C4.5 DECISION TREE IMPLEMENTATION IN SISTEM INFORMASI ZAKAT (SIZAKAT) TO AUTOMATICALLY DETERMINING THE AMOUNT OF ZAKAT RECEIVED BY MUSTAHIK

David Bayu Ananda and Ari Wibisono

Faculty of Computer Science, Universitas Indonesia,
Kampus Baru UI Depok, Jawa Barat, 16424, Indonesia

E-mail: david.bayu@ui.ac.id

Abstract

In general, Zakat Information Systems is established to manage the zakat services, so that the data can be well documented. This study proposes the existence of a feature that will determine the amount of zakat received by Mustahik automatically using C4.5 Decision Tree algorithm. This feature is expected to make the process of determining the amount of zakat be done easy and optimal. The data used in this study are the data taken from Masjid An-Nur, Pancoran, South Jakarta. The experiment results show that the proposed feature produces an accuracy rate over 85%.

Keywords: *C4.5 algorithm, classification, data mining, decision tree, SiZakat*

Abstrak

Pada umumnya Sistem Informasi Zakat dibangun untuk mengelola pelayanan zakat, sehingga data dapat terdokumentasi secara baik. Penelitian ini mengajukan adanya suatu fitur penentuan jumlah zakat yang akan diterima Mustahik secara otomatis dengan memanfaatkan algoritma *C4.5 Decision Tree*. Dengan adanya fitur tersebut diharapkan proses penentuan jumlah zakat dapat dilakukan secara mudah dan optimal. Data yang digunakan dalam penelitian ini merupakan data yang diambil dari Masjid An-Nur Pancoran Jakarta Selatan. Hasil dari eksperimen yang dilakukan menunjukkan fitur yang diajukan menghasilkan tingkat akurasi lebih dari 85%.

Kata Kunci: *Algoritma C4.5, data mining, decision tree, klasifikasi, SiZakat*

1. Introduction

Zakat is the third pillar of Islam that must be fulfilled by every Muslim in the world who are qualified. Zakat is divided into two types, namely: Zakat Fitrah and Zakat Maal (wealth). Zakat Fitrah is an obligatory zakat issued by Muslims during Eid Ramadan. Zakat Maal is issued by Muslims associated with the property or anything that can be owned and utilized. In general, zakat is also a form of manifestation of social solidarity, binding kinship between the people and the nation, and may unify the gap between strong and weak groups.

Management of Zakat can be seen in two major processes. The first process is the process of collecting zakat paid by the payer or referred to muzakki. Muzakki can pay through mosques around him or through Unit Pelayanan Zakat (UPZ). After that, Zakat will be managed through the recording and the calculation for every mosque committee and UPZ. The second process is the process of distributing zakat to the people or entities that have rights to receive zakat or commonly referred to mustahik.

There are 8 groups that are eligible to receive zakat [1] :

- Fakir, people who do not own property and do not have adequate income to meet their needs;
- Miskin, people who own property and decent income for him, but their income is not sufficient to meet their needs;
- Amil Zakat, people who carry out the activities of collecting zakat, administrative management, and utilization of zakat;
- Mualaf, people who recently embrace Islam and need assistance to further strengthen their faith in Islam;
- Hamba sahaya, slaves who want to liberate themselves;
- Gharimin, people who owes money for his own good or for the community in

order to implement obedience and goodness;
- Sabilillah, efforts and activities of an individual or entity that aims to uphold the interests of the good of religion or race;
- Ibnu Sabil, people who run out of stock or the cost of the trip which purpose is for the good of society and the religion of Islam.

In planning the quota of Zakat Fitrah per family, Zakat Fitrah committee should plan a fair share of this zakat. Data of Mustahik is needed for planning allocation of Zakat Fitrah which can be used as a reference. Zakat Fitrah's allocation per family will be very difficult for the committee to complete the planning process because there are a lot of data and the number of mustahik families. In addition, in order to be fair, zakat committees should have a provision in the distribution of zakat.

