Automatic Infogram Generation For Online Journalism

Farah Khouzam, Nada Sharaf, Madeleine Saad, Caroline Sabty and Slim Abdennadher
Computer Science and Engineering Department
The German University in Cairo
Cairo, Egypt
{farah.khouzam@gmail.com},
{nada.hamed, madeleine.saad, caroline.samy, slim.abdennadher}@guc.edu.eg

***Abstract*—Infographics is a tool for data visualization. It makes data easy to understand and interpret. An infographic is defined as a visual representation for data like a chart or a diagram. Infographics can help in many fields including education. This is due to the fact that information can be easily memorized and understood if was given in a visual form. Infographics are found everywhere and used in many different fields. Facebook timeline is considered as an infographic. On another hand, online journalism is increasingly gaining popularity. It is also considered as a source of big data that is rapidly expanding. Online newspapers and magazines provide a large population with daily important information. The aim of the work is to use data visualization, infographics and Natural Language Processing (NLP) techniques in online journalism. The aim of the work is to automatically visualize the information in an article in the form of infographics.**

I. Introduction

Visualization aims at having a clear communication for information by using a visual context [3]. This enables the human brain to identify relationships and patterns [3].

Similar to how headlines and photos attract readers, a good and an expressive infographic can also attract a user to read an article rather than just making it clearer to readers.In addition, a business can increase traffic to their website by 12 % if they publish infographics [3].

The aim of the work is to implement a platform that is able to automatically generate and create infograms without having to use an external tool. The article is automatically scanned to extract possible visual representations (not necessarily graphical) for the topics of the article. The application is based on a Natural Language Processing (NLP) engine that is able to extract information. Afterwards, data visualization techniques are applied to visualize the important topics in an article and represent them through infographics.

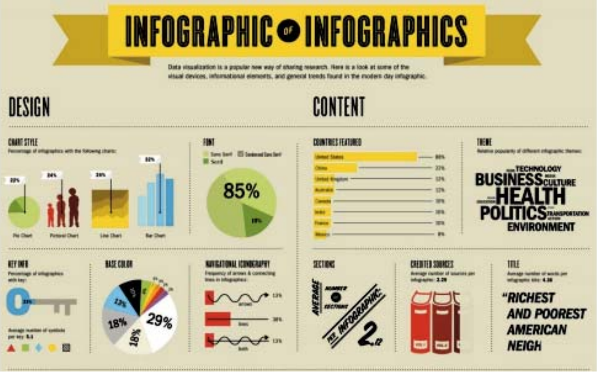


Fig. 1: The attempt to explain what infographics are by example of one of them [4].

Natural language processing (NLP) analyzes texts through computers. It involves knowledge on how human beings understand and use a language [1].

NLP has different applications such as machine translation, automatic summarization, sentiment analysis. NLP technologies are also used to create user-friendly systems for non-expert users [2].

II. Methodology

The proposed methodology is to apply the concepts and techniques of NLP to provide the visualization of articles. Visualization was decided to be done through *chart.js* [1]. However, the aim was to find interesting data

[1] https://www.chartjs.org/

2375-0138/19/$31.00 ©2019 IEEE
DOI 10.1109/IV.2019.00019

that should contribute to the visualization. The first step is to analyze the article. The article is analyzed and filtered using NLP techniques and algorithms to retrieve important information that should be visualized. As a basic step of the analysis, numbers available in an article were taken into consideration as a source of inforgram. Available numbers were used along with the analysis and filtering of the whole text file.

The text was converted to a csv file (comma-separated) to be able to apply tokenization to the words. Tokenizing the file or the comma-separated words is basically removing what is unnecessary form the words in the text file (the article) to start the classification. The unecessary words were removed through different steps. First the punctuation was removed from the text file. Afterwards, the stopwords were removed. The stopwords are words that do not add meaning to a sentence and thus they do not add to the visualization. A list of clean and supposedly useful words are then passed to the next step. This new step uses an algorithm that groups the similar words together.

In addition to the available numbers in the article, the important words/sentences in the article needed to be indentified.

Textrazor API [2] was used along with IBM Watson [3] to provide a clear idea about the concepts, categories and entities. They were able to give accurate information about the topic covered in the article.

Different tools were used for producing the infograms. Chart.js was used in order to draw the filtered information plotting a graph in form of a bar chart and pie chart representing the most two topics covered with corresponding numbers. In addition, Google charts was integrated in the application, in order to visualize data in a different way.

A. The first approach

Different functionalities of Watsons natural language understanding were used. The first was extracting the entities of the article using the producd csv file. Keywords and concepts were also extracted.

