INTELLIGENT MINING ASSOCIATION RULES

Sarjon Defit

Faculty of Computer Science University Putra Indonesia "YPTK" Padang,
West Sumatera
sarjond@yahoo.co.uk

ABSTRACT

Association rules is one of data mining methods for discovering knowledge from large amounts of data in databases. In this paper, we propose an intelligent method for discovering association rules, called IMAR. IMAR is designed through three main phases, i.e., preprocessing, processing and post processing. It has been experimented using three domain data sets, i.e., Australian Credit Card (ACC), Jakarta Stock Exchange (JSX), and Cleveland Heart Diseases (CLEV) data sets. Our experimental results show that IMAR can (i) discover association rules from large inconsistent databases intelligently and accurately, and (ii) reduce the number of generated interesting association rules without losing information and with higher accuracy.

KEYWORDS

Database, Data Mining, Association Rules, Knowledge, Information

1. INTRODUCTION

Association rule is one of data mining methods for discovering knowledge from large amount of data in databases. It is a rule in the form of [1, 2, 3, 4, 5].

$$X_1, X_2, \ldots, X_m \rightarrow Y_1, Y_2, \ldots, Y_n \quad [S,C] \qquad (1)$$

where X_i and Y_i are items. The S and C are support and confidence of rules respectively. It has become more and more popular since its introduction in 1993. Today, it is still one of the most popular methods in data mining [1, 2, 3, 4, 5, 6, 7].

A number of promising association rule methods have been studied and developed. For instance (i) a close algorithm [7], (ii) a closet algorithm [8], (iii) online analytical mining association rules [4], (iv) mining association rules with multiple Minimum Item Support [6], and (v) discovery of knowledge at multiple level concepts [9]. These methods have given great advantages for users in order to generate rules from large amounts of data. However, association rules methods still have some of the following weaknesses and need further improvement. First, accuracy of association rules method [4, 6, 7, 8, 9]. The association rule method should portray the contents of the database accurately. The noise and uncertainty should be cleaned elegantly. It is one of the important tasks in data preparation in order to identify which data in databases are inaccurate or missing values. Second, usefulness of association rules method [4, 6, 7, 8]. The association rules method should be useful for certain application and generate association rules from data which are represented at higher levels concept. The raw data should be transformed from raw data into higher levels concept. It allows users to find association rules deeply and view database contents

DOI : 10.5121/ijcsit.2012.4409

at different abstraction levels. Third, identification of interesting rules [6, 7, 8, 9].The association rules method should generate more interesting rules accurately. The generated rules should be identified in order to reduce the number of interesting rules without losing information. Fourth, using of prior domain knowledge [4, 6, 7, 8, 9]. The association rules method should generate interesting rules intelligently. It allows us to generate association rules from large amount of data in databases intelligently and automatically. Lastly, intelligent hybrid association rules method [4, 6, 7, 8, 9]. The association rules method should generate knowledge from various domains and solve the complex data mining problems. It can generate more interesting rules and reduce the number of rules without loss of information.

In order to cater these association rule problems, in this paper, we propose an Intelligent Mining Association Rules, called IMAR.

2. INTELLIGENT MINING ASSOCIATION RULES (IMAR) GENERAL ARCHITECTURE

In this section, we describe the Intelligent Mining Association Rules (IMAR) general architecture as given in the appendix. Generally, Intelligent Mining Association Rules (IMAR) consists of three main phases including preprocessing, processing and post processing. The first phase has two processes including data cleaning and data transformation while the main process of the processing is rule generation. The first two phases, i.e., preprocessing and processing, have two main steps including training and running steps. The first step is conducted for creating neural network knowledge based of data cleaning, data transformation and association rules. The generation of learned complete data, transformed data and interesting association rules are done in the running step. Next, the generated interesting rules are applied in real world problems in order to create crucial business decisions.

Intelligent Mining Association Rules (IMAR) is designed based on combination of several intelligent techniques, i.e, rough set, association rules and neural networks knowledge based. The purpose of the combinations of these intelligent techniques is to create a method for solving the data mining complex problems, i.e., data cleaning, data transformation and association rules. In order to support IMAR, we also propose data cleaning and data transformation methods. However, this paper is focused on data transformation only. The details of proposed data cleaning and association rules methods can be found in [10] and [11].

3. DATA TRANSFORMATION

In this section, we describe the basic and extended data transformation algorithms. The details of these data transformation algorithms are given in the following sub sections 3.1 and 3.2.

