IT Journal Research and Development (ITJRD)
Vol.7, No.1, August 2022, E-ISSN : 2528-4053 | P-ISSN : 2528-4061
DOI : 10.25299/itjrd.2022.8912

Classification Analysis Of Unilak Informatics Engineering Students Using Support Vector Machine (Svm), Iterative Dichotomiser 3 (Id3), Random Forest And K-Nearest Neighbors (Knn)

Hadi Sunaryanto[1], MHD. Arif Hasan[2], Guntoro[3]
Faculty of Computer Science, Lancang Kuning University
sunaryantohadi2@gmail.com[1], m.arif@unilak.ac.id[2], guntoro@unilak.ac.id[3]

Article Info

Article history:

Received Feb 05, 2022
Revised Mar 29, 2022
Accepted Aug 11, 2022

Keyword:

Classification, Study Period
Support Vector Machine (SVM)
Itterative Dichotomiser 3
Random Forest
K-Nearest Neighbor (KNN)

ABSTRACT

This research is entitled "Classification Analysis of the Study Period of Informatics Engineering Study Program Students at Unilak with the Support Vector Machine (SVM), Iterative Dichotomiser 3 (ID3), Random Forest and K-Nearest Neighbors (KNN) method. An attempt to understand whether there are factors that influence the length of a student's study period. Basically, the length of the study period is not a measure of a student's non-academic academic ability, but most people judge that students with a study period of more than 8 semesters or long are not good. Therefore, the researcher chose to classify the factors that affect the length of the student's study period at the Faculty of Computer Science, Lancang Kuning University. This study uses 4 (four) calculation methods. With the several methods used, the authors can compare the results of the four calculation methods so that they can determine which method is better calculated. The result of this research is a comparison between 4 (four) calculation methods in determining which method has good classification ability.

Corresponding Author:

MHD. Arif Hasan
Faculty of Computer Science
Lancang Kuning University
Jalan Yos Sudarso Km. 8 Tassel, Pekanbaru
Email: Info@unilak.ac.id

1. INTRODUCTION

Higher education is a level of education after secondary education which includes diploma, bachelor, master, specialist and doctoral education programs organized by universities. Graduation is the final result of the process of teaching and learning activities while attending lectures at universities. Lancang Kuninig University (UNILAK) is a private university in Riau with 9 (nine) faculties. One of them is the Faculty of Computer Science, which is abbreviated as Fasilkom. Fasilkom consists of undergraduate study programs, namely Informatics Engineering and Information Systems. The length of study for the undergraduate program according to the academic regulations of Lancang Kuning University is scheduled for 8 semesters (4 years) or less than 8 semesters (4 years) and no later than 14 semesters (7 years). Every year, Lancang Kuning University holds a graduation ceremony in 2 periods, namely April and October. In 2 graduation periods, the number of graduates with the number of new students is not comparable. This causes the number of students to increase. The length of a student's study period may be influenced by many factors. Factors that are estimated to influence on time graduation include Graduation Achievement Index (GPA), gender, scholarships, part time work, organization, and university entrance paths.

In this research, the writer tries to compare the classification of student study period with four methods, namely Support Vector Machine, Iterative Dichotomiser, Random Forest and K-Nearest neighbor.

2. RESEARCH METHOD

The classification stages of each method are different, the algorithm in calculating the student study period is determined from the test data that is tested with test data.

In this study, the authors use 4 (four) methods in classifying student study periods, namely Support Vector Machine, Iterative Dichotomiser 3, Random Forest and K-Nearest Neighbor.

2.1. Support Vector Machine

Support Vector Machine (SVM) is a learning that leads to quadratic programming with linear constraints. Based on the structured risk minimization principle, SVM seeks to minimize the upper bound of generalization error instead of empirical error, so that the new predictive model effectively avoids the over-fitting problem[1]. In addition, the SVM model works in a high-dimensional feature space formed by the nonlinear mapping of the N-dimensional input vector x into the K-dimensional feature space (K > N) through the use of a nonlinear function (x).

