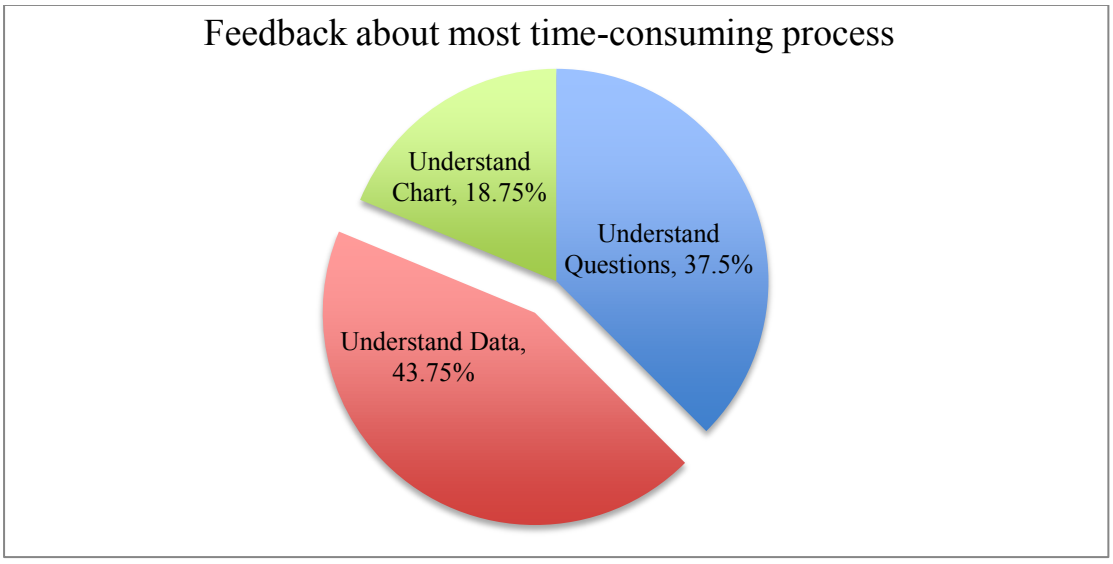

Figure 34: Time consumption over the assessment procedure according to the provided feedback.

5.5 Summary

This section gave a thorough description of the evaluation methodology applied in this project. The overview of the methodology intended to inform the reader about the reasons for applying this methodology and the research that it is based on, meaning the visual tasks.

Consequently, these tasks were defined, using examples of their application on visualisations like the ones evaluated. Also, the assessment experiment of the evaluation methodology was analysed as for its steps and its form.

Furthermore, the results of this assessment experiment were analysed. In addition, the statistical testing was defined, which was later applied for the analyses of the results. These analyses involved the examination of the time the participants spent to finish the assessment procedure for each visualisation. Also, they included the correctness of the answers retrieved from the questionnaires. These two measures defined the performance of each visualisation, which was as well analysed for each visualisation and all of them together.

Finally, the last step of the evaluation methodology involved the analysis of the provided feedback by the participants. This feedback offered an intuition about the visualisations and the experiment, independent of the rest results. Overall, this evaluation methodology designated the effectiveness of the three time-series visualisations regarding knowledge retrieval.

6 Discussion

This section discusses the outcomes of this research. Firstly, it remarks the observations about the used visualisation methods and reports the aborted visualisation ideas. Secondly, it comments about the assessment experiment and interprets its results. This section intends to state and criticise the final accomplishments of this project based on the initial objectives.

6.1 Visualisation remarks

6.1.1 Used Methods

For the enhancement of trends discovery, interactive visualisations were implemented for TV related Twitter content and subtitles data. Before the decision of using libraries that allow the creation of visualisation via programming, the popular technologies that offer an interface for the same purposes were analysed. This analysis brought out the weaknesses of such visualisation methods, like restrictions about the data that can be visualised.

The libraries that were used in the end are developed in JavaScript and are free to use. In total, more than the three libraries analysed in Section 4 were tested. However, those three were selected, as they were the most adequate for the efficient visualisation of large-scale data. Three facts determined their efficiency:

- The offer of the five interaction and distortion techniques, analysed in Section 2
- The rich gallery of different visualisations
- The user-friendly visualisations designed for large-scale data

Other JavaScript libraries that have been tested are: *D3.js*, *Protovis*, *jqPlot* and *JIT*. In brief, these libraries were either not as flexible as necessary to visualise the given data or they did not offer sufficient documentation for their efficient use. In contrast, the used libraries were very well documented.

Finally, after testing different ideas and technologies, the data of this project were in depth analysed. Therefore, for this data the *Information Visualisation* can reveal fashions about:

- The time of day when people tweet the most
- The preferred device for tweeting
- Sentiment and volume tendencies of each gender

However, as determined in Section 4, it cannot bring out valuable insights about the location of Twitter users, since they tend to keep such information private. In other words, the available data for location is not sufficient for an accurate visualisation of the volume and the average sentiment of tweets.

6.1.2 Aborted ideas

In the process of finding the use cases that could augment trends discovery, some ideas about visualisation were discarded. However, the testing of different ideas was vital for the exploration of the most contributing ideas for knowledge retrieval. That is because visualising the data led to a better understanding of the available information and to an exploration of the ways this information could be visualised.

One idea that was aborted was the one of visualising the co-occurrences of *Concepts* in tweets for different TV series. This idea was an inspiration from a similar visualisation about the co-occurrences of characters in different chapters of the book *Les Misérables* (Victor Hugo)

developed by Mike Bostock[18] with the *D3.js* library. This was not contributing for the company's goals, because they were mostly interested in revealing trends about the relation of tweets with *Concepts* over time and not patterns in general. The logic behind this decision is that the end users of any visualisation will be TV producers and they would be keener to view people's reaction on what is played on TV than observing other patterns.