Listing 1: Snapshot of extracted entities, categories, keywords and concepts when processing an article about ”earthquakes in Mexico”.

```
entities:
['Organization', 'PoliticalDistrict',
'AdministrativeDivision', '
    GovernmentalJurisdiction',
'OlympicHostCity', 'City', '
    PoliticalDistrict',
'GovernmentalJurisdiction', '
    FilmCinematographer',
'Country', 'Politician', 'Governor',
'AwardPresentingOrganization', '
    AwardWinner',
'StateOrCounty', 'City', 'City']
categories:
[['', 'travel', 'tourist destinations', '
    mexico and
central america'], ['', 'science', '
    geology',
'seismology', 'earthquakes'], ['', 'home
    and garden',
'home furnishings', 'sofas and chairs']]
keywords:
['Mexico\xa0City', 'Mexico City
    authorities', 'people',
'collapsed Enrique Rebs men', 'Mexico
    City Mayor',
'Puebla governor Jos ', 'Pe a Nieto', '
    bare hands',
'Enrique Pe a Nieto', 'education
    undersecretary Javier']
concepts:
['Earthquake', 'Mexico City', 'Mexico', '
    Greater Mexico
City', 'Political divisions of Mexico']
```

In addition, the sentences containing the numbers and not only the numbers were extracted. The engine chooses the numbers/information that is of more value and importance with respect to the context of the article to be graphically represented.

The next step was to relate the sentences containing numbers with the context of the whole article. The text analysis and processing step was directed towards creating classification of the sentences containing numbers and relating the classification of these sentences with the entities available in the whole article.

Sentences containing numbers were compared using Watson to check whether their entities are related. They were also checked against the common entities of the whole article to make sure that the sentence with a number is actually relevant.

Accordingly, the system was able to decide which corresponding sentences were more important and relevant to the context of the article. Unnecessary and unwanted

[2] https://www.textrazor.com/

[3] https://www.ibm.com/watson

numbers are then removed based on the numbers distance from the words related to the main idea of the uploaded file.

The system allows the user (journalist) to edit the shown values. Figures 2, 3 and 4 show the visualized results, the edit option and the new visualization correspondingly.

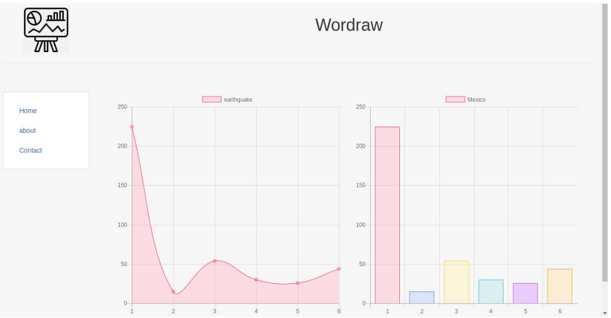
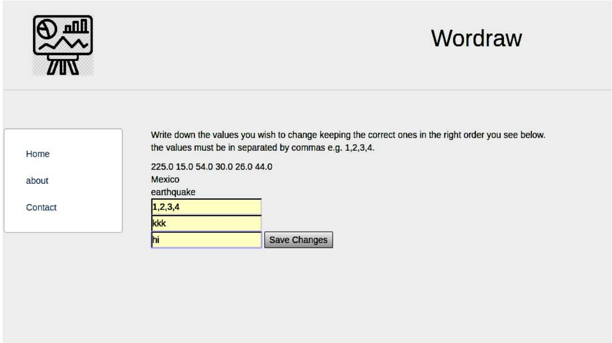


Fig. 2: This is a snapshot of the information that was visualized from the ”earthquakes in Mexico” article. The first chart has as a title the words ”earthquake” which is the most common word spoken of in the article, and we can tell it’s w valid information.The second chart is labeled ”Mexico” as this was the second most spoken of word in the article, which is also a valid information. As for the y values in both charts, they both represent same the numbers extracted from the article, and at last, the x values are the values coming from the enumerate function mentioned above.



Write down the values you wish to change keeping the correct ones in the right order you see below.
the values must be in separated by commas e.g. 1,2,3,4.

225.0 15.0 54.0 30.0 26.0 44.0
Mexico
earthquake
1,2,3,4
kkk
hi

Fig. 3: This is a snapshot of the ”the edit values” html page, where the user can edit the values of the titles or the numbers displayed by following the steps mentioned at the top of the page.

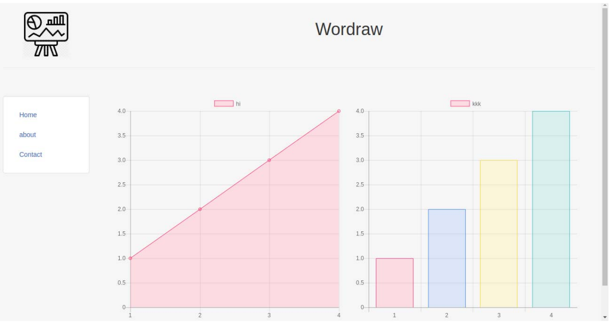


Fig. 4: This is a snapshot of the charts after the writer was done editing the values as shown in the above figure.

