International Journal of Grid and Distributed Computing
Vol. 9, No. 7 (2016), pp.43-52
http://dx.doi.org/10.14257/ijgdc.2016.9.7.05

Data Cleaning Utilizing Ontology Tool

Jing Ting Wong and Jer Lang Hong

School of Computing and IT, Taylor's University
jingting.wong@taylors.edu.my, jerlang.hong@taylors.edu.my

Abstract

Recent advancement in Internet Technologies has made web browsing increasingly easy and user friendly. From the traditional method of desktop web browsing and the birth of dial up modem connection, users nowadays are able to enjoy a fast and reliable web browsing via high speed wireless Internet connection and portable mobile devices. Browsing a web has become much easier with the state of the art search engines such as Google, which provide much functionalities which could make browsing easier such as improved crawler, easy to use search interface, web personlization, Web 3.0 support and integration and many more. In order to build a robust and reliable search engine, the developer needs to integrate all the data and present them in a meaningful format for user's viewing convenience. Integrating these data is a tedious task as data usually occur in numerous format, and layout. Furthermore, web developers usually present the data content in various languages of their choice, which made the processing of these data increasingly difficult. There is also no standard convention to represent the data format and even a standardize rule to process this data has not been developed. To resolve this issue, researchers develop data extractor which could effectively extract data from web sources, tabulate them, and used it for further processing. However, not all data are correctly extracted, they may either missed certain valuable information or contain additional unnecessary information. In the case of unnecessary information, researchers use a cleaning method to remove them such that the data extracted are free of errors. Removing these data is important as unnecessary information may affect the accuracy of subsequent extractor tools, hence may eventually prevent the tool from performing its task efficiently. In this research proposal, we embark on a data cleaning tool to clean data using ontology tools. Experimental results show that our tool is highly efficient in data cleaning and is able to outperform existing state of the art tools.

Keywords*: Data Cleaning, Ontology, Deep Web*

1. Introduction

Recent advancement in Internet Technologies have ease the users in their daily tasks. Users can now easily browse a web using their smartphones in any location of their choice. With the development of high speed processor and wide accessibility of Internet connection, users are able to use their devices to perform various tasks from checking their mail, routing information through GPS, and even programmed their microwave oven for baking purposes. To provide all these data in integrated form for users' viewing, developers need to develop an integrator tool, which includes crawler, extractor, alignment, and cleaning components.

Crawler is usually used to search for data across a vast pool of internet sources. Searching across the huge resources of Internet domain is usually highly impractical, developers usually resorted to focus specifically on specific domains of their choice, by narrowing down their searches to domain which are relevant to their searches, using web indexing and classification methods. Once a specific domain is identified, developer then use extractor tool to extract data from the domain. Extracting data from the domain

ISSN: 2005-4262 IJGDC

possess another fundamental problem, as the data from different domains are usually presented in different formats and layouts. Therefore, it is very difficult to identify the schema or template of these data and extract them out correctly. Even they have been successfully extracted, they may contain extra unnecessary information or missing relevant information.

