

# **PREDIKSI KEPUTUSAN MENGGUNAKAN METODE KLASIFIKASI *NAÏVE BAYES, ONE-R, DAN DECISION TREE***

## ***DECISION PREDICTION USING CLASSIFICATION METHOD NAIVE BAYES, ONE-R, AND DECISION TREE***

**Bahrawi As'ad**

Balai Pengkajian dan Pengembangan Komunikasi dan Informatika Manado

Jalan Pumorow No.76 Manado – Sulawesi Utara

email : bahrawi@kominfo.go.id

(Diterima: 21 Mei 2016; Direvisi: 10 Juni 2016; Disetujui terbit: 27 Juni 2016)

### **Abstrak**

Proses mencari pola atau informasi yang berguna pada suatu kumpulan data dengan menggunakan metode tertentu, saat ini telah menjadi topik yang menarik. Salah satu manfaatnya, yaitu dapat menunjang pengambilan keputusan dalam suatu organisasi baik itu organisasi profit maupun non profit. Pada makalah ini akan dilakukan pengujian terhadap sekumpulan data yang diambil dari kejadian nyata untuk diolah, guna mendapatkan informasi atau pola yang dapat berguna untuk penentuan pengambilan sebuah keputusan. Pengujian pada makalah ini merupakan prediksi terhadap pengguna jasa sebuah operator seluler akan kehadirannya pada suatu acara berdasarkan beberapa indikator, cuaca, jarak relatif terhadap lokasi acara, serta apakah pengguna jasa tersebut merupakan termasuk pelanggan *pasca* bayar atau tidak. Pengujian dilakukan dengan menggunakan tiga metode klasifikasi, yakni *naïve bayes*, *decision tree*, dan *oneR*. Hasil dari percobaan ini bisa menunjukkan prediksi dari setiap percobaan dengan tingkat akurasi prediksi yang berbeda-beda disetiap metode yang digunakan.

Kata kunci: penambangan data, klasifikasi, *naïve bayes*, *oneR*, pohon keputusan.

### **Abstract**

*The searching process for a pattern or useful information in a data set using a particular method, has now become a interesting topics. One of benefits is to support decision making in an organization event profit organizations or non-profit organizations. In this paper will be tested to a dataset taken from real events to be processed, in order to get information or a pattern that can be useful for the decision-making. Testing in this paper is the prediction users mobile service of his presence at an event based on several indicators, the weather, the distance relative to the location of the event, as well as whether the user services are included postpaid subscribers or not. Testing conducted by using three classification methods; Naive Bayes, Decision Tree, and Oner. The results of the experiments are able to show prediction of each trial with different accuracy on every method used.*

*Keywords : data mining, classification, naïve bayes, oneR, decision tree.*

## **PENDAHULUAN**

1. Pada umumnya manusia mampu memprediksi atau memperkirakan suatu kejadian yang belum terjadi hanya dengan seringnya melihat pola dari suatu kejadian secara berulang, contohnya adalah prediksi terhadap pertandingan sepak bola antara tim A melawan tim B, dilihat dari sejarah pertandingan dari kedua tim tersebut dari 2 tahun terakhir, tim A lebih sering menang terhadap tim B. Dengan kenyataan tersebut maka kita bisa memprediksi pertandingan berikutnya akan dimenangkan oleh tim A. Kemampuan ini jika diterapkan kedalam suatu sistem yang berupa perangkat lunak maupun perangkat keras, akan sangat berguna dalam banyak hal. Contoh aplikasinya adalah otomatisasi dalam mengklasifikasikan objek atau barang dalam proses industri, analisis citra satelit, pencarian data citra di dalam halaman web atau basis data, peninjauan kualitas barang, dan lain-lain.
2. Dalam makalah ini akan dibuat sebuah percobaan penambangan data yang diambil dari asumsi contoh kasus nyata dalam kehidupan manusia yaitu memprediksi apakah seseorang yang merupakan pengguna jasa sebuah operator seluler akan hadir pada suatu acara berdasarkan cuaca, jarak relative terhadap acara, serta apakah pengguna jasa tersebut termasuk pelanggan pasca bayar atau tidak, dengan mempelajari kejadian yang serupa pada waktu sebelum percobaan prediksi dilakukan.
3. Penelitian ini dilaksanakan dengan melakukan pengujian dengan menggunakan tiga metode klasifikasi, yaitu *naïve bayes*, *decision tree*, dan

