World of Computer Science and Information Technology Journal (WCSIT)
ISSN: 2221-0741
Vol. 9, No. 5, 22-27, 2019

A Content-Based Schema Matching Tool

Farid Bourennani
Dept. of Computer Science and Artificial Intelligence,
Faculty of Computer Science and Engineering,
Jeddah, Saudi Arabia

Mike Bourque
Faculty of Business and Information Technology,
University of Ontario Institute of Technology,
Oshawa, ON, Canada

Abstract— Schema matching (SM) is a fundamental task of data integration and data warehousing. Often SM is performed manually which is time consuming and error prone. Furthermore, existing SM tools do not scale well to large schemas. To alleviate these challenges, a novel tool is proposed for automated schema mapping based on the content by matching data entities exclusively based on the content. The resulting topology is convenient to visually explore the relationship among database entities even in large volume. Also, a post-processing algorithm based on data types is proposed for further enhancement clustering results. We present a case study to demonstrate the efficiency and the practicality of the proposed tool.

Keywords- Schema Matching; Schema Mapping; Data Integration; Content-Based Schema Matching; Instance-Based Schema Matching; Visualization.

I. Introduction

Schema matching consists of identifying semantic correspondences between two or more attributes of differing schemas and is the first step to data integration completion [1]. Also, schema matching is central to many other database applications such as data warehousing, data exchange, information sharing, schema migration, and others. Often, however, schema matching is performed manually which makes this task time consuming [2]. In addition, due to the basic nature of the problem, it is unavoidable that mistakes will be made [1]. Therefore, automated or semi-automated tools are more appropriately suited to the task.

Several schema matching methods have been proposed by researchers in the last decade. Extensive literature reviews can be found in [3]. These methods can be divided into two major groups schema-based matching and instance (content) based matching. Schema-level matching methods are based on schema-related information such as name of attributes, description of attributes, data types, constraints, and structure information. Instance-based methods for schema matching are based on database element instances, statistical similarities, and other content-related information.

Most previous works have focused on the development of schema-based matching tools rather than focusing on data content while data content can reveal more important insights into semantic relation between attributes [4]. In addition, in most schema matching operations there is a human involved in the process; therefore, it is important to have a graphical tool for large schemas [1]. Furthermore, the content-based schema matching methods are based on machine learning algorithms which offer a high level of recall measure and suffer from diminished precision [5] [6].

To address these challenges, this paper proposes the following contributions.

- The development of an automated purely content-based schema matching tool.
- A visual tool capable of handling visually large data schemas.
- A post-processing algorithm for the enhancement of precision of clustering results by focusing attributes data type information.

II. Litterature Review

There has been limited work done in relation to instance-based schema matching [4], [7], [8]. In this section, we present the most important schema-matching related works which are based either on content or hybrid (element and content) schema matching.

A. 2.1 Pure Content Approaches

In [7], a content-based approach has been utilized with google similarity and regular expression libraries resulting in high accuracy ranging between [93%, 98%]. However, there are two limitations to their work is the low number of instances and the restriction to 1-1 matchings. In [8], a pure content-based schema matching was used based on the entropy measure using neural networks resulting an increase of the precision by 7% and the recall by 17%. However, the disadvantage of a neural network is that it needs to be trained with some input data. So, if the training data is very different from the processed data, e.g. heterogeneous domains, the results would probably diverge. In [9], an instance-based schema mapping approach was proposed using a heterogeneous data mining method which consists of simultaneously processing two or more data types. The recall measure doubled while the precision slightly decreased.

B. Holistic Approach

In the previous subsection, the previous works focused on element level data; however, as we focus on instance-level data, we also reviewed hybrid element/instance level works. The first successful hybrid work was called the holistic schema matching [10] which integrates an instance-based approach and an element-based approach. In [11], it is proposed to perform schema matching of individuals stored in different repositories using in sequence instance-level and schema-level approaches. This sequential combination resulted in the enhancement of the recall measure by [15%, 75%] without "substantial loss of precision" [11]. More recently [12], the holistic approach was used for open data schema matching problems with promising results.

Although, holistic works led to promising results, a limited work has been conducted since then probably due to the complex task of combining results from instance-based approach and element level-approach which lead to poorer results. But, one thing is sure is that the amount of instance-level data is significantly larger than element-level data; so, if instance-level data is analyzed properly, it could lead to more accurate results. So, developing automated tools based on data mining and machine learning to process both instance and element level information could lead to promising results, and that’s what we propose in the next section.