One the other hand, another idea that was discarded was the one of visualising live data. For this rejection the main reason was the data. In order to have a live stream of data for the visualisation of trends over time, both datasets about tweets and *Concepts* should be available simultaneously. Besides not having the ability to connect to the company's continuously updated databases, it would be infeasible to reassure that the data for both datasets would be synchronised. Also, even if the data were in sync, there would be no point of visualising continuously updated visualisations for trends discovery. The latter stands because of the fact that observing a visualisation in order to retrieve knowledge requires some time and cannot be as fast as the data flows into the system.

Finally, an idea that was not completely rejected, but did not contribute much was the visualisation of sentiment analysis and volume per UK location. This idea was a proper use case and was implemented, as described in Section 4. However, after discussion with the company and experimenting with the data, it was clear that no valuable information could come out from the location related to tweets. That is because, as aforementioned, the data for location is insufficient. Therefore any visualisation regarding location would not be representative of all tweets. However, this visualisation has been included in the visualisation toolbox, because for other social network data with more information about location, this information retrieval would be valuable.

6.2 Evaluation resolution

Although the results of the assessment procedure were noted, it is still necessary to point out the essential conclusions about them. In this part of the document, the choice about the most adequate time-series visualisation is stated. Likewise, my proposition to Backflip for a solution to their problem is described. Finally, from the provided feedback the assessment experiment is appraised.

6.2.1 Visualisation choice

The assessment experiment was based on the visual tasks by Wehrend and Lewis (1990), evaluating the level that participants achieve to interact with each visualisation. This level was tested by questionnaires for each visualisation, asking questions related to the visual tasks. These questionnaires resolved three measures: time of completion, correctness of answers and performance.

Each of these measures were analysed in Section 5. Regarding the time of completion, the third visualisation (Figure 13) achieved the best time on average and its difference from the rest results was also statistically significant. As for the correctness of answers, again the third visualisation delivered the best results and the *t-tests* revealed that its relevant outcomes were significantly better than the ones of the rest. Finally, the results declared that the third visualisation fetched the highest performance that was, as the rest measures, significantly better than the performances of the rest. From all the above, it is clear that the third visualisation is the most adequate for time-wise trends discovery regarding Twitter content. Notably, this is also the easiest visualisation to understand and handle, as ascertained from the provided feedback.

[18] *Les Misérables* Co-occurrence, http://bost.ocks.org/mike/miserables/

Moreover, the experiment revealed good results for the first visualisation (Figure 12) regarding all the measures. These results were significantly better than the equivalent for the second visualisation. Because of this, it was worth including this visualisation in the final experiment, as it achieved the second best results. However, the second visualisation was the people's choice, according to the feedback, as the preferred visualisation from the three. This supports my choice to evaluate this visualisation. The fact that people mostly preferred the second visualisation could be a motivation for editing it to achieve better results or using it on other data for knowledge retrieval.

My recommendation to Backflip in order to identify trends more efficiently is to use a visualisation toolbox, as the one I implemented, correlating all visualisations regarding a TV channel or series in the proved most effective time-series chart. Specifically, I suggest that the outcome for a query about a channel for a time-interval returns a *Bubble Chart* (Figure 13 – evaluated as the most efficient) visualising the TV series per parts of day. From this visualisation each element (TV series) should link to the according sentiment and volume analysis over time visualised by the second best time-series visualisation (Figure 12). The latter time-series chart correlates the sentiment and volume of tweets with the *Concepts* dataset and allows the discovery of trends according to Backflip needs. Furthermore, all the rest visualisations about demographics should be available for any picked time-interval in each case, revealing more trends, like devices used, etc.

6.2.2 Appraisal of the assessment experiment

The evaluation methodology was one of the challenges of this project, as explained in Section 3. This approach for evaluating visualisations required the composition of questions for the testing of visual tasks. Although the method was similar to the one of Morse et al. (2000), which was argued effective for information visualisation, the questions were different and could have an effect on the final outcomes. Therefore, the provided feedback was used to reveal insights about this assessment experiment. These insights are analysed, as they could be valuable for a future similar approach evaluating visualisations.

As stated in Section 5, most people submitted that they spent most of their time trying to understand the visualised data. This shows that the data should be more effectively explained, before the participant proceeding to the experiment. A lot people submitted as well, that they consumed a lot of time trying to understand the questions. This acknowledges the unavoidable weakness of this method that the outcomes depend on how the questions are formed. Also, it designates that making the questions simpler would make the experiment more pleasant for the participants, which could have again an effect on the final outcomes.

Although the visual tasks are easy in practice, creating questions that require the user to perform them is a hard process. The cultivation of each question would probably resolve more confident results, but this was not possible for this project due to time limitations.

7 Conclusion

This study examined the *Information Visualisation* regarding TV related activity from the social network Twitter. Specifically, it analysed the research around this field, describing the visualisation approaches that can enhance trends discovery and information retrieval from large-scale data. For this analysis different visualisations for this data and for TV subtitles have been developed. Such visualisations on Twitter data in combination with TV subtitles data attempted to improve the information retrieval regarding the reactions of people on what is played on TV.