The best separating hyperplane (decision boundary) between the two classes can be found by measuring the margin of the hyperplane and finding its maximum point[2]. Margin is the distance between the hyperplane and the closest data from each class. This closest data is called the support vector[3]. The solid line in the image above shows the best hyperplane, which is located right in the middle of the two classes, while the data of circles and squares that are crossed by the margin line (dotted line) is a Support Vector. The effort to find the location of this hyperplane is the core of the Support Vector Machine (SVM) training process. According to Sentosa[4]SVM can be connoted to

$$f(x) = w^T x + b \quad (1)$$

2.2. Iterative Dichotomiser 3

The ID3 Gain algorithm formula measures how well an attribute separates the training example into the target class[5]. The attribute with the highest information will be chosen in order to identify the gain, first we use the idea of information theory called entropy[6], [7]. Entropy measures the amount of information contained in an attribute. The following is the equation for the entropy formula for the ID3 .

$$Entropy\ (S) = \ -p + log_2 p - p + log_2 p \quad (2)$$

Description :

S :sample space (data) used for training
P+ :the number of positive solutions (supporting) the sample data for certain criteria
P- :the number of negative solutions (supporting) the sample data for certain criteria

The formula to calculate the Gain on the ID3.

$$Gain\ (S, A) = Entropy(S) - \sum \frac{|Sv||Sv|}{|S||S|} \quad (3)$$

Description :
S= sample space (data) used for training
A= attribute
V= a possible value for attribute A
|Sv|= number of samples for the value of V
|S|= total sample data

2.3 Random Forest

The random forest method is an extension of the CART method. Classification And Regression Tree (CART) (Han, 2012). CART uses information gain to measure the selection of attributes to be used at each node of a tree. Let's say N is the node that will be used to separate each

class by using attributes from dataset D. The attribute with the highest information gain will be used to split the N nodes. The formula for finding the information gain value can be found as below.

$$Gain(A) = Info\ (D) - Info_A(D) \quad (4)$$

Where is the value$Info\ (D)$searched by using the formula:

$$Info\ (D) = \sum_i^m pi\ log_2\ (pi) \quad (5)$$

Description:
m : number of target classes
pi : probability of occurrence of class i on partition D

While value $Info_A(D)$searched by using the formula:

$$Info_A(D) = \sum_{j=1}^{v} \frac{|Dj|}{|D|}\ x\ Info\ (Dj) \quad (6)$$

Description:
v : number of partitions
Dj : total partition to j
D : number of tuples/ rows in all partitions

2.4 K-Nearest Neighbor

The KNN method algorithm is very simple, works based on the shortest distance from the query instance to the training sample to determine the KNN. The training sample is projected into a multidimensional space, where each dimension represents a feature of the data. This space is divided into sections based on the training sample classification. A point in this space is marked class c if class c is the most common classification found in the k closest neighbors of that point. Near or far neighbors are usually calculated based on the Euclidean Distance which is represented as follows:

$$D(a,b) = \sqrt{\sum_{k=1}^{d} (a_k - b_k)^2} \quad (7)$$

3. DISCUSSION

In the early stages of this study, the authors collected data on students who were in semester 1 (one) to semester 8 (eight) with the limitation of Informatics Engineering students from the 2016 to 2017 class and the factors that influence the length of the student's study period using a questionnaire. The data taken is in the form of the last GPA for 7 (seven) and 8 (eight) semester students. While the data from the questionnaire table in the form of several choices as follows.

Table1. Questionnaire score

Criteria	Yes	Not
Are you involved in campus organizations?	1	2
Are you a scholarship recipient?	1	2
Do you work while studying (part time)?	1	2

3.1 Support Vector Machine

In the Support Vector Machine method, the first step is to look for HyperPlane or the highest point in finding the dividing line between data. The HyperPlane formula is:

$$\text{Hyperplane} = w_1 \cdot x_1 + w_2 \cdot x_2 + b = 0 \quad (8)$$

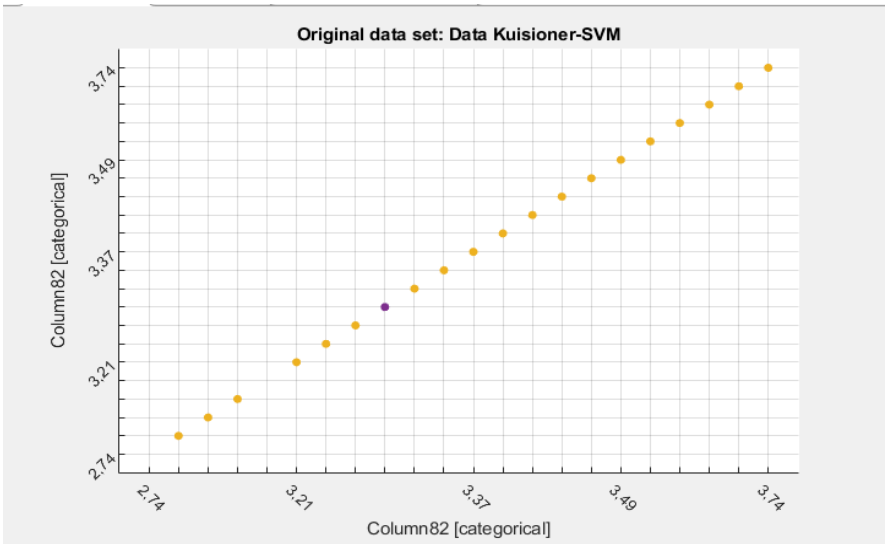

Figure1. Support Vector Machine method deployment form in Matlab

Search using the Matlab application, where the initial data distribution is located where the VarName16 line is the dividing line between the vector values of VarName 15 and VarName 17. The classification value of the Support Vector Machine method is 94,4%.

3.2 Iterative Dichotomiser

Iterative Dichotomiser method is a calculation method based on a decision tree. Where the value of each branch of the decision tree determines every possibility of each branch of the decision tree. The main purpose of the ID3 calculation / algorithm is to determine the size of the gain value of an entropy. Where entropy is a measure of the amount of information in the attribute. Entropy can be formulated as follows:

$$Entropy\ (S) = \ -p + log_2 p - p + log_2 p \quad (9)$$

Table2. Number of Initial Data Value

	Total	Yes	Not
Organization	145	86	59
Scholarship	145	61	84
Part Time	145	63	82

Table 3. Initial Data Value

List of Questionnaire Table Values				GPA score
1	1	2	1	3.50
1	2	2	1	3.45
1	2	2	1	3.74
2	1	1	1	3.68
2	2	2	2	3.41

2	1	2	2	3.21
2	2	1	1	3.50
1	2	1	1	3.45
2	2	1	1	3.68
2	2	1	1	3.41

Table 4. Entropy Data Formula

$^{86}/_{145}\ \log 2\ ^{86}/_{145} + ^{59}/_{145}\ \log 2\ ^{59}/_{145}$
$^{61}/_{145}\ \log 2\ ^{61}/_{145} + ^{84}/_{145}\ \log 2\ ^{84}/_{145}$
$^{63}/_{145}\ \log 2\ ^{63}/_{145} + ^{82}/_{145}\ \log 2\ ^{82}/_{145}$

After get the entropy value, then look for the gain value from the entropy. Gain is the selection of information by measuring how well an attribute separates the training data into target classes. Gain can be formulated as follows:

$$Gain\ (S, A) = Entropy(S) - \sum \frac{|Sv||Sv|}{|S||S|} \quad (9)$$

Then the formula for gain with entrophy and existing data can be written as:

Table 5. Gain Data Formula

(-0.974842133) – 86/145*(-0.974842133) + (-0.974842133) – 61/145*(-0.974842133) + (-0.974842133) – 63/145*(-0.974842133)
(-0.981773634) – 86/145*(-0.981773634) + (-0.981773634) – 61/145*(-0.981773634) + (-0.981773634) – 63/145*(-0.981773634)
(-0.987578748) (-0.987578748) – 86/145*(-0.987578748) + (-0.987578748) – 61/145*(-0.987578748) + (-0.987578748) – 63/145*(-0.987578748)

From the results of the search for entrophy and gain, the following values are obtained:

Table 6. Entropy and Gain Data Value

	Entropy	Gain
Organization	-0.974842133	-0.380595975
Scholarship	-0.981773634	-0.387527476
Part Time	-0.987578748	-0.393332591

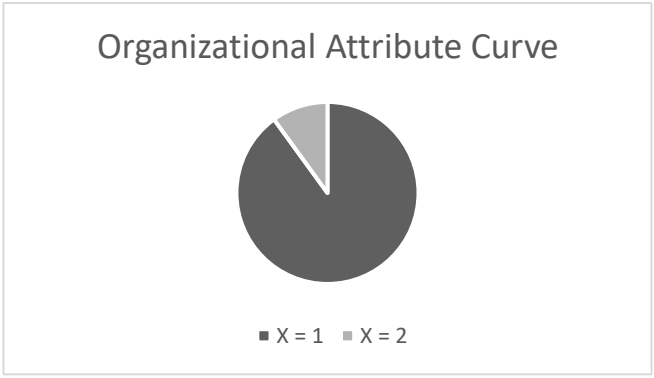

Figure 2. Organisation Atribute Diagram

From the results of the comparison of the dataset and training data on the ID3 method, the results obtained are 90 %