B. *The second approach*

Unfortunately, the previous approach did not yield satisfactory results when tried with different articles. The output may have only given the user a general idea about the article through the two extracted words in the article. In addition, having the x-axis coordinates filled with ordered numbers did not provide any useful information. The Chart Gallery[4] offered by Google charts was investigated for the aim of finding a suitable chart type that could a clear idea about the purpose of the data when visualized. At first, the wordtree and the treemap were found easy to use with our data. They both can display the words in a tree in a descending order according to the number of occurrences and the relevance score of each word. Accordingly, common words were matched with the article lines containing numbers. The returned were displayed as important information. Each line was shown in the form of a chat bubble.

The resulting charts were still confusing. Luckily, Google charts offers numerous chart types among which is the piechart with its different shapes and forms. However, the filtering process needed to be updated.

Accordingly, the lines were then separated by the (”.”) as a separator. This required a pre-processing step for decimal numbers. In this step, the dot in any decimal number (matched with a regex expression) was rounded to the nearest integer.

The focus for filtering was also on numbers in the article. In this approach different types of numbers are classified. If a certain number represents a year, it was appended to a list ”years” which will be returned as an output of the step. Along with the extracting the years, the the lines

[4] https://developers.google.com/chart/interactive/docs/gallery

containing these years were also found to be interesting. The final of this step were two list of numbers along with their sentences (in case of relevance to the article). One of the lists represented years and the other represented other numbers.

In this approach, an article about world population was taken as the evaluation sample for all the functions. Figure 5 shows the output of visualizing the years' information.

Accordingly, Google charts time line was chosen to display these two lists.

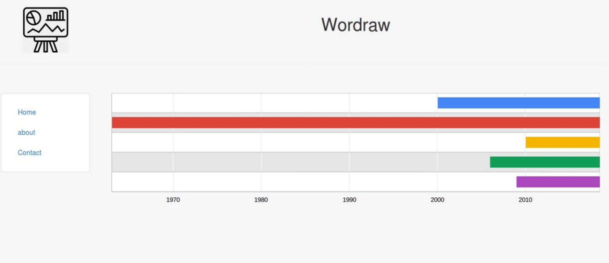
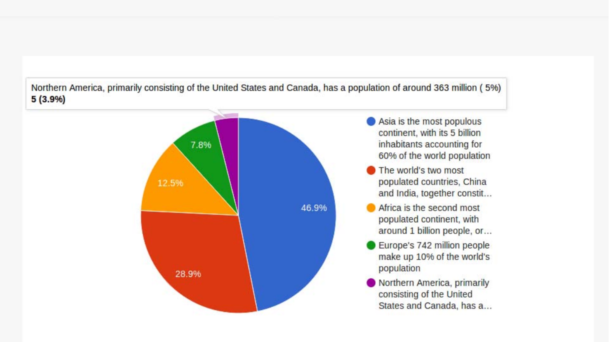


Fig. 5: This is a snapshot of time line of the world population article. Each color represents a different year and when hovering on the bar, the corresponding line pops up.

The second type of interesting numbers were percentages. Similar to years, the engine was able to extract percentages by looking for any number followed by the percentage sign or the word ”percent”. The output of this step builds on the previous step producing two lists for percentages and the rest of the numbers. The result of visualizing data with percentages is shown in Figure 6.

After implementing the function, Google pie chart was fond suitable to display the data resulting from the ”is-percent” function. Using the same drawchart() with different options, the two lists were passed to the drawchart function. The function was tested and the results were accurate and credible to the user.

TextRazor functions were used to filter the output lines containing numbers (whether the numbers are years, percentages or another type). The TextRazor functions used were the entities extraction, the concepts extraction, the categories extraction and the keywords extraction. The sequence was as follows: a list having as element the percentage sentences or the years sentences, was passed to any of these functions. Then, the concepts, categories, entities and keywords were retrieved from every element in the list. All retrieved words from all elements in the list were then compared. The elements having similar words were given as an output. This was ensured that the sentences present in the time line, the piechart or even the bubbles showing important information, were related.



Northern America, primarily consisting of the United States and Canada, has a population of around 363 million (5%)
5 (3.9%)

● Asia is the most populous continent, with its 5 billion inhabitants accounting for 60% of the world population
● The world's two most populated countries, China and India, together constit...
● Africa is the second most populated continent, with around 1 billion people, or...
● Europe's 742 million people make up 10% of the world's population
● Northern America, primarily consisting of the United States and Canada, has a...