Furthermore, a visualisation toolbox has been developed that gathers the most valuable visualisations for trends discovery. This toolbox provides query methods that allow the search of Twitter content regarding TV activity and reveal all the relevant visualisations. This system has been developed with the ability to adjust any Twitter data regarding information retrieval. Moreover, visualisations over time, also known as time-series charts, have been developed for a better inspection of Twitter activity over time and in relation with TV subtitles.

Finally, the time-series visualisations have been evaluated via an assessment experiment on the visual tasks, defined by Wehrend and Lewis (1990). This experiment revealed the most adequate time-series chart for information retrieval on large-scale data. In general, this research showed the abilities of *Information Visualisation* on Twitter data given the modern visualisation technologies and appraised the used assessment experiment regarding the evaluation of visualisations.

7.1 Future work

One of the main deliverables of this project is the visualisation toolbox, which attempts to provide a tool for trends discovery from Twitter content. In this research this toolbox has not been evaluated regarding its contribution on *Information Visualisation*. Therefore, a possible future research could be done for its appraisal, testing whether it is better for all the visualisations to be gathered in one tool, when trying to reveal fashions on social network activity.

Lastly, another possible enhancement of this project could be the improvement of the assessment experiment for the time-series visualisations. This experiment could have TV producers as participants, who will be the end users of such interfaces. As a result, the results would concern only the end users and would be more specific for this domain. However, this was not possible during this research, because it was not feasible to find such subjects for the experiment.

8 Bibliography

Bhatnagar, A., Ghose, S. (2004). Online information search termination patterns across product categories and consumer demographics. *Journal of Retailing*, *80*(3), pages 221–228.

Bifet, A., Frank, E. (2010). Sentiment Knowledge Discovery in Twitter Streaming Data, *Discovery Science 6332*, pages 1–15.

Boer, T. D., Verkooij, K. (2011). Interactive Visualization Toolkits for Rich Internet Publications. Technical Report, Ultrecht University, Department of Information and Computing Sciences.

Boulos, M. N. K., Warren, J., Gong, J., & Yue, P. (2010). Web GIS in practice VIII: HTML5 and the canvas element for interactive online mapping. *International journal of health geographics*, *9*, 14.

Boyd, D. M., Ellison, N. B. (2007). Social Network Sites: Definition, History, and Scholarship. *Journal of Computer-Mediated Communication*, *13*(1), pages 210-230.

Cheong, M., Lee, V., Stiller, B., Boyle, S. (2009). Integrating Web-based Intelligence Retrieval and Decision-making from the Twitter Trends Knowledge Base. *In Proceedings of CIKM 2009 Co-Located Workshops: SWSM 2009*, pages 1–8.

Chuah, M. C., Roth, S. F. (1996). On the Semantics of Interactive Visualizations. *In proceedings of the IEEE Symposium on Information Visualisation (InfoVis'96)*, pages. 26-36.

Dix, A. (2011). Information Visualization. PROMISE Winter School 2012, Information Retrieval meets Information Visualization. Zinal, Valais - Switzerland 23 - 27 January 2012 Dundee, 24-26 Oct 2011.

Ellis, G., Dix, A. (2006). An explorative analysis of user evaluation studies in information visualisation. *In proceedings of the 2006 AVI workshop on BEyond time and errors novel evaLuation methods for Information Visualization (BELIV), ACM, 2006*.

Faisal, S., Cairns, P., Blandford, A. (2007). Challenges of Evaluating the Information Visualisation Experience, *In proceedings of the 21st British HCI Group Annual Conference on People and Computers: HCI...but not as we know it – Volume 2, BCS-HCI,* pages 167-170.

Google Chart Tools (2012, September 17). Handling Events. Retrieved September 17, 2012, from https://developers.google.com/chart/interactive/docs/events

Grudin, J., Viégas, F. B., Wattenberg, M. (2005). Tag Clouds and the Case for Vernacular Visualization, *ACM Interactions, 15(4),* pages 223-252.

IFFGD Glossary (2012, September 22). P-value definition. Retrieved September 22, 2012, from http://www.iffgd.org/site/learning-center/glossary

Jackson, S.L. (2005) Research Methods and Statistics: A Critical Thinking Approach (2 edition). Wadsworth Publishing.

JSON.org (2012, September 18). Introducing JSON. Retrieved September 18, 2012, from http://www.json.org

Keim, D. a. (2002). Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, *8*(1), pages 1-8.

Leskovec, J., Adamic, L. a., & Huberman, B. a. (2007). The dynamics of viral marketing. *In proceedings of 7th ACM Conference on Electronic Commerce.*

Machlis, S. (2011, April 20). 22 free tools for data visualization and analysis. *ComputerWorld*. Retrieved April 19, 2012, from http://www.computerworld.com/s/article/9215504/22_free_tools_for_data_visualization_and_analysis

Makice, K. (2009). Twitter API: Up and Running Learn How to Build Applications with the Twitter API, O'Reilly Media, Inc.

Mazza, R. (2009). Introduction to Information Visualisation. Springer, London, UK.

Morse, E., Lewis, M., Olsen, K. (2000). Evaluating visualizations: using a taxonomic guide. *International Journal of Human-Computer Studies*, *53*(5), pages 637–662.

Northman, J. (2008). Learning Named Entity Recognition from Wikipedia. BSc Thesis, The University of Sydney, Australia.

Pak, A., Paroubek, P. (2010). Twitter as a Corpus for Sentiment Analysis and Opinion Mining. Computer. *In proceedings of the 7th conference on International Language Resources and Evaluation (LREC'10)*, European Language Resources Association (ELRA), Valletta, Malta

Shedroff, N. (1994). Information interaction design: a unified field theory of design. In Jacobson, B.: Information Design. MIT Press (2000).