III. Proposed Tool

An important part of the proposed work is done at the pre-processing level which is described in the next sub-section. Then, we describe briefly the utilized clustering method called SOM, the proposed visualization tool, and the post-processing algorithm as well as its impact on clustering and visualization results.

A. Data Pre-Processing

The pre-processing phase consists in preparing the data for processing via several steps. First, the relational database entities originating from heterogeneous data repositories are named using the following naming convention column@table@database. For example, a table, from a database 1, is named student and encompasses student information, it is composed of the following columns: id, name, dateBirth, etc. The extracted elements are as follows: “id@student@1”, “dateBirth@student@1”, “name@student@1”, etc.

Then these columns are transformed in vector space model (VSM) commonly used in text mining [13] using the TF-IDF measure [14]. Due to the large dimensionality, we use the Random Projection (RP) which is simple and offer comparable results to PCA [15].

B. Processing: SOM-based Clustering

The clustering is done via Self-Organizing Maps (SOMs) which is an unsupervised learning technique. The most remarkable capability of SOM is its ability to produce, as shown in Fig. 1, a mapping of high dimensional input space onto a low-dimensional (usually two dimensional) map, where similar input data can be found on nearby regions of the map. The resulting map offers improved insight into the interrelationships among the input data which in this work are data entities (columns).

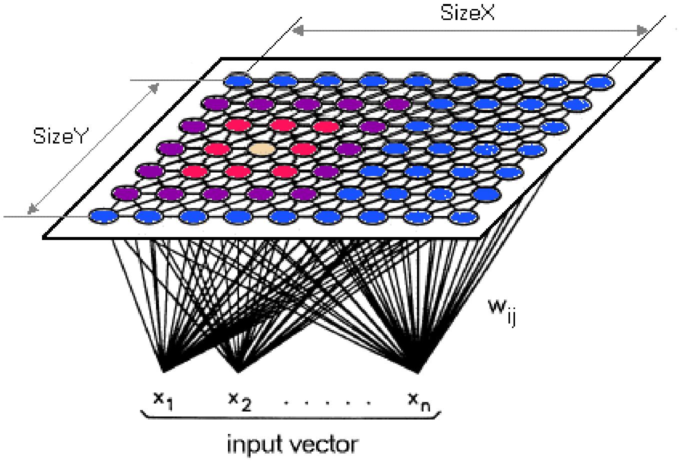


Figure 1: SOM map which projects of high dimensional data into two dimensional space [16].

The map display [17] has the following advantages:

- The ability to convey a large amount of information in a limited space
- The facilitation of browsing and the perceptual inferences on retrieval interfaces
- The potential to reveal semantic relationships of terms and documents

All these advantages are demonstrated through a map display generated by SOM in this work. As such, SOM map has been selected in this paper for schema matching and relationship exploration between data entities.

C. Post-Processing

Most information retrieval algorithms offer a high recall measure but typically a lower precision [6]. SOMs suffer from similar results. Therefore, we propose a post processing algorithm for the enhancement of SOM’s clustering results by improving the precision, i.e. every cluster found by a SOM is broken into sub-clusters based on the data types of the entities.

IV. Experiments

In this section we present case study to demonstrate the practicality of the proposed content-based methodologies. As illustrated in Fig. 2. The first scenario uses only the SOM-based clustering while the second scenario integrates the proposed post-processing algorithm, as second step, to the first scenario. So, the first part called “Document Encoding” is data pre-processing phase described in the previous section. The Second phase “Construction of SOM Map” is the data processing phase which includes visualization. The third phase “Post-Processing Optimization” is optional and serves to enhance the precision of the SOM clustering results as well as to enhance the quality of the visualization tool.

