Novel Data Oriented Structure Learning Approach for the Diabetes Analysis

Adel THALJAOUI
Department of Computer Science and Information, College of Science at Zulfi
Majmaah University, Al-Majmaah 11952
Saudi Arabia

***Abstract*—Diabetes mellitus is considered a significant disease an ever rising epidemic. Accordingly this disease represents a worldwide public-health-crisis. Several classification techniques have been recently employed for diabetes diagnosis, however only few researches have been dedicated to facilitating its analysis based on knowledge representation using probabilistic modelling. Bayesian Network as a probabilistic graphical model is considered as one of the most effective techniques of classification. Bayesian Network (BN) is widely employed in several domains like risk analysis, medicine, bioinformatics and security. This probabilistic graphical model represents an effective formalism to reason under uncertainty. The construction of the BN model goes through two learning phases of structure and parameter. The first learning phase of BN skeleton has been assessed as complex problem (NP-hard problem). Accordingly, several methods have been introduced amongst which the score based algorithms that are considered as one of the most powerful methods of structure learning. In this paper, we introduce a novel algorithm based on graph theory and the information theory combination. The proposed algorithm called GIT algorithm for Parents and children detection for BN structure learning. In addition, we evaluate the obtained results and using the reference networks, we prove the efficiency of the proposed GIT algorithm in terms of accuracy. Furthermore, we apply our algorithm in a real field, especially for detecting the interesting dependencies which are useful for the diabetes analysis.**

***Keywords*—Classification; Bayesian Network; structure learning; score oriented approach; diabetes analysis**

I. Introduction

In this century, the diabetes mellitus represents a serious health problem [1] [2]. The International Diabetes Federation (IDF) reveals that by 2040 it is expected to have 642 million adults who are diabetic and during the next two decades, our world will attend an important increase of 10.4%. The estimated percentage of the undiagnosed diabetes is about 0.497 of all affected people where the highest values were discovered respectively in Africa (70), South-East of Asia (60) and regions of Western Pacific (54) [3]. Consequently, and considering the importance of these diagnostics, correct and rapid analysis for diabetes detection using an intelligent technique has been considered as a crucial necessity [1] [4] [5].

The BN is a classification technique based on the graphical representation mode and the probabilistic reasoning. This technique is deemed as consistent formalism for making a model for the complex systems [6]. This classification technique is included in the most extensively used category of probabilistic-graphical models [7]. Therefore, and because of its potent abilities in reasoning using graphical representation, the BN has been effectively applied in several research areas like image processing [8], risk analysis [9], medical diagnosis [10], image processing [11], bioinformatics [12], etc. The construction of the BN model consists of two learning stages of its structure and its parameter. The structure learning phase allows the specification of the dependency set between the variables (random). In fact, it permits to create a directed acyclic graph (DAG), which consists of nodes and edges representing the dependency relation between Parents and children nodes while the parameter learning stage allows these dependencies' quantification.

The aim of the first learning phase is to generate the optimal structure which is judged as an NP-hard problem due to the intractable space of search [13]. In order to solve this problem, two main methods have been proposed: data oriented method and the second one is based on the expert knowledge [14] which is time consuming. In this paper, we propose a score driven algorithm, called GIT algorithm, which is based on the Information Theory IT (precisely the Mutual Information MI exchanged between nodes) and the Graph Theory GT. The proposed BN was experimented while assisting in the exploration of the medical database for diabetes diagnosis in the Zulfy hospital of Saudi Arabia [5].

Section II is dedicated to the fundamental concept's introduction which is useful for our proposal description. Section III is devoted to the proposed GIT algorithm representation. In Section IV, we represent an illustrative example. Section V introduces the experimental results and its evaluation using the well-known benchmarks. In Section VI, we present the application of the proposed data oriented method in the medical field, precisely for the diabetes diagnosis. In the last section which is our conclusion, we summarize the main ideas of the paper.

II. Preliminaries

Since our purpose is to propose a novel BN structure learning algorithm based on the IT and the GT, we have to introduce the interrelated notions before highlighting the proposed idea.

A. Structure Learning of the BN

The Bayesian network is a Direct Acyclic Graph (DAG) that allows the representation of the distribution of the conditional probabilities over a set of variables. The DAG is composed of a set of nodes (random variables) and edges representing the dependencies between the nodes. The BN is illustrated as a couple (G, P) where the G is the directed graph and P designates the probabilities distribution, and the graph can be denoted as a couple (N, E), in which N is the nodes (or the random variables) and the E designates the edges between the nodes. The variable value can be discrete or continuous. As indicated in the following equation, the joint probability distributions are computed as the product of the local conditional probabilities:

$$P(N_1, N_2, ..., N_n) = \prod_{i=1}^{n} P(N_i \mid P_a(N_i)) \tag{1}$$

Where N_i is the node i and $P_a(N_i)$ represents its parent.