3.1. Basic Data Transformation Algorithm

This algorithm is unsupervised data transformation. Number of classes is derived based on number of observed data. Each attribute in databases has the same number of classes and each class has the same intervals. The basic concept of this algorithm is given in definition 1.

Definition 1: Let βv and γv be the largest and the smallest values. The total value range, αv, is defined as $\alpha_v = \beta_v - \gamma_v$. Let ηc be the number of classes, defined as $\eta_c = 1 + 3.3\,Log(n)$ where

n equals to number of observations. The Interval, denoted as χv, is defined as $\lambda_v = \frac{\alpha_v}{\eta_c}$, and the cut point, denoted as ϕv, defined as $\theta v = \gamma_v + \lambda_v(i) \quad for\, i = 1,2,3,...,\eta_c - 1$.
Based on definition 1, the basic data transformation algorithm is given in figure 1.

Input : Complete Databases
Output : Transformed Data
Method:

1) Estimation of the total value range.
2) Estimation of the number of classes
3) Derivation of interval value
4) Compute the cut point
5) Code the observed values based on set of cuts, i.e., 0, 1, 2,…,n-1, where n is number of class
6) Generate indiscernibility relation based on conditional attribute
7) For each indiscernibility relation, check inconsistent data
8) *If inconsistent data is found, replace decision value by modus decision value*
9) *This process is proceeded until all data are consistent*

Figure 1: The Basic Data Transformation Algorithm

Figure 1 shows a basic data transformation algorithm. It is used for generating a transformed data from complete data in the training step and target output of transformed data in the running step. The input and output of this algorithm are complete and transformed data respectively.

3.2 Extended Data Transformation Algorithm

In this section, we give an extended data transformation algorithm. It is used for generating learned transformed data which is supported by neural network knowledge based. In this algorithm, the creation of neural network knowledge based of data transformation and target output of transformed complete testing data are supported by basic data transformation algorithm. The illustration of extended data transformation algorithm is given in the following figure 2.

Step 1: Training

Input : The complete training data which is generated at training step of data cleaning
Output : Knowledge of data transformation

Methods

1) Transform complete training data, A, using basic data transformation algorithm, and then saved as transformed training data, B.
2) Merge transformed training data, B, and complete training data, A, into merged data, C.
3) Split the merged data, C, into training and testing data sets, D and E respectively.
4) Train these data sets, D and E, using neural network .
5) Merge the condition part of merged data and trained data in order to create knowledge of data transformation, N.

Step 2: Running

Input : Learned complete testing data which is generated at running step of data cleaning; Knowledge of data transformation

Output : Learned transformed learned testing data

Methods

1) Transform learned complete testing data, I, using basic data transformation algorithm and then saved it as transformed testing data, O.
2) Merge the learned complete testing data and transformed testing data, and I and O, into merged data, P
3) Suppose the merged data, P, and knowledge of data transformation which is generated at training step, N, as testing and training data sets.
4) Learn training and testing data sets, P and N, using neural network.
5) Merge the condition part of merged data with the learned data in order to create new knowledge of data transformation, Q.
6) Compare the merged data with the new knowledge of data transformation. When the condition parts are matcedh, then replace the action part of the merged data with the new knowledge of data transformation.
7) Add new knowledge into previous knowledge of data transformation.

Figure 2: The Extended Data Transformation Algorithm

Figure 2 shows the extended data transformation algorithm. It consists of two main steps including training and running steps. The training step is conducted for generation neural network knowledge based of data transformation while the running step is used for generating learned transformed testing data. The input and output of the training step are complete training data which is obtained in the first step of data cleaning and neural network knowledge based of data transformation. Compared to the first step, the input of running step is the learned complete testing data which is obtained in the second step of data cleaning while the output is the learned transformed testing data.

4. IMAR EXPERIMENTAL RESULTS

We have tested IMAR by using three different domain data sets. They were Australian Credit Card (ACC), Jakarta Stock Exchange (JSX) and Cleveland Heart Diseases (CLEV). In the following, we describe the IMAR performances in both preprocessing and processing phases.

