ATLANTIS PRESS
Atlantis Highlights in Engineering (AHE), volume 2
International Conference on Industrial Enterprise and System Engineering (IcoIESE 2018)

Deduplication for Data Profiling using Open Source Platform

Margo Gunatama
Information System Department
School of Industrial and System Engineering, Telkom University
Bandung, Indonesia
margogunatama@student.telkomuniversity.ac.id

Tien Fabrianti Kusumasari
Information System Department
School of Industrial and System Engineering, Telkom University
Bandung, Indonesia
tienkusumasari@telkomuniversity.ac.id

Muhammad Azani Hasibuan
Information System Department
School of Industrial and System Engineering, Telkom University
Bandung, Indonesia
muhammadazani@telkomuniversity.ac.id

***Abstract*—Many companies still do not know of the importance of data quality for the company's improvement. Many companies in Indonesia, especially BUMN and Government companies have only single application with single database, which cause a problem related to duplication of data between columns, tables and applications when the application is integrated with other applications. This problem can be handled by doing the data preprocess, one of the data preprocess method is data profiling. Data profiling is the process of gathering information that can be determined by process or logic. The process of profiling data can be done with various tools both paid and open source tools, each has advantages both in performance and in data processing according to the desired case study. In this study, the main focus is on data analysis by conducting data profiling using deduplication method. The results of the profiling will be implemented in logical form in open source application and will do comparisons between open source applications.**

Keywords—data preprocess, data governance, levensthein distance

I. INTRODUCTION

Data is an important component in a company. Many new companies realize that data quality can lead to a profit both in terms of time and cost. A good data quality must be accurate, relevant, complete and easy to understand. Lack of data content management can occur a loss to the company, so now many companies are starting to look for a tool to help optimize the content for data quality to fit the desired company [1]. The results of Barchard & Pace analysis conducted in 2011 on 195 randomized people can prove that there is an average error in the data entry once without re-checking that is around 12.03, while when doing data entry twice that means can be done checking data i.e. around 0.34 [2].

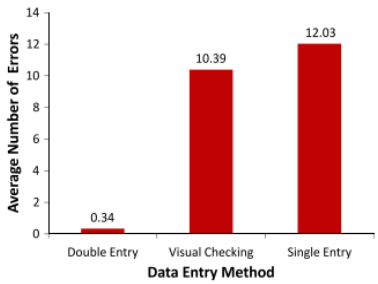

Fig. 1. Average data entry errors by humans [2]

Previous research by Tien and Fitria using an open source tool from Google, OpenRefine in one of the cases in BPOM where the data that were processed are Number of Edible Permits and Company Name. The business rule applied in this research are that the Permit Number can not be empty , must be unique to each entity and have similarities in alphanumeric patterns. The result of the research shows that the Permit Number has 70 patterns on 5000 rows of data. Duplication analysis needs to be combined with other elements because one production with a single license number can be duplicated if the factory location, volume and weight of the package are different [3].

Column Name	Duplicates (%)	Blank(%)	Cluster	Pattern
NIE	46	0	2	70
Company name	79	1	120	-

Fig. 2. Profiling analysis results using OpenRefine [3]

There is also previous research by Febri on profiling clustering data by implementing fingerprint algorithm using BPOM dataset and tested by comparative test with result of every algorithm implemented in each application have difference. The comparative results are that Pentaho found 602 lines from 4482 lines, Talend Open Studio found 502 lines from 4482 lines and Google OpenRefine found 562 lines from 4482 lines. The differences occurs in results between each application is because the

Google OpenRefine application can not be done one process that existed on Pentaho Data Integration [4].

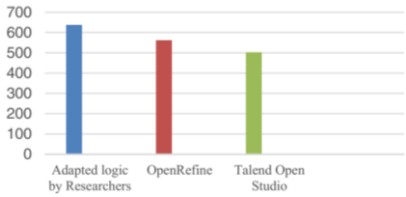

Fig. 3. Comparison result of clustering between Google OpenRefine and Talend Open Studio [4]

Profiling data over one column can be generalized to multiple columns. Multi-column profiling plays an important role in performing data cleansing [5]. For example in accessing frequent data disturbances in multiple column combinations [6]. Multi-column analysis is a statistical method and a data mining approach for generating meta data based on event values and values depedencies between attributes [5].

The pre-process data is the action taken before the data analysis process begins. The purpose of data analysis to find and present it well, data problems can be overcome by prevention using data analysis tools to generate acceptable data. The number of applications has the need for more than one time to perform preprocess data [7].

