

A Binary Matrix Synthetic Data and Its Bi-set Ground Truth Generator

Mujiono Sadikin*

Faculty of Computer Science,
Universitas Mercu Buana, Jakarta Indonesia

Abstract— Due many reasons, such the privacy aspects, availability, and less flexibility in parameter adjustment, the real data set is becoming a main barrier in various researches. Facing to the difficulties to access the real data set, a synthetic dataset is a viable alternative as a complementary to a real data set. Bi-clustering to binary categorical data is one of the most recent challenging researches which are applied to many areas such: DNA expressions, tourism, commerce and marketing, event sports. This paper presents the useful algorithm to generate synthetic binary matrix data and its bi-set as the bi-clustering results ground truth. In generating this synthetic data, it uses some parameters includes: initial size, injected bi-set size, number of overlapping size and bi-set position. Some of the generated synthetic data sets are applied to two bi-set searching algorithms, Bibit and Bimax. The experiment result shows that the synthetic data set has capabilities to evaluate methods or algorithm relates to binary matrix data set.

Keywords— Synthetic dataset, Bi-set, Bi-clustering, Binary matrix, Match Score

I. INTRODUCTION

Synthetic data are very widely used by the research community in widespread domains such: privacy protection, health care, pattern recognition, data mining, economy and socials [1], [2], and information analysis as well [3]. There are many reasons why the demand for synthetic data in various researches is very high and its existence is crucial. We cannot fully depend on the real data since privacy reasons (i.e. to protect the privacy of the individual)[1]. The lack of real data availabilities which is match with required certain criteria is the other reason. Synthetic dataset can be useful in many cases, includes: to provide repeatable new method testing, to adjust any parameter for the testing, or to determine any characteristics data test required [2]. Given of those situations, a synthetic dataset is a viable alternative as a complementary to real dataset [3], [4]. In IT industry for example, the synthetic data set is required for: *regression testing*, *secure application development*, and *testing of data mining applications*. Regression testing is performed for testing enterprise applications, secure application development is needed to develop enterprise application in real condition, but it is actually not real, while the testing of data mining is developed to gauge whether data mining tools can discover those characteristics with known characteristics of the dataset.

Many researchers have published the study results regarding to synthetic data for many purposes. Some of them are generated because the privacy reasons [5], [6], [7], while the others are due the requirement of specific of dataset characteristics [8], [9], and [3]. In data mining are, especially in bi-clustering task, synthetic data set is used in almost all of the study to validate new proposed method, or to compare many methods regarding to their performance, such proposed in [10], [11], [12], [13].

In this study, we propose a simple synthetic data generator to provide a binary categorical matrix data set. The purpose of this kind of data is to validate certain method or algorithm to find a bi-cluster in such a matrix. The bi-cluster is represented by a bi - set which will be explained later in the next section. Some of the synthetic data resulted from the method are used to test two methods to find bi-sets contained in the matrix data.

When you submit your paper print it in one-column format, including figures and tables. In addition, designate one author as the “corresponding author”. This is the author to whom proofs of the paper will be sent. Proofs are sent to the corresponding author only.

A. Related Study

Synthetic Data

In response to the high demand of the synthetic data set, many researchers have proposed various methods to generate the data. Some of the methods are described in this section. Mathematical tools such probability distributions and linear transformation as a baseline of data synthetic generator is proposed by Pei Y as published in [8]. The usage of those tools provides a flexibility mechanism in controlling the parameters required. To simulate data test for *Statistics on Income and Living Conditions of Europe Union* (EU-SILC), A. Alfons et al [2] proposes such approach that's based on synthetic reconstruction and combinatorial optimization. Synthetic reconstruction normally involves sampling from conditional distributions derived from published contingency tabulations, while combinatorial optimization uses reweighting of existing micro data sets [2]. The interpolation mechanism to provide character-images data set is proposed in [9]. In this study, the authors use this interpolation to provide a character - image (font) through two steps. In the first, the parameter values are listed for each font and differences between the same parameter in each font were calculated. The second step is there are created nine interpolations, the difference was divided by 10 and one tenth of this value was added to or subtracted from the first font until the second font parameters is reached. Mark A.

Whiting et al [3] proposed synthetic data of threat stream to evaluate a Software of Information Analysis through some phase includes: to determine the aspects of the information analytics process to be tested, to define scenario development, to decide the type of dataset will be created, and to generate and to review the data set through a combination of automated and manual processes. Some methods and usages of synthetic data which are illustrated in this section show that the methods and synthetic may vary depend on a certain purpose.

Bi-clustering application and algorithm

Many researches have proposed many approaches and algorithms regarding to bi-clustering challenges. Some of them are described in this section, and two of those methods will be used to validate the synthetic dataset that is provided by this study. S. Kaiser et al describe some of bi-clustering method such: *bimax*, *CC bi-clustering*, *ISA*, *Plaid*, *Quest*, *Spectral*, *sv4d*, *Xmotifs*, and *CTWC*. While J.K. Gupta et al provide some useful codes in Matlab for various Bi-Clustering approach like: Cheng Church (CC) [14], *Bipartite Spectral Graph Partitioning Algorithm* [15], *Bimax* [10], *FLoc* [16], and *OPSM (Order Preserving Sub Matrix)* [17]. Bozdag et el [18] to do a comparison of *HARP*, *CPB (Correlated Pattern Bicluster)*, and *OPSM (Order Preserving Sub Matrix)* algorithms.