Singh, S. (2009). Social Media Marketing for Dummies. Indiana: Wiley Publishing, Inc. pp. 140-141.

Spence, B. (2007) Information visualization: design for interaction, 2nd Edition, Prentice Hall.

Twitter Team (2012, March 21). Twitter Turns Six. Twitter Blog. Retrieved April 19, 2012, from http://blog.twitter.com/2012/03/twitter-turns-six.html

Vaughan, L. (2003). Statistical Methods for the Information Professional: a Practical, Painless Approach to Understanding, Using, and Interpreting Statistics. Information Today, Medford, NJ.

Ware, C., Mire, B., Snyder, C. (2000). Information Visualisation: perception for design, Morgan Kaufmann Publishers Inc., San Fransisco, CA.

Wasserman, S., Faust, K. (1994). Social Network Analysis: Methods and Applications. Cambridge University Press, Cambridge, UK.

Wehrend, S., Lewis, C. (1990). A Problem-oriented Classification of Visualisation Techniques. *In proceedings of the 1st conference on Visualisation '90 (VIS '90)*. Pages 139-143. IEEE Computer Society Press Los Alamitos, CA, USA.

WordNet Team (2012, September 20). WordNet search: Cartesian coordinate system. Retrieved September 20, 2012, from http://wordnetweb.princeton.edu/perl/webwn?s=cartesian%20coordinate%20system

Zhang, A. (2006) Advanced Analysis of Gene Expression Microarray Data. Danvers: World Scientific Publishing Co.

APPENDIX A: Datasets

In this appendix, examples of all the used datasets in this project are given.

Table 4: Example data from the Tweets dataset. For space economy the tweets are shortened with brackets defining the missing parts.

TV Series	TV Episode	TV Channel	Transmission Start Time	Transmission End Time	Creation Time	From User	Twitter Text	Sentiment Analysis Score	Gender	Location	Device Used
The Big Bang Theory	The Peanut Reaction	E4	2012-06-25 01:00:00	2012-06-25 01:25:00	06/25/2012 00:30:50+0100	RyanHorne94	tele would be useless [...] #howimetyourmother #bigbangtheory	-0.2178			Web
The Big Bang Theory	The Peanut Reaction	E4	2012-06-25 01:00:00	2012-06-25 01:25:00	06/25/2012 00:31:59+0100	brunanegrona	Daora que pra converter tbbt t😴😴 a meia hora em 99%	0.0000			Web
The Big Bang Theory	The Peanut Reaction	E4	2012-06-25 01:00:00	2012-06-25 01:25:00	06/25/2012 00:32:00+0100	pumpsDISorder	Nw Big bang theory <3 this show!	0.1241			iPhone
The Big Bang Theory	The Peanut Reaction	E4	2012-06-25 01:00:00	2012-06-25 01:25:00	06/25/2012 00:32:30+0100	niallsmofos69	big bang theory marathon ;) sunday nights	0.2833			iPhone
The Big Bang Theory	The Peanut Reaction	E4	2012-06-25 01:00:00	2012-06-25 01:25:00	06/25/2012 00:33:05+0100	MZhe11oKITTY	Bwahaha Sheldon ain't shit #TheBigBangTheory	-0.6772			Web
The Big Bang Theory	The Peanut Reaction	E4	2012-06-25 01:00:00	2012-06-25 01:25:00	06/25/2012 02:05:18+0100	Friz1992Ro	[...] Big Bang Theory Rubix make your own 12.99 [...] #TBBT #BBT #THINKGEEK #SHELDON	-0.0161			Web

Table 5: Example data from the Concepts dataset.

Concept	TV Channel	TV Episode	Sentence Text	Creation Time	Transmission Start Time	Transmission End Time
branches	E4	Will's Birthday	CRIES) '(BRANCHES CRACK)' Aaaaggggghhh!	06/25/2012 22:04:30.274349+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01
crack	E4	Will's Birthday	SOBS) Uncle, I'm sorry.	06/25/2012 22:04:26.501534+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01
uncle	E4	Will's Birthday	(Uncle) SOBS) 'Eragon?	06/25/2012 22:03:54.219194+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01
uncle	E4	Will's Birthday	Uncle!	06/25/2012 22:03:51.665976+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01
uncle	E4	Will's Birthday	ROARS) Uncle?	06/25/2012 22:03:51.665976+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01
uncle	E4	Will's Birthday	Take me down now!	06/25/2012 22:03:39.765718+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01
take	E4	Will's Birthday	Ag!	06/25/2012 22:03:01.433839+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01
ag	E4	Will's Birthday	Please put me down now!	06/25/2012 22:02:46.453239+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01
please	E4	Will's Birthday	Hold on!	06/25/2012 22:02:38.920966+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01
hold	E4	Will's Birthday	CRIES) '(BRANCHES CRACK)' Aaaaggggghhh!	06/25/2012 22:04:30.274349+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01
mango	E4	Will's Birthday	They like the mango, oh, oh, ohe They like the mangoe Mango, mangoe Man-mango, mango, oh, ohe Who like the mango?	06/25/2012 22:13:44.525585+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01
let	E4	Will's Birthday	Let's give that a go.	06/25/2012 22:13:41.5119+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01
clarityn rapide with	E4	Will's Birthday	Now I use Clarityn Rapide With fast-melt technology, it dissolves on your tongue in an instant.	06/25/2012 22:13:05.446335+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01
check	E4	Will's Birthday	Check.	06/25/2012 22:13:05.278218+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01
check	E4	Will's Birthday	Check!	06/25/2012 22:13:04.205862+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01
rapide	E4	Will's Birthday	Clarityn Rapide?	06/25/2012 22:13:04.205862+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01
clarityn rapide	E4	Will's Birthday	Clarityn Rapide?	06/25/2012 22:13:04.205862+01	2012-06-25 22:00:00+01	2012-06-25 22:35:00+01