*oneR* terhadap kasus yang akan diprediksi. Pengujian tersebut juga dilakukan dengan menggunakan bantuan tools mesin pembelajaran yang disebut “WEKA (*Waikato Environment for Knowledge Analysis*)”. Tujuan utama penelitian ini adalah untuk memahami, menganalisa dan menerangkan secara ilmiah hasil dari pengujian yang dilakukan serta mengetahui tingkat akurasi dari teknik-teknik penambangan data yang digunakan dalam percobaan.

## **LANDASAN TEORI**

### ***Decision Tree***

*Decision Tree* adalah sebuah struktur pohon, dimana setiap node pohon merepresentasikan atribut yang telah diuji, setiap cabang merupakan suatu pembagian hasil uji, dan node daun (*leaf*) merepresentasikan kelompok kelas tertentu. Level node teratas dari sebuah *Decision Tree* adalah node akar (*root*) yang biasanya berupa atribut yang paling memiliki pengaruh terbesar pada suatu kelas tertentu. Pada umumnya *Decision Tree* melakukan strategi pencarian secara *top-down* untuk solusinya. Pada proses mengklasifikasi data yang tidak diketahui, nilai atribut akan diuji dengan cara melacak jalur dari node akar (*root*) sampai node akhir (daun) dan kemudian akan diprediksi kelas yang dimiliki oleh suatu data baru tertentu.

Beberapa contoh pemakaian *Decision Tree*, yaitu :

- a) Diagnosa penyakit tertentu, seperti hipertensi, kanker, stroke dan lain-lain.
- b) Pemilihan produk seperti rumah, kendaraan, komputer dan lain-lain.

- c) Pemilihan pegawai teladan sesuai dengan kriteria tertentu.
- d) Deteksi gangguan pada komputer atau jaringan computer seperti Deteksi Entrusi, deteksi virus (Trojan dan varians), dan lain-lain.

### **Naive Bayes**

*Naive Bayes* merupakan pengklasifikasian dengan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai *teorema Bayes*. *Teorema* tersebut dikombinasikan dengan "naive" di mana diasumsikan kondisi antar atribut saling bebas [1]. Pada sebuah dataset, setiap baris/dokumen  $I$  diasumsikan sebagai vector dari nilai-nilai atribut  $\langle x_1, x_2, \dots, x_n \rangle$  dimana tiap nilai-nilai menjadi peninjauan atribut  $X_i$  ( $i \in [1, n]$ ). Setiap baris mempunyai label kelas  $c_i \in \{c_1, c_2, \dots, c_k\}$  sebagai nilai variabel kelas  $C$ , sehingga untuk melakukan klasifikasi dapat dihitung nilai probabilitas  $p(C=c_i|X=x_j)$ , dikarenakan pada Naïve Bayes diasumsikan setiap atribut saling bebas, maka persamaan yang didapat adalah sebagai berikut :

Peluang  $p(C=c_i|X=x_j)$  menunjukkan peluang bersyarat atribut  $X_i$  dengan nilai  $x_i$  diberikan kelas  $c$ , dimana dalam Naïve Bayes, kelas  $C$  bertipe kualitatif sedangkan atribut  $X_i$  dapat bertipe kualitatif ataupun kuantitatif.

### **Weka**

*Weka (Waikato Environment for Knowledge Analysis)* adalah aplikasi *Data Mining Open Source* berbasis *Java*. Aplikasi ini dikembangkan pertama kali

oleh Universitas Waikato di Selandia Baru. *Weka* terdiri dari koleksi algoritma *machine learning* yang dapat digunakan untuk melakukan generalisasi/formulasi dari sekumpulan data sampling. Algoritma ini bisa diterapkan secara langsung kedalam data set atau bisa juga dipanggil dari kode *java* kita sendiri. *Weka* memiliki *tools* untuk data *re-processing*, *classification*, *regression*, *clustering*, *association rules*, dan *visualization*. *Weka* mengorganisasi kelas-kelas kedalam paket-paket dan setiap kelas dipaket dapat mereferensi kelas lain dipaket lain.