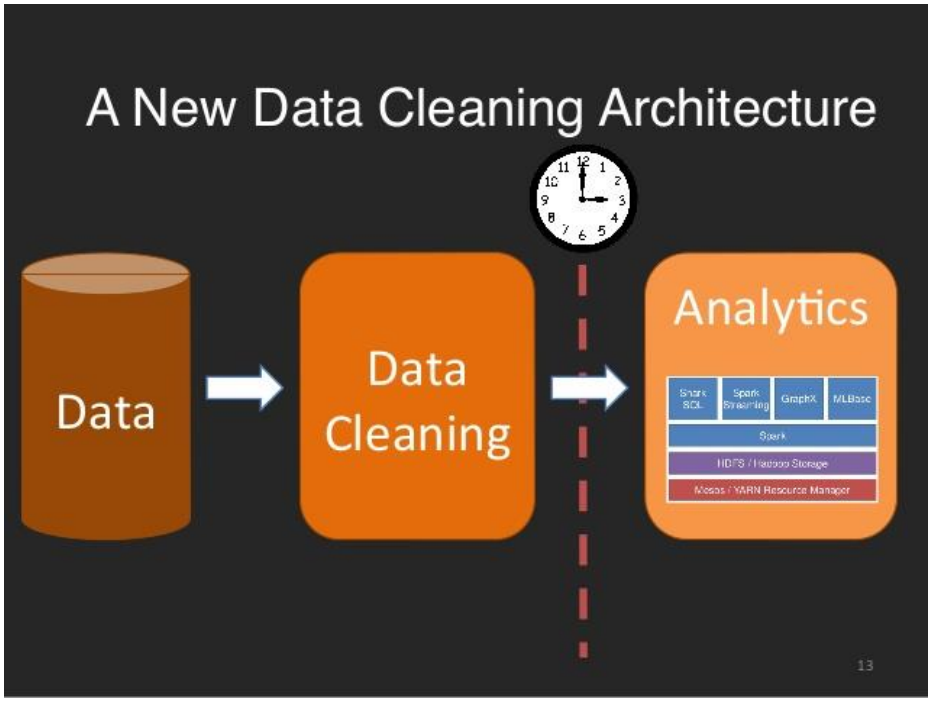

Figure 1. Data Cleaning Process

To resolve this issue, developers usually extract as much information as possible, on the assumption that unnecessary information can be removed later in the processing stage. They can't afford to miss out the relevant information during the extraction stage simply, as it is certainly hard to obtain this information in the later part of the processing stage. To remove unnecessary information from the extracted data, developers use data cleaning tool, which checks for the similarity across all the individual data, and performs cleaning work where necessary. Once data cleaning is carried out, they can then be used for further processing.

Our research proposal embarks on data cleaning tools. Several existing state of the art tools exist for data cleaning, all of them are efficient but not without problems. In particular, all of these tools are not able to check the content of the extracted data, instead they rely on the structure and visual layout of the extracted data for cleaning purposes. Our key observations on the content of the data show that the data content contains invaluable information for data cleaning, in addition to their structure and layout. We are of the opinion that data structure and layout have important part in the data cleaning process, however if we were to incorporate the semantics of data as part of our data cleaning methods, the accuracy could be significantly improved. We propose a novel data cleaning tool utilizing Ontology as the primary component of our methods in addition to data structure and layout. Our method works by checking the semantics of data using WordNet in addition to DOM Tree Structure and its underlying visual information. Unlike other existing techniques, our tool provides multilingual support and is able to correct broken languages as well as matching the data at template level in a fully automated environment. We hope that using ontology tool we can then improve the accuracy of the data cleaning tools significantly. Experimental results show that our tool is highly

efficient in data cleaning and is able to outperform exisitng state of the art tools. After data is cleaned, it can then be used for further processing such as data analytics. Figure 1 shows the process of data cleaning operation.

2. Related Work

Research on data cleaning first emerged in [13]. Since then, extensive research has been made on the ETL tools [1], where support for data transformation, merging, and repairing of data are proposed. Recent research focused on data consistency, for example the work of FDs [17] and its extension CFDs [6], [9], MDs [10], and INDs [4]. However, these works suffer from several drawbacks. Their level of expression is of first level order and does not represent real life scenarios [15]. To resolve this issue, researcher has developed an interface where users can express their thoughts and have better control of the cleaning process.

Most data cleaning techniques rely on similarity and matching operators, hence they tend to process beyond first order logic. Though these techniques are not able to represent the level of expressiveness completely, they can at least be integrated into other research domains such as record matching, entity resolution, and duplicate detection [8]. The work of [2] proposed a two steps solution for data cleaning, match and merge, where match identifies duplicates data and merge combines the duplicate data into one.

Several other data cleaning methods have been proposed. The works of [4], [6], [12], [13], [15], [18] use heuristic rules based on the FDs [4] and INDs [4], CFDs [9], and editing rules [13]. The work of [4], [6] employs a predefined value set by the users to guide the repairing process. This approach however, is manual and labor intensive, hence not suitable for large scale data processing. The work of [] detects the missing data by statistical calculations [15]. On the other hand, the works of [15], [18] require user guidance for cleaning process. Similar to the work of [4], [6], these approaches are semi supervised and may not be suitable for large scale data processing. For further information on data cleaning, the users may refer to the work of [11], where record matching and data matching are studied in details.