APPENDIX B: JSON object with retweet data

The following text shows the content of a JSON object with retweet data.


```
{"twitter":
      {"retweeted":
            {"source": "web",
             "created_at": "Thu, 31 May 2012 23:01:14 +0000",
             "id": "208332336914374657",
             "user": {"lang": "en",
                  "utc_offset": -18000,
                  "id_str": "139848943",
                  "statuses_count": 32324,
                  "name": "\u043c\u03b9\u00a2\u043d\u0454\u2113\u2113\u0454\u2122",
                  "friends_count": 3375,
                  "created_at": "Mon, 03 May 2010 21:32:29 +0000",
                  "time_zone": "Quito",
                  "followers_count": 4853,
                  "screen_name": "BelieberNYBaby",
                  "location": "The great city of London ",
                  "geo_enabled": true,
                  "listed_count": 32,
                  "id": 139848943,
                  "description": "Me and Juju fucked for 30 seconds in a bathroom at the
staples center & we                  have a baby together. Justin followed me. You mad?
Stay mad."
            }
      },
      "id": "208359116480065536",
      "retweet":
            {"count": 48,
             "text": "\"A Fish Called Wanda\" Don't forget her idiot husband Cosmo and her
son Poof.",        "created_at": "Fri, 01 Jun 2012 00:47:39 +0000",
            "source":           "<a           href=\"http://twitter.com/#!/download/iphone\"
rel=\"nofollow\">Twitter for          iPhone</a>",
            "user":
                  {"lang": "en",
                   "utc_offset": -10800,
                   "statuses_count": 165,
                   "name": "Violet Harmon",
                   "friends_count": 168,
                   "url": "http://shoottothrill-playtokill.tumblr.com",
                   "created_at": "Sat, 17 Mar 2012 04:06:32 +0000",
                   "time_zone": "Brasilia",
                   "followers_count": 38,
                   "screen_name": "eatastar",
                   "id_str": "527075467",
                   "id": 527075467,
                   "description": "I love Skins, One Direction, The Hunger Games, Parmore,
La Vela Puerca,                The Mortal Instruments, Amy Winehouse."
            },
            "id": "208359116480065536"
      }},
      "interaction":
            {"author":
                  {"username": "eatastar",
                   "link": "http://twitter.com/eatastar",
                   "name": "Violet Harmon",
                   "avatar": "http://a0.twimg.com/profile_images/
      2255981740/430503_3155533701626_1663723938_2616496_1634873351_n_normal.png",
                  "id": 527075467},
            "created_at": "Fri, 01 Jun 2012 00:47:39 +0000",
             "content": "RT @BelieberNYBaby: \"A Fish Called Wanda\" Don't forget her idiot
husband            Cosmo and her son Poof.",
             "source": "Twitter for iPhone",
            "link": "http://twitter.com/eatastar/statuses/208359116480065536",
            "type": "twitter",
            "id": "1e1ab8363210a780e07479116b7f8542"},
      "salience":
            {"content":
                  {"sentiment": -4}},
            "language": {"tag": "en"},
      "klout": {"score": 10}
}
```