The application of CC Bi-clustering algorithm for corpora is proposed by Aljeber et al, [19]. In the study, a certain matrix is performed to represent documents as its row and the term of words as its column. The author reported that compared with HAC and K-Means algorithm, the accuracy of Bi-clustering is better, but it is not too stable. In the experiment applied to gene of yeast, mouse, and drosophila as reported by Bozdag. et al [18] among of CC, HARP, and CPB, CPB algorithm gives the best result in its capabilities to identify the similarities of local pattern and pattern shifting finding. The improvement to bi-clustering basic algorithm is also studied by Kaiser et al, Wei Shen et al [20], and Mihee Lee et. al [13]. In [15], the residue of CC bi-clustering algorithms is improved by the using of GA (Genetic Algorithm) model, while in [20] it is performed a reduction of high dimensional matrix with *SSVD (Sparse Singular Value Decomposition)* method.

Bi-clustering to binary matrix

As the synthetic dataset generator proposed in this study is applied for binary matrix to find certain bicluster, in this section we introduce a brief of this blustering aspect. The objective of bi-clustering applied to binary matrix is to find sets of bi-set.

Definition

Let $S = \{s_1, s_2, \dots, s_m\}$ is a set of sample and $G = \{g_1, g_2, \dots, g_n\}$, is a set of attributes/properties

Bi-Clustering Objective is:

To find a set (K) of Bi-Set $B = \{b_1, b_2, \dots, b_K\}$, $b_i = (S_i, G_i)$, $S_i \subseteq S$, $G_i \subseteq G$, is a set of sample s_i that contain or are influenced by a set of properties G_i . b_i is defined as Boolean vector:

$$\langle s_i \rangle, \langle g_i \rangle = \langle s_{i1}, s_{i2}, \dots, s_{im} \rangle, \langle g_{i1}, g_{i2}, \dots, g_{in} \rangle,$$

$$s_{ik} = 1, \text{ if } s_k \in S_i, \text{ else } s_{ik} = 0$$

$$g_{ik} = 1, \text{ if } g_k \in G_i, \text{ else } g_{ik} = 0$$

As an example to the definition above, let there is a binary matrix as below:

	g1	g2	g3	g4	g5
s1	1	1	0	1	0
s2	0	0	1	0	1
s3	0	1	1	1	0
s4	0	1	1	0	1
s5	0	1	1	0	1
s6	1	1	0	1	0

Fig. 1 Binary Matrix M

Then, the Bi-set (s) as the result of *Bi-Clustering* process are:

$$b_1 = \langle s_4, s_5 \rangle \langle g_2, g_3, g_5 \rangle = \langle 0, 0, 0, 1, 1, 0 \rangle \langle 0, 1, 1, 0, 1 \rangle$$

$$b_2 = \langle s_1, s_6 \rangle \langle g_1, g_2, g_4 \rangle = \langle 1, 0, 0, 0, 0, 1 \rangle \langle 1, 1, 0, 1, 0 \rangle$$

....

....

$$b_K = \dots\dots\dots$$

The application of bi-clustering, especially for binary categorical data, covers many areas such: marketing, tourism, or event sport [21].

The most popular applications is in biological data, [14], [22]–[24]. To overcome the bi-set finding in binary categorical data, many approaches and algorithms have been proposed. Pensa et al [25] proposed a framework to compute a bi-partition (bi-set) from a collection of local pattern which is represented by binary matrix. The other two approaches, Bibit and Bimax, are presented in this section since both of them will be used to validate the synthetic data. Bit-Pattern Bi-clustering Algorithm (BiBit) is proposed by Rogriguez et al [11] and Bimax is studied by Prelic et al. [10]. BiBit approach consists of two steps: first is encoding, and the second is bi-set searching which its work mechanism is based on the similarities of binary number and binary matrix. The encoding is performed to encode (convert) binary number to decimal number. And the bi-set searching is executed as follow: Every two rows of binary matrix are used to perform the preliminary bi-set, if this preliminary bi-set is not contained yet in the bi-set finding list then the bi-set is listed in the set of Bi-Set list. Afterward, the rest of rows in the matrix are compared to the finding bi-set to find the other new bi-set. The other input of this algorithm, beside binary matrix B , is the minimum row and minimum column number which is predefined for each bi-set generated. In this study, we implement the Bibit algorithm in Matlab script.