The construction of the BN model consists of two learning phases of the structure and the parameter. The learning of the BN structure allows us to obtain the graphical representation of the qualitative knowledge in which we are focusing on in this paper. The structure learning phase aims to represent explicitly the causal relationship among the random variables (is the answer of the “what if?” question) [15].

In the last two decades, several algorithms of BN structure-learning have been proposed which can be categorized into score oriented approach, conditional independency based approach and hybrid approach. (1) The conditional independency based methods perform a qualitative study of the variables dependency, and the generated skeleton represents these dependency relationships. The well-known algorithms of this approach are the PC (Predictive Causation) algorithm and IC (Inductive Causation) algorithm. (2) The score oriented approach is based on the score metric and it aims to determine the learned graph that maximizes the used score. This metric is defined as a fit measure between the data and the graph. The main goal of the algorithms based on the score is to produce the structure having the highest score. For instance, we can cite the MWST (Maximum Weight Spanning Tree) algorithm, GS (Greedy Search) algorithm, SEM (Structural Expectation Maximization) algorithm and K2 algorithm. Different score metrics have been proposed such as the BIC (Bayesian Information Criterion), BDe (Bayesian Dirichlet Equivalent), MDL (Minimum Description Length) and the AIC (Akaike Information Criterion). (3) The hybrid approach combines the advantages of both cited methods in learning the correct BN skeleton. As examples, we can cite the MMMB (Max Min Markov Blanket) algorithm and MMPC (Max Min Parents Children) algorithm.

In the last decade, many researches proved that the methods based on score represent the widely used algorithms like Amirkhani et al. [16] (2016), Tabar et al. [17] (2018) and Benmohamed et al. [18] (2019) [19](2020), Accordingly, the present study introduces a novel score based algorithm as explained in the following section.

III. Proposed Scored-oriented-algorithm

The proposed GIT method (Fig. 1) is divided into two main phases. The first phase allows the extraction of the dependencies between the different random variables within the dataset in order to create an undirected acyclic graph. To obtain this latter, we have to avoid the cyclic structure and the weak dependencies between the nodes. Henceforth, the extracted graph is used to determine the list of parent and children nodes. The generated oriented acyclic graph is not only important for correct parameter learning and instances classification but it also provides a graphical representation of useful knowledge that allows a better analysis of the dataset.

For extracting the undirected acyclic graph, we used the information theory IT, precisely the calculation of the mutual information MI in order to eliminate the cyclic structure. Basing on the graph theory GT, we start by avoiding the cycle between each tree nodes forming the graph because of the acyclic characteristic of the BN. In our algorithm called IT and GT based structure learning algorithm (GIT), the MI is calculated to determine the weak dependencies for erroneous edge elimination. Between each two variables A and B, the MI noted I(A,B) is calculated as follows:

$$I(A,B) = H(A) - H(A \mid B). \tag{2}$$

H(A) is the entropy of A, and H(A|B) represents the conditional entropy of A given B. The entropy of the variable A is defined by:

$$H(A) = -\sum_{i=1}^{n} P(A_i)\log(P(A_i)) \tag{3}$$

Mathematically, the H(A|B) is defined by the following equation:

$$H(A \mid B) = \sum_{i=1}^{n}\sum_{j=1}^{m} P(A = a_i, B = b_j) \times \log(P(A = a_i, B = b_j)) \tag{4}$$

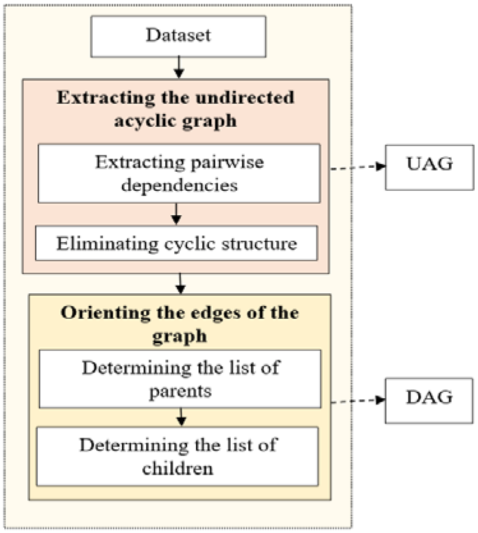
Dataset
Extracting the undirected acyclic graph
Extracting pairwise dependencies
Eliminating cyclic structure
UAG
Orienting the edges of the graph
Determining the list of parents
Determining the list of children
DAG

Fig. 1. Schematic Representation of the Proposed GIT Algorithm.