3.3 Random Forest

The Random Forest method is a collection of several trees[8]. Where each tree depends on the pixel value in each vector which is taken randomly and independently. Random Forest does not tend to overfit and can process quickly, making it possible to process as many trees as the user wants[9]. In this study, the author uses a tool in the form of WEKA software. The test results show that the Random Forest method is complete by using 10 trees. The estimation result of out of bag error = 0%. The results of the evaluation of the split test are shown in the image below:

Correlation coefficient	1.6	
Mean absolute error	0	
Root mean squared error	0	
Relative absolute error	100	%
Root relative squared error	100	%
Total Number of Instances	71	

Figure 3. Random Forest Classification

With these results, it can be ascertained that the accuracy of this method on training data is 100%.

3.4 K-Nearest Neighbor

The K-Nearest Neighbor (KKN) algorithm is an algorithm used to classify an object, based on the k value of the training data that is closest to the object. The condition for the value of k is that it cannot be greater than the number of training data and the value of k must be odd and more than one (1).

The KNN method algorithm is very simple, works based on the shortest distance from the query instance to the training sample to determine the KNN[10]. The training sample is projected into a multidimensional space, where each dimension represents a feature of the data. This space is divided into sections based on the training sample classification. A point in this space is marked class c if class c is the most common classification found in the k closest neighbors of that point. Near or far neighbors are usually calculated based on the Euclidean Distance which is represented as follows:

$$D(a,b) = \sqrt{\sum_{k=1}^{d} (a_k - b_k)^2} \quad (11)$$

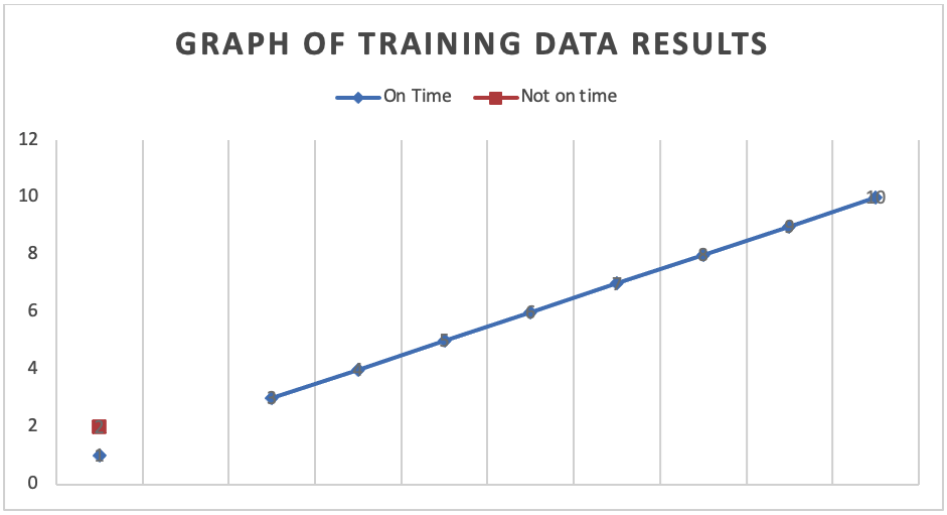

Figure 3. After obtaining the results, it can be seen that the test results have an accuracy rate of 90%.

4. CONCLUSION

The conclusion of this study is that each of the Support vector machine, Iterative dichotomiser 3, Randome forest and K-nearest neighbor algorithms is able to classify the existing data. The Random forest algorithm is the algorithm with the best level of accuracy compared to the other three algorithms with a percentage of data accuracy of 100%. Following the Support vector machine algorithm with 94.4% and Iterative dichotomiser and K-nearest neighbor with a percentage value of 90% each.

REFERENCES

[1] R. Purnamasari, M. Fairuzabadi, and A. Riyadi. "Sistem Pengecekan Plagiasi Judul Tugas Akhir Menggunakan Algoritma Winnowing di Fakultas Sains dan Teknologi Universitas PGRI Yogyakarta." In *Seri Prosiding Seminar Nasional Dinamika Informatika*, vol. 5, no. 1, 2021.

[2] R. K. Wibowo, and K. Hastuti. "Penerapan Algoritma Winnowing Untuk Mendeteksi Kemiripan Teks Pada Tugas Akhir Mahasiswa." *Techno. Com*, vol. 15, no. 4, pp. 303-311, 2016.