Recently, ontology has been used in data cleaning research. The work of [5] uses clustering based ontology where its work in a semiautomatic environment. Dimitris *et. al.*, [14] uses an NLP based approach for data cleaning where it uses Linked Data for cleaning data. In 2005, Xin *et. al.*, [16] uses ontology to describe data and domain represented by the classes and their attributes. On the other hand, Stefan and Fabian [3] uses ontology for Data Quality Management and Julian *et. al.*, [7] uses an ontology based approach for data cleaning which is able to scale across big data in the information extraction field.

3. Problem Formulation

Cleaning the data after extraction is a non-trivial task. First of all, most of the extracted data exist in unformatted form, hence distinguishing the text content in these data is difficult due to the lack of information available. The only feasible way of checking the unnecessary information is through the semantic properties of the text content. However, not all the text are written in English language, hence if a researcher is to use semantic based tools for processing, that tools must be able to support multilingual format. Finally, text written in many webpages is categorized by sentences, followed by paragraphs. All the words exist in these text are neither related to each other by their structure nor visual layout information, hence existing approaches using DOM Tree and its underlying visual properties may not work well on data cleaning.

4. Motivation

Inspired by the challenges in data cleaning research and the recent advancements in Ontology Research, we are of the opinion that ontology based tools are the best option to solve all the problems in data cleaning. Though ontology tools are slow in operation, we feel that this is not a major drawback due to the recent processing power made available by any personal desktop manufacturers. Unlike the early days which uses mainframe computer, modern computers are not only fast in processing powers, but they are also smaller in size and portable across regions. It is very convenient to move around and perform daily tasks with handheld devices nowadays, which are capable of doing almost all the jobs such as web browsing. Furthermore, due to the promising results obtained in ontology research, researchers have since worked on extending the ontology tool to support other languages. Multilingual support is an important factor to our work for handling webpages written in different languages. Furthermore, ontology tools are capable of handling text content better than DOM Tree based tools, particularly text content in extracted data, where there exists no structural and visual relationship between all the words.

ANIMALS

T_1	1	2	2	4
	Cat	Fel	ine	Last Updated: 1/6/2014
T_2	1	2	3	
	Dog	Canine	Last Updated: 30/6/1988	
T_3	1	2	3	2
	Fish	Pe	Last Updated: 5/8/2014	T

Figure 2. An Extracted Data in HTML Table

ANIMALS

T_1	1	2	3	4
	Cat	Feline	Last Updated: 1/6/2014	
T_2	1	2	3	4
	Dog	Canine	Last Updated: 30/6/1988	
T_3	1	2	3	4
	Fish	Pet	Last Updated: 5/8/2014	

Figure 3. Pre-Processed Table

We use common ontology WordNet to clean a webpage. Inspired by the extensive libraries provided by WordNet and the multilingual as well as cross platform supports, we believe that using WordNet is the best choice for cleaning a webpage efficiently. Though research on using ontology tools for cleaning a webpage is still in its infancy, ontologies research has shown a dramatic increase in its application unlike other approaches. This is because ontology tools are able to achieve higher precision than other conventional methods. Secondly, recent research on ontologies provides support for higher intelligence that is they provide more semantic capabilities and larger knowledge domains for the researchers. SUMO for example, gave upper ontology support for the common ontology WordNet. On the other hand, research has been widely carried out to provide multilingual

support for WordNet. This is a significant advantage for us, as different languages have different syntaxes and structures in their representation, making them complicated to be included in the WordNet library. An alternative implementation is therefore required which involves complete reimplementation from the syntaxes, synsets, structure, and word relationships. Due to the fact that WordNet contains multilingual support for 90% of the languages in this world, it is certainly possible for us to use ontology to clean a webpage without much difficulty. We are able to clean most webpages if not all, written in any languages having different structure and layout. Figure 2 and Figure 3 illustrate an example of how a data is structured before data cleaning and after the preprocessing stage.

5. Proposed Solution

5.1. Initial Stage

Data is parsed from tabulated data extracted from the World Wide Web. The tabulated data is generated using ICE Browser which reads sample pages and categorizes the content into tokens. These tokens exist in two forms: HTML command tag and text token. HTML tags are text that are enclosed in '<' and '>'. The rest is considered text tokens. These data is stored in a DOM tree and an excel file is produced with the corresponding data. Data in the excel file is parsed into the data cleaning system and a linked list is generated.

Before data cleaning is carried out, we first need to extract out the data from deep webpages. To do this, we use WISH wrapper to extract data. Once data is extracted, they are then further aligned into tabular form. The tabulated data is then used for further processing in data cleaning stage.