In this study, the number of companies in Indonesia, especially BUMN and Government companies that have single application with single database, thus when the application are integrated with another application, duplication of data both between columns, tables and databases occurs. Because each application has its own database which can lead to irrelevant data when determining the business rule of a company due to standardization errors of the company that will impact on poor quality data. Poor quality data will affect the data governance. Data governance are planning, oversight, and control over management of data and the use of data and data-related resources [8]. Data governance involves processes and controls to ensure that information at the data level raw alphanumeric characters that the organization is gathering and inputting is true and accurate, and unique. It involves data cleansing, data inaccurate, or extraneous data and data deduplication, to eliminate redundant occurrences of data [9].

In connection with this problem it is necessary that the data is clean due to the master data management where to perform data warehousing required data that is clean, unique and have a uniform standardization in one organization. With the number of tools that provide solutions, the data profiling are required for the better the data quality. This study uses an open source tool that refers to Google OpenRefine. Application logic that will be implemented in open source tools will be comparable to comparative decisions in determining open source tools.

II. Deduplication Algorithm for Profiling

Deduplication is designed to eliminate data redundancy in storage systems. Deduplication aims to streamline all types of stored data [10]. Deduplication has several techniques: Phonetic Matching Techniques, Pattern Matching Techniques and Dictionary-Based Matching Techniques. Phonetic Matching Techniques are a fast matching technique by finding most of the correct matches but have low precision due to the incorrect results produced, Phonetic Matching Techniques aims to convert strings into codes that are easy to understand verbally. Pattern Matching Techniques search by calculating the character spacing per character that are commonly used to estimate string matches. Dictionary-Based Matching Techniques use dictionaries to identify name variations that will be matched to all the variations contained in the dictionary. Each technique has a algorithm adopted on the technique, in Phonetic Matching Techniques adopts the Soundex, Daitch-Mokotoff Soundex, Metaphone, etc. for Pattern Matching Techniques adopts the Levenshtein Distance, N-gram, Jaro-Winkler, etc. [11].

Levenshtein distance is a technique of Pattern Matching Techniques which performs the number of operations (insert, delete and substitution) required to convert one string to another string [12], the greater the Levensthein Distance the more different the string will be operated. In general, Levensthein Distance spacing can be calculated between words of the same writing, so to compare two different names one name must be translated to another name [11].

		r	t	s	h	a	r	d
	0	1	2	3	4	5	6	7
r	1	0	1	2	3	4	5	6
i	2	1	1	2	3	4	5	6
c	3	2	2	2	3	4	5	6
h	4	3	3	3	2	3	4	5
a	5	4	4	4	3	2	3	4
r	6	5	5	5	4	3	2	3
d	7	6	6	6	5	4	3	**2**

Fig. 4. Levensthein Distance operation [11]

In the N-gram algorithm, conditional probability of the next word is calculated by *(n-1)-th* with the previous *(n-1)* word strings as states. When *n* increases by 1, the total number of parameters typically becomes tens of thousands of times, and there can be an explosion due to dimensionality that can't accommodate with increasing number of *n*. In other words, the performance of n-grams is highly dependent on the amount of text available and the accuracy to detect non-dictionary words is limited by the amount of text available [13].

Jaro-Winkler is a distance variation that allows better lexical measurement of similarity between two character strings and is particularly suitable for short

sequence comparisons like names or password. Jaro-Winkler is an increased function that returns the real number belonging to the interval [0, 1]. Since this 1 tends to metrics means there is a high similarity between the two strings as compared, if required, the metric tends to 0, there is no similarity [12].

Based on the Levenshtein Distance, N-gram and Jaro-Winkler algorithms, the researchers chose the Levenshtein Distance algorithm due to the lack of performance on N-grams that have limitations on the large datasets, in Jaro-Winkler, the data that has an excessive length will display results that don't accordingly, Levenshtein Distance has advantages both in the amount of data and data length although in the profiling process takes a little longer time. Fig. 4 is a process for finding Levensthein Distance with richard and rtshard examples with the result that the minimum distance between two concurrent one "t" is replaced by "i" and "s" replaced by "c". The process of applying Levensthein Distance on Pentaho Data Integration by using Fuzzy Match function.

III. METHOD

The research method used to find duplicate data between columns or tables or databases is divided by 3 stages, the same method performed with Febri method in the previous research [4]. The first stage is mapping the function between deduplication algorithm logic with Pentaho Data Integration function. The second stage is design and configuration functions used in Pentaho Data Integration and the last stage is by evaluation, analysis and comparison between Pentaho Data Integration results with Google OpenRefine.

First stage, mapping function by Pentaho Data Integration by analyzing the flow algorithm and customize the components in Pentaho Data Integration accordingly. The flow of algorithm can be seen in Fig. 5 where it has similarity of function with the components on Pentaho Data Integration. Deduplication method focus on standardization pattern of string and find duplicate data or unique data, so the components used in Pentaho Data Integration by researcher are the components that related to such functions such as related to function such as string operation, unique rows and fuzzy match.

