Information Visualization for CSV Open Data Files Structure Analysis

Paulo Carvalho[1,2], Patrik Hitzelberger[1], Benoît Otjacques[1], Fatma Bouali[2] and Gilles Venturini[2]
[1]*Gabriel Lippmann Public Research Center, 41 Rue du Brill, L-4422 Belvaux, Luxembourg*
[2]*University François Rabelais of Tours, Tours, France*

Keywords: CSV, Information Visualization, InfoVis, Open Data, Table structure, Tabular Information Visualization.

Abstract: New and different information sources have appeared over the past years (e.g. Blogs, Media, Open Data, Scientific Data and Social Networks). The variety of these sources is growing and the related data volume increases exponentially. Open Data (OD) initiatives and platforms are one of the current major data producers, also because the topic seems to be important for many governments world-wide. Given the many fields and sectors involved, OD brings high business and societal potential. The amount and diversity of available information is high. However, analysing and understanding OD in order to exploit is far from being an easy task. Several problems and constraints must be solved. Information Visualization (InfoVis) can help to give a graphical idea of the processed files structure. Given that OD is provided very often as tabular data, this paper focuses on OD CSV files. It presents an overview on the analysis of tabular information. Finally, the paper describes the role of Information Visualization and the way it may help the end-user to understand quickly the structure and issues of OD CSV files.

1 INTRODUCTION

The amount of information being generated and released every day on the Internet is enormous. This fact results mainly from the growing number of new information sources: Open Data (OD), Social Networks and Media (e.g. Twitter, Facebook), blogs, scientific data, commercial data, and so on. OD is a current active movement in terms of data generation and release. This is mainly due to the pressure made by governments in order that data produced and maintained by public entities should be accessible to the global public. The economic value of OD has been estimated at 40 billion, per year, in Europe alone (EuropeanCommission, 2014). At the same time, OD represents an important element of the way towards more IT supported participation and transparency (Janssen, 2011) in the public sector. Many actors of both the private and the public sector are involved in OD publication: agriculture, economy, health, culture, transports, education, etc. This diversity is one of the factors that explain why OD has a high value and an important potential for the society and the economy in general. Every data, and in particular, every OD dataset has its own business/social value. To realize the maximum potential of datasets, however, it may be necessary to combine them. We have been exploring this area in order to create adequate charts to support OD integration processes. However, before combining data, it is necessary to understand it. One of the major problems concerning OD is the lack of a common and unique standard used by organizations to publish their datasets. This leads to datasets being published and accessible in different formats and shapes: files (in different formats), RSS, APIs, etc. Understanding how datasets are organized is a major issue considering all kinds of dataset formats that may exist. This problem is known as the ”table recognition problem” (Ng et al., 1999). It consists in identifying, in a given data file, the important parts of data tables such as headers (columns and rows), data values and their types (string, numeric, etc), missing values and other information in the file. Some files might contain more than one table. Once the table structure is understood, the table can be further processed and combined with other information: it can be integrated in a data warehouse, or it can be processed by a data mining or a visualization tool. This paper focuses on OD CSV files. The reason of this choice is explained in a later section. We support the idea that Information Visualization is a promising candidate to ease the understanding of OD CSV files structure. The paper tries to answer the challenge of presenting an intuitive and efficient manner to visualize the structure of such datasets. The rest of the paper is structured as follows: First, we give a brief introduction of table structure

Carvalho P., Hitzelberger P., Otjacques B., Bouali F. and Venturini G..
Information Visualization for CSV Open Data Files Structure Analysis.
DOI: 10.5220/0005265301010108
In *Proceedings of the 6th International Conference on Information Visualization Theory and Applications* (IVAPP-2015), pages 101-108
ISBN: 978-989-758-088-8

analysis, followed by an overview on the state of the art in this field. Several chart types we have worked on during our analysis are presented and their advantages and limitations are discussed. Our chosen chart solution - *Piled Chart* - is described in detail. Finally, we present the results and conclusions of using Information Visualization to support CSV files structure analysis.

2 TABLE STRUCTURE ANALYSIS

In 2011, OD implementation across Europe was still emerging. Only a small percentage of datasets was published under open (1.8%) and machine-readable (11.8%) formats (Reggi, 2011). PDF was the most spread format. The only open format used was CSV. In the next section we discuss how things have evolved and our choice of focusing our work in the CSV format.