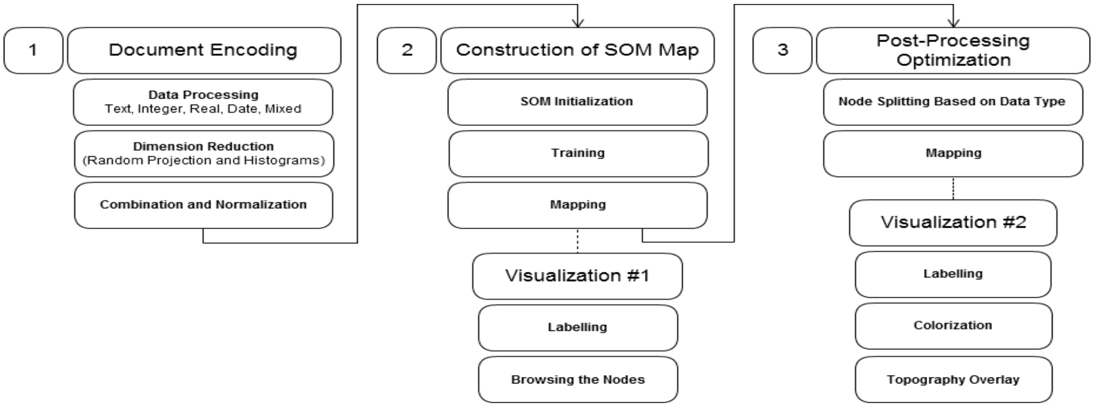


Figure 2: Proposed methodologies.

A. *Input Data*

The tests are run by using the Northix [18] repository which is a schema matching of two demo repositories namely Northwind and Sakila [19]. Table 1 shows the composition of the Northix [20] database.

Table 1: The Norhtix database properties

Data Set	**Data entities**	**Terms (tokens)**	**Classes**
Northix	115	21805	33

4.2 Measures

In order to measure performance of the proposed methodologies, the F-measure is used [13]. It is calculated with respect to the known classes for each document, and it is based on Precision and Recall weights. The inverse relationship between precision and recall with regards to information processing are explained by the following formulas.

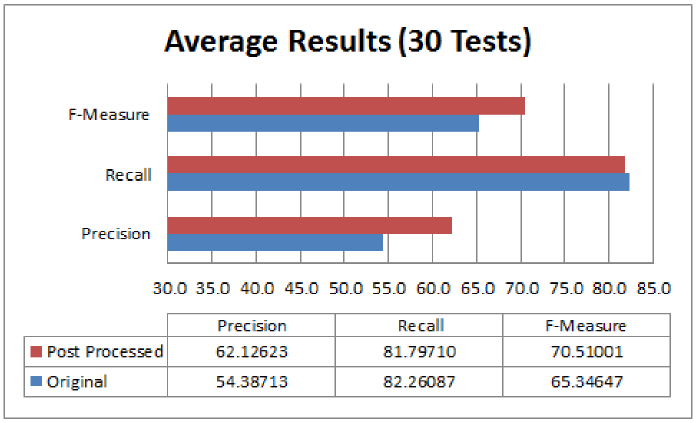
$$\text{Precision} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|}$$

$$Recall = \frac{|relevant\ documents\ \cap retrieved\ documents|}{|\{total\ number\ of\ relevant\ documents\}|}$$

$$F - measure = 2\,.\frac{precision\ .\ recall}{precision + recall}$$

The above formulas show that precision is a fraction or ratio of documents that are both relevant and retrieved over the total number of retrieved documents. The recall formula, however, shows that recall is a fraction or ratio of documents that are both relevant and retrieved over the total number of relevant documents.

4.3 Results



	Precision	Recall	F-Measure
🟥 Post Processed	62.12623	81.79710	70.51001
🟦 Original	54.38713	82.26087	65.34647

Figure 3: Experiment Results – Average measures of 30 Runs.

As shown in fig. 3, the results are presented for both the original processing and the post-processing algorithm over a series of 30 tests. Matlab was utilized to conduct the experiments. For each of the 30 tests, we compared the Precision, Recall and F-Measures. As seen in Fig 3, the average F-Measure for the 30 tests is 65% and it is improved by 5% after the post-processing algorithm is applied. This is due mainly to the precision enhancement which represents the number of retrieved data objects that are relevant to a specified query. The precision increases because of the improved homogeneity of the post-processed nodes since each node contains only data objects of a singular data type. This improved homogeneity necessarily increased the overall number of nodes since a heterogeneous node from the original SOM topography may be split into multiple homogeneous nodes based on the number of unique data types existing on the original node.