Second stage is performed accordance to Fig. 5, starting by building transformation algorithms in the Pentaho Data Integration. Preparation of a master database, configuration, testing the connection and testing the data preprocess. The making of transformation is done gradually for each step according to deduplication algorithm and tested on every steps. If the test results of every step of the deduplication algorithm is in accordance with the existing transformation on Pentaho Data Integration then proceed to the final stage.

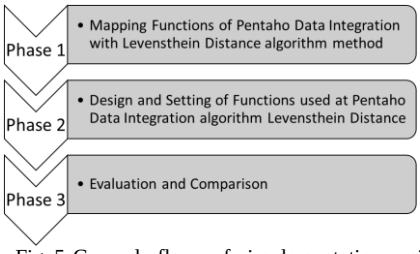

Fig. 5. General flow of implementation using Pentaho Data Integration

Final stage is to conduct comparative evaluation of the result from Pentaho Data Integration with the results from Google OpenRefine. The evaluation is carried out with the same amount of data and the same flow process. For this research the data used are from two connected databases but from a two different sources of data and is being processed using the same execution deduplication method.

IV. DEDUPLICATION IN PROFILING DATA WITH PENTAHO DATA INTEGRATION

Data profiling analysis between columns with deduplication method is using Levenshtein Distance algorithm that can be done with various applications, for example Pentaho Data Integration, the use or search based on Levenshtein Distance algorithm can be determined by string matching or in the Pentaho Data Integration is called Fuzzy Match [14].

Fig. 5 is the flow of the Levenshtein Distance search process, the Levensthein Distance process will be mapped according to the flow in the Pentaho Data Integration tools described in TABLE II.

TABLE I. Mapping Deduplication algorithm to Pentaho Data Integration Component

Deduplication Algorithm	**Pentaho Data Integration Component**
Receive data	Table input
Standardization Character	String operations
Specifies the pattern of each letter	Fuzzy match
Checking suitability between letters	
Display profiling of deduplication	

The implementation of the deduplication method logic to Pentaho Data Integration components can be seen in Fig. 6.

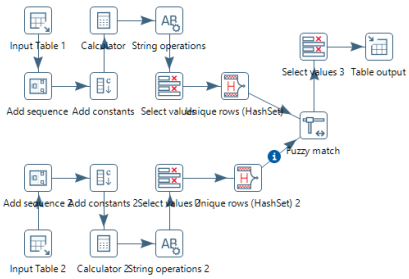

Fig. 6. Implementation of logic multi-column deduplication algorithm

Based on Fig. 5 and the selection of the components available on the Pentaho Data Integration, and the adjustment of the configuration with the algorithm flow. Settings of the components can be seen in TABLE II.

TABLE II. Implementation Levenshtein Distance algorithm on Pentaho Data Integration components

PDI Component	Function	Setting
Input Table	Specify column for data pre-process	SQL query Select
Add Sequence	Transformation counter to get sequence	-
Add constants	Give value for specify table 1 or table 2	TB1 or TB2
Calculator	Combine two streams become one output	A + B (sequence + constant)
String Operations	Trim the contents of particular column in order to be changed or to be in one format	Upper
Select Values	Transformation on what stream for the next process	-
Unique rows (HashSet)	Process transformation to be compare between field or row	-
Fuzzy Match	Lookup between main stream and lookup stream	Levenshtein algorithm
Table Output	Display output profiling to database	Connection MySQL

This implementation is done by using the dataset contained in the MySQL database so it can connect to Pentaho Data Integration application. The next is with the flow of Uniqerows (HashSet) so that the process can only be passed by unique data (no data redundancy) and the last is by doing a string matching process using the Fuzzy Match referring to the Levensthein algorithm flow and the final result will be inputted to the initial database by creating a new table.

V. EVALUATION AND DISCUSSION

The dataset used is a government agency dataset where there are tables 1 and 2. Table 1 is the master database of application X and Table 2 is the master database of application Y where these two applications will be compared by company name and company address. The business rule of the company is a unique address search or no duplication, because there are companies that have branches, after obtaining a unique address and then made a duplicate or duplicate company name search, search company name by doing search between tables in the name field companies in table 1 and table 2.

The results of the test of output comparisons between open source tools of Pentaho Data Integration and Google OpenRefine have a far differences in deduplication method, seen in Fig. 7. The search results of deduplication show a large differences due to Pentaho Data Integration search process using Unique Rows (HashSet) where only unique lines can continue the next process, while Google OpenRefine tools run duplication facet process without any characterization process before the deduplication process.

