Processing incoherent open government data: A case-study about Romanian public contracts funded by European Union

Bogdan Ghita[1] , Octavian Rinciog[1], Vlad Posea[1]

[1] Politehnica University of Bucharest, Computer Science Department
bogdan.ghita93@gmail.com,
{octavian.rinciog,vlad.posea}@cs.pub.ro

Abstract. Lately, many governments have adopted policies and mechanisms for making open data available to citizens, in order to increase the transparency of state administration and institutions. The usage of these data is hampered by the incorrect, incomplete and incoherent nature of the information.

The purpose of this paper is to summarize the general steps that are needed in order to transform raw open data that contain errors to consistent data. These steps are used to correct the open data published by the Romanian government regarding public contracts funded by European Union, supporting entities interested in using these data.
Keywords: Open Governemt Data, Error Correction, Error Detection

1 Introduction

In recent years the popularity of open data has increased worldwide. Many governments have adopted policies and mechanisms for publishing open data to citizens, in order to increase the transparency of their public administration and institutions. The open data publishing and usage benefits include the increase of citizens' confidence in government administrations, as well as corruption decrease.

Despite the states' efforts to provide access to open data, their publishing is not sufficient to achieve the expected benefits. The data usage is burdened by the incorrect, incomplete and unstructured nature of the information.

This paper analyzes error types that may appear in published open government data and provides techniques to correct them. The goal of this paper is to provide structured and clean data for citizens' usage. These techniques of error correction are applied to curate the data released by the Romanian government about contracts financed by European Union. At the end of the paper, we show how the cleaned data are used for analyzing these type of contracts.

2 Open Government Data

A lot of papers have analyzed both benefits and disadvantages of open data[1][3][5][8]. Their authors studied how the usage of this type of information helps both economic growth and confidence in public administration and minimizes corruption in public institutions, increasing transparency [2][7].

One aspect that attracts a lot of attention worldwide is finding solutions that can minimize corruption, as it is a prominent issue in multiple countries. Making data available is one fact that may facilitate the discovery of corruption evidences. An example of corruption is the so called "conflicts of interest" through which a person within the administration can use its influence and position for its own benefit. In these situations, data availability about public contracts may assist civil society or journalistic efforts to discover this type of illegalities.

Although the theory indicates many benefits of open data, often the publishing process can be difficult, and the users may face obstacles in gaining information. Often, public administrations believe that simply publishing data will bring you straight benefits. This is not entirely true, because without data standardization and proper description of the information, open data cannot be used. In many cases, published data can become useless, without proper help from qualified personnel, who know what the data are about. [6]

In [4], Futia et all discussed about using errors occurred in Italian procurement documents and they developed a semantic framework to overcome the error that they discovered.

The above-mentioned problems indicate that the first step in processing the published data is to standardize them by correcting the various errors that may occur and to describe them in a consistent manner.

3 Romanian public contracts funded by European funds

In Romania, lots of public projects are financed by European Union. For a project to obtain funding from this institution, the potential recipient must first register a project plan to European Commission and must meet the minimum eligibility requirements for participation. The Commission verifies that the proposal is eligible and admissible, then the project is evaluated by a group of specialists.

After signing the grant agreement, the beneficiary can start implementing the project using its own money, guided by a financing contract and after a period, may submit applications for reimbursement of the spent money. Data about Romanian projects financed by European Union, containing information on applications for funding and reimbursement projects between 2009 and 2016 are published by the Romanian Government on the site dedicated to open data.

3.1 Analyzed dataset

The analyzed dataset consists of 59 documents found on the Romanian open data portal site[1,2,3]. The documents are in XLS, CSV, XML and ODT formats, totaling almost 1GB of information, and can be divided into 3 categories based on the information that they contain.

a) Financing contracts - 27 files (26 XLS, 1 CSV), 316.7 MB
   Each row of each file contains information about one project, such as: project unique code (PUC), financing contract number, date, customer, region, county, city, total budget, eligible budget
b) Reimbursement of financing contracts - 22 files (21 XLS, 1 CSV), 543.7 MB
   Each row of each file contains information about a refund, such as: project unique code (PUC), customer, outsourcer (CompanyID), region, county, city, application's date, required amount of money (RequestedM), reimbursement authorization number (ReimbursementNum), reimbursement date and reimbursement authorized amount of money (AuthorizedM). There is a relation between the two money amounts (RequestedM >= AuthorizedM), because some payments can be not eligible for reimbursement.
c) Nomenclatures - 10 files (5 XLS, 3 XML, 2ODT), 426 KB
   Each document is a list of all possible values for a particular type of information that can be found in a dataset about cities, counties or customer information.