### **OneR**

*OneR* adalah algoritma klasifikasi sederhana yang membangun pohon keputusan satu level (tingkat). (Widodo 2014). Algoritma *OneR* pertama kali dibuat oleh Robert C. Holte pada tahun 1993. Algoritma *OneR* membuat ranking atribut rerata eror pada data training seperti pengukuran entropi pada C4.5 (Holte 1993). *OneR* memperlakukan nilai numerik atribut sebagai nilai continuous dan menggunakan metode sederhana untuk membagi cakupan nilai ke dalam beberapa interval terpisah. *OneR* membuat satu aturan untuk masing-masing atribut dalam data training kemudian memilih *rule* dengan *error* terkecil yang disebut satu aturan (*one rule*) (Buddhinath & Derry 2005).

Algoritma *OneR* yang menghasilkan *rule* yang sederhana menghasilkan akurasi yang sedikit lebih rendah (3,1%) dibandingkan dengan akurasi algoritma klasifikasi yang terkenal, C4.5. Algoritma C4.5 menghasilkan yang lebih besar dibandingkan dengan OneR. Rule sederhana seringkali dijadikan

alternatif dari sistem rule yang lebih kompleks (Holte 1993). (Widodo 2014)

**METODE PENELITIAN**

Penelitian ini berbentuk pengujian yang dilakukan dengan tiga kali percobaan terhadap satu kasus atau kondisi. Data dari pengujian adalah merupakan asumsi yang diambil dari kejadian nyata yang direpresentasikan dalam bentuk tabel dengan atribut dan record. Pengujian prediksi dilakukan dengan menggunakan tiga teknik klasifikasi yang berbeda yakni *OneR*, *Naïve Bayes* dan *Decision Tree* atau Pohon Keputusan. *Tools* yang digunakan dalam melakukan pengujian adalah *tools* pembelajaran mesin “*WEKA*”.

Prediksi yang dilakukan adalah menentukan apakah seseorang akan tetap datang ke sebuah *event/acara* dengan kondisi berikut:

1. cuaca : apakah berawan, cerah atau hujan,
2. jarak : apakah jarak orang tersebut dari lokasi acara dekat, jauh atau sedang,
3. pemakaian : orang tersebut adalah pelanggan salah satu operator seluler dengan tingkat pemakaian yang normal atau tinggi,
4. pelanggan : orang tersebut adalah merupakan jenis pelanggan pasca bayar atau bukan.

dari keempat kondisi serta beberapa indikator tersebut di atas kita akan memprediksi apakah seseorang tersebut datang ke acara atau tidak, dengan terlebih dahulu mempelajari kejadian sebelumnya dengan berbagai kondisi yang serupa dengan kondisi yang akan diprediksi. Akan disediakan data informasi kejadian serupa sebanyak 14 kejadian di mana data

informasi yang ke 15-lah yang akan kita prediksi.

Berikut data hasil digitalisasi dari kejadian riil pada masing-masing pengujian yang akan dilakukan prediksi :

Tabel 1. Data set pengujian

Cuaca	Jarak relative	Pemakaian	Pelanggan pasca bayar	Datang ke acara
Berawan	Dekat	Normal	Ya	Yes
Berawan	Jauh	Normal	Tidak	No
Berawan	Jauh	Tinggi	Tidak	No
Berawan	Sedang	Tinggi	Ya	Yes
Cerah	Dekat	Normal	Tidak	Yes
Cerah	Jauh	Tinggi	Tidak	No
Cerah	Jauh	Tinggi	Ya	Yes
Cerah	Sedang	Normal	Ya	Yes
Cerah	Sedang	Tinggi	Tidak	No
Hujan	Dekat	Normal	Tidak	No
Hujan	Dekat	Normal	Ya	Yes
Hujan	Sedang	Normal	Tidak	No
Hujan	Sedang	Tinggi	Ya	Yes
Hujan	Sedang	Tinggi	Tidak	Yes
Cerah	Dekat	Tinggi	Tidak	?