APPENDIX C: Time-series charts source code

In this appendix the code for *scatter plot* by the Highcharts library (Figure 11) is given. This is an example of the way that the evaluated visualisations were implemented.

```
<!--


First time-series visualisation (Figure 11)
File: Time_Series_Vis1.php
Description: It develops the scatter plot by
Highcharts library for the Twitter dataset.

-->
<!DOCTYPE HTML>
<html>
	<head>
		<meta http-equiv="Content-Type"
content="text/html; charset=utf-8">
<?php

	//Connect to remote MySQL database
	require "../connect.php";

	//Get the name of the episode that was queried
	$episode = $_GET["q"];

	//Retrieve the Twitter data about females for the queried
episode
	$query = 'SELECT ttext, Round(sentiment,4) as sentiment,
MID(created_at, 4, 2) AS Day, MID( created_at, 12, 2 )
	AS Hour, MID( created_at, 15, 2) AS Minutes FROM Tweets
WHERE Program ="'.$episode.'" AND gender LIKE "%female%"
	ORDER BY created_at LIMIT 0,1000';

	$data = mysql_query($query)
	or die(mysql_error());

	$i1=0;

	while($res = mysql_fetch_array( $data ))
	{
		$females[$i1]['Hour'] = $res['Hour'];
		$females[$i1]['Min'] = $res['Minutes'];
		$females[$i1]['Sent'] = $res['sentiment'];
		$females[$i1]['Day'] = $res['Day'];
		$females[$i1]['Text'] =str_replace("\n", '
',$res['ttext']);
		$i1++;
	}

	//Retrieve the Twitter data about males for the queried
episode
	$query = 'SELECT ttext, Round(sentiment,4) as sentiment,
MID(created_at, 4, 2) AS Day, MID( created_at, 12, 2 )
	 AS Hour, MID( created_at, 15, 2) AS Minutes FROM Tweets
WHERE Program ="'.$episode.'" AND gender LIKE "%male%"
	 AND gender NOT LIKE "%female%" ORDER BY created_at LIMIT
0,1000';

	$data = mysql_query($query)
	or die(mysql_error());

	$i2=0;

	while($res = mysql_fetch_array( $data ))
	{
		$males[$i2]['Hour'] = $res['Hour'];
		$males[$i2]['Min'] = $res['Minutes'];
		$males[$i2]['Sent'] = $res['sentiment'];

		$males[$i2]['Day'] = $res['Day'];
		$males[$i2]['Text'] = str_replace("\n", '
',$res['ttext']);

		$i2++;
	}
?>
<title>Avg Sentiment by Gender for <?php echo $program?></title>
```

```
<script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min
.js"></script>

<!-- Implement the visualisation using the Highcharts library -->
<script type="text/javascript">

$(function () {
	var chart;
	$(document).ready(function() {
	chart = new Highcharts.Chart({
		chart: {
			renderTo: 'container',
			type: 'scatter',
			zoomType: 'xy'
		},
		title: {
			text: "Tweets for <?php echo $episode?>
visualised by sentiment, gender and time."
		},
		subtitle: {
			text: 'Source: Backflip'
		},
		xAxis: {
			type: 'datetime',
			title: {
				enabled: true,
				text: 'Hours ([0-24h].[0-60mins])'
			},
		},
		yAxis: {
			title: {
				text: 'Sentiment (N. Bayes) %'
			}
		},
		tooltip: {
			formatter: function() {
				var time = new Date(this.point.x);

				var hour = time.getHours();
				if(hour=="00")
					hour = "23";
				else if(hour=="01")
					hour = "00";
				else
					hour = hour-1;
				var n = "\"" + time + "\"";
				var x = n.split(" ");
				x = x[4];
				n = x.split(":");
				var minutes = n[1];
				var text = this.point.t;
				var tLength = text.length;
				var newText = new Array();

				if(tLength<40){
					newText[0] = text;
				}
				else if(tLength >= 40 &&
tLength<80){
					newText[0] =
text.substring(0,40) +'<br />';
					newText[1] =
text.substring(41,tLength-1);
				}
				else if(tLength>=80 &&
tLength<=120){
					newText[0] =
text.substring(0,40)+'<br />';
					newText[1] =
text.substring(41,80)+ "\n";
					newText[2] =
text.substring(81,tLength-1);
				}
				else {
				newText[0] =
text.substring(0,40)+'<br />';
				newText[1] =
text.substring(41,80)+'<br />';
				newText[2] =
text.substring(81,120)+'<br />';
				newText[3] = text.substring(121,
tLength-1);
				}

				var returnText= 'Time: '+ hour +
':'+ minutes +'<br />' +
				'Sentiment: ' + this.point.y + '<br
/>' +
					'Text: ';
```

```
                var inbetweenText = '';
                for(i=0; i<newText.length; i++)
                {
                        inbetweenText+= newText[i];
                }

                returnText+=inbetweenText;

                return returnText;
                }

        },
legend: {
        layout: 'vertical',
                align: 'left',
                verticalAlign: 'top',
                x: 24,
                y: 1,
                floating: true,
                backgroundColor: '#FFFFFF',
                borderWidth: 1
        },
        plotOptions: {
                scatter: {
                        marker: {
                        radius: 5,
                        states: {
                        hover: {
                        enabled: true,
                        lineColor:
                'rgb(100,100,100)'
                        }
        }
        },
                        states: {
                        hover: {
                        marker: {
                        enabled: false
                        }
                        }
        },
        cursor: 'pointer',
        events: {
                click: function(event) {
                if (window.XMLHttpRequest)
                                        {// code for IE7+, Firefox, Chrome,
Opera, Safari
                                                xmlhttp=new
XMLHttpRequest();
                                        }
                                        else
                                        {// code for IE6, IE5
                                        xmlhttp=new
ActiveXObject("Microsoft.XMLHTTP");
                                                                                }

        xmlhttp.onreadystatechange=function()
                                        {
                                        if (xmlhttp.readyState==4 &&
xmlhttp.status==200)
                                        {

        document.getElementById("txtHint").innerHTML=xmlhttp.resp
onseText;
                                        }
                                }
                                var d = new Date(event.point.x);

        xmlhttp.open("GET","getConcepts.php?q="+d+"&e=<?php echo
str_replace("\\","",$program)?>",true);
                                xmlhttp.send();
                        }
                        }
        },
        },

                //Feed the visualisation with the retrieved data

                                series: [{
                                        name: 'Female',
                                        color: 'rgba(223, 83, 83, .5)',

                                        data: [<?php
                                        for($j=0;$j<$i1;$j++)
                                        {
                                                if($females[$j]['Hour'][0]
                                        == "0")
                                                {
                                                        echo '{g: "f", x:
Date.UTC(2012, 5, '.$females[$j]['Day'].',
```

```
'.$females[$j]['Hour'][1].','.$females[$j]['Min'].'),y:'.$females
[$j]['Sent'].', t: "'.$females[$j]['Text'].'"}';
                                        }
                                        else
                                                echo '{g: "f", x:
Date.UTC(2012, 5,
'.$females[$j]['Day'].'.$females[$j]['Hour'].','.$females[$j]['Mi
n'].'),y:'.$females[$j]['Sent'].', t:
"'.$females[$j]['Text'].'"}';

                                        if(($j+1)!=$i1)
                                        {
                                        echo ",";
                                        }
                                }
                                ?>
                                ]},
                                {
                                name: 'Male',
                                color: 'rgba(119, 152, 191, .5)',
                                data: [
                                        <?php
                                        for($j=0;$j<$i2;$j++)
                                        {

        if($males[$j]['Hour'][0] == "0")
                                                {
                                                echo '{g: "m", x:
Date.UTC(2012, 5, '.$males[$j]['Day'].
','.$males[$j]['Hour'][1].','.$males[$j]['Min'].'),y:'.$males[$j]
['Sent'].', t: "'.$males[$j]['Text'].'"}';
                                        }
                                        else
                                        echo '{g: "m", x:
Date.UTC(2012, 5, '.$males[$j]['Day']. ',
'.$males[$j]['Hour'].','.$males[$j]['Min'].'),y:'.$males[$j]['Sen
t'].', t: "'.$males[$j]['Text'].'"}';

                                        if(($j+1)!=$i2)
                                        {
                                                echo ",";
                                        }
                                }
                                ?>
                                ]}]
```

```
                                        });
                                });

                        });
                        </script>
</head>
<body style=" background-image:none; background-color:#333;">

<!-- Import the Highcharts library -->

<script src="../js/highcharts.js"></script>
<script src="../js/exporting.js"></script>
<link rel="stylesheet" type="text/css" href="../visualiseit.css"
/>
<link rel="stylesheet" type="text/css" href="../ddsmoothmenu.css"
/>

<!-- Render the visualisation -->
<div id="container" style="max-width: 1000px; height: 400px;
margin: 0 auto"></div>
<div id="txtHint"><b>Concepts will be listed here</b></div>
        </body>
</html>
```