2.1 CSV Format

Several initiatives encourage the use of machine-readable formats for OD. For the prevailing tabular datasets, the simplest and widest-spread format is CSV (comma-separated values). Its usage in OD contexts is commonplace. The Open Government Partnership (OGP) was launched in 2011 (Harrison et al., 2012) to stimulate the creation of open governments worldwide. Currently, OGP has 64 participating countries (OpenGovernmentPartnership, 2014). Members of this organization are required to respect various open-governance standards (Yu and Robinson, 2012), concerning e.g. the type of information released or the format of the published data. In general, datasets are required to be released in a machine-readable format. In Europe, the Financial Transparency System (FTS) of the European Commission provides information related with European Union projects since the year of 2007 (Martin et al., 2014). All the datasets can be downloaded as CSV files. In the Netherlands, Zuiderwijk and Janssen made a study of the OD policy of seven countries (Zuiderwijk and Janssen, 2014). It shows that standard formats, including CSV, are used most of the time. In 2014, Veljkovi et al. presented a benchmark proposal regarding OD available in the United States OD portal (Data.Gov, 2012). In this study, it has been concluded that most of the OD datasets were available in CSV, XLS and PDF files (Veljković et al., 2014). Finally, a recent study of OD policies in five different countries (United States, United Kingdom, Netherlands, Kenya and Indonesia) has confirmed that CSV is used in all involved countries except Indonesia, where datasets are only available as PDF files (Nugroho, 2013). All these facts demonstrate the importance of the CSV format for OD, and why we focus our research on CSV files. This choice is also linked to the bellow reasons:

- CSV is a machine-readable format;
- CSV is a basic, non-proprietary format;
- CSV is a simple tabular format.

However, the advantage of the extreme simplicity of the format is not without issues. The semantic and syntactic interpretation of CSV files can be difficult. Furthermore, there is no common standard used for publishing OD (Rivero et al., 2012). In order to profit from the presumptive high potential business-value of OD, data must be made usable, meaningful and exploitable. Getting an overview of the structure and the content of a CSV file can be a hard task. Despite its popularity, this format does not have a formal specification. The informal de-facto standard is RFC 4180 (Shafranovich, 2005) that specifies the following syntatic rules for CSV files:

- Records are located on a separated line delimited by a line break;
- Last record of the file may not have an ending line break;
- An optional header line may exist on the first line of the file;
- Within the header and each record, there may be one or more fields separated by commas;
- Each field may or not be enclosed in double quotes;
- Fields containing line breaks, double quotes and commas should be enclosed in double-quotes;
- In fields where double-quotes are used to enclose them, a double-quote must precede another one appearing inside a field to escape it.

2.2 Visualizing the Structure of a Table

Despite its importance and barriers faced, the visualization of the structure of tabular data does not seem to be a subject of deep study during the last years. Many visualization techniques use tabular data as an input. They consider the structure of the table as clearly identified. Among the table visualization methods (Hoffman and Grinstein, 1997), some approaches can directly represent the structure of a table, like Table Lens (Rao and Card, 1994) and Tableplot Graphics (Malik et al., 2010).
Table Lens is a technique to visualize and understand

the meaning of large tables using a *fisheye* approach. The idea behind this methodology is based on a distortion applied by the user where the centre of the distortion becomes zoomed-in while the other regions displayed are zoomed-out (Sundararajan et al., 2011). Table Lens is more appropriate for the analysis of precise and small regions of a table. It can be used to find correlations existing in a given context. On its side, Tableplot Graphics is a method to represent graphycally the cell values of a tabular dataset. It does not analyse and show the type of data analysed. Other related work on this subject has been found: Sopan at al. Exploring Distributions - Design and Evaluation (Sopan et al., 2010). However, data types were also not taken into account.

Although its global utility, these methods are not convenient regarding the type of CSV structure analysis we want to achieve. We support the idea that it is necessary to create a more specific solution to assess the structure of a CSV file. Our objective is to provide the user with an intuitive and global overview of the structure of a CSV file and the type of data it contains, in order to determine what pre-processing must be performed on such a table. We also want to furnish a tool that is able to identify possible errors in CSV files in order to correct them if possible. This is why we have created the concept of a tabular *Piled-Chart*. This chart will be described further section in this document.