For learning the BN structures, we used in the algorithm the mutual information MI which is significantly used in the literature for structure learning. The proposed sub-algorithm for undirected graph extraction (Algorithm 1) included two phases, for the detection of the dependencies between nodes and for cyclic structure elimination as described in the rest of this section.

Algorithm 1 Undirected graph learning

Input: Dataset

Output: UAG (dag)

```
Repeat for i from 1 to N do
    Repeat for j from i+1 to N do
        K= 1,   Ok=True
        if  $i \neq j$ then
            Repeat while OK==True and k<=N do
                if $(k \neq i)$ && $(k \neq j)$ then
                    MIij ← calc_mutual_information(i, j);
                    MIki ← calc_mutual_information(k, i);
                    MIjk ← calc_mutual_information(j, k);
                    if MIij > MIki || MIij > MIjk   then
                        ok ← True;
                    else
                        ok ← False;
                    end if
                end if
                k=k+1;
            end while
            if (k>N) then
                dag(i,j) ←  1
            end if
    end for
end for
```

As shown in Algorithm 1, for each pairwise (X, Y) we verify the existence of a third node Z forming a cycle. As defined in the graph theory, the path can form a cycle if the start node (ni) represents the final node in the path (with the number of nodes is greater or equal to 1). In order to avoid the circuit formed by the three nodes X, Y and Z, we eliminate the weak dependency. In fact, if the condition cannot be reached, the specified nodes will form a cycle and the edge between the original pairwise should be eliminated as shown in Fig. 2:

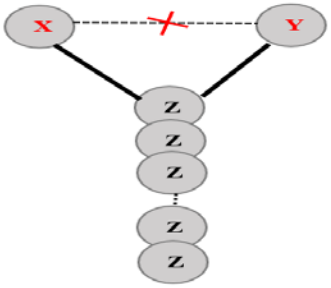

Fig. 2. Elimination of the Edge with Weak Dependency.

As a matter of fact, we use the obtained graph to determine the orientation of each edge. Indeed, the lists of parents and children can be created the final graph which are acyclic and oriented. These tasks are reached by Algorithm 2 which consists of two phases for parent and children detection. In [20], the dependence criteria used to determine the parent or the children node between two dependent nodes (X and Y) is calculated as:

$$OE(Y \rightarrow X) = \frac{H(X|Y)}{H(X) \times |X|}, X \neq Y \tag{5}$$

Where OE determines the orientation of the edge from Y to X (Y node represents the node parent) and the possible values' number of the variable X is $|X|$. In addition, we combine this criteria with the condition based on the value of the Maximum Mutual Information (MMI) for the edge orientation. For the determining of the parent and child, we propose an amelioration of the OE metric as follows:

The node X represents the parent if this condition is satisfied:

$$\frac{H(Y|X)}{H(Y) \times |Y|} + I(X) > \frac{H(X|Y) + I(Y)}{H(X) \times |X|} + (\alpha \times MMI(X))$$

Where $\alpha \geq 0.5$

If the node Y is the child and the orientation of the edge is $(Y \rightarrow X)$, the following condition should be satisfied:

$$\frac{H(X|Y)}{H(X) \times |X|} + I(Y) > \frac{H(Y|X) + I(X)}{H(Y) \times |Y|} + (\alpha \times MMI(Y))$$

These steps is explained in the algorithm2:

Algorithm 2 Orienting the edges

Input: Undirected Acyclic Graph (UAG)

Output: set of candidate parents and children

```
for i from 1 to N-1
    for j from i+1 to N do
        calculateMaxMI(i)  ← Max(MI(i,j))
    end for
end for
for i from 1 to N-1 do
    for j from i+1 to N do
        OEIJ  ← calc_OE(i,j);
        OEJI ← calc_OE(j,i);
        MIij ← calc_mutual_information(i, j);
        if (OEIJ + MIij > OEJI + α * MaxMI[i]) then
            CandidateParents [j] ←  i;
            CandidateParents [i] ←  j;
        else if (OEJI + MIij > OEIJ + α * MaxMI[j]) then
            CandidateParents [i] ←  j;
            CandidateParents [j] ←  i;
        end if
        end if
    end for
end for
```

In this section, we describe our method for learning the BN structure through data oriented approach. The learned dependencies obtained by the execution of the first algorithm are oriented basing on the second algorithm for generating the final directed acyclic graph. The main idea for the structure learning is based on the mutual information and the graph theory. The following section will be dedicated to the representation of the experimental results.