APPENDIX D: Experiment Questionnaires

In this appendix the questionnaires, which were not included in the Section 3 are given.

Table 6: The questionnaire for the evaluation of the time-series visualisation illustrated in Figure 11. For each question the corresponding order in the questionnaire (No.) and the relevant visual task (Task) is given.

No.	Task	Question
1	Compare within entities	At 00:01 which gender posted the most positive tweet?
2	Associate	Which of the two times 00:39 and 01:03 is associated with more concepts?
3	Distinguish	At 01:16 a tweet has sentiment analysis close to 1 (positive). This point belongs to:
4	Rank	For females rank times 01:25, 01:30, 01:33 with respect to their sentiment analysis value.
5	Cluster	At 01:05 which sentiment analysis value corresponds to more than one tweet?
6	Correlate	At 00:25 what is common between all genders regarding sentiment analysis?
7	Locate	During which of the following time periods people tweeted more frequently?
8	Categorize	Most points after 01:30 are regarding their sentiment analysis:
9	Identify	Which gender is the last to tweet about "The Money Pit"?
10	Compare between relations	At 01:28 if you wanted tweets with sentiment analysis equal to -1 which tweet would you ignore?

Table 7: The questionnaire for the evaluation of the time-series visualisation illustrated in Figure 13. For each question the corresponding order in the questionnaire (No.) and the relevant visual task (Task) is given.

No.	Task	Question
1	Compare within entities	For which TV series people tweeted the most during the dawn?
2	Associate	Which TV series (singular) is associated with more tweets than the rest?
3	Distinguish	Which TV series (singular) has average sentiment analysis equal to 0?

No.	Task	Question
4	Rank	Rank the TV series Crimewatch Roadshow, Bargain Hunt, Panorama according to their sentiment analysis.
5	Cluster	During which part of the day did people tweet the least?
6	Correlate	Which part/s of the day has/ve only TV series with negative sentiment?
7	Locate	One TV Series has the most positive sentiment analysis compared to the rest TV Series. Where is it located?
8	Categorize	Crimewatch Show... [sentence follows that is related to a category, e.g. TV series on air during dawn and with positive sentiment.]
9	Identify	Which two TV series have the same volume of tweets during the dawn?
10	Compare between relations	If you wanted to observe parts of day that have only TV series with sentiment analysis <=0, which part/s of day would you ignore?

APPENDIX E: Experiment results

In this appendix the experiment results are given as further reading. Table 8 includes the results regarding the questionnaires and Table 9 the results about the provided feedback.

Table 8: All the results from the assessment experiment about the time-series visualisations. The first four columns show the demographics of each participant and the rest show the number of correct answers, the time and the answer-trace this participant achieved for each visualisation.

English	Education	Gender	Year Born	Correct 1st	Time 1st	Trace 1st	Correct 2nd	Time 2nd	Trace 2nd	Correct 3rd	Time 3rd	Trace 3rd
Advanced	Higher (University)	Male	1977	9	00:12:07	TFTTTTTTTT	8	00:06:16	TFTTFTTTTT	6	00:05:29	TTTTFTFT_F
Advanced	Higher (University)	Female	1969	4	00:17:17	TFTFFFFTT_	3	00:16:28	F__FFFTTFT	6	00:08:12	TFTFFTTT_T
Advanced	Higher (University)	Male	1987	8	00:19:28	TFTFTTTTTT	5	00:21:15	TFTFFFTTTF	7	00:07:47	TTTTFFTT_T
Intermediate	Primary/Secondary	Male	1988	7	00:18:24	TTT_TT__TT	6	00:16:22	TFTTFTF_TT	6	00:08:24	TTFTFTTF_T
Advanced	Higher (University)	Female	1971	9	00:17:46	TTTT_TTTTT	6	00:17:09	TTT_FFTTFT	10	00:06:27	TTTTTTTTTT
Advanced	Higher (University)	Male	1988	10	00:10:57	TTTTTTTTTT	6	00:13:37	TTTFFFTTTF	7	00:06:51	TTTT_FTT_T
Advanced	Higher (University)	Female	1986	5	00:08:35	TFFFTTTTFF	6	00:09:20	TFTFFTTTTF	7	00:18:11	TTTTTFTT_F
Advanced	Higher (University)	Male	1985	8	00:16:21	TFTTTTTTFT	6	00:23:12	TFTFFFTTTT	6	00:16:21	T_TTFFTTFT
Advanced	Higher (University)	Female	1989	7	00:08:14	TFTFTTTT_T	8	00:10:39	TTTFTTTTTF	8	00:04:41	TTTTTFTT_T