The following entities can be identified in a dataset: public financed project (uniquely identified by PUC), reimbursement request, (uniquely identified by the combination between PUC and ReimbursementNum), customer (does not have an unique identifier), outsourcer (uniquely identified: CompanyID), region, county, locality (identified by unique codes, which are found in the nomenclatures).

Relations between these entities are the following:

- A customer is associated a region, a county and a locality, and may have several different financing contracts for different projects.
- For a financing contract, there is one customer and several reimbursement requests, each of them corresponding to one implementation step of the project.
- Each reimbursement request is made for a particular financing contract and has one outsourcer - the entity that implemented this part of the project.

4 Data Quality Errors

In the process of automatic document processing, possible errors or inconsistencies in data representation must be considered. After analyzing the above mentioned docu-

[1] http://data.gov.ro/dataset/transparentzare_smis
[2] http://data.gov.ro/dataset/informati-derulare-fonduri-europenesmis
[3] http://data.gov.ro/dataset/informati-smis-csnr

ments, several types of errors appeared along with problems that may arise when processing data.

In order to achieve the goal of obtaining clear and consistent data, all these cases should be investigated. Depending on their type, errors may be mitigated at different stages of processing. This involves error detection and correction, which if not possible, the affected data must be invalidated.

Next are detailed all types of errors and in the data processing section, more possibilities for detecting and mitigating them will be analyzed.

4.1 Document titles

Information needed for automatic data processing is the type and date of each document content (type is useful for content interpretation and date for time correlation). The analyzed files have this piece of information in their name, which does not respect a fixed format. That's why auto extraction of the type and date of the document cannot be 100% accurate, without any heuristics. For example, 3 documents about financing contracts are named like:

- financingcontractsdec2015.xls (Contains financing contracts from December 2015)
- financing contracts-dec2014.xls (Contains financing contracts from December 2014)
- financing contracts-05.08.20161.xls (Contains financing contracts from July 2016, published in 5th August 2016)

4.2 Missing headers

All the analyzed documents have tabular structure, but some of them do not contain the headers of the table, by lacking the first row in which the column name should be specified. The effect of this missing information is that it cannot determine what type of information is contained in each column. Another problem that comes with automatic file processing is determining whether or not a document has a header.

4.3 Inconsistency of headers

The names of the columns varies from one document to another, even if its type is the same. This can lead to problems in the standardization and aggregation processes. Inconsistencies are of two types : writing errors and alternate names (PUC (project_unique_code) vs UCP(unique_code_of_project) ; AuthorizedM vs RAAM - reimbursement authorized amount of money – both referring to the same concept).

4.4 Order columns

Not all documents follow the same column order. This is not a problem for documents where headers are present, but for files without headers, it is not possible to determine what type of information each column contains.

4.5 Name inconsistencies for the same entity

For data to be relevant it is necessary for the same entity to have the same unique name across each entry from each document. For entities that are identified using a unique ID this problem is only manifested by variations in the description or properties (eg for the same PUC, there can be different regions and city values). Instead, the entities which are identified only by the name can be written in multiple ways (eg "Praha" and "Prague" can be considered different entities, which is not true both of them describing Prague town.). There are two possible reasons for these inconsistencies: changing the name or the entities' properties or writing mistakes.

4.6 Format of date-type fields

The representation of the date field type is not consistent. In some documents, dates are specified as XLS specific format (the number of days since 1 January 1900) and in others they are formatted as strings (e.g., DD-MMMM-YY).

4.7 Missing information

Within a document, some rows may have missing columns. The problem that arises is how to determine if a missing value is the result of a shifting/formatting error or if the fact that it misses is correct (eg: the completion date of financing contract is missing because the contract is not yet finalized).

4.8 CSV structure inside one cell of XLS documents

Some documents contain information structured in a different format than the rest of the document. This is a kind of information which is found in a tabular format having as value separator \t and \n for new row in one single cell of XLS documents. This format variation is not detected by the XLS parser used in Microsoft Excel or LibreOffice Calc, fact which leads to erroneous data interpretation. This causes errors in the data interpretation process and leads to the loss of information from the differently formatted block as well as the invalidation of the row in which there is a CSV format cell present.