IV. Experimental Results

A. Used Datasets

To test the proposed GIT algorithm on the well-used benchmark networks, firstly we represent theses datasets and then the used performance measures. Thus, the test of the novel algorithm, is done using the three well-known datasets: ASIA, CANCER and ALARM. The algorithm is executed on an Inteli5-5300U with 8G of memory (64-bitsystem). In the following table, we expose the datasets description (Table I):

In the next sub-section, the evaluation metrics of the GIT algorithm’s performance and the gained results will be described.

B. Used Metrics and Experimental Results

The evaluation of the proposed algorithm, which is based mainly on the MI and the GT demonstrates its efficiency in resolving the structure learning problem. Moreover, to present the obtained results, we use the metrics shown in Table II.

The experimental results, shown in Fig. 3, Fig. 4 and Fig. 5, will be described using the difference between the original structure and the learned one (terms: CE, AE, DE, RE, SD).

To verify the effectiveness of the GIT algorithm, it was executed on the datasets shown in Table II for 1000, 2000, 3000, 5000 and 10000 cases. Fig. 3, Fig. 4 and Fig. 5 show the experimental results basing on structures difference between the obtained skeleton and the original. In Fig. 4, we show the gained results for ASIA network for different cases which are seven correct edges, one reversed edge and zero deleted and added edge. In fact, for this latter, we obtain sensitive values when changing the number of cases from 1000 to 10000. In addition, for CANCER network, we obtain four correct edges for 1000, 2000 and 3000 cases with one added edge. However, for ALARM network, our method can correctly detect thirty four edges, four edges are wrongly deleted, one reversed edge, and eleven are added incorrectly for ALARM-1000, ALARM-2000 and ALARM-3000. For the rest of the cases, our GIT algorithm produces just one reversed edge, four deleted edges and twelve added edges; accordingly, we obtain seventeen erroneous edges in comparison to the original ALARM network.

TABLE I. Used Datasets in the Experiments

Dataset	Number of cases	Number of nodes	Number of edges
CANCER	1000/2000/3000/ 5000/10000	5	4
ASIA	1000/2000/3000/ 5000/10000	8	8
ALARM	1000/2000/3000/ 5000/10000	37	46

TABLE II. The used Evaluation Metrics

Metric	Description
RE	The reversed edge: is the correct edge with the inversed orientation
CE	The correct edge: exists in the original and the learned graph
AE	The added edge: does not exist in the original network
DE	The deleted edge: exists in the learned structure and does not exist in the original graph
SD	The erroneous edges: represent the structural difference (RE+AE+DE).