We proposed implementation of C4.5 decision tree algorithm to determine the amount of zakat received by Mustahik, so that the allocation of zakat will be done automatically. We choose C4.5 decision tree because based on the results of preliminary experiments conducted using the data mining tool Weka [2] and the results of research that has been done by several other researchers, namely Surbhi Hardikar et al. [3] and Aman Kumar Sharma et al. [4], the level of accuracy and timing modeling owned by C4.5 algorithm is better than the other algorithms and has lower error rate.

2. Literature Review

2.1 Model-View-Controller (MVC) in CodeIgniter

CodeIgniter is a PHP framework that exists today. CodeIgniter is developed by Rick Ellis [5]. The purpose of the CodeIgniter framework is to produce a framework that can be used in website development so the development of a website is faster than by coding manually. This framework is supported by many libraries that have been provided in the CodeIgniter, so the people who develop website using CodeIgniter framework can focus on website development by minimizing the number of lines of code needed for various purposes of making a website.

CodeIgniter apply the concept of Model-View-Controller (MVC). MVC has a goal to create dynamic websites. As the name implies, the concept of MVC consists of three parts, namely Model, View, and Controller. The pattern of this design has been around since 1979 when it was first described by Trygve Reenskaug from Norway [5].

Model represents data structures. All processes related to retrieval, additions, and changes to the information in the database are done on this file. View is intended to display the user information that is derived from the model. View usually consists of a full web page containing HTML. However, in CodeIgniter, a view can contain snippets of web pages (fragment) from the HTML. Controller is the liaison between the model and the view. Controller performs data changes to the model and displays dynamic information obtained through the model into the view.

2.2 Decision Tree

Decision tree and decision rule are data mining methodology that are widely used to search for a solution within a classification problem. Decision tree method changes a very large data into a decision tree model that represents the rule. A decision tree is a structure that can be used to divide a large data set into sets of smaller record by applying a set of decision rules [6]. A decision tree models is used to divide up a collection of heterogeneous data to homogeneous groups with smaller specific target variables [6].

Decision tree classification technique is a supervised learning. The class labels or categories are already defined in the beginning and in the process of making a model using the training data to classify new data. Decision tree itself consists of several parts of the node [7] :

- **Root node**, a node that is at the top of the tree, this node has no incoming branches and has more than one branch; sometimes it does not have a branch at all. This node is usually the most attributes that have the greatest influence on a particular class.
- **Internal node**, a branching node that only has one incoming branch, and has more than one branch coming out.
- **Leaf node**, an end node that only has one incoming branch, and has no branches at all. It also marks the node as a class label.

Figure 2.1 shows three different attributes: X, Y, and Z (oval shape). X is a root node that has the most influence in the tree. Y and Z is an internal node that has decision rule like root node. The square shape is the leaf node that shows the class label.

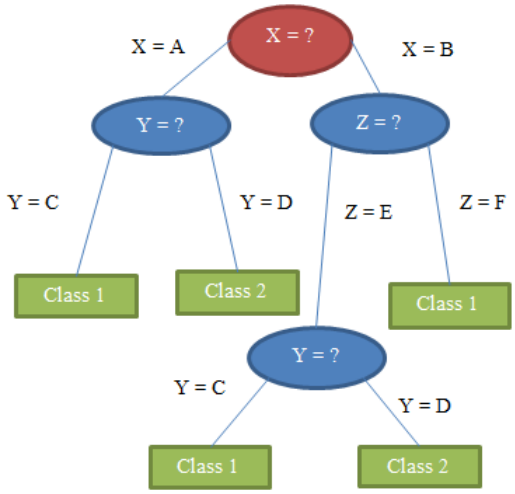


Figure 2.1. Decision Tree Model Illustration

2.3 C4.5 Algorithm

C4.5 algorithm is an algorithm developed by J. Ross Quinlan in 1993 [8]. The C4.5 algorithm is a continued development of the previous algorithm, namely the ID3 algorithm. Therefore, the actual ID3 and C4.5 algorithms have the same basic principles. Some developments are done on the algorithm C4.5 algorithm that makes the C4.5 different from its predecessor, that is:

- Ability to handle attributes with discrete or continuous type.
- Ability to handle empty attribute (missing value).
- Can do pruning on branches.
- The selection is done using a calculation attribute Gain Ratio.