The second approach will be used to evaluate our synthetic dataset is Bimax. The bi-set searching algorithm as proposed by Prelic et al is one of the most popular approach in this area and it acts as reference based model for comparison and evaluation of various bi-clustering method as described by [26], [27], [28], and [29]. In the earlier, this algorithm was used to search Bi-Set in gene expression data. A bicluster Bi-Set (G, C) correspondence with a subset of gene G , with $G \subseteq \{1, \dots, N\}$ which simultaneously corresponding to a subset of sample C , $C \subseteq \{1, \dots, m\}$. Or in other words a pair (G, C) is a sub matrix E with all of its element value is 1. Bimax algorithm work mechanism is based on *divide and conquers*. In this study, we use the Bimax tools box as are developed by Gupta [26].

B. Overview of Our Approach

The approach of the binary matrix synthetic dataset generator that is presented in the study is based on some parameters, such: the complexity of the data, the maximum size of the bi-set provided, and the density of the matrix element which its value is 1 (one). All of those parameters are injected manually to give the user a high flexibilities to set up the characteristics of the resulted dataset as they required. The detail explanation of our framework is presented in the section 2 of this paper.

C. Contribution and Organization

The main contribution of this study is to provide the binary matrix synthetic dataset generator algorithm which has high flexibilities in parameters adjustment. This kind of data set will be useful as data test for various research purposes in a wide range of real applications. The rest of this paper is organized as follows: Section 2 describes the approach and framework to overcome the challenges of the binary matrix synthetic data set preparation. Sample data set and its bi-set, and the approach algorithm are also described in the section. The experiments scenario is elaborated in section 3. Section 4 illustrates the evaluation mechanism that is used to validate the proposed approach. This study uses the *Match Score* as the evaluation parameter. Section 5 contains the explanation of the experiment results and its analysis, while section 6 explains the achievement, the prospects, and the shortcoming of this study. It is also discussed several potential exploration in the future study.

II. OBJECTIVE AND FRAMEWORK

A. Objective

The objectives of the binary matrix synthetic data set study are:

- To provide the dataset and the ground truth of the bi-set contained in the data,
- To evaluate the performance of the dataset by applying them as data test for two of the bi-clustering algorithm

B. Framework

Figure 2 below illustrates the framework of the synthetic data generator and its evaluation. The characteristic of the dataset is described by 4 attributes: its complexity, its matrix dimension, its rectangular size of bi-set contained in, and its overlapping maximum size of the bi-set. The data set complexities consist of two kinds, disjoint bi-set and overlapping bi-set. Both of them have two same attributes, and one attribute is specific for overlapping bi-set. The values of those attributes are predefined by user. Matrix dimension (*size*) parameter is clear; it is the size of row and column of the matrix as a rectangular matrix. The rectangular size of bi-set (M) is an initial size of bi-set which is defined as a rectangular sub matrix with all of the element value are 1 (one). The initial position of this sub matrix, upper-left starting point, is determined randomly by the system. The maximum number overlapping (O) is the value of row or column overlapping of two of the bi - set. The real value of overlapping is generated randomly with maximum value as the value which is predefined. The real value of maximum overlapping will affect the size of the matrix and maximum size of contained bi-set, since the value will be added to the rectangular bi-set and matrix size as well. The synthetic dataset generator will generate tree output: the dataset itself, the contained bi-set ground truth, and the density.

Those three values, afterwards, will be used in experimental step which will apply two bi-set searching algorithms to those datasets. To evaluate the performance of both of this algorithm, we use *Match Score* (MS) parameter which is explained later in the next section.

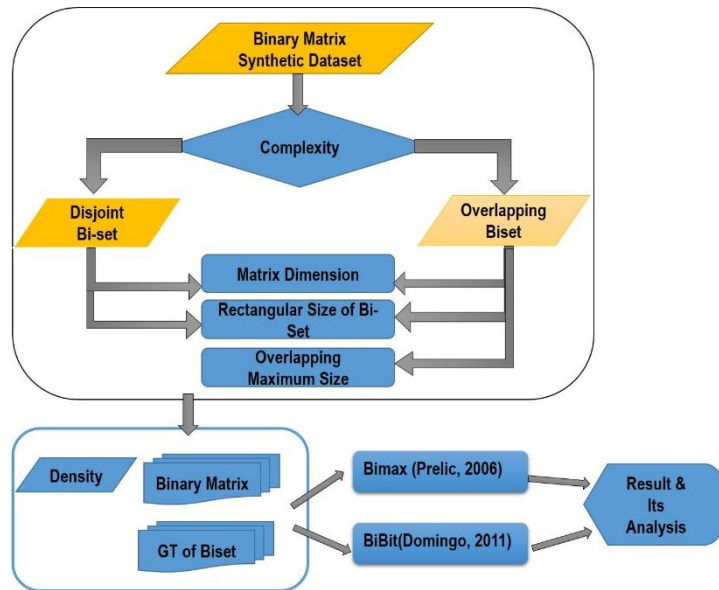


Fig. 2 The binary matrix synthetic data set study framework

C. Dataset and Its Bi-Set

To get a better understanding about the data set generator mechanism, dataset, and the bi-set, the figure 3 can be referred. This figure presents the example of dataset matrix generated and the bi-set which is contained in the data. The matrix is resulted by setting $size = 25$, $M = 5$, and $O = 3$. The bold line borders in the matrix are markers for each bi-set contained in the matrix. The lists of some bi-set candidates are presented in the figure 4.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0
6	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
7	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
8	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
11	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
12	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
17	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