[3] M. N. Khidfi, and J. Y. Sari. "Rancang bangun aplikasi pendeteksian kesamaan pada dokumen teks menggunakan algoritma enhanced confix stripping dan algoritma winnowing." *no. September* 2018.

[4] A. H. Purba and Z. Situmorang. "Analisis Perbandingan Algoritma Rabin-Karp Dan Levenshtein Distance Dalam Menghitung Kemiripan Teks." *Jurnal Teknik Informatika UNIKA Santo Thomas*, vol. 2, no. 2, pp. 24-32. 2017.

[5] J. Pierce and C. Zilles. "Investigating student plagiarism patterns and correlations to grades." In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, pp. 471-476. 2017.

[6] A. H. Pratomo, and A. P. Suryotomo. "Implementasi pengecekan plagiarisme proposal tugas akhir mahasiswa teknik informatika UPN Veteran Yogyakarta." In *Seminar Nasional Informatika (SEMNASIF)*, vol. 1, no. 1, pp. 221-229, 2020.

[7] N.I. Kurniati, A. Rahmatulloh and R. N. Qomar. "Web scraping and winnowing algorithms for plagiarism detection of final project titles." *Lontar Komputer: Jurnal Ilmiah Teknologi Informasi*, *10*(2), pp.73-83. 2019.

[8] N. Alamsyah. "Deteksi Plagiarisme Tingkat Kemiripan Judul Skripsi dengan Algoritma

Winnowing." *Technologia: Jurnal Ilmiah.* Vol. 8, no. 4, pp. 205-13, Oct, 2017.
[9] N. F. Ulfa, M. Mustikasari. "Pembuatan Aplikasi Pengukuran Tingkat Kemiripandokumen Berbasis Web Menggunakan Algoritma Winnowing." *Jurnal Ilmiah Informatika Komputer*. Vol 21, no. 2, Apr, 2017.
[10] A. P. Tjiawi, D. E. Herwindiati, and L. Hiryanto. "Perancangan aplikasi pendeteksi tingkat kesamaan antar dokumen dengan algoritma winnowing." *Computatio: Journal of Computer Science and Information Systems*, vol. 2, no. 1, pp. 36-44, 2017.
[11] S. Sunardi, A. Yudhana, and I. A. Mukaromah. "Implementasi Deteksi Plagiarisme Menggunakan Metode N-Gram Dan Jaccard Similarity Terhadap Algoritma Winnowing." 2018.
[12] A. Setiawan. "Implementasi algoritma winnowing untuk deteksi kemiripan judul skripsi studi kasus stmik budidarma." *Informasi dan Teknologi Ilmiah (INTI)*. vol 4, no. 2, 2017.
[13] M. Maskur, D. Q. Putra, and N. Hayatin. "Deteksi Kemiripan Dokumen Pengajuan Proposal Menggunakan Algoritma Biword Winnowing Pada Sistem Informasi Penelitian Dan Pengabdian." *Jurnal Repositor*, vol. 2, no. 5, pp. 571-582, 2020.
[14] N. Nurdin and A. Munthoha. "Sistem Pendeteksian Kemiripan Judul Skripsi Menggunakan Algoritma Winnowing." *InfoTekJar: Jurnal Nasional Informatika dan Teknologi Jaringan*.;vol. 2, no. 1, pp. 90-97, Sep, 2017.
[15] F. R. N. Wulan, A. Kunaefi, and A. Permadi. "Deteksi Plagiasi Dokumen Skripsi Mahasiswa Menggunakan Metode N-Grams Dan Winnowing." *Simetris: Jurnal Teknik Mesin, Elektro Dan Ilmu Komputer*, vol 9. no. 2, pp. 1021-1032, 2018.
[16] L. Sugiarto, C. Mulyadi, S. Rihastuti. "Analisa Algoritma String Matching Dan Winnowing Untuk Deteksi Kemiripan Judul Tugas Akhir Perguruan Tinggi." *Jurnal Teknologi Informasi*, vol. 6, no. 2. pp. 97-106, 2020.
[17] Y. Nurdiansyah and F. N. Muharrom. "Implementation of winnowing algorithm based K-gram to identify plagiarism on file text-based document." In *MATEC Web of Conferences*, vol. 164, p. 01048. EDP Sciences, 2018.
[18] F. Abror. "Implementasi Algoritma Winnowing pada Deteksi Penjiplakan." PhD diss., Universitas Muhammdiyah Jember, 2016.