Data yang terdapat pada tabel di atas selanjutnya disebut dataset yang terdiri dengan jumlah data *record* 14 dengan 5 atribut, yakni cuaca {berawan,cerah,hujan}, Jarak Relatif {dekat, jauh, sedang), Pemakaian {normal, tinggi}, Pelanggan Pasca Bayar {ya, tidak}, Datang ke acara {yes. No}. Data yang akan diprediksi adalah data record terakhir (data ke-15) yang diberi ‘tanda tanya’, predikasinya adalah menentukan apakah orang yang tersebut datang ke acara dengan kondisi cuaca cerah, jarak yang relatif dekat dari tempat acara, serta pengguna jasa operator seluler yang tinggi namun bukan pelanggan *pasca* bayar.

**Model Penelitian**

Untuk percobaan pertama dengan menggunakan algoritma *classifier OneR*, *pseudocode* atau langkah-langkah yang dilakukan adalah sebagai berikut :

1. Pada data training, hitung kelas C yang mempunyai nilai C untuk atribut A : simpan informasi ini dalam matrik 3 dimensi, COUNT (C,V,A).
2. Kelas default adalah kelas yang memiliki nilai terbanyak pada training. Akurasi kelas default adalah jumlah training pada kelas default dibagi jumlah seluruh data training.
3. Untuk masing-masing *numeric*, A, buat versi nominal dari A dengan menentukan interval nilai yang tetap. Interval ini menjadi nilai versi nominal A.
4. Untuk masing-masing atribut A,
  - Buat sebuah hipotesis yang melibatkan atribut A dengan memilih, untuk masing-masing nilai V dari A, sebuah kelas optimal untuk V.
  - Tambahkan hipotesis yang dibangun ke sebuah aturan yang dinamakan hipotesis. Aturan ini berisi satu hipotesis untuk masing-masing atribut.
5. Pilih aturan yang dari hipotesis memiliki akurasi tertinggi pada data data training (jika ada beberapa, pilih secara acak)

Untuk algoritma *naïve bayes* proses kerjanya melalui dua tahap, sebagai berikut :

1. Learning (Pembelajaran) *Naïve Bayes* adalah algoritma yang termasuk ke dalam *supervised learning*, maka akan dibutuhkan pengetahuan awal untuk dapat mengambil keputusan. Langkah-langkah:
  - Step 1 : Bentuk vocabulary pada setiap dokumen data training.

- Step 2 : Hitung probabilitas pada setiap kategori P(vj).
- Step 3 : Tentukan frekuensi setiap kata wk pada setiap kategori P(wk | vj).

2. *Classify* (Pengklasifikasian)  
Langkah-langkahnya adalah:
  - Step 1 : Hitung  $P(vj) \prod P(wk | vj)$  untuk setiap kategori.
  - Step 2 : Tentukan kategori dengan nilai  $P(vj) \prod P(wk | vj)$  maksimal.

Untuk *classifier decision tree* proses pembuatan *tree* dilakukan sebagai berikut :

```

Algorithm: Generate_decision_tree. Generate a decision tree from the training tuples of data
partition D.
Input:
    • Data partition, D, which is a set of training tuples and their associated class labels;
    • attribute_list, the set of candidate attributes;
    • Attribute_selection_method, a procedure to determine the splitting criterion that "best"
      partitions the data tuples into individual classes. This criterion consists of a splitting_attribute
      and, possibly, either a split_point or splitting_subset.
Output: A decision tree.
Method:
(1) create a node N;
(2) if tuples in D are all of the same class, C then
(3)   return N as a leaf node labeled with the class C;
(4) if attribute_list is empty then
(5)   return N as a leaf node labeled with the majority class in D; // majority voting
(6) apply Attribute_selection_method(D, attribute_list) to find the "best" splitting_criterion;
(7) label node N with splitting_criterion;
(8) if splitting_attribute is discrete-valued and
      multiway splits allowed then // not restricted to binary trees
(9)   attribute_list ← attribute_list - splitting_attribute; // remove splitting_attribute
(10) for each outcome j of splitting_criterion
      // partition the tuples and grow subtrees for each partition
(11)   let D_j be the set of data tuples in D satisfying outcome j; // a partition
(12)   if D_j is empty then
(13)     attach a leaf labeled with the majority class in D to node N;
(14)   else attach the node returned by Generate_decision_tree(D_j, attribute_list) to node N;
endfor
(15) return N;
    
```

Gambar 1. Algoritma *Decision tree* (Sunjana 2010)  
Penjelasan proses pembuatan *tree* sebagai berikut (Jiawei Han 2011).