English	Education	Gender	Year Born	Correct 1st	Time 1st	Trace 1st	Correct 2nd	Time 2nd	Trace 2nd	Correct 3rd	Time 3rd	Trace 3rd
Advanced	Higher (University)	Female	1988	7	00:13:42	TFTTTFTTFT	6	00:19:22	TTTFFFT_TT	7	00:08:13	TTTTFFTTFT
Advanced	Higher (University)	Male	1989	3	00:14:47	FFTFTFFFTF	6	00:28:44	TTFFFTTTTF	6	00:09:10	TFTTTFFT_T
Advanced	Primary/Secondary	Male	1988	9	00:10:04	TTTFTTTTTT	8	00:06:26	TTTFTTTTTF	9	00:04:52	TTTTTTTT_T
Proficient	Higher (University)	Male	1978	5	00:20:26	TFFTTTTFT_	6	00:16:39	TFTFFTTTTF	6	00:10:21	TTTT_TFF_T
Advanced	Higher (University)	Male	1988	10	01:14:12	TTTTTTTTTT	8	01:06:55	TTTFTTTTTF	9	01:03:42	TTTTTTTT_T
Intermediate	Higher (University)	Male	1987	8	01:09:47	TFTFTTTTTT	4	01:02:24	TFFFTFTFTF	8	00:10:24	TTTTTTFT_T
Advanced	Higher (University)	Male	1985	6	00:22:08	TFTFTTFT_T	7	00:17:35	TFTTTFTTTF	6	00:11:20	T_TTFFTT_T
Proficient	Higher (University)	Male	1979	9	00:19:47	TTTTTTTTFT	6	00:13:57	TFFTFTFTTT	10	00:12:13	TTTTTTTTTT
Advanced	Higher (University)	Female	1988	7	00:23:18	TTTT_TTT__	6	00:17:09	TTT_FFTTFT	10	00:06:27	TTTTTTTTTT
Advanced	Higher (University)	Male	1986	10	00:10:57	TTTTTTTTTT	6	00:13:37	TTTFFFTTTF	7	00:06:51	TTTT_FTT_T
Proficient	Higher (University)	Male	1984	8	00:19:28	TFTFTTTTTT	5	00:21:15	TFTFFFTTTF	7	00:07:47	TTTTFFTT_T
Advanced	Higher (University)	Female	1988	7	00:18:24	TTT_TT__TT	8	00:27:22	TFTTTTFTTT	9	00:12:13	TTTTTTTTFT
Intermediate	Primary/Secondary	Female	1977	9	00:43:37	TTTTTTTFTT	5	00:32:14	TTT_FFT_FT	8	00:16:30	TTTFTTTT_T

English	Education	Gender	Year Born	Correct 1st	Time 1st	Trace 1st	Correct 2nd	Time 2nd	Trace 2nd	Correct 3rd	Time 3rd	Trace 3rd
Advanced	Higher (University)	Male	1989	6	00:13:11	TT_TT__TFT	8	00:22:10	TTT_TTTTTF	10	00:11:58	TTTTTTTTTT
Advanced	Higher (University)	Male	1988	6	00:08:21	TFTFTTT_TF	5	00:11:32	TFTFFFTTTF	7	00:05:32	TT_TFTTT_T
Advanced	Higher (University)	Male	1986	9	00:24:11	TTTTTTT_TT	7	00:27:13	TFTTFTFTTT	9	00:15:07	TTT_TTTTTT
Advanced	Higher (University)	Female	1989	5	00:08:17	T_TT_T_FFT	6	00:12:18	TTTFFFTTFT	5	00:05:08	TFT_TFFTFT
Advanced	Higher (University)	Female	1988	7	00:19:10	TTT_TTFT_T	3	00:11:02	FFTFFFTT_F	7	00:10:49	TTTT_FTTFT
Proficient	Higher (University)	Female	1985	8	00:37:56	T_TTTTTFTT	7	00:31:24	TFT_TFTTTT	9	00:25:33	TTTTTTTTFT
Advanced	Higher (University)	Male	1982	4	00:06:57	TFFFTTFFFT	2	00:05:22	FFFTFTFF_F	6	00:08:33	TT_TT_TFFT
Intermediate	Higher (University)	Male	1988	7	00:26:32	T_TT_TTFTT	5	00:31:39	T_TFFFTTFT	8	00:15:41	T_TTTTT_T
Advanced	Higher (University)	Female	1985	10	00:33:14	TTTTTTTTTT	9	00:42:18	TTTTT_TTT	10	00:17:32	TTTTTTTTTT
Proficient	Higher (University)	Female	1989	7	00:25:13	T_T_TTTTFT	7	00:38:29	TTTFTFTT_T	9	00:19:22	TTTTTTT_T

Table 9: All the results retrieved from the *Feedback* page of the assessment experiment.

Preferred visualisation	**Easier to understand visualisation**	**Easier to handle visualisation**	**Most time spent to understand the...**
third	third	third	questions
second	first	third	data
second	first	third	data
second	third	third	chart
first	first	second	chart
second	second	third	questions
first	first	first	data
first	third	first	data
third	third	third	questions
third	third	third	questions
second	second	third	data
second	second	second	questions
third	third	third	data
third	third	third	data
third	third	third	chart
second	third	third	questions