4.1 Preprocessing of IMAR Experimental Results

We have studied and analyzed the IMAR data cleaning experimental results. In this research, the IMAR data cleaning performance is measured based on three different error measurements, called Mean Difference Error (MDE), Mean Relative Error (MRE) and MSE (Mean Square Error) which are defined as follows

$$MDE = \frac{1}{n}\sum_{i=1}^{n}(va_i - vp_i) \quad (2)$$

$$MRE = \frac{1}{n}\sum_{i=1}^{n}\frac{(va_i - vp_i)}{vp_i} \quad (3)$$

$$MSE = \left[\frac{1}{n}\sum_{i=1}^{n}(va_i - vp_i)^2\right]^{1/2} \quad (4)$$

Where

va_i and vp_i = the observed and predicted values of output respectively
n = the number of observations

The performance of IMAR data cleaning compared with mean substitution data cleaning method is given in the following table 1 and figure 3.

Table 1: The IMAR Data Cleaning Performance Compared With Mean Substitution Data Cleaning Method

Data Set	IMAR Data Cleaning			….	Mean Substitution		
	MDE	MRE	MSE	…	MDE	MRE	MSE
ACC	0.004	0.0013	0.0160	…	0.0127	0.0013	0.4033
JSX	0.260	0.0012	7.00	…	0.108	0.0032	10.987

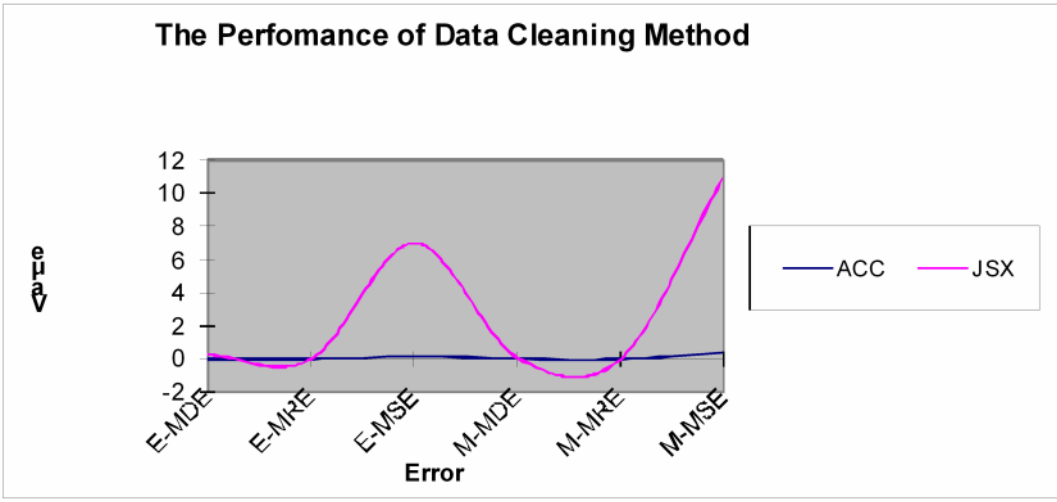


Figure. 3: The IMAR Data Cleaning Performance Compared With Mean Substitution Data Cleaning Method

Table 1 and figure 3 show the performance of IMAR data cleaning compared with mean substitution data cleaning method. For example, JSX data set, the Mean Difference error (MDE) of IMAR data cleaning and mean substitution are equal to 0.2600 and 0.1080 respectively. While the Mean Relative Error (MRE) are equal to 0.0012 and 0.0032 respectively. The Mean Square Error (MSE) of these methods is equal to 7.000 and 10.987 respectively. Based on these experimental results, we concluded that these methods give better results for handling uncertainty and missing data. Thus, IMAR data cleaning offers several advantages as given in table 2.

Table 2: The Advantages and Disadvantages of IMAR Data Cleaning

Advantages	**Disadvantages**
i) It can handle the missing and uncertainty data elegantly and intelligently ii) It give more accurate values iii) It decrease the error including MDE, MRE and MSE	i) It is limited for numeric data

Table 2 shows the IMAR data cleaning advantages and disadvantages. This method can (i) handle the missing and uncertaint data in databases, (ii) it can fill in the missing values with the accurate values, and (iii) it can decrease the error measurements including mean differences error (MDE), Mean Relative Error (MRE) and Mean Square Error (MSE). Although IMAR data cleaning has great advantages, this method is still limited for handling the missing values form numerical data. It still needs further improvement for handling uncertainty and missing data from another data type, i.e., categorical and text data

4.1.2 IMAR Data Transformation Experimental

In this research, the performance of IMAR data transformation is evaluated based on number of generated classes of transformed data. The performance of IMAR data transformation compared with two previous data transformation, i.e., Yongjian Fu's and equal binning frequency data transformation methods is given in table 3 and figure 4.