1. Algoritma dipanggil dengan 3 parameter: D, *attribute\_list*, dan *attribute\_selection\_method*. D merupakan partisi data. Awalnya D merupakan keseluruhan record pada *training* set beserta kelas labelnya. Parameter *attribute\_list* merupakan list atribut yang mendeskripsikan record. *Attribute\_selection\_method* menspesifikasikan prosedur heuristik yang digunakan untuk memilih atribut yang terbaik untuk

- mendiskriminasikan *record* berdasarkan kelas. Prosedur ini menggunakan *attribute selection measure*, seperti *information gain*, *gini index*, atau *minimum descriptor length (MDL)*. Apakah *tree* harus binary atau tidak ditentukan oleh *attribute selection measure*. Beberapa *attribute selection measure*, seperti *gini index* mengharuskan *tree* yang dihasilkan binary.
2. *Tree* dimulai dengan node tunggal, N, merepresentasikan *record* dalam *training set D* (langkah 1)
  3. Jika *record* dalam D semua berada dalam *class* yang sama, maka node D menjadi daun dan diberi label dengan *class* tersebut (langkah 2 dan 3). Langkah 4 dan 5 adalah merupakan *terminating conditions*.
  4. Jika *record* dalam D tidak semua berada dalam 1 kelas yang sama. Algoritma memanggil *attribute\_selection\_method* untuk menentukan *splitting criterion*. *Splitting criterion* merupakan atribut yang digunakan untuk tes pada node N dengan menentukan cara terbaik untuk memisahkan atau mempartisi *record* dalam D ke kelas individual (langkah 6). *Splitting criterion* juga menentukan cabang mana yang harus dibuat dari node N sesuai dari hasil output dari tes. *Splitting criterion* ditentukan sehingga, idealnya, partisi yang dihasilkan semurni mungkin. Partisi disebut murni bila semua *record* yang berada di dalamnya berada dalam *class* yang sama.
  5. Node N diberi label dengan *splitting criterion*, yang berfungsi sebagai tes pada node tersebut (langkah 7). Cabang dibuat dari node N untuk setiap hasil dari *splitting criterion* dan *record D* dipartisi sesuai dengan *splitting* tersebut (langkah 10-11).
  6. Algoritma menggunakan proses yang sama secara berulang untuk membentuk *Decision tree* untuk *record-record* pada partisi yang dihasilkan,  $D_2$ , dari D.
  7. Partisi berulang ini berhenti ketika memenuhi *terminating condition*:
    7. Partisi berulang ini berhenti ketika memenuhi *terminating condition*:
      - Semua *record* pada partisi D (direpresentasikan pada node N) tergabung dalam kelas yang sama (langkah 2 dan 3)
      - Tidak ada atribut lagi di mana *record-record* dapat dipartisi lebih lanjut (langkah 4). Pada kasus seperti ini node N diubah menjadi daun dan diberi label dengan nilai *class* yang paling banyak di D (langkah 5). Alternatif lain distribusi *class* pada *record* di node tersebut dapat disimpan.
      - Tidak ada *record* pada cabang yaitu ketika partisi  $D_2$  kosong (langkah 12). Pada kasus ini, daun dibuat dengan *class* yang paling banyak muncul di D (langkah 13)
  8. Hasil dari *Decision tree* dikembalikan (langkah 15).

## HASIL PENELITIAN DAN PEMBAHASAN

Pengujian prediksi dilakukan dengan menggunakan tiga teknik klasifikasi yang berbeda, yakni *OneR*, *Naïve Bayes* dan *Pohon Keputusan*. Berikut adalah hasil pengujian *Training (full training set)* terhadap dataset.