In order to achieve the purpose of the paper, which is error correction and data standardization for information containing the public contracts financed by European Union, it is necessary to process them in order to bring them from the current form (XLS and CSV files) to the final one (database). Data analysis results in a large number of errors that need to be handled and require processing at different levels: from

the interpretation and formatting of numerical values and data to the elimination of duplicate entries.

Due to the quality of the data set and its size, the processing phase is one of high complexity, which is why it will be divided into several stages, which will keep the intermediate results.

5 Architecture

Dataset processing will be done in 4 steps. Each stage is based on several scripts that are processing units and a database in which the results will be stored. The Figure 1 shows an overview of the architecture used for data processing.

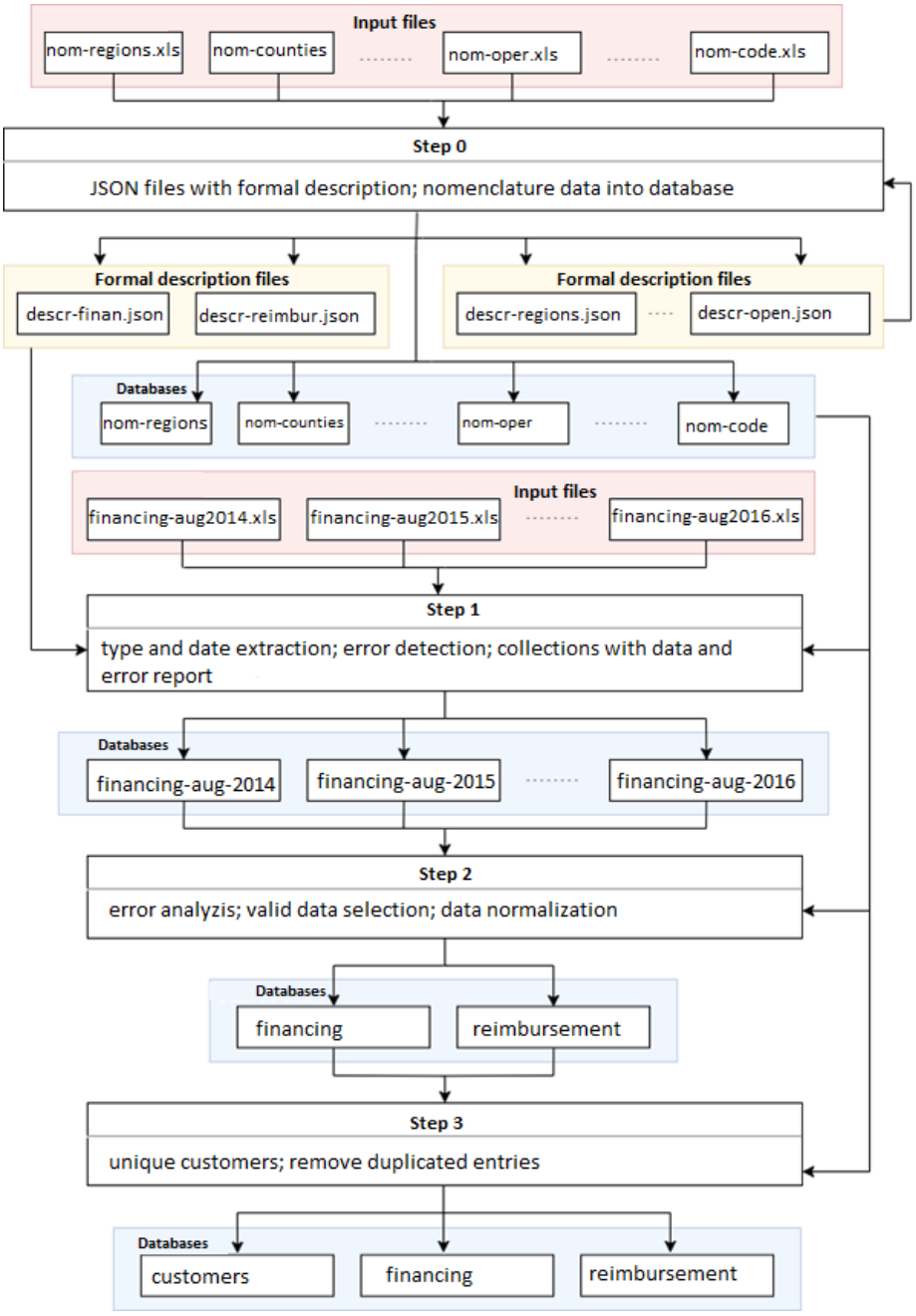

Figure 1. Steps for data processing

5.1 Step 0

Step 0 involves creating formal description files for each type of document and creating databases from the nomenclatures. The first step in data standardization is creating a formal description of the structure for each document type. This is necessary to treat cases where headers names varies and also to determine the type of values on each column from the documents that do not have headers. Creating a description file about each column from the file is useful also for the validation step, when values from columns are compared with the ones from nomenclatures.