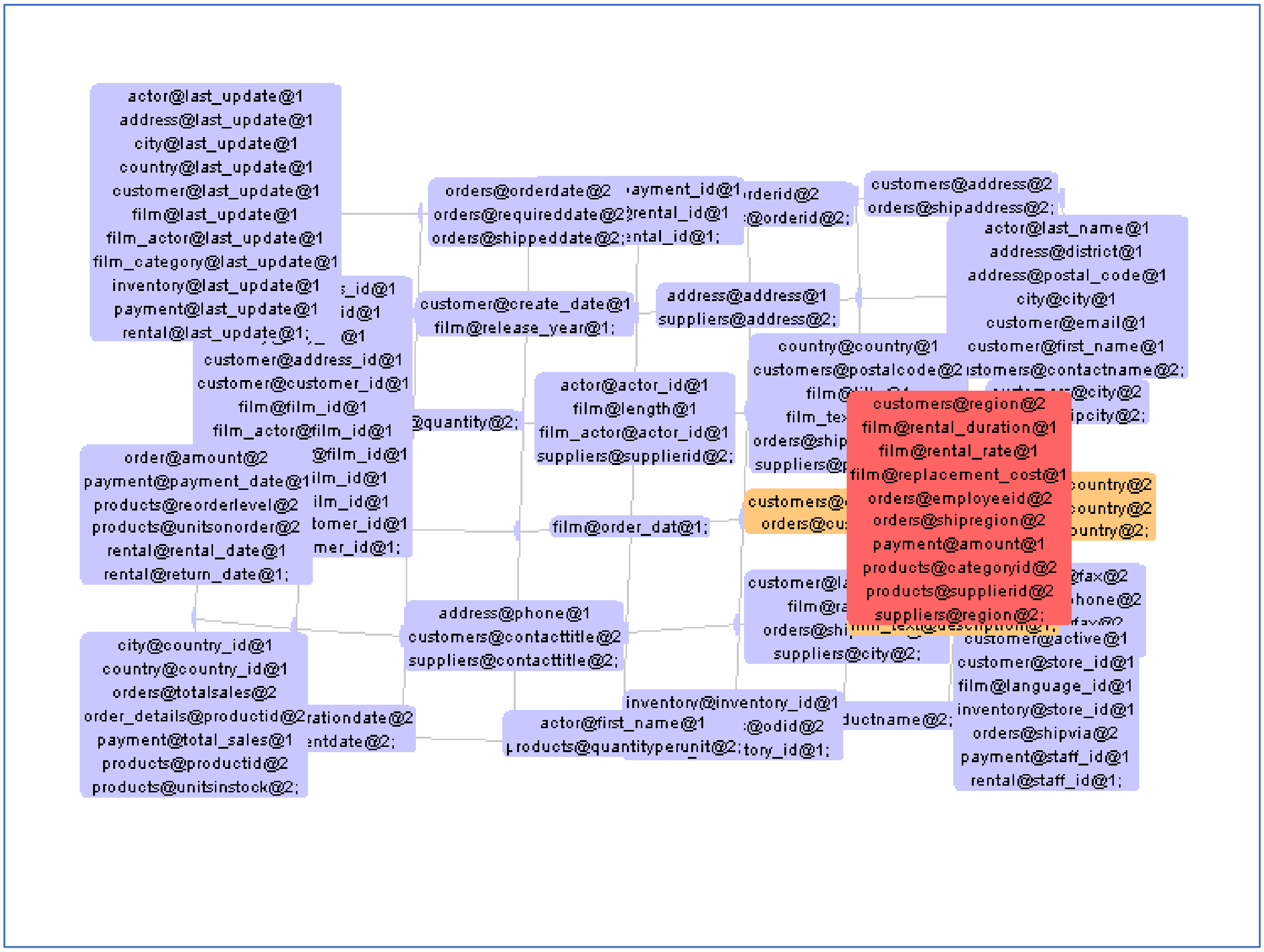


Figure 4: SOM-based Visualization map without post-processing

Figure 4 shows a visualization result of a Self-Organizing Map. The data is represented by a 2-dimensional grid with data objects grouped and assigned to nodes based on results from the SOM-based clustering discussed earlier in this paper. Each node is labeled with none, one, or more data entities. The selected node shown in red contains 10 data entities identified to be similar for schema matching by the SOM algorithm, while the orange nodes are its neighboring clusters.