5.2. Template Identification

We first clean the data by matching and comparing all the data located in the same column of the table. We are under the assumption that at least 90% of the data are aligned using similar template. As such, we try to identify all the templates in the table before we clean the data in the next stage. We provide a list of predefined templates to match with the data aligned. The most common templates that we have obtained are date format, price, age, name, country code, and phone number. These templates follow certain predefined format which can be formulated using regular expression rules. Other than predefined templates, we also create templates which have predefined orders. For example, companies usually present certain data using certain order, such as name of a person (first name followed by last name). We determine the templates of each of the data column using our predefined templates. If we are not able to determine the template of the data, we treat the data column as not defined.

5.3. First Stage Cleaning

The information in the linked list is processed with a post tagging algorithm. This algorithm leverages on regular expression to remove punctuations that will get in the way of efficient data cleaning. The only punctuation allowed is space ' ' which will be used to tokenize the data into arrays. Information such as date and price in which punctuations matter will be retained to prevent the loss of integrity.

Once data columns are assigned certain template, we use the template to identify the text contents. As mentioned previously, our research deals with unnecessary information, not missing information. Therefore, for every text contents we handled, we assumed that not all the text contents contain words which are relevant to the data column aligned. To remove the irrelevant words from the text contents, we use the template assigned to the text contents as a guide. For example, in the column "Price", there exist text content

"Hurry! Buy Now at RM 56.00", our method identify that this data column has price template assigned to it. Hence, we know that the text "RM 56.00" is relevant to our work, while the text "Hurry! Buy Now at" may not be relevant at all to our work and should be removed.

5.4. Second Stage Cleaning

Unfortunately, removing and cleaning data using template may not be as easy as we perceived. Most users write the content of their websites using broken language. Moreover, some users may use certain slang, symbols, and terms to represent a particular phrase or definition. For example, the word "Nike" represents the world well known Shoe Maker while the word "☺" may represent a person happiness in regard to a particular topic under discussion. To resolve this issue, we propose a spell checker which could ultimately spell check a particular word and correct it automatically. Once a word is corrected, we then use stemmer which is able to stem a word to its root. To detect the special character of a word, we create predefined character sets. Using these predefined character sets, we then matched it against the data column text contents.

Each word in the array will be checked in terms of spelling with an open source Java Spell Checker. Suggester Spell Check uses Basic Suggester as a spell checker and is a 100 % pure Java library. The Suggester provides a list of recommendations, if any, for words that are unknown to the Suggester. For efficiency sake, the first suggestion will be selected as it is the highest in possibility in terms of accuracy.

The Suggester is intelligent in the sense that it uses shortest Edit-distance measure combined with Metaphone algorithm and Fuzzy-machine algorithm to rank suggestions. Dictionary search and suggestions provider algorithms are perfected to take up only 0.002/0.05 ms per word for word search and 40 ms per set of suggestions for each unknown word in the user query on Pentium M 1.4Gz. Of course, the quality of suggestions is the highest which translates to having one of the suggestions being selected by the user.

Spell-checked words are then stemmed to its root form using Porter Stemmer. It is an algorithm to remove the commoner morphological and inflexional suffixes from words in English. Terms with common stem usually have similar definition, for example:

Connect
Connected
Connecting
Connection
Connectivity

These terms will be stemmed into a single term which is "Connect" which is done by removing various suffixes –ED, -ING, -ION and –IVITY. The removal of suffixes will reduce the size and complexity of the data in the system.

5.5. Third Stage Cleaning

The previous two steps use predefined templates and libraries for matching purposes. In some cases, these steps may be impractical to generally solve the data cleaning problem. First of all, English languages are huge and extensive, with most of the words interrelated to each other. Therefore, it is hard and time consuming to create a predefined library to check the data column efficiently. For example, two words in English language may be the synonyms, hence they should be treated as similar. Fortunately, researcher has since developed a semantic based lexical dictionary called WordNet which is able to conceptualize the English words and present them in a meaning format. To check whether two words are similar or not, we use the Word Similarity Check by Jiang and Conrath to check the word similarity. If two words are at least 75% similar, they are treated as

belonging to the same template. Otherwise, the two words are removed as they are treated as unnecessary information. Some text contains word disambiguation. For example, the word "interest" in the sentences "Interest in book" and "High interest rate in bank" are having entirely different meaning. For such a case, we use Adapted Lesk algorithm to differentiate the meaning between these two keywords. Adapted Lesk algorithm detects the semantic of two similar keywords by checking their neighboring words and matched those neighboring keywords with WordNet similarity check. Since the two sentences mentioned previously have highly dissimilar keywords (*e.g.*, book, bank), it is concluded that the two sentences are not semantically similar, hence chosen for removal.

