



## Selektivitas Pemilihan Komputer dengan Pendekatan Pohon Keputusan

A. Maarip<sup>1</sup>, S.N.M.P. Simamora<sup>2\*</sup>

<sup>1,2</sup>Program studi Teknik Informatika, IDE LPKIA, Bandung

<sup>1</sup>190914018@fellow.lpkia.ac.id, <sup>2</sup>snmpsimamora@lpkia.ac.id \*

### Abstract

*The important role of a computer in work activities is solving cases that require fast and precise computing processes. However, the limitations of the required computer specifications must sometimes be taken into account according to the specified needs. Cases that are commonly found in terms of aspects of the availability of computer specifications are related to multimedia, and one that is quite prominent in the current 4G technology trend is e-sport activities, namely online games. The main requirement for running games is video-memory capacity, namely image requirements. The image is based on a dictionary-color data that can be saved, so for a display with fidelity characteristics it certainly requires a large video-memory capacity, so that when the game is running it doesn't get stuttering. A fairly simple method with a decision-tree approach by applying certain techniques or algorithms, in order to obtain the specifications of the computer; based on the component attributes it has (such as processor, main-memory, and video-memory), and the operating system (such as software) is whether it is compatible or vice versa. There are two operating systems involved in the testing process, namely: Windows 10 and Windows 11; the Windows family was chosen for the reason that the majority of users use it and is adapted to games that are commonly run on this variant of the operating system. The decision tree approach is based on the C4.5 algorithm. And in this research a series of tests have been carried out on a number of computers with various characteristics and specifications; with the field of objects tested being games, without specifically limiting whether they run on-line or off-line; but on this research used on-line. There were three sample games involved in the testing with the number of computers processor being selected using seven different specifications. The results show that from a number of computer specifications involved in testing the three game samples, some are compatible. And the results obtained showed that the machine learning model designed was optimal.*

*Keywords: c4.5 algorithm, decision tree approach, selective*

### Abstrak

Peran penting sebuah komputer dalam aktivitas pekerjaan adalah penyelesaian kasus yang membutuhkan proses komputasi yang cepat dan presisi. Namun keterbatasan spesifikasi komputer yang disyaratkan kadangkala harus diperhitungkan menyesuaikan dengan kebutuhan yang ditentukan. Kasus yang umum ditemukan dalam hal aspek ketersediaan spesifikasi komputer berkaitan dengan multimedia, dan salah satu yang cukup menonjol pada *trend* teknologi 4G saat ini adalah kegiatan *e-sport* yakni *game on-line*. Kebutuhan yang utama untuk menjalankan *game* adalah kapasitas *video-memory*, yakni kebutuhan citra. Citra didasar pada kamus-data warna yang dapat disimpan, sehingga untuk tampilan dengan karakteristik *fidelity* tentu membutuhkan kapasitas *video-memory* yang besar, sehingga saat *game* dijalankan tidak tersendat. Metode yang cukup sederhana dengan *decision-tree approach* (pohon keputusan) dengan menerapkan teknik atau algoritma tertentu, sehingga diperoleh untuk spesifikasi komputer tersebut; berdasar komponen atribut yang dimilikinya (yakni: *processor*, *main-memory*, dan *video-memory*), dan sistem operasi (*software*) adalah apakah telah tepat (*compatible*) atau sebaliknya. Ada dua sistem operasi yang dilibatkan dalam proses pengujian yakni: Windows 10 dan Windows 11; dipilihnya keluarga Windows dengan alasan mayoritas penggunaan dan disesuaikan dengan *game* yang umum dijalankan pada varian sistem operasi ini. Pendekatan pohon keputusan didasar pada algoritma C4.5. Dan pada penelitian ini telah dilakukan serangkaian pengujian terhadap sejumlah *local* komputer dengan berbagai karakteristik dan spesifikasinya; dengan bidang objek yang diuji adalah *game*, tanpa spesifik membatasi apakah berjalan *on-line* atau *off-line*; namun pada penelitian ini digunakan yang berjalan secara *on-line*. Ada tiga *sample game* yang dilibatkan dalam pengujian dengan jumlah *processor* komputer yang dilakukan selektivitas sebanyak tujuh spesifikasi berbeda. Hasilnya menunjukkan dari sejumlah spesifikasi komputer yang dilibatkan untuk uji-coba terhadap tiga *sample game* tersebut ada beberapa *compatible* dan sebaliknya. Dan diperoleh hasil bahwa model pembelajaran mesin yang dirancang sudah optimal.