Table 3: The IMAR Data Transformation Performance Compared with YongjianFu's and Equal Binning Frequency Data Transformation Methods

Data Sets	The Generated Number of Classes (Mean)		
	Y-DT	EBF-DT	IMAR-DT
JSX	14	3	7
ACC	14	3	11
CLEV	14	3	8

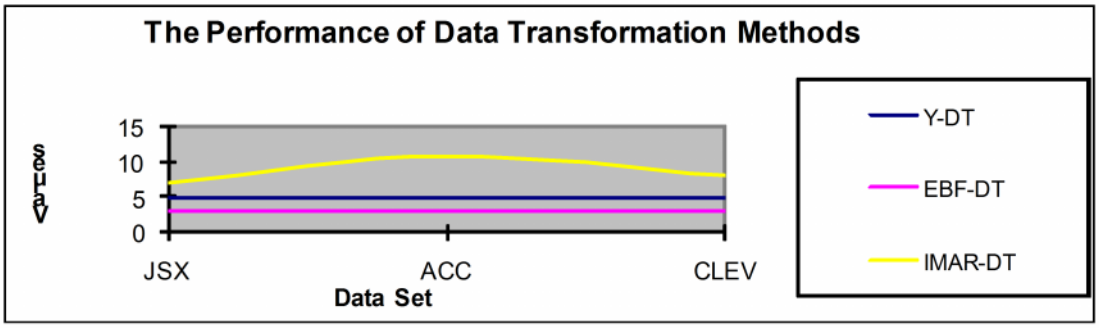


Figure 4: The IMAR Data Transformation Performance Compared with YongjianFu's and Equal Binning Frequency Data Transformation Methods

where:

Y-DT =Yongjian Fu's Data Transformation
EBF-DT=Equal Binning Frequency Data Transformation
IMAR-DT= IMAR Data Transformation

Table 3 and figure 4 show the IMAR data transformation performance compared to Yongjian Fu's and equal binning frequency data transformation methods. We show that, different methods generate different number of classes. For instance, Yongjian Fu's data transformation method, generate number of classes that depends on number of classes threshold, i.e., 2, 3, …, n. While equal binning frequency generates each attribute in databases into static number of classes. In other words, all attributes in databases could have the same number of classes. For instance, attributes in JSX, ACC and CLEV data sets are transformed into 3 (three) classes. Compared to Yongjian Fu's and equal binning frequency data transformation, the IMAR data transformation

could transform raw data in databases into different number of classes. For instance, “Price” and CapVal” attributes in JSX data set are transformed into 9 and 3 classes respectively.

Based on these experimental results, we conclude that IMAR data transformation method offers several advantages as given in table 4.

Table 4: The Advantages and Disadvantages of IMAR Data Transformation

Advantages	Disadvantages
i) It can transform raw data into transformed data accurately and intelligently. ii) It can transform raw data into various number of classes of transformed data	i) It is limited to transforming raw data from numerical data. ii) it cannot transform raw data into more higher levels.

Table 4 shows the advantages and disadvantages of IMAR data transformation method. This method can (i) transform raw data into transformed data accurately and intelligently, and (ii) transform raw data into transformed data with differences in number of classes. Although IMAR data transformation could transform raw data into transformed data more accurately, this method still has the following limitations (i) it is limited to transform raw data into discrete values and need to be improved to be able to transform raw data into higher levels, and (ii) it is limited to transforming raw numerical data and need a further study for transforming raw data from other data types.