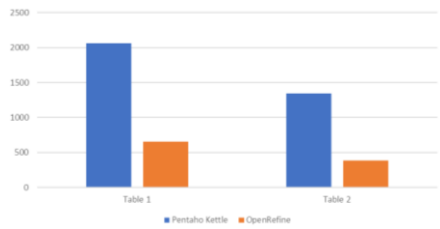

Fig. 7. Result comparison process deduplication using open source

The final result of the comparison can be found in the factory table that in terms of deduplication table 1 tools Pentaho Data Integration found 2059 lines, for tools Google OpenRefine found 656 lines, in table 2 tools Pentaho Data Integration found 1346 lines, for tools Google OpenRefine found 382 lines.

TABLE III. The final result of analysis of the use of logic on tools Pentaho Data Integration

Table Name	***Deduplication: single-column* (%)**	***Deduplication: multi-column* (%)**
Table 1 (4482 *rows*)	45%	27%
Table 2 (2890 rows)	46%	

The result of the final multi column analysis: deduplication using Pentaho Data Integration tools can be found that the problems in the tested dataset is an industry problem where a lot of dirty data is duplicated due to the absence of standard on the company. Deduplication multi-column performs checks between two tables, the results can be seen on TABLE III.

Differences in the number of duplications in Pentaho Data Integration applications with Google OpenRefine because of the process flow between Pentaho Data Integration with Google OpenRefine one of the process between tables that can't be done on Google OpenRefine, then the other process is the difference path algorithm, the Pentaho Data Integration each configuration can be selected using each algorithm, the Google OpenRefine algorithm selection based on the OpenRefine language or called GREL is a language that can only be run on the Google OpenRefine application. Lastly on Pentaho Data Integration company address search using Uniquerows where the search process is based on unique or non-redundant data while in Google OpenRefine the search process uses a duplicate facet which only summarizes the data and instantly displays duplicate results [15].

VI. CONCLUSION

Transformation by performing data pre-process with data profiling and also comparing result produced by the implementation of deduplication either to one column or to multi column with the result produced while using Google OpenRefine are so different, thus makes Levensthein Distance algorithm is one important factor to enable many companies to understand the importance of the quality of data which can affect the business process and data governance going forward.

REFERENCES

[1] J. E. Olson, *Data Quality—The Accuracy Dimension*. Elsevier Science, 2003.

[2] K. A. Barchard and L. A. Pace, "Preventing human error: The impact of data entry methods on data accuracy and statistical results," *Comput. Human Behav.*, vol. 27, no. 5, pp. 1834–1839, 2011.

[3] T. F. Kusumasari, "Data Profiling for Data Quality Improvement with Openrefine," 2016.

[4] F. Dwiandriani, "Fingerprint Clustering Algorithm for Data Profiling using Pentaho Data Integration," pp. 358–362, 2017.

[5] Z. Abedjan, L. Golab, and F. Naumann, "Profiling relational data: a survey," *VLDB J.*, vol. 24, no. 4, pp. 557–581, 2015.

[6] T. Dasu, J. M. Loh, and D. Srivastava, "Empirical glitch explanations," *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discov. data Min. - KDD '14*, pp. 572–581, 2014.

[7] E. S. Fazel Famili, W. M. Shen, R. Weber, "Data Pre-Processing and Intelligent Data Analysis," *Int. J. Intell. Data Anal.*, vol. 18, no. 6, pp. 1–29, 1997.

[8] P. Cupoli, S. Earley, D. Henderson, and Deborah Henderson, "DAMA-DMBOK2 Framework," p. 26, 2014.

[9] A. Yulfitri, "Modeling Operational Model of Data Governance in Government," 2016.

[10] A. Acronis and W. Paper, "How Deduplication Benefits Companies of All Sizes," pp. 2000–2009, 2009.

[11] A. Hassan, "Technique using Novel Modified Levenshtein Distance," pp. 204–209, 2015.

[12] H. Gueddah, A. Yousfi, and M. Belkasmi, "The filtered combination of the weighted edit distance and the Jaro-Winkler distance to improve spellchecking Arabic texts," *Proc. IEEE/ACS Int. Conf. Comput. Syst. Appl. AICCSA*, vol. 2016–July, pp. 1–6, 2016.

[13] Y. Ikegami, E. Damiani, and R. Knauf, "Flick : Japanese Input Method Editor using N-gram and Recurrent Neural Network Language Model based Predictive Text Input," 2017.

[14] G. Navarro, "A guided tour to approximate string matching," *ACM Comput. Surv.*, vol. 33, no. 1, pp. 31–88, 2001.

[15] F. Rabe, "Faceting," 2014. [Online]. Available: https://github.com/OpenRefine/OpenRefine/wiki/Faceting. [Accessed: 11-Jun-2018].