### Klasifikasi One R

Berikut di bawah ini adalah hasil pengujian menggunakan *training set*,

pembelajaran mesin dilakukan terhadap data yang berjumlah 14 data *record*:

=== Evaluation on training set ===	
Correctly Classified Instances	12
	85.7143 %
Incorrectly Classified Instances	2
	14.2857 %
Kappa statistic	0.72
Mean absolute error	0.1429
Root mean squared error	0.378
Relative absolute error	29.0909 %
Root relative squared error	76.3638 %
Total Number of Instances	14

Berikut di bawah ini adalah hasil pengujian terhadap *supplied test*, di mana data yang dimasukkan adalah data yang akan diprediksi (data ke 15):

=== Evaluation on test set ===	
Correctly Classified Instances	1 100 %
Incorrectly Classified Instances	0 0 %
Kappa statistic	1
Mean absolute error	0
Root mean squared error	0
Relative absolute error	0 %
Root relative squared error	0 %
Total Number of Instances	1

Berikut di bawah ini adalah tabel *confusion matrix* untuk klasifikasi *OneR* pada saat *test set*.

Tabel 2. Hasil confusion matrix (test set) *OneR*

<b>OneR</b>	
=== Confusion Matrix ===	
<i>a b</i> <-- classified as	
0 0	<i>a</i> = yes
0 1	<i>b</i> = no

Berdasarkan data yang terdapat dalam confusion matrix diatas, dapat terlihat prediksi yang benar adalah (bb) atau dengan kata lain data yang dimasukkan bernilai 'no' untuk data record ke-15 dianggap sebagai prediksi yang tepat. Hasil pengujian dengan *OneR* :

Dengan kondisi *cuaca=cerah, jarak relatif=dekat,pemakaian=tinggi, pelanggan pasca bayar=tidak*” maka orang cenderung tidak datang ke *event*.

### Klasifikasi Naïve Bayes

Berikut di bawah ini adalah hasil pengujian *training set* terhadap data yang berjumlah 14 data record:

=== Evaluation on training set ===	
Correctly Classified Instances	12 85.7143 %
Incorrectly Classified Instances	2 14.2857 %
Kappa statistic	0.72
Mean absolute error	0.2502
Root mean squared error	0.3097
Relative absolute error	50.9469 %
Root relative squared error	62.5736 %
Total Number of Instances	14

Berikut di bawah ini adalah hasil pengujian terhadap *supplied test*, di mana data yang dimasukkan adalah data yang akan diprediksi (data ke 15):

=== Evaluation on test set ===	
Correctly Classified Instances	1 100%
Incorrectly Classified Instances	0 0 %
Kappa statistic	1
Mean absolute error	0.4404
Root mean squared error	0.4404
Relative absolute error	78.2889 %
Root relative squared error	78.2889 %
Total Number of Instances	1

Berikut di bawah ini adalah tabel *confusion matrix* klasifikasi *Naïve Bayes* pada saat *test set*

Tabel 3. Hasil confusion matrix (test set) *Naïve Bayes*

<b>NaïveBayes</b>	
= Confusion Matrix =	
<i>a b</i> <-- classified as	
0 0	<i>a</i> = yes
0 1	<i>b</i> = no

Berdasarkan data yang terdapat dalam *confusion matrix* di atas, dapat terlihat prediksi yang benar adalah (bb) atau dengan kata lain data yang dimasukkan bernilai 'no' untuk data *record* ke-15 dianggap sebagai prediksi yang tepat. Hasil pengujian sama dengan klasifikasi *OneR*: Dengan kondisi *cuaca=cerah, jarak relatif=dekat, pemakaian=tinggi, pelanggan pasca bayar=tidak* maka orang cenderung tidak datang ke *event*.