In this way, each document is assigned a formal file, containing the description of its structure and other specific properties for each type of information that characterizes an entry in the document. The format of this formal file is JSON, so it can be easily described, modified and parsed in the data processing further steps.

The input data of this step are the files containing the nomenclatures. At the end of this stage will result 7 collections based on nomenclatures' data, one for each nomenclature and 2 more formal description files, one for each document type.

5.2 Step 1

Step 1 will receive the documents regarding finance and reimbursement data and will use the information from the nomenclatures collections which resulted in the previous step. This step aims to extract information from each file (type, date and content), entity standardization, error detection and correction of certain types, resulting in preliminary data organized in multiple collections (a collection for each file) that contain data together with the error report.

Information regarding the type and date of each document can be found, as we mentioned in its name, but it does not follow a fixed format. The type of documents can have two values: financing or reimbursements contracts. The date specified in document's name represents the month and the year of each entry from the file.

A document is considered valid if a tuple of <one single document type, one month, and one 4-digit number, between 2009 and 2016> is found. The documents for which such a tuple was not found will not be processed and are reported with error so they can be processed manually.

The next step is to read the data from tabular files. Because values belonging to the same type of logical information can be represented in different formats (eg: numbers as string or as number) we normalized the format across each column.

Some cells may contain values in CSV format separated by \t and \n. After a manual analysis it turns out that a single grouping in this format can contain hundreds of valid inputs, which implies treating these cases, rather than data invalidation, even with the risk of affecting other adjacent inputs correctly represented. The detection of these data groups is done by checking whether a value contains the \t character and parses this value.

These validation steps result in an error report for each entry from each document. The types of errors presented in the report are: missing fields (there is no such field in

one entry), field nulls (exist but are null) and invalid values (resulting from the validation process with nomenclatures).

Table 1. Entries statistics after processing Step 1

	Financing	Reimbursement
Total entries	337 685	1 132 656
Values in CSV format	845	1002
Obtained inputs after parsing values in CSV format	7329 (2.17% increase)	9838 (0.86% increase)
Total errors	12176 (3.6%)	101416 (8,9%)
Missing values	9188 (75.45% of total errors)	54275 (53.5% of total errors)
Invalid values (nomenclatures)	2674 (0,7%)	3287 (3,2%)
Common Missing Values	POC (project_unique_code)	last_payment_date (78%)

The total number of entries extracted from documents is 1 470 341. Of these, 1847 represent values in CSV format that, following the parsing process, resulted in 17,167 valid entries, recording a 1.16% increase in data volume. As for the number of errors, a total of 7.7% of entries contain quality errors. If in the case of funding documents the percentage is small (3.6%), the data about reimbursements contain 8,9% errors, over 78% of errors are due to the absence of last_payment_date field.

5.3 Step 2

Step 2 starts from the collections with the nomenclatures and the results from the previous stage. At this stage, data is filtered to keep only the valid entries.

The first criterion used for defining an entry as invalid is the presence of values, which cannot be corrected nor found in nomenclatures. This is due to the fact that all the fields present in the nomenclatures are key information for later aggregations and filters.

The second criterion is dependent on the document type and involves defining the mandatory keys for a document type. Thus, are selected only the entries in which mandatory properties are not missing. For documents regarding funding data, the required fields are POC and Customer, while for reimbursement data the required fields are POC, Customer and ReimbursementNum.

Any data that does not meet these two criteria are ignored and will no longer be used, but will remain in the intermediate database for archiving.

In order to be able to perform searches and aggregations over these data, it is necessary to normalize all the values. For this purpose the following operations are carried out:

a) Empty fields are deleted altogether

b) All unwanted characters, such as ‘\n’, ‘\r’, ‘"’ or leading or trailing whitespaces are removed from all string values

c) Fields having a date value are standardized to ISO8601 [9] format, thus all strings, numbers or "dd-mm-yy" date format are converted to “yyyy-mm-dd” format.