Here are the three principles of the work done by the C4.5 algorithm according to [9] and [8]:

- **First**, perform decision tree construction. The purpose of this decision tree construction algorithm is to create a model of a set of training data that will be used to predict the class of a new data.
- **Second**, the decision tree pruning. Since the results of decision tree construction can be bulky and not easy to "read", the C4.5 algorithm can simplify the decision tree with pruning based on the value of the level of confidence. Pruning also aims to reduce the prediction error rate on new data.
- **Third**, making the rules for the decision tree that has been constructed. The rules are in if-then form that derived from the decision tree by tracing from the root node to the leaf node.

Basic algorithms used by the C4.5 algorithm for decision tree induction is a greedy algorithm that builds decision tree from top to bottom (top-down) recursively by divide and conquer [9][8]. Below is the pseudo code of the C4.5 algorithm:

```
Algorithm: Generate_decision_tree
Input: data training samples; list of attributes;
attribute_selection_method.
Output: decision tree.
Method:
(1) create a node N,
(2) if samples has the same class, C, then
(3)     return N as leaf node with class C label;
(4) if list of attributes is empty then
(5)     return N as leaf node with class label
        that is the most class in the samples.
(6) Choose test-attribute, that has the most
    GainRatio using attribute_selection_method;
(7) give node N with test-attribute label;
(8) for each ai pada test-attribute;
(9)     Add branch in node N to test-attribute =
        ai;
(10)    Make partition for sample si from samples
        where test-attribute = ai;
(11)    if si is empty then
(12)            attach leaf node with the most
                class in samples;
(13)    else attach node that generate by
        Generate_decision_tree (si, attribute-
        list, test-attribute);
(14) endfor
(15) return N;
```

Algorithm 2.1. Decision Tree Model Ilustration

Based on Algorithm 2.1, decision tree model can be illustrated as follows. Assuming that there is one set of training data samples T that have the attributes (A1, A2, A3, ...) and classes consisting of (K1, K2, K3, ...). C4.5 algorithm will run as follows:

• If T is not empty and all the samples have the same class of Ki, then the decision tree for T is a leaf node with label Ki.

• If the attribute is empty then the decision tree contains a leaf node with label Kj where Kj is the highest class in the training samples T.

• If T consists of a sample that has a different class of the partition T into T1, T2, T3, ... Tn. Training samples T partitioned by distinct values of attribute Ak, which at the time became the parent node. Suppose Ak consists of 3 types of values that are: n1, n2, n3, then T will be partitioned into three subsets, namely the value of Ak = n1, n2 = Ak, and Ak = n3.

This process continues recursively with the base case of step 1 and step 2. Attribute that will serve as the parent node or attribute that will

partition the data is done by calculating the gain. Gain is used to select the attributes to be tested based on information theory concepts of entropy.

- **Entropy**

Entropy is a measurement based on the probability that is used to calculate the amount of uncertainty. Info (T) is also known as the entropy of T that is explained by equation 2.1 where T is data training, T_i is subset of T that is partitioned with X attribute.

$$Info_x(T) = \sum_{i=1}^{n} \left(\left(\frac{|T_i|}{|T|} \right) \times Info(T_i) \right) \quad (2.1)$$

$Info_x(T)$ is an important information to classify a tuple of T that based on the partition with X [9].

- **Information Gain**

Information gain in C4.5 algorithm is the change in entropy that occurs after partitioning the data based on an attribute. Entropy can be used to determine the purity of the data partition's result.

$$\boldsymbol{Gain(x) = Info(T) - Info_X(T)} \quad (2.2)$$

Info (T) is the entropy of the training data before it is partitioned by attributes X, and $Info_x$ (T) is the entropy of the training data after it is partitioned by attributes X.