3 PILED CHART TECHNIQUE

Processing, analysing and understand large amounts of information is a complex task. Creating and developing a new technique of information visualization that is able to provide better results than existing methods and respecting the properties presented above (Section 1.1) is a hard job. With all the different forms, graphs and shapes already existing to represent information (e.g.: Pie charts, Ellimaps - use nested ellipses of various sizes to build graphics (Otjacques et al., 2009), Treemaps, Geographical Treemaps, etc.), being innovative and bringing better results becomes more and more challenging. Table Lens is a technique that is used to analyse tabular information. However, it is based on a *fisheye* approach which is not appropriate to give a view of the entire data. Our aim is to create a visualization technique that is able to give a global overview of an OD CSV file structure:

- To estimate the size (number of rows/columns) of the CSV file;
- To provide information on the data type of CSV file cells;
- The possibility to compare the same type of datasets over time periods. This can be very useful to detect problems, errors and incoherencies because OD datasets are often published periodically.

Additionally, the new chart should have the following characteristics:

- Be efficient - in order to deliver a rapid idea of the CSV file form. Users with limited or even poor knowledge about the analysed information should be capable of getting a global idea of the CSV file structure in an effective and intuitive manner;
- Be user-friendly provinding e.g. zoom-in/zoom-out functions; legends; tooltip information; etc.

With the focus on the objectives presented, respecting constraints listed above and taking into account the peculiar nature of tables (their columns/rows structure; their size which may vary from small to large; the cells may contain diverse data types; etc.), the *Piled Chart* technique has been created. Before creating *Piled Chart*, we focused our studies on several solutions which fulfilled our needs partly, and were important for the development of *Piled Chart* rise. This is why they will be presented briefly.

3.1 Previous Attempts

Before arriving to a solution that is able to present graphically tabular data, we have worked on several solutions (e.g. the *Sphere Chart*; the *Circle Chart*). They acted as a starting point of our research but were not retained because of several weaknesses:

- They were not able to present the structure of more than one file simultaneously - the user can only visualize the structure of a region of one CSV file;
- Some limitations regarding their understanding and perception;
- The user did not have the view of the entire structure of the file. Mouse interaction was needed in order to view other parts of the structure.

The *Circle Chart* (Figure 1) is a chart that allows to represent the structure of a CSV file. Its main drawback is its unreadability when the number of columns and rows of the analysed file is considerable.

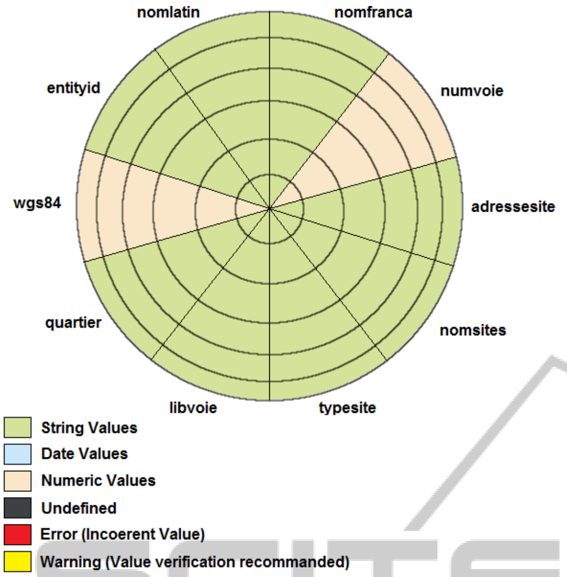


Figure 1: Circle Chart example.

Figure 2 shows the *Sphere Chart*. The aim of the *Sphere Chart* is to show the structure of a CSV file giving the ability to view its cells content, detect missing values and potential errors. However, it has an important limitation: the user needs to rotate the sphere in order to view parts of the file structure. It does not give an entire image of the complete file structure.