Based on the invalid data selection process, from a total of 1,487,508 entries 6183 were discarded, representing a negligible percentage of 0.41%.

Table 2. Entries statistics after processing Step 2

	Financing	Reimbursement
Total entries	345 014	1142494
Invalid entries	2688 (0.78%)	3495 (0.30%)

5.4 Step 3

Step 3 performs high-level data processing over the information obtained at the previous step to uniquely identify the customers recipients and perform more data validation steps, including duplication removal. The output of this step will be a new collection dedicated to the customers and the final version for funding and reimbursements data.

This final step of data processing involves creating a collection with each unique customer and a high-level data analysis by deleting redundant data. The input for this step is the data obtained from the previous step: the two collections (financing and reimbursements) with normalized data that does not contain errors.

One usage of these data is to uniquely identify each customer. As we already mentioned, in the analyzed dataset there is no unique customer code. In our case, we cannot rely on customer name, due to the fact that this can change across time (for example one school has changed its name 6 times in 4 years). Instead, we can rely on the project_unique_code (POC), because each project can have only one customer. So, from each entry from financing and reimbursement data, we extract the following properties: POC, customer name, region, county, locality and build a dictionary with: <POC> as key and List<customer_name, region, county, locality> as value.

The use of lists instead of a single value is given by the fact that the same customer can have multiple values for certain fields (e.g., different customer names). The number of customers in this situation is 189 and represents 1.89% of the total 9973 uniquely identified beneficiaries.

Another part of this step is to remove duplicate entries from reimbursement collection, which can appear due to multiple appearance of the tuple <POC, ReimbursementNum> during several months. We identified a total number of 10395 of this type of duplication, representing 0,9% from the total number of entries.

6 Data usage

A number of Romanian institutions specialized in fraud detection use this curated date in order to analyze public contracts funded by European Union.

A first example of data usage is to obtain the period between the submission date of the reimbursement request and the time this request has been authorized and the money has been reimbursed. To determine this information we plotted on Figure2 the average waiting time calculated for each outsourcer.

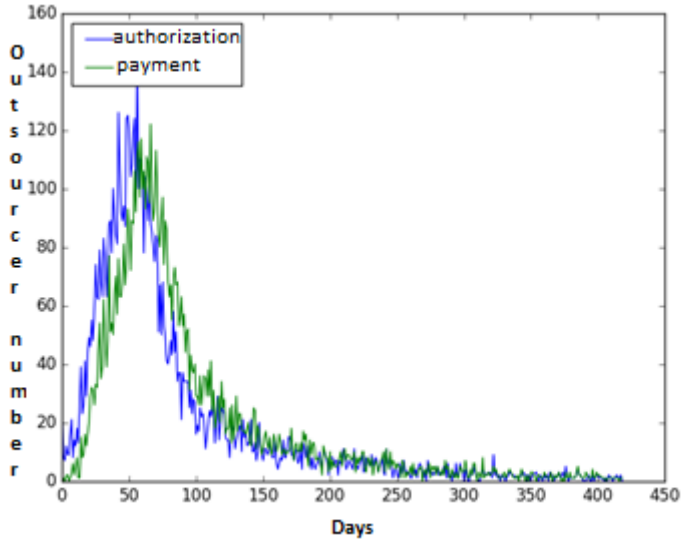

Figure 2 The graph represents the number of outsources as a function of the average waiting time. The Ox axis contains the number of days and the Oy axis the number of outsources who have waited an average number of X days to authorize the reimbursement request (blue), respectively to receive the reimbursement (green).

Another interesting statistics is tracking how these reimbursement requests are spread across different Romanian regions. This information is useful to observe the trend of growth or decrease of the total European Union funds allocated for certain regions or at national level.

The data for reimbursements is grouped according to the month and year taken from reimbursement_authorization_date and for each month is calculated the sum of AuthorizedM for each region. The results can be seen in the figure 3.