- **Gain Ratio**

The calculation of the information gain still has a number of deficiencies [9]. The use of information gain in ID3 algorithm focuses about testing that produces a lot of output. In other words, attributes that have lots assess are selected as attributes that would partition the training data. Gain Ratio is added to the C4.5 algorithm to overcome the deficiencies in the information gain.

$$\boldsymbol{GainRatio(X) = \frac{Gain(X)}{SplitInfo(X)}} \quad (2.3)$$

X is the attribute of the training data. Gain (X) is the information gain of attributes X. SplitInfo (X) is split information on attribute X that can be derived from equation 2.4.

$$\boldsymbol{SplitInfo_X = -\sum_{i=i}^{n} \left(\frac{D_i}{D} \right) \times \log_2 \left(\frac{|D_i|}{D} \right)} \quad (2.4)$$

SplitInfo expresses entropy or potential information generated by partitioning training data T, into a number of different variables which are owned by the attribute X.

- **Continuous Attribute**

Continuous value is handled by sorting the distinct value from continuous attribute on data training T (v_1, v_2, …, v_n) in ascending order. Threshold $\boldsymbol{t = \lfloor \frac{v_i + v_{i+1}}{2} \rfloor}$ will be calculated from i = 1 until i = n-1. GainRatio will be applied to each threashold so that the chosen threshold is the threshold with biggest GainRatio.

- **Missing Value**

GainRatio(X) will involve the probability of a known value (denoted by F). GainRatio formula (X) can be written as follows [8]:

$$\boldsymbol{GainRatio(X) = F \times \frac{Gain(X)}{SplitInfo(X)}} \quad (2.5)$$

$$\boldsymbol{F = \frac{|known\ value\ of\ X|}{|NilaiX|}} \quad (2.6)$$

- **Pruning**

C4.5 algorithm uses postpruning method. This pruning algorithm is performed from the bottom of the decision tree. Pruning is done by calculating the degree of prediction error in the subtree, another subtree that branches out from the subtree, or leaf nodes of the subtree. If the prediction error rate of the subtree is lower, then the subtree can be replaced with a branch that comes out from the subtree or leaf nodes of the subtree.

2.4 Weka

Weka (Machine Learning Group at the University of Waikato, nd) is a data mining tool that can be used to solve data mining tasks, such as to perform classification, regression, clustering, and association rules. In this study, we use Weka to compare decision tree model and get the accuracy result of SiZakat. We use Weka as a comparison of classification models because Weka is an open source data mining tool and has many algorithms that will be used in determining the amount of zakat prediction.

3. Analysis and Result

3.1 Problem Identification

Zakat fitrah allocation per family must be planned as fair as possible by zakat committee based on the existing provisions. Moreover, zakat management process conducted by every mosque and UPZ should also be transparent and accountable. Currently most of the management of zakat is still conventional, i.e. by recording zakat transaction through the books one by one,

then recapitalizing and re-recording for report generation, all of which are done manually. This method has a high probability of wrong decision due to high amount of data.

We use data from Masjid An-Nur Pancoran, South Jakarta to develop predictive decision tree for zakat allocation to be received by each mustahik. The data consist of 3 attributes: age, Mustahik status, and number of family that will be classifed to 5 classes.

TABLE I
CLASS DESCRIPTION

Class name	Description
Class one	One bag of rice and Rp 100.000,-
Class two	Two bags of rice and Rp 150.000,-
Class three	Three bags of rice and Rp 200.000,-
Class four	Four bags of rice and Rp 250.000,-
Class five	Five bags of rice and Rp 300.000,-

3.2 System Requirement

There are two main actors in this system, Admin and Staff. Administrators manage the Mustahik and the period. Staff has a role in Zakat calculation and recapitulation. In general, the system is built to meet the following specifications:

- The system is able to manage Mustahik
- The system is able to manage period
- The system is able to calculate Zakat
- The system is able to make Zakat recapitulation.

The other requirement is done in previous study.

3.3 Sytem Design

The design of the system is represented in several forms as follows:

- **Logical view**, represented in the class diagram. Logic programming is using the concepts of CodeIgniter MVC (model-view-controller). Therefore, the class diagram is separated into Model and Controller Class Diagram Class Diagram.