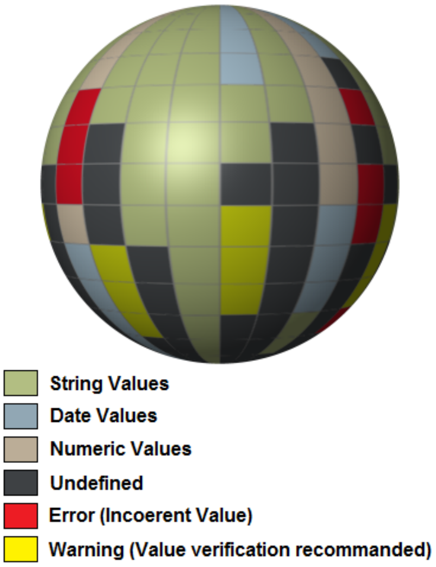


Figure 2: Sphere Chart example.

3.2 CSV Cell Chart

Succeeding previous chart solutions, *CSV Cell Chart* was created (Figure 3). *CSV Cell Chart* was a step forward to our solution - *Piled Chart* is completely based on this approach. *CSV Cell Chart* consists merely in showing every column, row and cell in the same order as they are in the CSV file. In other words, it gives to the user an image of the CSV file structure, using a colour code to represent the data type of each cell. This colour code is also used to show possible warnings/errors.

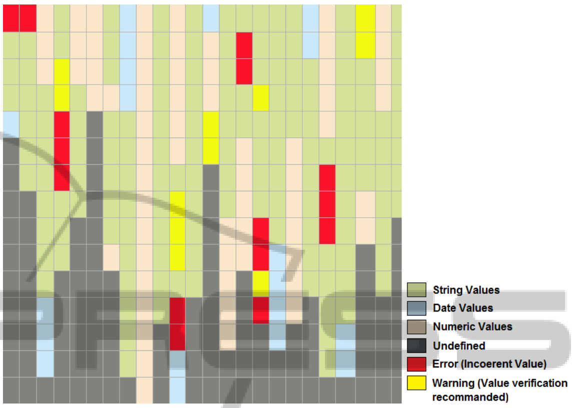


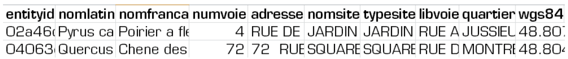
Figure 3: CSV Cell Chart example.

3.3 Piled Chart

CSV Cell Chart has an important weakness: it displays all rows/columns, even if they have the same structure. This fact is not optimal in terms of space usage and of the visualization. To eliminate this inconvenience, retaining however the advantages of the *CSV Cell Chart*, the *Piled Chart* was created. Its main characteristics are:

- Rows with the same structure - rows with the same data types on the same column index - are grouped (piled) into an unique row;
- Columns with the same structure - columns with the same data types on the same row index - are grouped (piled) into an unique column;
- A colour code is used to represent the data type for every cell. For now the data types supported are limited (String, Number, Date and Percentage). The colour code may also be used to represent possible errors and/or warnings.

With this approach, the structure of a CSV file can be represented on a reduced area - because rows and columns with a similar formation are grouped. The figure below (Figure 4) shows a part of a OD CSV file which contains information about the trees in the parks of the Versailles city (Île-de France, 2014).



entityid	nomlatin	nomfranca	numvoie	adresse	nomsite	typesite	libvoie	quartier	wgs84
02a46	Pyrus ca	Poirier a fle	4	RUE DE	JARDIN	JARDIN	RUE A	JUSSIEU	48.80
04063	Quercus	Chene des	72	72 RUE	SQUARE	SQUARE	RUE D	MONTRE	48.80

Figure 4: OD CSV file example - Trees existing on the parks of Versailles.

It is a small and simple CSV file with a structure of 10 columns and 3 rows. Using a *Piled Chart* to represent the file structure will help the user to determine easily the types of the cells present on the file. The advantage of using *Piled Chart* grows with the complexity of the analysed CSV file.

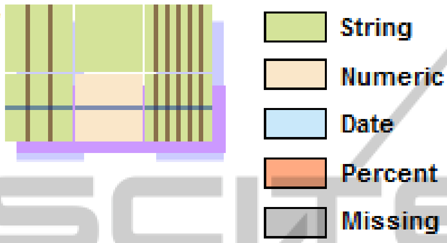


Figure 5: Piled Chart example.

The chart (Figure 5) is composed of three columns (two piled-columns + one ”normal” column) and two rows (one normal row + one piled-row). By a quick look at the chart, with a special focus on the piled-rows and piled-columns, it is possible to extract following information:

- The cells are all composed by Strings or Numbers (based on the colour code presented);
- The first row has only String values in every cell;
- All the others rows (2 rows) have the same structure;
- The file is composed - from left to right - of 3 columns with the same structure (cells with a String value), followed by 1 column and finally ends with a set of 6 columns with the same structure (cells with a String value).