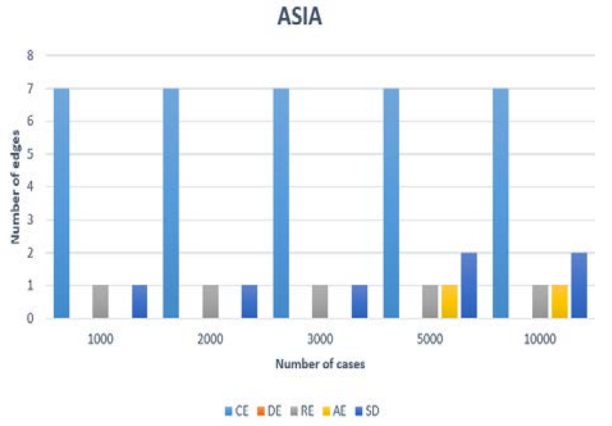

Fig. 3. Experimental results for ASIA network

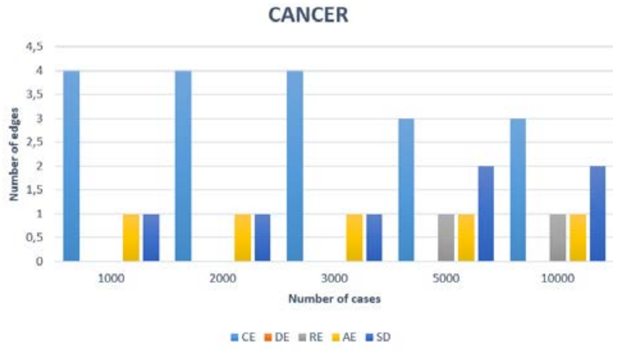

Fig. 4. Experimental results for CANCER network

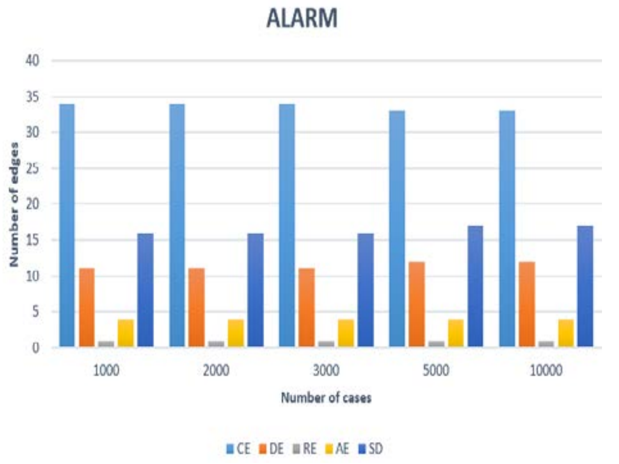

Fig. 5. Experimental results for ALARM network

Furthermore, we use the accuracy metric (Acc) [21] in order to present the performance of our GIT algorithm and this factor is defined as follows:

$$Acc = \frac{CE}{CE + SD}$$

In the following figure, we introduce the values of accuracy for the 1000, 2000, 3000, 5000 and 10000 cases for CANCER, ASIA and ALARM networks.

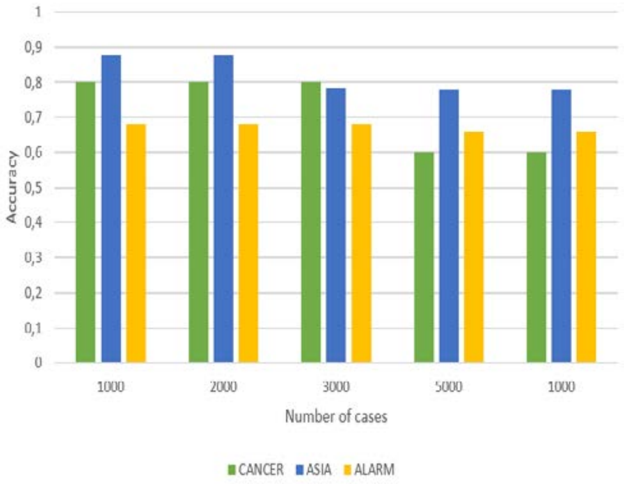

Fig. 6. Accuracy for Various Datasets.

As shown in Fig. 6, the proposed method produces better accuracy values for ASIA network which are respectively for ASIA (1000, 2000 and 3000) 0.875 and for ASIA (5000 and 10000) 0.778. In addition for CANCER network, for 1000, 2000 and 3000 is 0.8 and for CANCER-5000 and CANCER-10000 equals to 0.6. As shown, for ALARM-1000, ALARM-2000, ALARM-3000, ALARM-5000 and ALARM-10000, all values are greater than 0.65 and decrease with the increase of the number of samples. The produced results allow the demonstration of the reliability of the proposed algorithm for learning the structure of CANCER, ASIA and ALARM networks. Besides, for GIT algorithm efficiency demonstration, we present a comparison section of the gained results with the other methods results for the resolution of the BN structure-learning problem.

V. Experimental Results

The gained results produced by the GIT algorithm are introduced in Table III and Table IV for respectively ASIA and ALARM networks. The terms CE, AE, DE and RE designate the number of correct edges and respectively the number of accidently added, deleted and reversed edges. The tables show the evaluation of our results compared to the NDPSO-BN [22], Ko and Kim [23], Tabar et al. [] and Ai [24] algorithms for ASIA and ALARM networks for 1000, 2000, 3000, 5000 and 10000 cases. To highlight the results, the bold value represents the best value and the starred value depicts the second potential value.

In Table III, we present the different experimental results describing the difference between the original topology and the learned structure using respectively our algorithm, NDPSO-BN, Ko and Kim, Tabar et al. and Ai algorithms. As exhibited, our GIT algorithm allows the gaining of the best or the second best results for ASIA-1000, ASIA-2000 and ASIA-3000. Comparing to the NDPSO-BN method, our algorithm produces the same number of erroneous edges, which equals to one edge. The GIT algorithm detects the eight correct edges but with one incorrect oriented edge while the NDPSO-BN cannot determine it for ASIA-1000 and ASIA-3000. In addition, the other methods give between three to six erroneous edges. In Table IV, we present the experimental results comparisons among the five algorithms for ALARM-1000, ALARM-2000, ALARM-3000, ALARM-5000 and ALARM-10000.