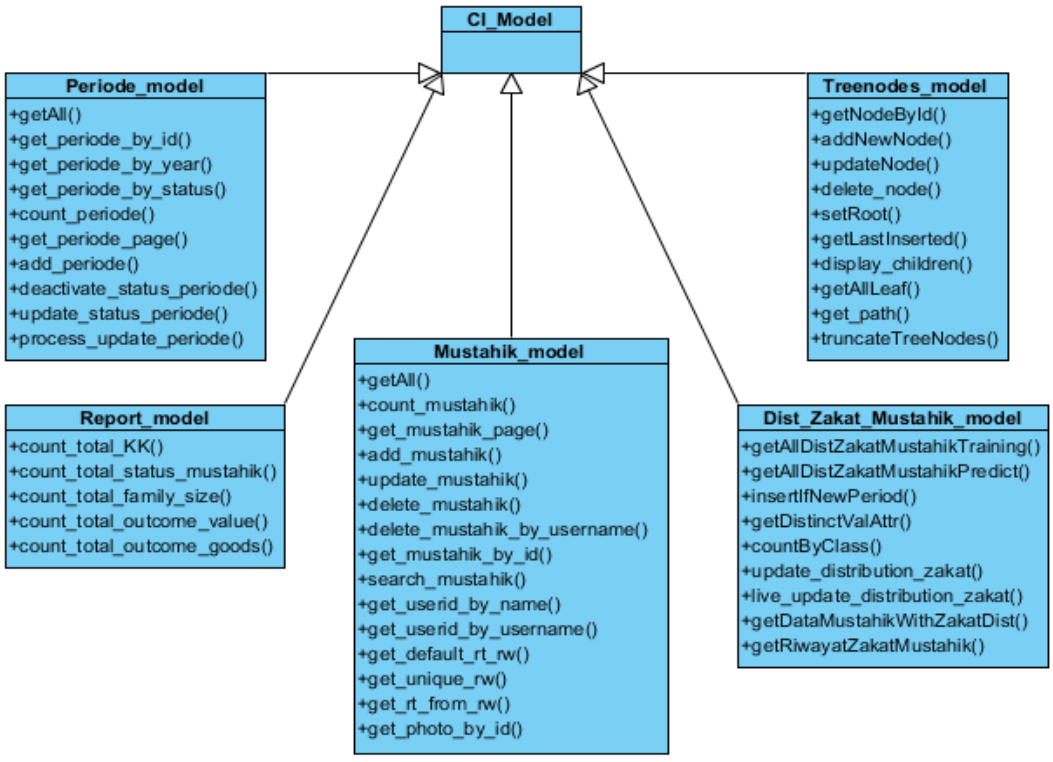


Figure 3.1 Class Diagram for Model

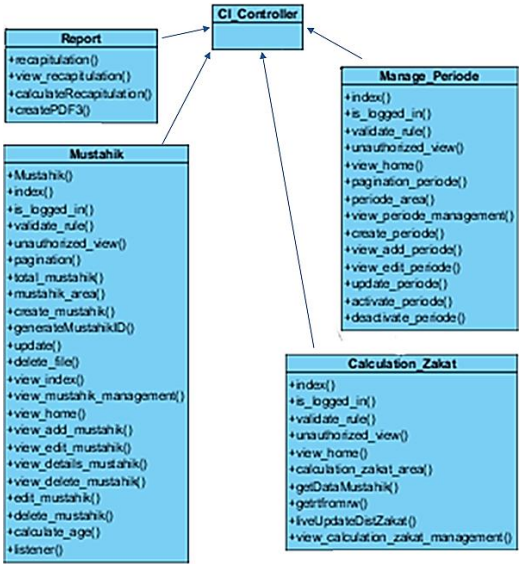


Figure 3.2 Class Diagram for Controller

- **Deployment view**, represented in deployment diagram.