In Figure 5, you can see the indicated node from Figure 4. The labeling of each of the 10 data objects on the node uses the format column@table@datasource. These entities should be matched together based on these results. Intuitively it can be seen that some entities (columns) are of different data types. Therefore, the proposed post-processing algorithm will be used to separate them based on content data types which results in generating new sub-clusters. Every sub-cluster will be of a singular specific data type. The advantage of this approach is that the original SOM topology is not modified; i.e., two neighboring nodes are suggested to have content based similarities without the recommendation to be matched together. In addition, every sub-cluster of a particular data type will be colored differently, further facilitating initial visual data exploration without having preliminary knowledge of the original schemas.

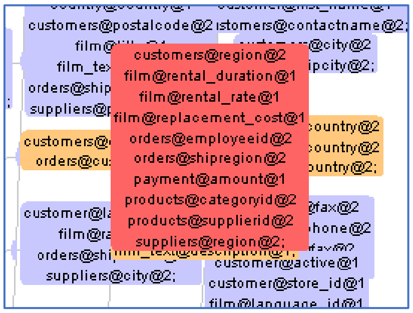


Figure 5: A Zoomed Node without post/processing

After application of the post-processing algorithm, the new multi-node topology is then visualized as an overlay upon the original two dimensional SOM map. Figure 6 shows a visualization result of the proposed post-processing method. For conformity, the same SOM Training presented in Figures 4 and 5 are also presented after post-processing in Figures 6 and 7. Here, you can see the original two-dimensional SOM map topology underneath the enhanced layer of labeled nodes. The number of individual nodes visible in the upper layer has increased, yet all new nodes are visually bound to the nodes in the original SOM from which they were derived.