TABLE III. Structures Comparisons among Five Algorithms on ASIA Network

Cases	Edge type	GIT	NDPSO-BN	Ko and Kim	Tabar et al.	Ai
1000	CE	**7**	-	5	4	4
	DE	**0**	1	0	0	1
	RE	**1***	0	3	4	2
	AE	**0**	0	1	0	3
	SD	**1**	1	4	4	6
2000	CE	**7**	-	5	5	4
	DE	**0**	-	0	0	1
	RE	**1**	-	3	3	2
	AE	**0**	-	1	0	3
	SD	**1**	-	4	3	6
3000	CE	**7**	-	5	5	4
	DE	**0**	1	0	0	1
	RE	**1***	0	3	3	2
	AE	**0**	0	1	0	3
	SD	**1**	1	4	3	6
5000	CE	**7**	-	5	5	4
	DE	**0**	-	0	0	1
	RE	**1**	-	3	3	1
	AE	**1**	-	1	1	3
	SD	**2**	-	4	4	6
10000	CE	**7**	-	5	6	5
	DE	**0**	-	0	0	1
	RE	**1**	-	3	3	1
	AE	**1**	-	1	1	3
	SD	**2**	-	4	4	5

TABLE IV. STRUCTURES COMPARISONS AMONG FIVE ALGORITHMS ON ALARM NETWORK

Cases	Edge type	GIT	NDPSO-BN	Ko and Kim	Tabar et al.	Ai
1000	CE	**34***	-	38	38	23
	DE	11	2	4	2	3
	RE	**1**	1	4	8	28
	AE	**4***	2	9	4	34
	SD	16	6	17	14	59
2000	CE	**34***	-	39	39	23
	DE	11	2	2	1	3
	RE	**1***	0	4	8	21
	AE	**4***	2	9	4	34
	SD	16	4	15	13	55
3000						
	CE	34	-	-	-	-
	DE	11	1	-	-	-
	RE	1	0	-	-	-
	AE SD	4 16	1 3	- -	- -	- -
5000	CE	33	-	40	41	24
	DE	12	1	2	1	2
	RE	**1***	0	4	8	21
	AE	**4***	1	13	7	32
	SD	17	2	19	16	55
10000	CE	33	-	40	41	24
	DE	12	-	2	1	2
	RE	**1**	-	4	8	20
	AE	**4**	-	15	7	30
	SD	**17***	-	21	16	52

From the first observation, we can demonstrate that the proposed algorithm produces sensitive results when increasing the dataset's size. Besides, we obtain the lowest or the second best number of accidently RE and AE for ALARM-1000, ALARM-2000, ALARM-5000 and ALARM-10000 (Table IV). The proposed GIT algorithm cannot extract the correct number of correct edges and gives thirty four or thirty three CE and eleven or twelve DE. For ALARM network for 1000 and 2000 cases, our method produces the second greatest number of CE. Moreover, the results presented above allows us to declare that the GIT algorithm learns the BNs' topologies with sensitivity to the increasing of the number of cases. In the next section, we will use the proposed BN to model the features of diabetes dataset and the extracted causality relationships which represent the main factor for helping in the diabetes diagnostics.

VI. APPLICATION OF GIT ALGORITHM FOR THE DIABETES DIAGNOSTICS

Data classification techniques have an important role in medical field. These intelligent techniques help physicians to analyse and explore large amount of datasets. In our case study, as explained above 47.7% of diabetes patients are not diagnosed. Therefore, basing on the probabilistic graphical model, we aim to represent the cause-effect relationships between the characteristic features in the diabetes dataset. In Table V, we introduce the features description and the used samples (for training and test) in the diabetes dataset.

The proposed GIT algorithm is implemented using FULLBNT project in Matlab.

The application of our proposal in the training diabetes dataset (368 samples) allows the generation of the structure shown in Fig. 7 in which the class node A1Cg represents the parent node for all given variables. These extracted dependency relationships means that being diabetic affects directly or indirectly the rest of the used features. In addition, the calculated K2 score of the generated graph is -99543,25.

The directed acyclic graph reported in Fig. 7 describes the dependency relationships between the used variables of the diabetic dataset. Specifically, the class variable (A1Cg) has a direct connection with CIMT, PT SPT, Ba and AgeG variables. The age, A1C and DiP nodes are connected to the A1Cg variable through CIMT node. Furthermore, the rest of the variables are connected to the main variable through the nodes 1 and 6. Once these connections are broken, the physician cannot analyse correctly the cause and effect relationships. The efficiency of our GIT algorithm has to be validated and improved basing on the expert knowledge. Indeed, completing the BN model construction with the conditional-probability tables allows the provability of our proposal's importance. Moreover, this latter can be demonstrated relating to the GIT algorithm's ability in the improvement of the classification of results.