4.2 The Processing of IMAR Experimental Results

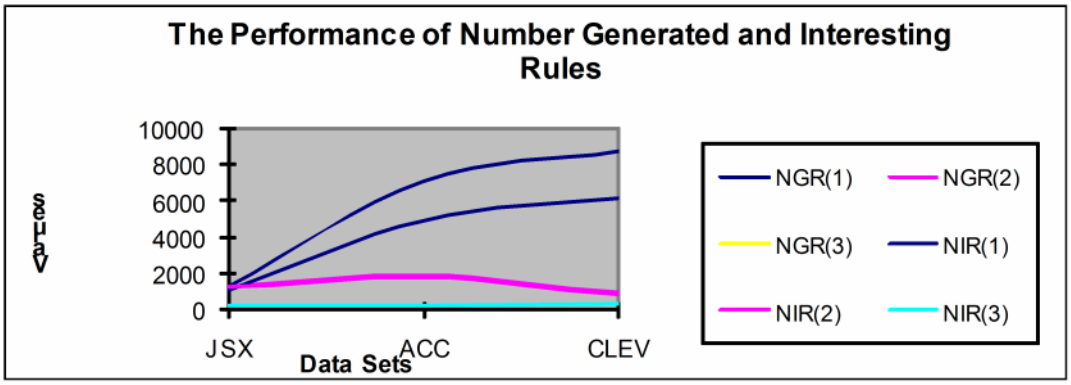
In this research, the processing of Intelligent Mining Association Rules (IMAR) performance is measured based on number and accuracy of generated interesting rules as given in table 5, figure 5.a and 5.b.

Table 5: The Processing of IMAR performance Compared With Association Rules Without Preprocessing and Basic Association Rules Methods

Methods	ACC			JSX			CLEV		
	NGR	NIR	Acc	NGR	NIR	Acc	NGR	NIR	Acc
AR-Without Pre Processing	1181	1004	90.13	7120	4941	78.71	8781	6161	69.95
BRG	1240	1236	99.5	1879	1758	96.7	953	860	90.4
IMAR-AR	202	189	91.4	232	231	99.6	272	271	99.8

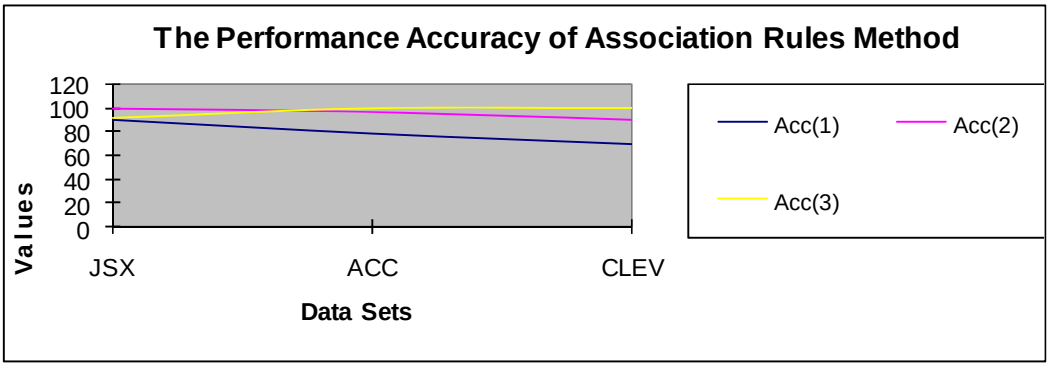
Where:

AR-Wihout-Pre = Association Rules Without Pre Processing Phase
BRG = Basic Rule Generation Method
IMAR-AR = Intelligent Association Rules
NGR = Number of Generated Rules
NIR = Number of Interesting Rules



NGR(1) = Number of Generated Rules Using Association Rules Without Pre Processing Phase
NGR(2) = Number of Generated Rules Using Basic Rule Generation
NGR(3) = Number of Generated Rules Using IMAR-Association Rules
NIR(1) = Number of Generated Interesting Rules Using Association Rules Without Pre Processing Phase
NIR(2) = Number of Generated Interesting Rules Using Basic Rule Generation
NIR(3) = Number of Generated Interesting Rules Using IMAR-Association Rules

Figure 5.a: The Performance of Number Generated and Interesting Rules



where:

Acc(1) = Accuracy of Association Rules Without Pre Processing Phase
Acc(2) = Accuracy of Basic Rule Generation
Acc(3) = Accuracy of IMAR-Association Rules

Figure 5.b: The Performance Accuracy of Association Rules Methods

Table 5, figure 5.a and 5.b show the performance of IMAR association rules compared to association rules without pre processing, basic rule generation association rules methods. For example, generated rules form JSX data set, using association rules without pre processing, basic rule generation and IMAR association rules are equal to 1181, 1240 and 202 while the number of interesting rules are equal to 1004, 1236, and 189 rules respectively. The accuracy of generated interesting rules are equal to 90.3, 99.5 and 91.4 %. Another example, generated rules from ACC data set, the number of generated rules using association rules without pre processing step, basic rule generation and IMAR association rules are equal to 7120, 1879 and 232 respectively while the generated interesting rules are equal to 4941, 1758 and 231 respectively. The accuracy of generated interesting rules are equal to 78.71, 96.7 and 99.6 respectively.