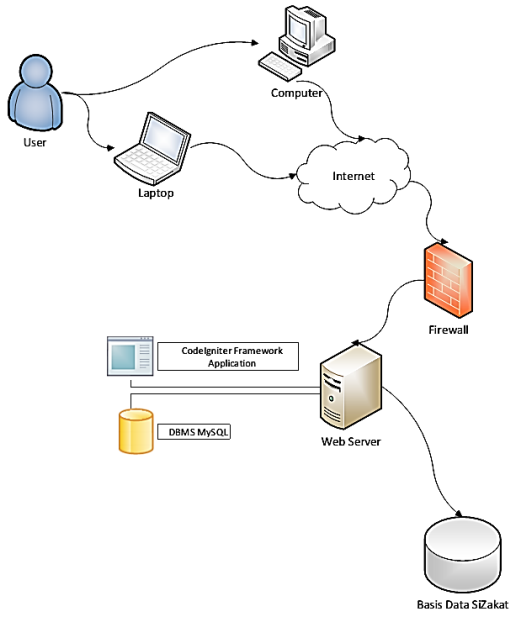


Figure 3.3 Deployment Diagram of SiZakat

- **Data view**, represented in the Enhanced Entity Relationship (EER) diagram.

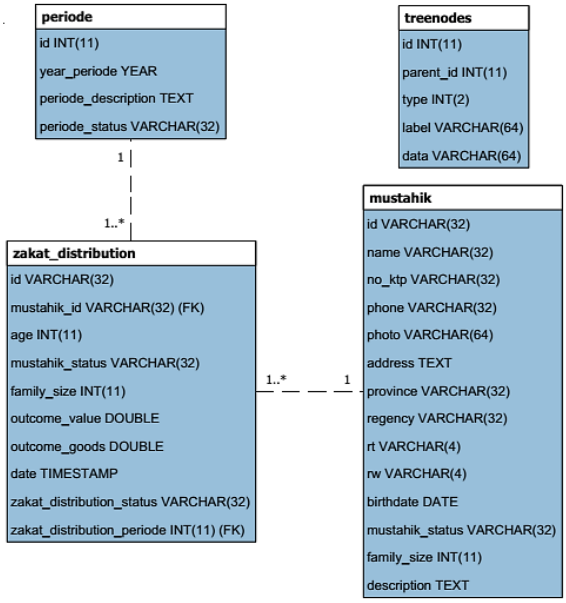


Figure 3.4 EER Diagram of SiZakat for addition feature

3.4 System Implementation

This sistem is implemented using PHP language in CodeIgniter framework. There are 4 modules that we developed after the old system: Period Management Module, Mustahik Management Module, Zakat Calculation Module, and Summary Report Module.

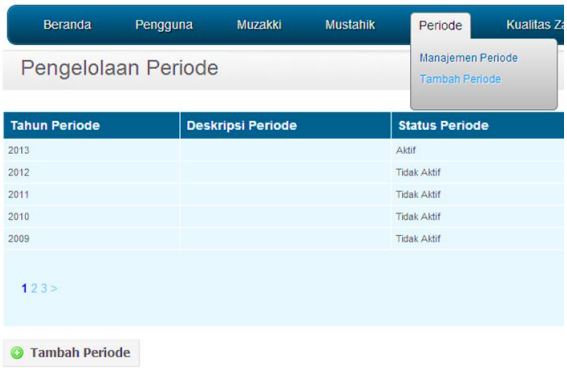


Figure 3.5 Period Management Modules

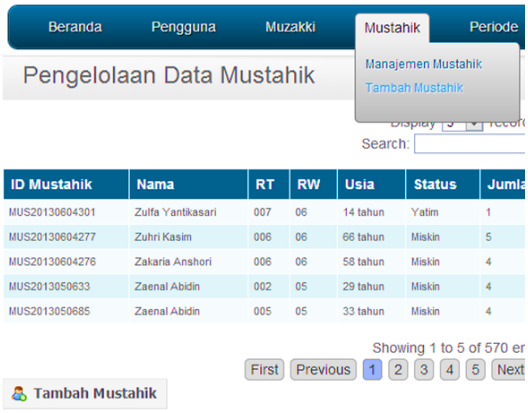
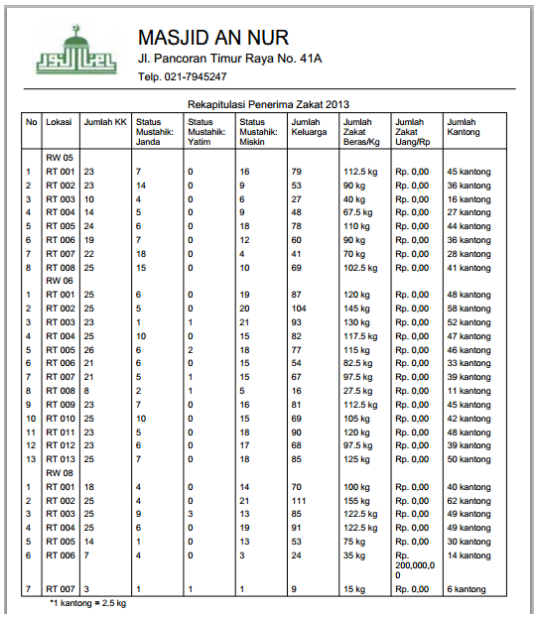