**Klasifikasi Pohon Keputusan J48**

Berikut di bawah ini adalah hasil pengujian *training set* terhadap data yang berjumlah 14 data record:

```

==== Evaluation on training set ====
Correctly Classified Instances 12 85.7143 %
Incorrectly Classified Instances 2 14.2857 %
Kappa statistic 0.72
Mean absolute error 0.2143
Root mean squared error 0.3273
Relative absolute error 43.6364 %
Root relative squared error 66.133 %
Total Number of Instances 14
    
```

Berikut di bawah ini adalah hasil pengujian terhadap *supplied test*, dimana data yang dimasukkan adalah data yang akan diprediksi (data ke 15):

```

==== Evaluation on test set ====
Correctly Classified Instances 1 100 %
Incorrectly Classified Instances 0 0 %
Kappa statistic 1
Mean absolute error 0.25
Root mean squared error 0.25
Relative absolute error 44.4444 %
Root relative squared error 44.4444 %
Total Number of Instances 1
    
```

Berikut di bawah ini adalah tabel *confusion matrix* klasifikasi pohon keputusan j48 pada saat *test set*.

Tabel 4. Hasil confusion matrix (test set) J48

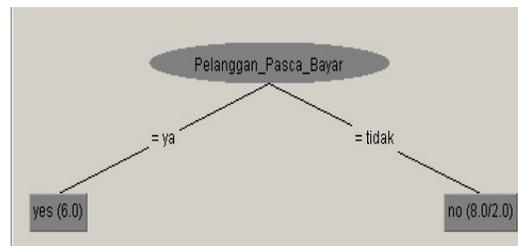
```

J48
=Confusion Matrix =
a b <-- classified as
0 0 | a = yes
0 1 | b = no
    
```

Berdasarkan data yang terdapat dalam *confusion matrix* di atas dapat terlihat prediksi yang benar adalah (bb) atau dengan kata lain data yang dimasukkan bernilai 'no' untuk data *record* ke-15 dianggap sebagai prediksi yang tepat. Jadi nilai 'Datang ke *Event*' pada data record ke-15 tersebut adalah 'no'.

No.	Cuaca Nominal	Jarak_Relatif Nominal	Pemakaian Nominal	Pelanggan_Pasca_Bayar Nominal	predictedDatang_Ke_Event Nominal	Datang_Ke_Event Nominal
1	cerah	dekat	tinggi	tidak	no	no

Gambar 2. hasil Prediksi *arffviewer*



Gambar 3. Hasil visualisasi *tree*

Gambar 2 dan 3 di atas adalah merupakan gambaran hasil prediksi menggunakan klasifikasi pohon keputusan. Perhitungan *gain* dan *entropy* pada metode pohon keputusan adalah sebagai berikut:

Tabel 5. Perhitungan *gain* klasifikasi pohon keputusan – j48

Attribut	Kelas	Jumlah kasus	Datang ke event		Entropy	Gain
			No	Yes		
Total			6	8		
Cuaca						-0.03
	Berawan	4	2	2	0.28	
	Cerah	5	2	3	0.34	
Jarak relative	Hujan	5	2	3	0.34	
	Jauh	4	3	1	0.23	
	Sedang	6	2	4	0.39	
Pemakaian	Dekat	4	1	3	0.23	
	Normal	7	3	4	0.49	0

	Tinggi	7	3	4	0.49	
Pelanggan pasca bayar	Ya	6	0	6	0	0.52
	Tidak	8	6	2	0.46	

Dari hasil tabel di atas dapat diketahui bahwa atribut dengan Gain tertinggi adalah 'pelanggan pasca bayar' yaitu sebesar 0.52. Dengan demikian 'pasca bayar' menjadi node akar. Ada 2 kelas dari atribut 'pasca bayar' yaitu 'ya' dan 'tidak'

Tabel 6. Perbedaan *average precision*, *average recall* dan *time taken*

Classifier	Average Precision	Average Recall	Time Taken (s)
OneR	0.893	0.857	0
Naïve Bayes	0.893	0.857	0
J48	0.893	0.857	0

Tabel di atas menunjukkan rata-rata kedekatan data, rata-rata pemanggilan kembali data serta lama waktu yang digunakan selama proses klasifikasi dari tiga *classifier* yang digunakan dalam percobaan ini.