Fig. 6: The piechart shows info on the article of world population shown above, google charts sums up the percentages given to the function that draws the charts and divides 100 by their sum in order to have a factor which is multiplied by the percent values and the output is displayed on the piechart as shown below, in order that all percent values in the piechart sum up to 100. This explains the difference between the percentages shown on the piechart and the ones on the right.

Furthermore, TextRazor entities extraction function was used for the purpose of improving the pie chart displayed. In more details, there was another function implemented ”entitylines”. This function was given the list of line containing the calculated important percentages as input. The main task of this function was to extract the entities of each sentence and its relevance score. The enitities with the highest relevant scores were chosen as labels for the pie chart as shown in Figure 7. This was considered as an improved verion of the piechart depending on how related are the lines containing percentages are.

Listing 2: This is a snapshot of the entities extracted by TextRazor and their relevance score. The entities with the highest relevance scores are also shown.

```
[[(’World population’, 0.1997), (’Asia
    ’, 0.1184),
(’Continent’, 0.1774)], [(’China’,
    0.05523), (’India’,
0.07155)], [(’Continent’, 0.07772), (’
    Africa’, 0.1351)],
[(’Europe’, 0)], [(’United States’,
    0.09353), (’United
States’, 0.05171), (’Northern America
    ’, 0.04434),
(’Canada’, 0.06538)]]

[[’World population’, ’Asia’, ’
    Continent’], [’China’,
’India’], [’Continent’, ’Africa’], [’
    Europe’], [’United
States’, ’United States’, ’Northern
    America’, ’Canada’]]

[[0.1997, 0.1184, 0.1774], [0.05523,
    0.07155], [0.07772,
0.1351], [0], [0.09353, 0.05171,
    0.04434, 0.06538]]

[’World population’, ’India’, ’Africa
    ’, ’Europe’, ’United
States’]
```

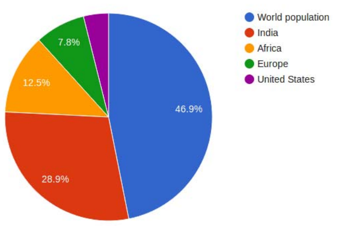
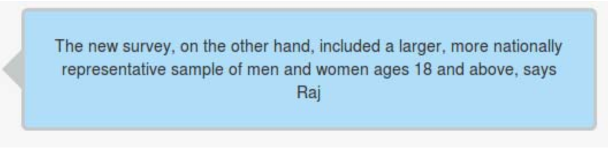


Fig. 7: This is a snapshot of the ”piechartwords”. It is another version of the piechart where the words replace the corresponding lines.

III. Conclusions and Future Work

Online journalism is becoming increasingly important and it provides all users the feasibility to read anywhere at any time. The research question was whether the generation og infographics for online journalism could be automated. A data visualization tool was build in the attempt of improving and simplifying the way an online reader or writer knows an information. Natural Language Processing (NLP) techniques were used with the help of integrated APIs to retrieve the important information from an article given by the user. Data visualization was used to put the information retrieved in a visual context, more specifically infographics. The analysis of different samples of articles using the tool showed that it is considered reliable with respect to meeting the definedconstraints. The reliability and the precision of the tool can be improved by increasing the text classification and the information retrieval functions used. The work shows that a fully automated tool needs more work in order to produce useful results. The solution to that was to introduce specific constraints on the types of numbers that are of interest. This could be generalized such that a journalist can feed into the system the type of numbers that they need to extract through a regular expression/format editor. Figure 8 shows some data that out approach would miss since the number does not fall into the defined categories. This emphasizes the need of more work into automating and relating the sentences extraction process.



The new survey, on the other hand, included a larger, more nationally representative sample of men and women ages 18 and above, says Raj

Fig. 8: This is the important information containing numbers that could not fit in any other visual form

References

[1] Vraj Shah Aditya Jain, Gandhar Kulkarni. Natural Language Processing. *International Journal of Computer Sciences and Engineering*, 6, 2018.

[2] Sethunya R Joseph, Hlomani Hlomani, Keletso Letsholo, Freeson Kaniwa, and Kutlwano Sedimo. Natural language processing: A review. *International Journal of Research in Engineering and Applied Sciences*, 6, 2013.

[3] Waralak V Siricharoen. Infographics: the new communication tools in digital age. In *The international conference on e-technologies and business on the web (ebw2013)*, pages 169–174. The Society of Digital Information and Wireless Communication, 2013.

[4] Mateusz Szotysik. Processes of creating infographics for data visualization. 2016.