Fig. 3 Sample of binary matrix dataset MS

b1		b2		b3		b4		b5		b6		b7		b8		b9		b10	
si	gj	si	gj	si	gj	si	gj	si	gj	si	gj	si	gj	si	gj	si	gj	si	gj
11	1	6	6	16	16	11	11	11	1	6	6	16	1	6	6	11	11	1	16
12	2	7	7	17	17	2	2	2	2	7	7	17	2	7	7	2	2	2	17
13	3	8	8	18	18	3	3	3	3	8	8	18	3	8	8	3	3	3	18
14	4	9	9	19	19	4	4	4	4	9	9	19	4	9	9	4	4	4	19
15	5	10	10	20	20	5	5	5	5	10	10	20	5	10	10	5	5	5	20
16	6	11	11	21	21	6	6	6	6	11	11	21	6	11	11	6	6	6	21
17	7			22	22	7	7	7	7	12	12	7	7	12	12	7	7	7	22
				23	23			8	8	13	13			16	16	13	13	8	16
								9	9	14	14			17	17	14	14	9	17
								10	10	15	15			18	18	15	15	10	18
								11	11	16	16			19	19	16	16	11	19
										17	17			20	20			21	20
														21	21			22	21
														22	22			23	22
														23	23				23

Fig. 4 Sample of bi-set collection

The bi-set, $b_i = \langle s_{ip}, \langle g_{jq} \rangle$ in this study follows two of the rules below:

1. $p > 1$, it is mean that there are a minimum of two samples that grouped in same bi - sets. No constraint for q
2. If there is another bi-set $b_j = \langle s_{jk}, \langle g_{jl} \rangle$, and $(\langle s_{ip} \rangle \subset \langle s_{jk} \rangle \text{ and } \langle g_{iq} \rangle \subset \langle g_{jl} \rangle)$ then b_i is not final bi-set since all of its samples and all of its properties are members of b_j .

Based on those two of rules $b_j = \langle 6,7,8 \rangle \langle 6,7,8 \rangle$, for example, is not the final bi-set since it does not satisfy rule number 2, all of its sample and properties member are the member of bi-set b_2 . Because the rule number 1, the bi-set b_5 is not final bi-set.

D. Algorithm

The figure 5 describes the main block of the algorithm. The algorithm consists of two blocks. The first block, line number 8 to line number 15, is used to provide the binary matrix and the rectangular sub matrix with its element value is one which also act as the rectangular ground truth bi-set. While the second block, line number 19 to line number 28, is used to create the overlapping ground truth bi-set. Input parameters of this function are the initial size of matrix, initial rectangular sub matrix that acts as ground truth as well, and maximum overlapping size. The outputs of this function are its density, list of bi-set ground truth, and binary matrix respectively. If the maximum overlapping number input is 0 (zero), then the second sub block is not needed anymore and the function finish, line 16- 18. The source code implementation of this algorithm is developed in Matlab R2010, and will available at <https://moedjionosadikin.wordpress.com/researchmaterial/> (will be uploaded when this paper is accepted)

```

1. function f_createSynthData(size_,M_,O_)=> [Density, Bi-setGT, MatrixSample]
   %Create overlap, random position, size of rectangular bi-set M_>1, random
   % with maximum O_ of overlap;
   %if O_ = 0, bi-set in binary matrix output are disjoint each other
2. nb <= floor((size_ / M_))
3. MatrixSample <= zeros(size_+ O_, size_+ O_)
4. R <= [] %List of row position of bi-set
5. C <= [] %List of column position of bi-set
6. Bi-setGT <= [] % List of Bi-set
7. Density <=0
8. for i=1 to nb
9.     find random r that is not in R, includes r in R
10.    find random k that is not in C, includes k in C
11.    find random n lte O_
12.    MatrixSample[r..r+M_+n-1][k..k+M_+n-1]<= 1
13.    assign bi-set bi: si= <r..M_+n-1>, gi=<k..M_+n-1>
14.    Density <= (M_ + n)^2
15.    Bi-setGT = Bi-setGT ∪ bi
16. If (N==0)
17.    Output [Density/( size_+ O_)^2, Bi-setGT, MatrixSample]
18.    return
   %To create Bi-set GT, which intersect between bi-set in Bi-setGT
19. for y=1 to |Bi-setGT| - 1
20.    Y<=Bi-setGT[y]
21.    for z=y+1 to |Bi-setGT|
22.        Z<=Bi-setGT[z]
23.        if |Y.si ∩ Z.si| > 1 %bi-set must satisfy number of row > 1
24.            Create NewBi-set <= < Y.si ∩ Z.si > <Y.gi ∪ Z.gi>
25.            Bi-setGT <= Bi-setGT ∪ NewBi-set
26.        if Y.gi ∩ Z.gi != ∅
27.            Create NewBi-set <= < Y.si ∪ Z.si > <Y.gi ∩ Z.gi>
28.            Bi-setGT <= Bi-setGT ∪ NewBi-set
29. Output [Density/( size_+ O_)^2, Bi-setGT, MatrixSample]