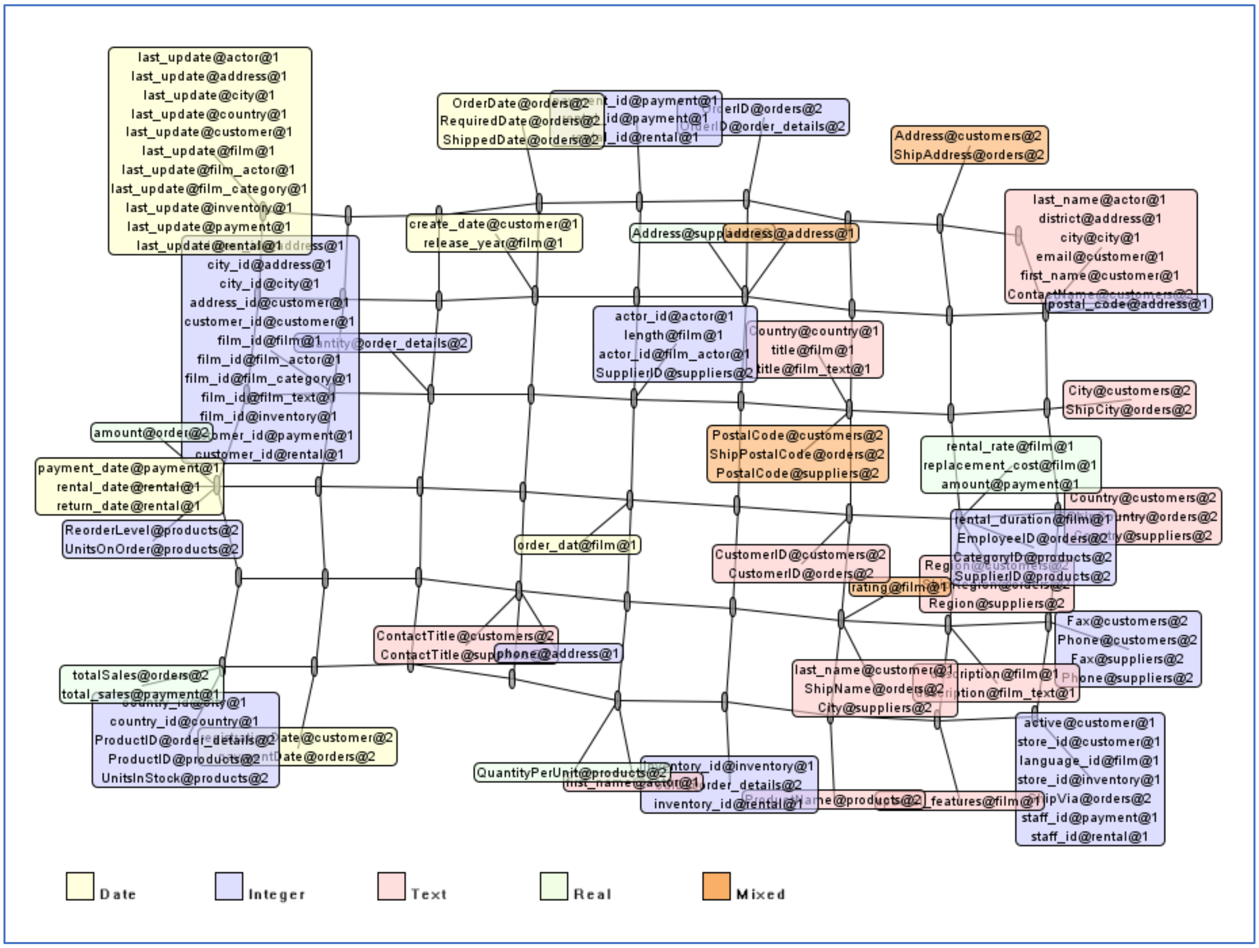


Figure 6: Visualization after Post-Processing Clustering based on Data-Types

Coloration has been introduced to further enhance the visual delivery of information to the user. Each unique color represents a unique data type. In the case of Figure 6, five distinct colors represent each of Date, Integer, Text, Real and Mixed (text and numbers i.e. address) data types. The user, looking at the visualization, can readily identify the data type for any of the enhanced nodes.

In Figure 7, the highlighted node from Figure 4 and Figure 5 has been isolated. Here you can see that each of the original 10 data objects assigned to the original node have been broken apart into three enhanced nodes containing four data objects of type Integer, three data objects of type Text, and three data objects of type Real. The three new (and now smaller) nodes remain attached to the original SOM topology for reference. These results are not only more accurate but also makes the schema matching/mapping tasks much simpler and faster through the enhance visualization tool.

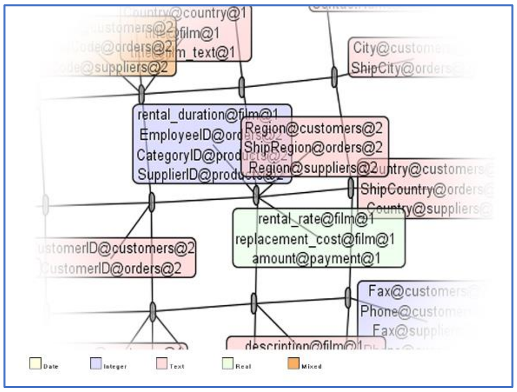


Figure 7: Post-Processed Visualization of Selected Node

I. CONCLUSIONS

This paper presented a visualization tool for automatic schema matching using a pure content-based approach. Self-Organizing Maps, an unsupervised clustering algorithm, was used for data clustering. The resulting map topology is very convenient for exploration of entities relationship originating from different repositories. In addition, a post processing algorithm was proposed for results enhancement by improving the precision. The advantage of this approach is that the schema matching results are improved in accuracy while preserving the original SOM topology. The clustering accuracy improvement is due mainly to the precision enhancement because of a stronger homogeneity of the post-processed nodes since each node contains data objects of a singular data type. In addition, every sub-cluster of specific type will be colored different to further facilitate visual data exploration and accelerate the schema matching process because a unique color represents a specific data type. The user, looking at the visualization, can readily identify the data type for any data entity on the SOM map which makes the schema matching process, simpler, faster, and more accurate.

References

[1] 1. Rahm, Erhard. The case for holistic data integration. s.l. : In East European Conference on Advances in Databases and Information Systems, Springer, pp. 11-27, 2016.

[2] 2. Anam, S., Kim, Y. S., Kang, B. H., & Liu, Q. Adapting a knowledge-based schema matching system for ontology mapping. s.l. : In Proceedings of the Australasian Computer Science Week Multiconference (p. 27). ACM., 2016. pp. 334-350.

[3] 3. Bernstein, P. A., Madhavan, J. , Rahm, E. Generic Schema Matching, Ten Years Later. s.l. : In PVLDB, Vol.4, No. 11, pp. 695-701, 2011.