Figure 3.6 Mustahik Management Modules

Figure 3.7 Zakat Calculation Modules



MASJID AN NUR

Jl. Pancoran Timur Raya No. 41A

Telp. 021-7945247

Rekapitulasi Penerima Zakat 2013

No	Lokasi	Jumlah KK	Status Mustahik: Janda	Status Mustahik: Yatim	Status Mustahik: Miskin	Jumlah Keluarga	Jumlah Zakat Beras/Kg	Jumlah Zakat Uang/Rp	Jumlah Kantong
	RW 05								
1	RT 001	23	7	0	16	79	112.5 kg	Rp. 0,00	45 kantong
2	RT 002	23	14	0	9	53	90 kg	Rp. 0,00	36 kantong
3	RT 003	10	4	0	6	27	40 kg	Rp. 0,00	16 kantong
4	RT 004	14	5	0	9	48	67.5 kg	Rp. 0,00	27 kantong
5	RT 005	24	6	0	18	78	110 kg	Rp. 0,00	44 kantong
6	RT 006	19	7	0	12	60	90 kg	Rp. 0,00	36 kantong
7	RT 007	22	18	0	4	41	70 kg	Rp. 0,00	28 kantong
8	RT 008	25	15	0	10	69	102.5 kg	Rp. 0,00	41 kantong
	RW 06								
1	RT 001	25	6	0	19	87	120 kg	Rp. 0,00	48 kantong
2	RT 002	25	5	0	20	104	145 kg	Rp. 0,00	58 kantong
3	RT 003	23	1	1	21	93	130 kg	Rp. 0,00	52 kantong
4	RT 004	25	10	0	15	82	117.5 kg	Rp. 0,00	47 kantong
5	RT 005	26	6	2	18	77	115 kg	Rp. 0,00	46 kantong
6	RT 006	21	6	0	15	54	82.5 kg	Rp. 0,00	33 kantong
7	RT 007	21	5	1	15	67	97.5 kg	Rp. 0,00	39 kantong
8	RT 008	8	2	1	5	16	27.5 kg	Rp. 0,00	11 kantong
9	RT 009	23	7	0	16	81	112.5 kg	Rp. 0,00	45 kantong
10	RT 010	25	10	0	15	69	105 kg	Rp. 0,00	42 kantong
11	RT 011	23	5	0	18	90	120 kg	Rp. 0,00	48 kantong
12	RT 012	23	6	0	17	68	97.5 kg	Rp. 0,00	39 kantong
13	RT 013	25	7	0	18	85	125 kg	Rp. 0,00	50 kantong
	RW 08								
1	RT 001	18	4	0	14	70	100 kg	Rp. 0,00	40 kantong
2	RT 002	25	4	0	21	111	155 kg	Rp. 0,00	62 kantong
3	RT 003	25	9	3	13	85	122.5 kg	Rp. 0,00	49 kantong
4	RT 004	25	6	0	19	91	122.5 kg	Rp. 0,00	49 kantong
5	RT 005	14	1	0	13	53	75 kg	Rp. 0,00	30 kantong
6	RT 006	7	4	0	3	24	35 kg	Rp. 200,000,00	14 kantong
7	RT 007	3	1	1	1	9	15 kg	Rp. 0,00	6 kantong

*1 kantong = 2.5 kg

Figure 3.8 Summary Report in PDF format.