The number of columns covered by a piled-column may be determined by counting the regions defined by the brown lines. At the same time, the number of rows covered by a piled-row may be determined by counting the regions defined by the dark-blue lines. It is also possible to obtain this information by just moving the mouse over the piled-region behind the piled-rows and/or the piled-columns (e.g. Figure 6).

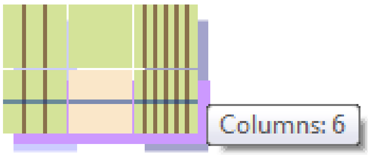


Figure 6: Tooltip example.

An additional characteristic of the *Piled Chart* is the possibility to expand a piled-column and/or a piled-row. The user has the possibility to click on the piled-area behind a piled-row/piled-column in order to expand it. This action will turn visible the entire structure of the selected piled-region. The following picture (Figure 7) shows the result of expanding the first piled-column of the graph.

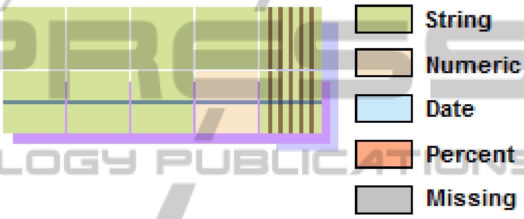


Figure 7: Piled-column expand.

It is possible to visualize the content of a given cell. Obviously, this functionality is not available for piled-columns and piled-rows. To do so, the user just has to move the mouse pointer over the wanted non-piled-cell and a tooltip with its value appears (e.g. Figure 8).

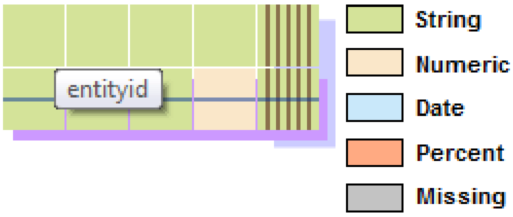
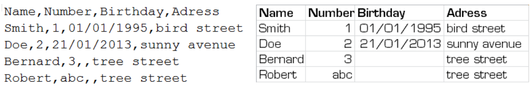


Figure 8: Cell value.

Finally, another useful functionality of the Piled Chart is its ability to detect and show missing values on a CSV file to the user. Every missing value is represented with a specific colour in the example, the grey colour is used. This scenario is demonstrated in the figures 9 and 10.



```
Name,Number,Birthday,Adress
Smith,1,01/01/1995,bird street
Doe,2,21/01/2013,sunny avenue
Bernard,3,,tree street
Robert,abc,,tree street
```

Name	Number	Birthday	Adress
Smith	1	01/01/1995	bird street
Doe	2	21/01/2013	sunny avenue
Bernard	3		tree street
Robert	abc		tree street

Figure 9: CSV file plain text format VS CSV file in Excel form.

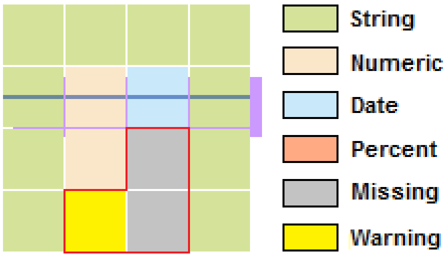


Figure 10: CSV file with missing value and possible error.

The difficulty which may occur when viewing a CSV file grows with the size of the CSV files and the complexity of its structure (missing values, many rows, many columns, irregular number of rows/columns, etc.). Looking at an image like the one in Figure 10, the user is able to conclude quickly that the analysed CSV file may have a problem in the cell located on its last row of the 2nd column because of the yellow colour. Looking at Figure 9, this cell has the value ”abc”, while all the others cells in this column are numbers (except the header). In certain cases, missing values can have an important impact on the process, so it may be crucial to detect them. Using the same kind of reasoning, the user is able to conclude that two values are missing on the 3rd column of the analysed CSV file. This effectiveness of usage is mandatory for the success of every visualization solution. The more complex the structure and the larger the size of a CSV file are, the more a *Piled Chart* is valuable. The figure below (Figure 11) shows the *Piled Chart* representation of a CSV file with 52 rows and 33 columns. Because many rows and columns have the same structure, the entire file structure is shown using two rows and nine columns reducing the area of analysis needed by the user to understand the file structure. The *Piled Chart* area size is reduced along as similar rows/columns exists.