Based on these experimental results, we see that IMAR offers several advantages as given in table 6.

Table 6: The Advantages and Disadvantages of IMAR Method

Advantages	Disadvantages
i) it can mine multi dimensional association rules from large databases accurately and intelligently ii) It can reduce the number of generated interesting rules without loss of information	i) It cannot mine more complex association rules

Table 6 shows the advantages and disadvantages of IMAR method. This method can (i) mine multi dimensional association rules from large inconsistent data intelligently and accurately, and (ii) it can reduce the number of generated interesting rules without loss of information. Although IMAR association rules could reduce the number of generated interesting rules with higher accuracy, this method should be extended for generating association rules from other types of rules since it is limited for generating rules from first predicate calculus form.

5. SUMMARY

We have proposed an intelligent method for discovering multi dimensional association rules from large inconsistent databases, called IMAR. IMAR is designed through three main phases, i.e., preprocessing, processing and post processing. IMAR has been experimented using three domain data sets, i.e., Australian Credit Card (ACC), Jakarta Stock Exchange (JSX), and Cleveland Heart Diseases (CLEV) data sets. Our experimental results show that IMAR can (i) discover multi dimensional association rules from large inconsistent databases intelligently and accurately, and (ii) reduce the number of generated interesting association rules without loss information and with higher accuracy.

References

[1] Sarjon, D., Mohd, N., Mining Association Rules Using Rough Sets and Association Rules Methods, Proceedings of International Conference on Artificial Intelligence in Engineering and Technology 2002 (ICAIET'02) , Kota Kinabalu, Sabah, Malaysia, June 17-18, 2002.

[2] Sarjon, D., Mohd, N., Mining Multiple Level Association Rules Using Rough Sets and Association Rules Methods, Proceedings of International Conference on Artificial Intelligence and Soft Computing 2002 (ASC'02), Banff, Canada, July 17-19, 2002.

[3] Sarjon, D., Mohd, N., Association Rules Using Rough Sets and Association Rules Methods, Proceedings of Pacific Rim International Conference on Artificial Intelligence 2002 (PRICAI'02), Tokyo, Japan, August 18-19, 2002.

[4] Hua, Z., Online Analytical Mining of Association Rules, MSC Thesis Simon Fraser, December 1998.

[5] Goebel, M., and Gruenweld, L., A Survey of Data Mining and Knowledge Discovery Software Tools, Proceedings of ACM SIGKDD, Volume 1, Issue 1, Page 20, June 1999.

[6] Bing, L., Wynne., et.al., Mining Association Rules With Multiple Minimum Support, Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-99), August 15-18, 1999.

[7] Pasquier, N., Bastide, Y., et.al., Discovering Frequent Closet Item Sets For Association Rules, 1997.

[8] Jian, P., Jiawei, H., et.al., Closet: An Efficient Algorithm For Mining Frequent Closet Item Sets, 1997.

[9] Yongjian, F., Discovery of Multiple Level Rules From Large Databases, PhD Thesis Simon Fraser, July 1996.

[10] Sarjon, D., Mohd, N., Intelligent Mining Association Rules From Large Indonsistent Databases, Journal of Information Technology, Volume 14, Number 2, December 2002
[11] Sarjon, D., Mohd, N., Intelligent Mining Association Rules Method", International Journal of Information Technology, 2002
[12] Huan, L., and Rudi, S., et.al., Chi2: Feature Selection and Discretization of Numeric Attributes, 1998.
[13] Wesley, W.C., and Kuorong, C., Abstraction of High Level Concepts From Numerical Values in Databases, 1997.

Biography

Dr. Sarjon Defit, S.Kom, MSc is a lecturer of Information System Department at Faculty of Computer Science University Putra Indonesia "YPTK" Padang, West Sumatera, Indonesia.

The President of Yayasan Perguruan Tinggi Komputer (YPTK) Padang appointed me as the Rector of University Putra Indonesia "YPTK" Padang on 9th May 2009, for a term of 4 years, until 8th May 2013

Education / Academic Qualification

Doctor of Philosophy (Ph.D), Universiti Teknologi Malaysia (UTM), August 2003
Master of Science (M. Sc), Universiti Teknologi Malaysia (UTM), August 1998
Bachelor of Science (B.Sc), University Putra Indonesia "YPTK" Padang, September 1993