5.6. Fourth Stage Cleaning

The method we used previously does not cater for other languages other than English. To date, there exist numerous webpages written in different languages. The earlier version of WordNet caters only for English language. Recently, research is carried out where support for other languages has been incorporated into WordNet. This is a significant advantage to our work as multilingual support provided by WordNet can be used to analyze the semantic properties of text data written in various languages. To check for semantic similarity between keywords written in other languages, we need to implement the similarity methods in WordNet to cater for other languages. Fortunately, it is not difficult to map the implementation of Word Matching in English to that of other languages of WordNet as the functionalities provided by WordNet across other languages are almost similar though the accuracy returned by all these different methods may not be exactly similar. For example, a match between Cat and Dog in English WordNet may return 75% similar while that of Chinese WordNet may return 73% similar. Once we have implemented all the similarity check methods for WordNet written in other languages, we repeat the similarity check procedure used previously.

5.7. Fifth Stage Cleaning

The fifth stage cleaning involves clustering the cleaned data into the relevant groups. To achieve this, we use multi objective clustering methods SOM. Due to the heterogeneous nature of the data, we choose SOM clustering method which is capable of grouping data based on multi objective factors. First, we group data using their semantic similarity. After that, we group data based on their data type. Data type is those with predefined formatting such as date (DD-MM-YYYY), and time (HH-MM-SS). Once data are grouped using their semantic, we further grouped data according to word disambiguation. We used Adapted Lesk algorithm to group data.

5.8. Data Cleaning Finalization

Once all the five stages of data cleaning are completed, we then revalidate the data using human labelling. We collect a random samples of 250 pages, and then run the experiments with a group of 3 researchers. Tabulated data is consider free of errors if at least 2 researchers agreed with generated results. We consider our data cleaning as completed if the human validation test returns an accuracy of 90%. Otherwise, the previous automated procedure of data cleaning is improved until an accuracy of 90% is achieved.

6. Experimental Tests

We conduct our experimental tests on a wide range of datasets. We collect a random sample of 250 pages from the deep web repositories for single language and another 200 pages for webpages written in different languages. The domain of our data collection varies, they are taken from various sources such as governmental sites, social networking

sites, blogs, forums, search engines, and online auctioning sites. Webpages that failed the preprocessing phase are taken out from our evaluation. In addition to that, we make full attempt to ensure that all webpages under evaluation are well formed and erroneous. This step is carried out to ensure that a proper DOM Tree is constructed together with its visual cue. We measure the effectiveness of our algorithm using precision and recall which are formulated as follows:

Recall=Correct/Actual*100
Precision=Correct/Extracted*100

Table 1. Experimental Tests (Single Language)

Terms	Our method	SemantiClean [7]
Actual	1214	1214
Extracted	1186	1012
Correct	1004	847
Recall	82.70%	69.77%
Precision	84.65%	83.69%

Table 2. Experimental Tests (Multiple Languages)

Terms	Our method	SemantiClean [7]
Actual	998	998
Extracted	864	678
Correct	825	451
Recall	82.67%	45.20%
Precision	95.49%	66.52%

Correct depicts the number of data items correctly identified. Actual is the actual number of data items in that webpage. Extracted depicts the number of data items extracted. We benchmark our work against the work of [7], a state of the art tool which utilize ontology for data cleaning. We conduct our experiments in two stages, where the first stage involves data cleaning for webpages written solely in English language while the second stage involves data cleaning for webpages written in broken language as well as other languages.

Table 1 and Table 2 show the experimental results of our test. Our system outperforms the work of [7] both in terms of recall and precision. This is due to the fact that our work incorporates state of the art ontology WordNet to detect and check the semantic properties of data. On the other hand, our system significantly outperforms the work of [7] both in terms of recall and precision when tested on multilingual datasets. The work of [7] is incapable to detect and check the semantic properties for webpages written in other languages. Our system, however, is designed to check the semantic properties of data written in most of the common languages in the world. Unlike the work of [7], our system has a flexible yet robust spell checker technique which is able to detect broken languages and special characters.