3.5 Testing

Testing of the system is done by conducting a blackbox testing in Zakat Calculation Module. There are 570 data trains used to conduct decision tree model. The decision tree model will be tested by 3 groups of data set (200 data for each group) that are taken randomly using List Randomizer [10].

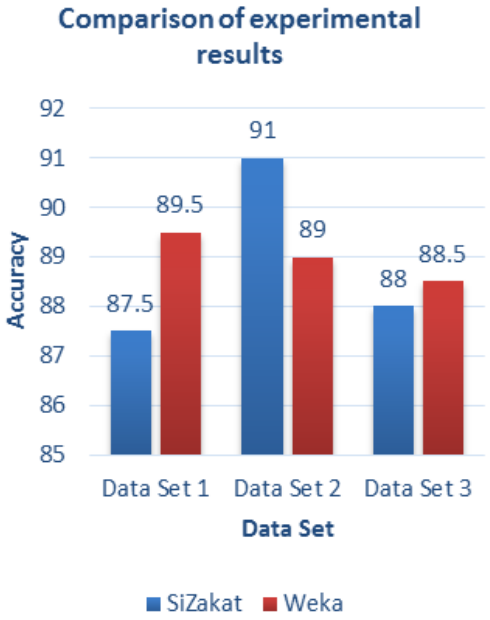


Figure 3.9 Comparison of experimental results.

The results of this test showed that there was no significant difference between the results of data classification using Weka and C4.5 algorithms in SiZakat. Although the results of the decision tree formed by the C4.5 algorithm in Weka and SiZakat are slightly different, but the results of data classification has a fairly high degree of similarity.

4. Conclusion

C4.5 decision tree algorithm is successfully implemented in PHP programming language and CodeIgniter framework. This implementation is a part of additional feature in SiZakat. This feature can determine the amount of Zakat received by Mustahiks. In testing phase, this feature produces an accuracy rate over 85%. It has no significant difference from Weka classifier tools.

We suggest that there should be a more complete data pre-processing before performing formation of decision tree. So, we can ensure that the data was already completed and free from noise or outliers to prevent over fitting.

Reference

[1] Bazis DKI Jakarta, "Petunjuk Praktis Bagi Mustahik," [Online]. Available: http://www.bazisdki.go.id/panduan/zakat12/85-petunjuk-praktis-bagi-mustahik. [Accessed 27 Mei 2013].

[2] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten, "The WEKA Data Mining Software: An Update," *SIGKDD Explorations*, vol. Volume 11, no. Issue 1, pp. 10-18, June 2009.

[3] S. Hardikar, A. Shrivastava and V. Choudhary, "Comparison between ID3 and C4.5 in Contrast to IDS," *VSRD International Journal of Comptuer Science & Information Technology*, vol. Vol. 2, no. 7, pp. 659-667, 2012.

[4] A. K. Sharma and S. Sahni, "A Comparative Study of Classification Algorithms for Spam Email Data Analysis," *International Journal on Computer Science and Engineering (IJCSE)*, vol. Vol. 3 No. 5, pp. 1890-1895, May 2011.

[5] D. Upton, CodeIgniter for Rapid PHP Application Development, Birmingham: Packt Publishing, 2007.

[6] M. J. Berry and G. S. Linoff, Data Mining Techniques For Marketing, Sales, Customer Relationship Management, Second ed., Indianapolis, Indiana: Wiley Publishing, Inc., 2004.

[7] P.-N. Tan, M. Steinbach and V. Kumar, Introduction to Data Mining, 1st ed., Boston: Pearson Addison-Wesley, 2006.

[8] J. R. Quinlan, C4.5: Programs for Machine Learning, USA: Morgan Kaufmann, 1993.

[9] J. Han and M. Kamber, Data mining Concepts and Techniques, Second ed., San Fransisco: Morgan Kauffman, 2006.

[10] Random.org, "RANDOM.ORG," 2012. [Online]. Available: http://www.random.org/lists/. [Accessed 8 June 2013].