Berdasarkan hasil data prediksi yang dihasilkan oleh ketiga teknik klasifikasi, ketiganya menghasilkan prediksi yang sama, yaitu bahwa yang menghadiri acara meskipun bukan pelanggan *pasca* bayar namun dengan pemakaian trafik tergolong tinggi dan bertempat tinggal relatif dekat dari lokasi acara tetap diprediksi tidak akan datang ke acara meskipun cuaca cerah. Dengan prediksi seperti itu maka yang paling memungkinkan datang ke acara adalah yang pelanggan *pasca* bayar dengan pemakaian trafik yang tinggi dan tempat tinggal berjarak tidak jauh dari lokasi acara juga dengan kondisi cuaca cerah.

## PENUTUP

### Kesimpulan

Berdasarkan hasil pengujian dan analisis dapat diambil kesimpulan sebagai berikut:

1. Percoobaan yang dilakukan dapat membuktikan bahwa pola kejadian yang terjadi pada kejadian riil dapat ditransformasikan dan diprediksi dengan menggunakan mesin pembelajaran.
2. Bahwa akurasi dan performansi dari mesin pembelajaran sangat tergantung dari data yang ada serta pemahaman akan metode yang diterapkan, dalam penelitian ini diperoleh akurasi prediksi dengan angka yang sama di setiap teknik yang digunakan.
3. Pembelajaran mesin untuk memprediksi suatu pola, akan lebih baik performansinya jika data yang dianalisis atau diolah sangat banyak.

## Saran

Perlu dilakukan pengujian lagi terhadap data yang lebih besar untuk dapat membandingkan performa akurasi teknik pembelajaran mesin yang digunakan.

## UCAPAN TERIMAKASIH

Penulis mengucapkan terima kasih kepada BPPKI Manado, pimpinan beserta jajarannya atas dukungannya pada penelitian ini. Keluarga, teman yang selalu ikut serta memberi motivasi terhadap penelitian ini. Penulis juga berterima kasih kepada staf pengajar Institut Teknologi Sepuluh Nopember Surabaya atas diskusinya yang bermanfaat.

## DAFTAR PUSTAKA

- Jiawei Han, Micheline Kamber, Jian Pei. *Data Mining: Concepts And Techniques 3rd Edition*. Urbana-Champaign: Morgan Kaufmann, 2011.
- Median Yuli Hartanto, Daniel Oranova S, Sarwosri. *Prediksi Umur Bug Dalam Poyek Pengembangan Perangkat Lunak*. Surabaya, 7 27, 2011.
- Munawar. *Pemodelan Visual dengan UML*,. Yogyakarta: Graha Ilmu, 2005.

- Natalius, Samuel. *Metoda Naïve Bayes Classifier dan Penggunaannya pada Klasifikasi Dokumen*. Makalah, Bandung: Makalah II2092 Probabilitas dan Statistika, 2010.
- Oded Z. Maimon, Lior Rokach. *Decomposition methodology for knowledge discovery and data mining : theory and applications*. World Scientific, 2005.
- Selvia Lorena Br Ginti, Reggy Pasya Trinanda. "Teknik Data Mining Menggunakan Metode Bayes Classifier Untuk Optimalisasi Pencarian Pada Aplikasi Perpustakaan (Studi Kasus : Perpustakaan Universitas Pasundan – Bandung." *Jurnal Teknologi dan Informasi UNIKOM, Volume 1 No 6*, 2014: 8-23.
- Silipo, Rosario dan Phil Winters. "big data, Smart Energy, and Predictive Analytics." *Time Series Prediction of Smart Energy Data*, 2013: 1-37.
- Silipo, Rosario. *KNIME and big data*. 2013. <http://www.dataminingreporting.com/blog/knime-and-big-data> (accessed September 28, 2014).
- Sunjana. "Aplikasi Mining Data Mahasiswa Dengan Metode Klasifikasi Decision Tree." *Seminar Nasional Aplikasi Teknologi Informasi 2010 (SNATI 2010)*. Yogyakarta: Universitas Widyatama, 2010. A24 - A29 .
- Weka 3. *Data Mining with Open Source Machine Learning Software in Java*. n.d. <http://www.cs.waikato.ac.nz/ml/weka/index.html> (accessed April 06, 2011).
- Widodo, Pudji. "Rule-Based Classifier Untuk Mendeteksi Penyakit Liver." *Bianglala Informatika Vol . II No 1*, 2014: 71-80.