7. Conclusions

The proposed solution on data cleaning begins with the realignment of data extracted using templates, follow by identification of relevant keywords for subsequent data cleaning context. After that, a built in spell check is carried out to correct broken languages. With the assistance of ontology-based tool WordNet, the spell-checked data will then be compared to evaluate their similarity. We then extended our solution further to support multilingual data. Finally, we validate the correctness of our test using human

validation to ensure the accuracy of the output. Experimental results show that our proposed solution is highly effective in data cleaning and is able to outperform existing state of the art techniques. Our system will be very helpful for other data intensive applications.

References

[1] C. Batini and M. Scannapieco. Data Quality: Concepts, Methodologies and Techniques. Springer, **(2006)**.
[2] O. Benjelloun, H. Garcia-Molina, D. Menestrina, Q. Su, S. E. Whang and J. Widom, "Swoosh: a generic approach to entity resolution", VLDB J., vol. 18, no. 1, **(2009)**.
[3] S. Brüggemann and F. Grüning, "Using Domain Knowledge Provided by Ontologies for Improving Data Quality Management", Networked Knowledge - Networked Media Studies in Computational Intelligence vol. 221, 2009, **(2009)**, pp. 187-203.
[4] P. Bohannon, W. Fan, M. Flaster and R. Rastogi, "A cost-based model and effective heuristic for repairing constraints by value modification", In SIGMOD, **(2005)**.
[5] D. Cherix, R. Usbeck, A. Both and J. Lehmann, "CROCUS: Cluster-based Ontology Data Cleansing", Proceedings of the 2nd International Workshop on Semantic Web Enterprise Adoption and Best Practice, **(2014)**.
[6] G. Cong, W. Fan, F. Geerts, X. Jia and S. Ma, "Improving data quality: Consistency and accuracy", In VLDB, **(2007)**.
[7] J. Dolby, J. Fan, A. Fokoue, A. Kalyanpur, A. Kershenbaum, L. Ma, W. Murdock, K. Srinivas and C. Welty, "Scalable Cleanup of Information Extraction Data Using Ontologie"s, The Semantic Web, **(2007)**.
[8] A. K. Elmagarmid, P. G. Ipeirotis and V. S. Verykios, "Duplicate record detection: A survey", TKDE, vol. 19, no. 1, **(2007)**.
[9] W. Fan, F. Geerts, X. Jia and A. Kementsietsidis, "Conditional functional dependencies for capturing data inconsistencies", TODS, vol. 33, no. 2, **(2008)**.
[10] W. Fan, X. Jia, J. Li and S. Ma, "Reasoning about record matching rules", PVLDB, vol. 2, no. 1, **(2009)**.
[11] W. Fan, J. Li, S. Ma, N. Tang and W. Yu, "Interaction between record matching and data repairing", In SIGMOD Conference, **(2011)**.
[12] W. Fan, J. Li, S. Ma, N. Tang and W. Yu, "Towards certain fixes with editing rules and master data", VLDB J., vol. 21, no. 2, **(2012)**.
[13] I. Fellegi and D. Holt, "A systematic approach to automatic edit and imputation", J. American Statistical Association, vol. 71, no. 353, **(1976)**.
[14] D. Kontokostas, M. Brummer, S. Hellmann, J. Lehmann and L. Ioannidis, "NLP Data Cleansing Based on Linguistic Ontology Constraints", The Semantic Web: Trends and Challenges, **(2014)**.
[15] C. Mayfield, J. Neville and S. Prabhakar, "ERACER: a database approach for statistical inference and data cleaning", In SIGMOD, **(2010)**.
[16] X. Wang, H. J. Hamilton and Y. Bither, "An Ontology-Based Approach to Data Cleaning", Technical Report, University of Regina, **(2005)**.
[17] J. Wijsen, "Database repairing using updates", TODS, vol. 30, no. 3, **(2005)**.
[18] M. Yakout, A. K. Elmagarmid, J. Neville, M. Ouzzani amd I. F. Ilyas, "Guided data repair", PVLDB, vol. 4, no. 5, **(2011)**.