```

Fig. 5 The algorithm of synthetics data generator

III. EXPERIMENT SCENARIO

The summary of experiment scenario is presented as figure 6. Based on contained bi-set characteristics, the synthetic dataset are divided into two groups: disjoint bi-set and overlapping bi-set. The disjoint bi-set contains four parameters that are matrix size, rectangular bi-set size, starting position of the bi-set, and its density. While the overlapping bi-set have two addition parameters, those parameters are the maximum number of overlapping and maximum rectangular bi-set size, which is the substitution of the rectangular bi-set size of the disjoint dataset. The first group of dataset consists of 7 variants; each variant is represented by one data set. While the second group consists of 18 variants and for each variant, there are generated 5 members of the dataset which are provided randomly for its starting position of bi-set and for its maximum number of overlapping. For all those 97 members dataset, afterward we apply two bi-set searching methods as is explained in section 1.1.

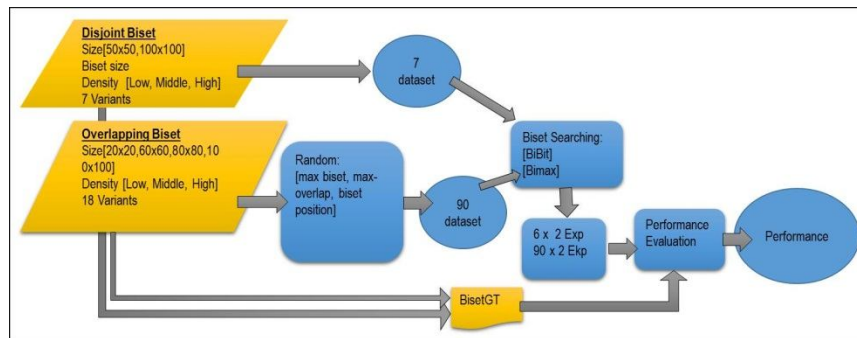


Fig. 6 Experiment Summary Scenario

IV. EVALUATION

The synthetic dataset is used as binary matrix input to two bi-set searching algorithms. To evaluate the resulted bi-sets we use *Match Score/MS* as used by Prelic et al [10] to evaluate Bimax algorithm and Domingo et al [11] to evaluate Bibit algorithm performance.

Definition

Referred from [10], MS is utilized as evaluation parameter to binary matrix as gene expression.

Let $P_1, P_2 \subseteq \{1, \dots, n\}$ are two sets of gen. The MS P_1 and P_2 is defined as a function:

$$S_G(P_1, P_2) = |P_1 \cap P_2| / |P_1 \cup P_2|$$

This score characterizes the correspondence between two of gen sets. This MS value is symmetric, i.e. $S_G(P_1, P_2) = S_G(P_2, P_1)$, and its value is ranging from zero (both of the sets are disjoint) to 1 (one) when both of the genes are identical. The MS in the study is modified by to include the attributes (properties) as:

Let there are two bi-sets $b_1 = (S_1, G_1)$, $S_1 \subseteq S$, $G_1 \subseteq G$ and $b_2 = (S_2, G_2)$, $S_2 \subseteq S$, $G_2 \subseteq G$, the *Match Score* of b_1 subject to b_2 is :

$$Sc(b_1, b_2) = |b_1 \cap b_2| / |b_1 \cup b_2|$$

$$= 1/2 \{ (| \langle s_{11}, \dots, s_{1m} \rangle \cap \langle s_{21}, \dots, s_{2m} \rangle | / | \langle s_{11}, \dots, s_{1m} \rangle \cup \langle s_{21}, \dots, s_{2m} \rangle |) + (| \langle g_{11}, \dots, g_{1n} \rangle \cap \langle g_{21}, \dots, g_{2n} \rangle | / | \langle g_{11}, \dots, g_{1n} \rangle \cup \langle g_{21}, \dots, g_{2n} \rangle |) \}$$

The MS two sets of bi-set B_1, B_2 , *MS* B_1 subject to B_2

$$Sc(B_1, B_2) = \frac{\sum_{b_1 \in B_1} \arg \max_{b_2 \in B_2} (Sc(b_1, b_2))}{|B_1|}$$

V. RESULTS AND ANALYSIS

Started with the simple data set one, implanted disjoint bi-set, the execution time and MS performance of both of those algorithms are presented as figure 7. The format data set DisjNxn_M means this data set matrix size is N row * N column and its maximum rectangular size of bi-set contained in this data is M. The actual size is generated randomly. The required inputs of minimum row and minimum column to the Bibit algorithm are 2. Both of these algorithms achieve the best results, i.e. MS = 1, for all of seventh variant data set. The execution time is slightly different. The differences are caused the differences of execution mode. While the Bibit algorithm is recoded in Matlab function and it is run in a script execution mode, the Bimax is executed by to run the matlab bi-clustering MTBA tools box [26] which is in binary matlab-c interfacing (mex) file. The first one, beside to execute the computation itself, also proceed the IO process to the display. So, the performance of execution time is not too fair to compare between both algorithms.