Kata kunci: algoritma c4.5, pendekatan pohon keputusan, selektivitas

## 1. Pendahuluan

Pemilihan komputer yang tepat sesuai kebutuhan kerja adalah hal yang patut diperhatikan dan diperhitungkan. Dan seiring dengan perkembangan teknologi *hardware* seperti *video-memory*, *processor* dan *main-memory*, juga diikuti dengan peningkatan kemampuan *operating-systems*; komputer memeran penting dalam dukungan aktivitas bisnis dan pekerjaan, bahkan pada bidang-bidang *high-risk*. Tuntutan meningkatkan spesifikasi teknis komputer juga diperlukan oleh alasan kompleksitas pekerjaan dan bidang yang dikerjakan. Misalkan dalam penyelesaian bidang *engineering*, *science*, *game-modeling* dan statistika yang tentu membutuhkan pengolahan data yang besar dan harus cepat. Terlebih saat ini berkembang adanya cabang olahraga yang disebut *e-sport*, yang tak lain kompetisi *game* secara *online* baik bersifat individu maupun berkelompok. Dan *trend game* saat ini kompleks dalam citra dan kualitas *sound* yang tajam serta jernih; oleh sebab itu dibutuhkan serangkaian metode dan analisis untuk melakukan selektivitas terhadap komputer yang tepat [1][2].

Untuk sejumlah data dengan beragam spesifikasi yang tersedia maka pembelajaran mesin dapat diterapkan untuk melakukan selektivitas [3]. Pembelajaran mesin merujuk pada perubahan sebuah sistem berdasar cara kerja *artificial intelligence* untuk melakukan *recognition*, *diagnosis*, *planning*, *robot control*, *prediction*; sehingga terbentuk sebuah pola yang baru dalam penyelesaian suatu persoalan [4][5]. Salah satu terapannya dalam aktivitas kegiatan bisnis adalah *data mining*. Sejumlah data dengan jumlah yang besar dilakukan selektivitas berbagai kategori yang ditentukan sehingga tujuan data yang ingin di-*filter* atau dicari dapat ditemukan. Kategori bisa dalam bentuk nama, identitas, alamat; atau bentuk/pola seperti: bulat, panjang, lebar, tinggi; dan warna seperti: merah, hijau, kuning, biru. Bahkan dapat mengenali *object* dalam bentuk tekstur dan kekenyalan (unsur *solid* dan *liquid*) [6].

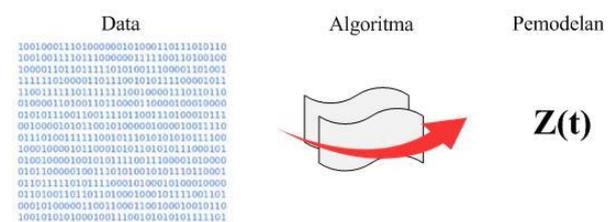
Demikian juga pada pemilihan komputer dengan spesifikasi mana yang *compatible* untuk menjalankan *game*. Saat ini *game* yang berkembang menuntut nilai atau tingkat tertentu dari spesifikasi komputer, seperti: jenis *processor*, *clock-rate processor*, kapasitas *main-memory*, dan *video-memory*. Demikian juga persyaratan sistem operasi, dimana level atau tingkat *hardware* juga harus diselaraskan juga dengan *software* yang digunakan. Saat ini mayoritas *end-user* menggunakan Windows 8/64-bit [7].

### 1.1 Pembelajaran Mesin

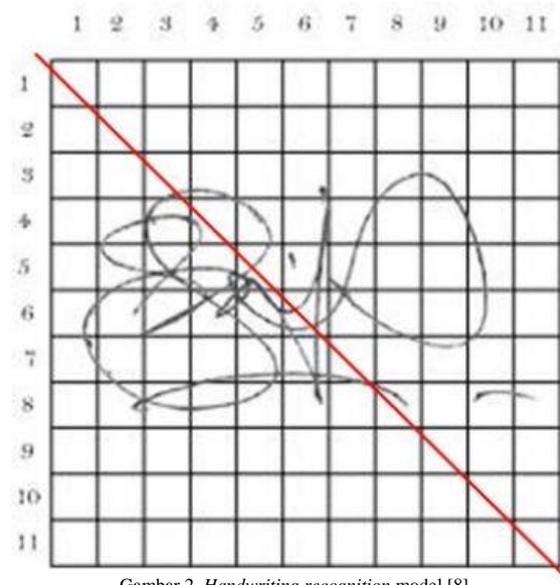
Bidang pembelajaran mesin telah berkembang sejak era 60-an dimulai sebuah artikel yang ditulis oleh seorang *engineer* bernama Arthur Samuel dengan judul artikel 'Some Studies in Machine Learning