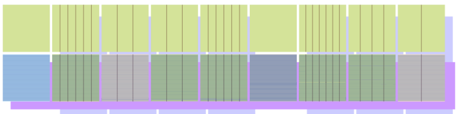


Figure 11: Understanding CSV files structure reducing user’s area of analysis.

The Figure 12 shows another example of the *Piled Chart* for a large CSV file, with 17 columns and 204 rows. This file gathers information related with the parks and the gardens of the city of Montpellier.

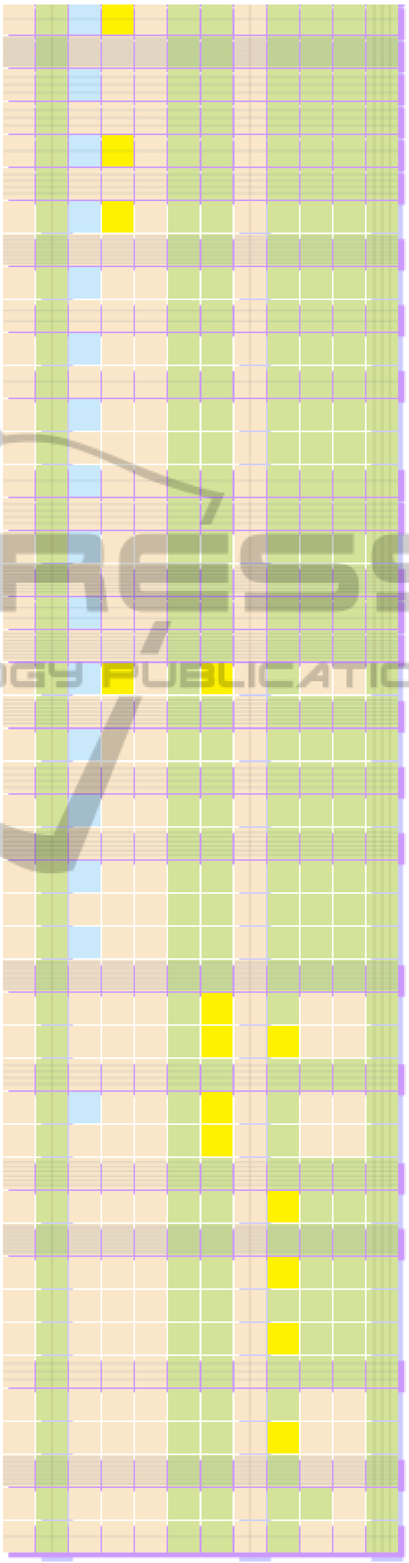


Figure 12: Understanding large CSV files structure reducing user’s area of analysis.

The following table summarizes the features of *Piled Chart*:

Table 1: Piled Chart features.

+	Global view of a CSV file's structure using the smallest area possible.
+	Possibility to visualize each cell type and each cell value.
+	Missing values detection.
+	User-friendly: zoom-in/zoom-out function; tooltips with information; etc.
-	Not able to analyse 2 different files simultaneously.
-	Limited to CSV files analysis.

4 CONCLUSIONS AND FURTHER WORK

In this paper we have presented the advantages of creating a new type of an information visualization chart that is able to describe the structure of CSV files. CSV files are not always easy to understand. Their structure may be very complex, with large number of columns and/or rows, non-regular number of columns in each row, different types of data in each cell and so on. With the growing OD trend, large amount of information is published over the Internet. Abundant variety and quantity of public data is now available for different usage scenarios. Many OD datasets are published in the CSV format. If the user cannot interpret efficiently OD CSV files, the potential of these datasets cannot be exploited and they may become useless. By providing better tools to the user that ease the understanding and exploitation of such data, we increase the potential use of these datasets. Beyond the possibility the user has to analyse more efficiently the files, potential errors can be detected - and corrected. These factors make the use of OD CSV files more convenient and efficient. According to the literature, it seems that there are not many techniques to analyse tabular information. Table Lens is one of them but has some weaknesses that were presented in this paper. That is the main reason why we have worked on a new type of information visualization technique for OD CSV files: *Piled Chart*. In our opinion, *Piled Chart* is promising, but still has potential for improvement. One current limitation of the *Piled Chart* is its incapability to inspect more than one file simultaneously. Many of OD datasets available on the web are published periodically. The possibility to evaluate several files at the same time would bring important advantages to compare rapidly the same kind of CSV files over periods of time (e.g. datasets with the data of the public budget for two different months). This would help to detect inconsistencies between file generations. Another challenging task is the scalability problem: can the technique cope with very large files? How will the system work when processing several large datasets? Finally, but not less important, the technique should be as user-friendly as possible. The user should be able to understand easily the information showed in the chart and the interaction must be easy and fluid. We are already taking this into account but there is still work ahead.