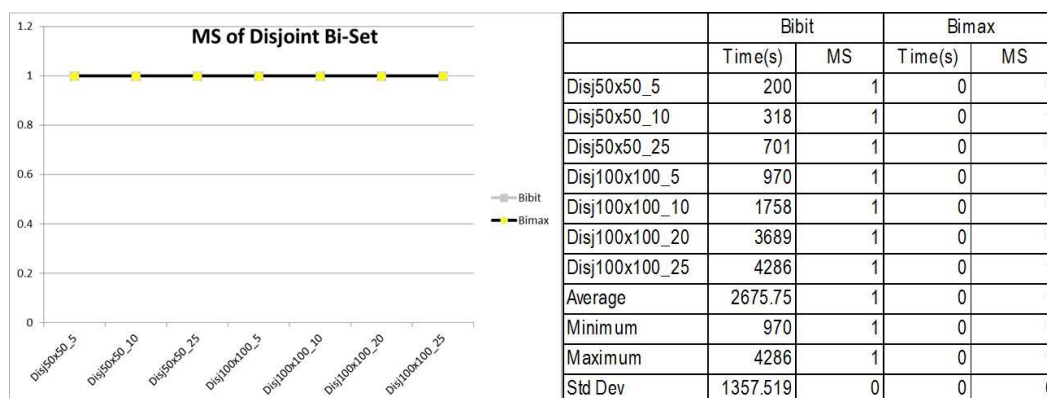


Fig. 7 MS Graph and table performance of disjoint bi-set

The results of the more complex implanted bi-set structure are illustrated as next figures. Figure 8 presents the average of each variant data set and average of all those averages. Each group kind of data set is characterized by its size, thus the four series of the graph represents the difference in size. Each size group consists of various numbers of data set member ranges from 2 to 6. And for each variant of data set, it is generated their 5 members randomly. The format data set DNxN_M_O means its size is N rows by N columns, its maximum rectangular bi-set is M, and its maximum size of overlapping is O.

By average and for all cases, Bimax achieves better performance. In some cases, the Bibit could not provide any bi-set since it is constrained by the requirement of minimum row and minimum column of the found bi-set that is allowed to include in the final set of bi-set. Table 1 shows one of detail results which presents some cases that Bibit could not find any bi-set, shown in yellow shading. Table 2, in the other side, illustrates the performance of Bibit and Bimax for each individual dataset members.

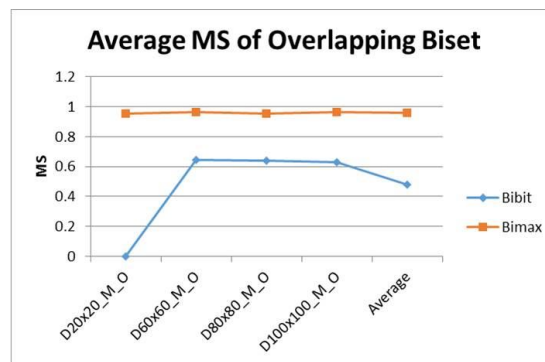


Fig. 8 The average MS performance

TABLE I - THE PERFORMANCE OF D80X80_M_O

	Bibit		Bimax	
	Time(s)	MS	Time(s)	MS
Exp1S80_M4_O2	1356	0.65194	0	0.95221
Exp2S80_M4_O2	1273	0.63104	0	0.9796
Exp3S80_M4_O2	1147	0.61625	0	0.93763
Exp4S80_M4_O2	1103	0.66401	0	0.92168
Exp5S80_M4_O2	1588	0.61657	0	0.96032
Exp1S80_M5_O3	6		0	0.94845
Exp2S80_M5_O3	6		0	0.92815
Exp3S80_M5_O3	6		0	0.91617
Exp4S80_M5_O3	0		0	0.97367
Exp5S80_M5_O3	0		0	0.95435
Exp1S80_M10_O8	4882	0.65906	0	0.95541
Exp2S80_M10_O8	4167	0.65021	0	0.9298
Exp3S80_M10_O8	6218	0.65547	0	0.94855
Exp4S80_M10_O8	83406	0.66983	0	0.95442
Exp5S80_M10_O8	5859	0.38301	0	0.8773
Exp1S80_M20_O18	9556	0.75218	0	0.95719
Exp2S80_M20_O18	10167	0.75957	0	0.96609
Exp3S80_M20_O18	9265	0.70817	0	0.92074
Exp4S80_M20_O18	84631	0.69101	0	1
Exp5S80_M20_O18	9193	0.50632	0	0.89122
Exp1S80_M40_O35	21		0	1
Exp2S80_M40_O35	21		0	1
Exp3S80_M40_O35	20		0	1