TABLE V. USED DATASET DESCRIPTION

Feature No	Feature description		Range
1	CIMT		0.40-0.90
2	A1C		4.5-11
3	AgeG		1-6
4	Age		18-72
5	RI		0.52-0.85
6	SPT		125-255
7	PT		0.59-0.92
8	DiP		0.50-0.78
9	ba		0.52-0.80
10	PPT		93.75-191.25
11	DT		218.75-446
12	SP		0.61-0.94
13	A1Cg		Class: Diabetic or Not diabetic
Sample	**Training**	368	80%
	Test	92	20%

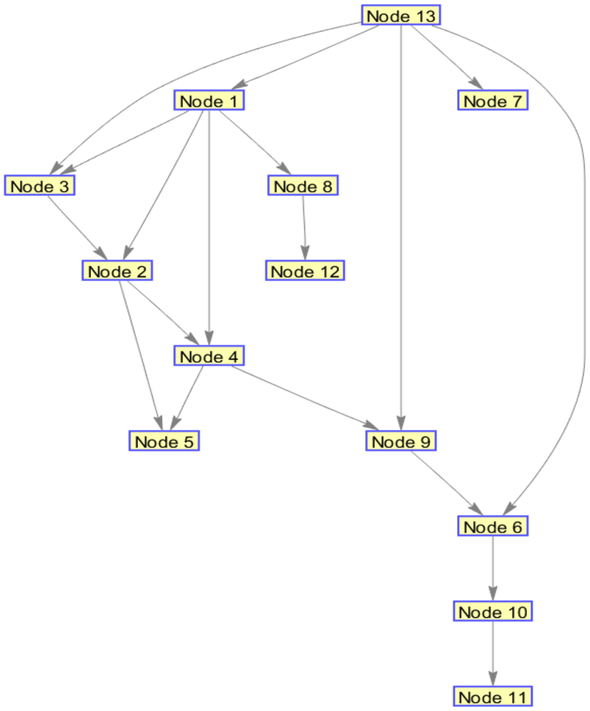

Fig. 7. DAG of the Diabetic Dataset.

VII. Conclusions

In this paper, we propose a novel algorithm based on the Information Theory and the Graph Theory in order to solve the Bayesian network structure learning problem. Furthermore, testing the GIT algorithm on the well-known networks produced important results which were compared to four proposed algorithms.

The gained results exhibit the efficiency of our method in learning the structure of small network (ASIA with 8 nodes).

For this network, our GIT algorithm is superior in terms of the correct edges detection and the erroneous edges learning. Besides, for ALARM network, our method generated acceptable results and comparing to the other algorithms, it produced the smaller number of reversed and added edges.

To summarize, our proposal represents an effective data driven method for BN structure learning. As for future works, we will complete to BN model construction to prove the effects of the learned skeleton on the classification of data. Moreover, we will concentrate on the model validation by the experts and we will further explain its efficiency for diabetes diagnostic.

References

[1] Singh, N., Singh, P., & Bhagat, D. (2019). A rule extraction approach from support vector machines for diagnosing hypertension among diabetics. Expert Systems with Applications, 130, pp.188-205, 2019.

[2] K. Kannadasan, E. Damodar, and K. Venkatanareshbabu, "Type 2 diabetes data classification using stacked autoencoders in deep neural networks," Clinical Epidemiology and Global Health, 2018.

[3] Nanditha, A., Ma, R. C., Ramachandran, A., Snehalatha, C., Chan, J. C. N.,Chia, K. S., et al. (2016). Diabetes in Asia and the Pacific: Implications for the global epidemic. Diabetes Care, 39(3), 472–485 Retrieved from. doi:10.2337/dc15-1536.

[4] Prasad, D. Venkata Vara, et al. "An efficient pre-processing method for improved classification of diabetics using decision tree and artificial neural network." AIP Conference Proceedings. Vol. 2161. No. 1. AIP Publishing LLC, 2019.

[5] Qawqzeh, Y. K. Neural Network-based Diabetic Type II High-Risk Prediction using Photoplethysmogram Waveform Analysis. IJACSA International Journal of Advanced Computer Science and Applications,Vol. 10, No. 12, 2019.