REFERENCES

Data.Gov (2012). The home of the u.s. governments open data. https://www.data.gov/. Last accessed on September 22, 2014.

EuropeanCommission (2014). Digital agenda for europe - a europe 2020 initiative - open data. http://ec.europa.eu/digital-agenda/public-sector-information-raw-data-new-services-and-products. Last accessed on September 16, 2014.

Harrison, T. M., Pardo, T. A., and Cook, M. (2012). Creating open government ecosystems: A research and development agenda. *Future Internet*, 4(4):900–928.

Hoffman, P. and Grinstein, G. (1997). Visualizations for high dimensional data mining-table visualizations.

Île-de France, R. (2014). Arbres dans les parcs de la ville de versailles. http://www.data.gouv.fr/en/dataset/arbres-dans-les-parcs-de-la-ville-de-versailles-idf. Last accessed on September 16, 2014.

Janssen, K. (2011). The influence of the psi directive on open government data: An overview of recent developments. *Government Information Quarterly*, 28(4):446–456.

Malik, W. A., Unwin, A., and Gribov, A. (2010). An interactive graphical system for visualizing data quality–tableplot graphics. In *Classification as a Tool for Research*, pages 331–339. Springer.

Martin, M., Stadler, C., Frischmuth, P., and Lehmann, J. (2014). Increasing the financial transparency of european commission project funding. *Semantic Web*, 5(2):157–164.

Ng, H. T., Lim, C. Y., and Koo, J. L. T. (1999). Learning to recognize tables in free text. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 443–450. Association for Computational Linguistics.

Nugroho, R. P. (2013). A comparison of open data policies in different countries.

OpenGovernmentPartnership (2014). Open government partnership. http://www.opengovpartnership.org/. Last accessed on September 22, 2014.

Otjacques, B., Cornil, M., and Feltz, F. (2009). Using ellimaps to visualize business data in a local administration. In *Information Visualisation, 2009 13th International Conference*, pages 235–240. IEEE.

Rao, R. and Card, S. K. (1994). The table lens: merging graphical and symbolic representations in an interactive focus+ context visualization for tabular information. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 318–322. ACM.

Reggi, L. (2011). Benchmarking open data availability across europe: The case of eu structural funds. *European Journal of ePractice*, 12.

Rivero, C. R., Schultz, A., Bizer, C., and Ruiz, D. (2012). Benchmarking the performance of linked data translation systems. In *LDOW*.

Shafranovich, Y. (2005). Common format and mime type for comma-separated values (csv) files.

Sopan, A., Freire, M., Taieb-Maimon, M., Golbeck, J., Shneiderman, B., and Shneiderman, B. (2010). Exploring distributions: design and evaluation. *University of Maryland, Human-Computer Interaction Lab Tech Report HCIL-2010-01*.

Sundararajan, P. K., Mengshoel, O. J., and Selker, T. (2011). Multi-fisheye for interactive visualization of large graphs. In *Scalable Integration of Analytics and Visualization*.

Veljković, N., Bogdanović-Dinić, S., and Stoimenov, L. (2014). Benchmarking open government: An open data perspective. *Government Information Quarterly*, 31(2):278–290.

Yu, H. and Robinson, D. (2012). The new ambiguity of'open government'. *Princeton CITP/Yale ISP Working Paper*.

Zuiderwijk, A. and Janssen, M. (2014). Open data policies, their implementation and impact: A framework for comparison. *Government Information Quarterly*, 31(1):17–29.