Exp4S80_M40_O35	0		0	1
Exp5S80_M40_O35	0		0	1
Average	9355.64	0.64098	0	0.95492
Min	0	0.38301	0	0.8773
Max	84631	0.75957	0	1
Std Dev	22287.7	0.09002	0	0.034

TABLE II - THE PERFORMANCE OF D60X60_M_O

	Bibit		Bimax	
	Time(s)	MS	Time(s)	MS
Exp1S60_M4_O2	5.6969986	0.8637931	0	0.94689
Exp2S60_M4_O2	5.8420029	0.8787356	0	1
Exp3S60_M4_O2	5.0270004	0.8643678	0	0.94042
Exp4S60_M4_O2	5	0.866092	0	0.93622
Exp5S60_M4_O2	5	0.8729885	0	1
Exp1S60_M5_O3	3.0820007	0.874353	0	0.9418
Exp2S60_M5_O3	3.2060013	0.8492495	0	0.90526
Exp3S60_M5_O3	2.8619995	0.8596014	0	1
Exp4S60_M5_O3	3	0.8566253	0	0.88628
Exp5S60_M5_O3	3	0.8564959	0	1
Exp1S60_M10_O8	1.1389999	0.8171977	0	0.94109
Exp2S60_M10_O8	1.060997	0.8232946	0	0.928
Exp3S60_M10_O8	1.0299988	0.9119706	0	0.94221
Exp4S60_M10_O8	2	0.8480963	0	0.94481
Exp5S60_M10_O8	1	0.8694731	0	0.91761
Exp1S60_M15_O12	1.1320038	0.8477788	0	0.96087
Exp2S60_M15_O12	0.4760056	0.8558045	0	0.95409
Exp3S60_M15_O12	0.5540009	0.8683095	0	1
Exp4S60_M15_O12	1	0.9371654	0	0.9132
Exp5S60_M15_O12	1	0.8542082	0	1
Exp1S60_M20_O18	1.7750015	0.8406955	0	1
Exp2S60_M20_O18	2.737999	0.9837354	0	1
Exp3S60_M20_O18	1.6289978	0.8323232	0	1
Exp4S60_M20_O18	1	0.8442159	0	1
Exp5S60_M20_O18	3	0.9005848	0	1
Average	2.490000305	0.867086223	0	0.96235
Min	0.476005554	0.817197712	0	0.886279
Max	5.842002869	0.98373538	0	1
Std Dev	1.654339338	0.034849095	0	0.036495

VI. CONCLUSION & FUTURE WORKS

One of the challenges in synthetic data set preparation has been trying to overcome in the study. The synthetic data set which could be generated by our proposed algorithm represents various characteristics and those characteristics are good enough to be used as a data test to validate methods or algorithm in binary categorical data bi-clustering. The proposed algorithm in this study also offers flexibilities for researchers in determining the requested data set profiles.

Not only is data set provided by the algorithm, but also ground truth bi-set as well. By providing such those advantages, the algorithm has capabilities to contribute in the future various researches in this area.

In the future study, we will explore still unknown potential improvements by to compare and to explore some real (experimental) binary matrix data set. Currently we are exploring the drug – drug interaction problem which the data set is a binary matrix as a representation of correspondence between drugs name and drugs component. We also are studying a new proposed method to find set of bi-set in the binary categorical data matrix.

ACKNOWLEDGEMENTS

The authors would like to thank to Mr. Anis Cherid, The Computer Science of Universitas Mercu Buana faculty member, who has supported to review and to correct the writing of this paper.