[6] O. Gevaert, F. De Smet, E. Kirk, B. Van Calster, T. Bourne, S. Van Huffel, Y. Moreau, D. Timmerman, B. De Moor, G. Condous, “Predicting the outcome of pregnancies of unknown location: Bayesian networks with expert prior information compared to logistic regression“, Human Reproduction, vol. 21, no. 7, pp. 1824–1831, 2006, https://doi.org/10.1093/humrep/del083.

[7] Wang, J., & Liu, S. A novel discrete particle swarm optimization algorithm for solving bayesian network structures learning problem. International Journal of Computer Mathematics, 96(12), 2423-2440, 2019.

[8] S. Nikolopoulos, G.T. Papadopoulos, I. Kompatsiaris, I. Patras, “Evidence-driven image interpretation by combining implicit and explicit knowledge in a bayesian network”, IEEE Trans. Syst. Man Cybern. Part B (Cybernetics) 41, pp. 1366–1381, 2011.

[9] B. Yet, A. Constantinou, N. Fenton, M. Neil, E. Luedeling, K. Shepherd, “A bayesian network framework for project cost, benefit and risk analysis with an agricultural development case study”, Expert Syst. Appl, pp. 141–155, 2016.

[10] P. Suchánek, F. Marecki, and R. Bucki, Self-learning bayesian networks in diagnosis, Procedia Comput. Sci. 35 (2014), pp. 1426–1435.

[11] L. Zhang and Q. Ji, A bayesian network model for automatic and interactive image segmentation, IEEE Trans. Image Proc. A Publ. IEEE Signal Process. Soc. 20(9) (2011), pp. 2582–2593.

[12] F. Liu, S.W. Zhang, W.F. Guo, Z.G. Wei, and L. Chen, Inference of gene regulatory network based on local bayesian networks, PLoS Comput. Biol. 12(8) (2016), pp. 1–17.

[13] W. Buntine, “A guide to the literature on learning probabilistic networks from data,” IEEE Trans. Knowl. Data Eng., vol. 8, no. 2, pp. 195–210, 1996.

[14] H. S. Sousa, F. Prieto-Castrillo, J. C. Matos, J. M. Branco, & P. B. Lourenço, “Combination of expert decision and learned based Bayesian Networks for multi-scale mechanical analysis of timber elements“. Expert Systems with Applications, vol. 93, pp. 156-168, 2018.

[15] M. Velikova, J.T. van Scheltinga, P.J. Lucas, M. Spaanderman, ”Exploiting causal functional relationships in bayesian network modelling for personalized healthcare”, Int. J. Approximate Reasoning 55 (1) , pp. 59–73, 2014.

[16] Amirkhani, H., Rahmati, M., Lucas, P. J., & Hommersom, A. Exploiting experts’ knowledge for structure learning of bayesian networks. IEEE transactions on pattern analysis and machine intelligence, 39(11), 2154-2170, 2016.

[17] V. R. Tabar,, F. Eskandari, S. Salimi, et al. “ Finding a set of candidate parents using dependency criterion for the K2 algorithm”. Pattern Recognition Letters, vol. 111, pp. 23-29, 2018.

[18] E. Benmohamed. H. Ltifi, M. Benayed. A Novel Bayesian Network Structure Learning Algorithm: Best Parents-Children. In proceeding. IEEE ISKE, the 14th International Conference on Intelligent Systems and Knowledge Engineering, 2019.

[19] E. Benmohamed. H. Ltifi, M. Benayed. Hybrid data analysis approach based on improved K2PC algorithm and expert knowledge Application for assessing the phosphate laundry effluents impact. In proceeding. ICDAR, International Conference on Document Analysis and Recognition, 2020.

[20] J. Jiang, J. Wang, H. Yu, H. Xu, Poison identification based on Bayesian network: a novel improvement on K2 algorithm via markov blanket. In International Conference in Swarm Intelligence. Springer, Berlin, Heidelberg, 2013, pp.173-182.

[21] M. Scutari, C. E. Graafland & J. M. Gutiérrez,“ Who learns better Bayesian network structures: Accuracy and speed of structure learning

algorithms“. International Journal of Approximate Reasoning, vol. 115, pp. 235-253, 2019.

[22] J. Wang et S. Liu, “A novel discrete particle swarm optimization algorithm for solving bayesian network structures learning problem”. International Journal of Computer Mathematics, pp. 1-18, 2019.

[23] S. KO et DW. KIM., “An efficient node ordering method using the conditional frequency for the K2 algorithm”. Pattern Recognition Letters, vol. 40, p. 80-87, 2014.

[24] X. Ai, “Node importance ranking of complex networks with entropy variation”, Entropy, vol. 19, no 7, pp. 303, 2017.