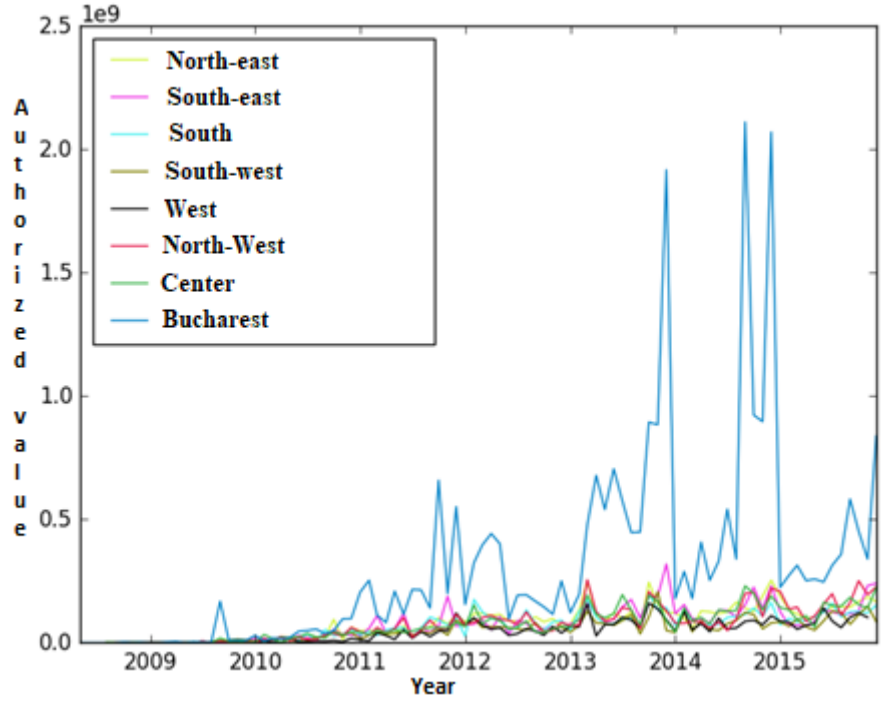

Figure. 3**.** Represents the evolution over time of the amount of authorized values for each region. The values on the Y-axis have as measurement unit millions, and the X-axis grain is monthly.

7 Conclusions

Open data provided by public institutions have the potential to increase transparency of the state's activities and to minimize corruption. Unfortunately, a large part of the open government data sets are incomplete and unstructured, making their usage difficult and reducing the expected benefits.

In this paper we analyzed the types of real errors that can occur in files published by governmental public institutions and how these types of errors can be eliminated. We have applied this methodology on data about Romanian public contracts funded by European Union, processing them to discover and eliminate errors. Based on the analysis of this dataset, we designed an automated data processing framework divided into several stages of processing data with different granularities. We applied a set of normalizing and correcting steps to obtain a structured and error-free data format that is currently being used by different institutions in order to detect frauds.

References

1. Berners-Lee, T., “Design issues: linked data”, available at http://www.w3.org/DesignIssues/LinkedData.html, accessed January 2018.
2. Bhatnagar, S. Transparency and corruption: Does e-government help?. DRAFT Paper prepared for the compilation of CHRI, 1-9.(2003)
3. Böhm, C., Freitag, M., Heise, A., et al.: Govwild: integrating open government data for transparency. In: Proceedings of the 21st International Conference on World Wide Web. pp. 321–324. ACM (2012)
4. Futia, G., Melandri, A., Vetrò, A., Morando, F., & De Martin, J. C. Removing Barriers to Transparency: A Case Study on the Use of Semantic Technologies to Tackle Procurement Data Inconsistency. In European Semantic Web Conference (pp. 623-637). Springer, Cham. (2017)
5. Heath, T., Hausenblas, M., Bizer, C., Cyganiak, R., Hartig, O.: How to publish linked data on the web. In: Tutorial in the 7th International Semantic Web Conference, Karlsruhe, Germany (2008)
6. Janssen M., Haralabidis Y., A. Zuiderwijk. Benefits, Adopton Barriers and Myths of Open Data and Open Government. Informaton Systems Management (ISM), vol. 29, no.4, pp. 258-268.(2012)
7. Rajshree, N., & Srivastava, B. Open government data for tackling corruption-a perspective. In Workshops at the Twenty-Sixth AAAI Conference on Artificial Intelligence (pp. 21-24) (2012)
8. Scharffe, F., Atemezing, G., Troncy, R., et al.: Enabling linked-data publication with the datalift platform. In: Proc. AAAI workshop on semantic cities (2012)
9. Wolf M, Wicksteed C. Date and time formats. W3C NOTE NOTE-datetime-19980827, August. 1998