REFERENCES

- [1] V. Ayala-rivera, P. Mcdonagh, and T. Cerqueus, “Synthetic Data Generation using Benerator Tool,” 2013.
- [2] A. Alfons, P. Filzmoser, S. Kraft, M. Templ, and F. Cs--, “Generation of synthetic EU-SILC data and simulation Generation of Synthetic EU-SILC Data and,” no. November, 2009.
- [3] M. A. Whiting, J. Haack, and C. Varley, “Creating realistic, scenario-based synthetic data for test and evaluation of information analytics software,” in Proceedings of the 2008 Workshop on BEyond time and errors: novel evaluation methods for Information Visualization, 2008, no. c, p. 8.
- [4] L. Piras and G. Giacinto, “Synthetic pattern generation for imbalanced learning in image retrieval,” *Pattern Recognit. Lett.*, vol. 33, no. 16, pp. 2198–2205, Dec. 2012.
- [5] Z. Huang and P. Williamson, “A comparison of synthetic reconstruction and combinatorial optimisation approaches to the creation of small-area microdata,” *Dep. Geogr. Univ. Liverpool*, 2001.
- [6] S.-F. Chin, A. Harding, R. Lloyd, J. McNamara, B. Phillips, Q. N. Vu, and others, “Spatial microsimulation using synthetic small-area estimates of income, tax and social security benefits,” 2005.
- [7] N. Boggs, H. Zhao, S. Du, and S. J. Stolfo, “Synthetic Data Generation and Defense in Depth Measurement of Web Applications,” in *Research in Attacks, Intrusions and Defenses*, Springer, 2014, pp. 234–254.
- [8] Y. Pei and O. Zaïane, “A synthetic data generator for clustering and outlier analysis,” *Dep. Comput. Sci. Univ. Alberta ...*, 2006.
- [9] J. Nonnemaker and H. S. Baird, “Using synthetic data safely in classification,” in *IS&T/SPIE Electronic Imaging*, 2009, vol. 7247, p. 72470G–72470G.
- [10] B. Peter, A. Prelić, S. Bleuler, P. Zimmermann, A. Wille, P. Bühlmann, W. Gruissem, L. Hennig, L. Thiele, E. Zitzler, A. Prelic, A. Wille, and P. B., “Comparison of Biclustering Methods: A Systematic Comparison and Evaluation of Biclustering Methods for Gene Expression Data,” *Bioinformatics*, vol. 22, no. 9, pp. 1122–112930, 2006.
- [11] D. S. Rodriguez-Baena, A. J. Perez-Pulido, and J. S. Aguilar-Ruiz, “A biclustering algorithm for extracting bit-patterns from binary datasets,” *Bioinformatics*, vol. 27, no. 19, pp. 2738–2745, 2011.
- [12] J. Gu and J. S. Liu, “Bayesian biclustering of gene expression data,” *BMC Genomics*, vol. 9 Suppl 1, p. S4, Jan. 2008.
- [13] M. Lee, H. Shen, J. Z. Huang, and J. S. Marron, “Biclustering via sparse singular value decomposition,” *Biometrics*, vol. 66, no. 4, pp. 1087–95, Dec. 2010.
- [14] Y. Cheng and G. M. Church., “Biclustering of expression data,” in *Proc. of the International Conference on Intelligent Systems for Molecular Biology*, 2000, pp. 93–103.
- [15] I. S. Dhillon, “Co-clustering documents and words using bipartite spectral graph partitioning,” *Proc. seventh ACM SIGKDD Int. Conf. Knowl. Discov. data Min. - KDD '01*, pp. 269–274, 2001.
- [16] J. Yang, H. Wang, W. Wang, and P. S. Yu, “An improved biclustering method for analyzing gene expression profiles,” *Int. J. Artif. Intell. Tools*, vol. 14, no. 05, pp. 771–789, 2005.
- [17] A. Ben-Dor, B. Chor, R. Karp, and Z. Yakhini, “Discovering local structure in gene expression data: the order-preserving submatrix problem,” *J. Comput. Biol.*, vol. 10, no. 3–4, pp. 373–384, 2003.
- [18] D. Bozdağ, A. S. Kumar, and U. V. Catalyurek, “Comparative analysis of biclustering algorithms,” *Proc. First ACM Int. Conf. Bioinforma. Comput. Biol. - BCB '10*, p. 265, 2010.
- [19] B. Aljaber, N. Stokes, J. Bailey, and J. Pei, “Document clustering of scientific texts using citation contexts,” *Inf. Retr. Boston.*, vol. 13, no. 2, pp. 101–131, Aug. 2009.
- [20] W. Shen, G. Liu, M. Zheng, Z. Li, and Y. Zhong, “A Novel Biclustering Algorithm and Its Application in Gene Expression Profiles □,” *J. Inf. Comput. Sci.*, vol. 9, no. 11, pp. 3113–3121, 2012.
- [21] S. Kaiser, “Biclustering: Methods, Software and Application,” *Ludwig-Maximilians, Munchen*, 2011.
- [22] S. C. Madeira and A. L. Oliveira, “Biclustering algorithms for biological data analysis: a survey,” *IEEE/ACM Trans Comput Biol Bioinform*, vol. 1, pp. 24–45, 2004.
- [23] C. Cano, L. Adarve, J. López, and a. Blanco, “Possibilistic approach for biclustering microarray data,” *Comput. Biol. Med.*, vol. 37, pp. 1426–1436, 2007.



- [24] S. C. Madeira and A. L. Oliveira, "A linear time biclustering algorithm for time series gene expression data," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2005, vol. 3692 LNBI, pp. 39–52.
- [25] R. G. Pensa, "A Bi-clustering Framework for Categorical Data," in In Proceedings of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD), 2005, pp. 643–650.
- [26] J. Gupta, S. Singh, and N. Verma, "MTBA: MATLAB Toolbox for Biclustering Analysis," in MTBA: MATLAB Toolbox for Biclustering Analysis, 2013, vol. 1, no. July, pp. 94–97.
- [27] D. S. Rodriguez-baena, A. J. Perez-pulido, and S. Jesus, "A biclustering algorithm for extracting bit – patterns from binary datasets," pp. 1–8, 2011.
- [28] H. C. Chen, W. Zou, Y. J. Tien, and J. J. Chen, "Identification of Bicluster Regions in a Binary Matrix and Its Applications," PLoS One, vol. 8, no. 8, 2013.
- [29] K. Benabdeslem and K. Allab, "Bi-clustering continuous data with self-organizing map," Neural Comput. Appl., vol. 22, pp. 1551–1562, 2013.