[4] 4. Junchi, Y., et al. A short survey of recent advances in graph matching. Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval. ACM : s.n., 2016.

[5] 5. Cormack, G. V., & Grossman, M. R. Scalability of continuous active learning for reliable high-recall text classification. s.l. : In Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, ACM, pp. 1039-1048, 2016.

[6] 6. Russell B., Yin H. and Allinson N. M. Document Clustering Using the 1 + 1 Dimensional Self-Organising Map. s.l. : Lecture Notes in Computer Science, Vol. 2412, Intelligent Data Engineering and Automated Learning — IDEAL, pp. 167-174, 2002.

[7] 7. O. A. Mehdi, et al. Exploring Instances for Matching Heterogeneous Database Schemas Utilizing Google Similarity and Regular Expression. s.l. : Computer Science & Information Systems, Vol. 15, No. 2 , 2018.

[8] 8. Yang, Y., Chen, M., and Gao, B. An Effective Content-Based Schema Matching Algorithm. s.l. : In Proceedings of the 2008 International Seminar on Future Information Technology and Management Engineering (FITME '08), Washington, DC, USA, pp. 7-11, 2008.

[9] 9. Bourennani, F., Pu, K. Q., and Zhu, Y. Visual Integration Tool for Heterogeneous Data Type by Unified Vectorization. s.l. : Proceedings of the 10th IEEE International Conference in Reuse and Integration (IRI'09), Las-Vegas, USA, pp. 132-137,, 2009.

[10] 10. He, B., & Chang, K. C. C. Statistical schema matching across web query interfaces. s.l. : In Proceedings of the 2003 ACM SIGMOD international conference on Management of data, ACM, pp. 217-228, 2003.

[11] 11. Nikolov, A., Uren, V., Motta, E., de Roeck, A. Overcoming Schema Heterogeneity between Linked Semantic Repositories to Improve Coreference Resolution. s.l. : The Semantic Web, Lecture Notes in Computer Science, Vol 5926, Springer Berlin / Heidelberg, ISBN: 978-3-642-10870-9, pp. 332-346, 2009.

[12] 12. Nargesian, Fatemeh, et al. Table union search on open data. s.l. : Proceedings of the VLDB Endowment Vol. 11, No. 7, pp. 813-825, 2018.

[13] 13. Baeza-Yates, R., and Ribeiro-Neto, R. Modern Information Retrieval. s.l. : Addison Wesley Longman, 1999.

[14] 14. Sebastiani, F. Machine learning in automated text categorization. s.l. : ACM Computing Surveys, Vol. 34, no 1,, 2002. pp. 1-47.

[15] 15. Fradkin, D., Madigan, D. Experiments with Random Projections for Machine Learning. s.l. : Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, D.C, USA, 2003. pp. 517 - 522.

[16] 16. Lin, X. Map displays for information retrieval. s.l. : J. Am. Soc. Inf. Sci., Vol. 48, pp. 40-54, 1998. pp. 40-54.

[17] 17. F., Bourennani. https://archive.ics.uci.edu/ml/datasets/Northix. [Online] [Cited: 4 01, 2019.]

[18] 18. http://dev.mysql.com/doc/sakila/en/sakila.html, Accessed: Nov.-2009.

[19] 19. http://www.microsoft.com/downloads/details.aspx?familyid=06616212-0356-46a0, Accessed: Nov. 2009.

[20] 20. Shvaiko, P. and Euzenat, J. A Survey of Schema-Based Matching Approaches, . s.l. : Journal on Data Semantics, 4, 2007. pp. 146-171.

[21] 21. Mehdi, O.A., Ibrahim, H., and Affendey, L.S. Instance based Matching using Regular Expression,. s.l. : Procedia Computer Science, Volume 10, pp. 688-695, 2012.

[22] 22. Mahdi, A.M. and Tiun, S.,. Utilizing WordNet and Regular Expressions for Instance-based Schema Matching. s.l. : Research Journal of Applied Sciences, Engineering and Technology, Vol. 8, No. 4, pp. 460-470, 2014.

[23] 23. Engmann D., and Massmann S. Instance Matching with COMA++. s.l. : BTW 2007 Workshop, pp. 28-37, 2007.