Using the Game of Checkers'. Pembelajaran mesin (*machine learning*) dapat diartikan sebagai sebuah pemrograman komputer untuk mengoptimalkan kriteria kinerja menggunakan contoh data atau pengalaman masa lalu, berdasar model yang telah ditentukan dari sejumlah parameter [8][9]. Pembelajaran mesin menggunakan data pelatihan atau pengalaman masa lalu untuk melakukan eksekusi program komputer dengan mengoptimalkan parameter model yang dibuat. Modelnya mungkin bersifat prediktif untuk membuat prediksi di masa depan, atau deskriptif untuk memperoleh keuntungan pengetahuan dari data, atau kombinasi keduanya [6,7].

Dengan pembelajaran mesin sebuah komputer dapat bekerja dan bertindak tanpa diprogram secara eksplisit, seperti ditunjukkan pada Gambar 1. Beberapa contoh terapan pembelajaran mesin seperti: *handwriting recognition learning problem*, *robot driving learning problem*, *chess learning problem*.



Gambar 1. Model sederhana sebuah pembelajaran mesin



Gambar 2. Handwriting recognition model [8]

Dalam pembelajaran mesin, data dibagi menjadi data *training* dan data *test*. Dari data *training*, diperoleh *model* dalam hubungan (t,r), lalu dari model yang ditetapkan tersebut dibangun *prediction*; lalu selanjutnya dibuat data *test* dalam hubungan (t,r) dan memungkinkan diperbaharui. Lebih lanjut tahapan ini dikenal dengan istilah *7 Steps of Machine Learning*,

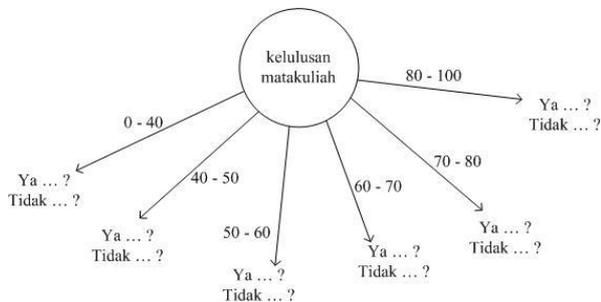
yakni: *Data Collection* ⇒ *Data Preparation* ⇒ *Choose a Model* ⇒ *Train the Model* ⇒ *Evaluate the Model* ⇒ *Parameter Tuning* ⇒ *Make Predictions*. Secara umum algoritma pembelajaran mesin terbagi atas kategori yakni: *supervised learning*, *unsupervised learning*, dan *reinforcement learning*[10] [11].

**1.2 Algoritma C4.5**

Algoritma C4.5 adalah sejumlah *steps* untuk menginduksi aturan klasifikasi dalam bentuk pohon keputusan dari sekumpulan data yang telah didefinisikan; dimana algoritma ini merupakan perpanjangan perangkat lunak dari algoritma ID3 (*Iterative Dichotomiser 3*) yang dasar rancangannya oleh Quinlan [5,6]. Oleh sebab kelebihanannya, algoritma C4.5 juga dimanfaatkan untuk bidang *data-mining*. Seperti hal konsep sebuah algoritma yakni menguraikan langkah-langkah umum dalam bentuk lebih sistematis dan terstruktur [9] [12].

Langkah-langkah algoritma C4.5 diuraikan sebagai berikut:

- i. Pilih satu atribut dari sekumpulan data *training*
- ii. Pilih *subset* awal dari data *training*
- iii. Gunakan atribut dan *subset* tersebut untuk membuat keputusan pohon
- iv. Gunakan data *training* lainnya (yang tidak termasuk dalam *subset* yang digunakan untuk konstruksi) untuk menguji keakuratan pohon yang dibangun
- v. Jika semua data *training* diklasifikasikan dengan benar → hentikan
- vi. Jika sebuah data *training* salah diklasifikasikan, tambahkan data *training* tersebut ke *subset* awal dan buat pohon baru
- vii. lakukan iterasi sampai memenuhi:
  - a. sebuah pohon dibangun yang mengklasifikasikan semua data *training* dengan benar, atau
  - b. sebuah pohon dibangun dari seluruh data *training*



Gambar 3. Model sederhana luaran algoritma C4.5

$$\text{Entropy}(S) = \sum_{i=1}^n f(-p_i * \log(p_i)); \tag{1}$$

dimana,  
S : himpunan kasus

P<sub>i</sub> : proporsi dari S<sub>i</sub> terhadap S  
N : jumlah partisi S

Sebuah *entropy* merujuk pada probabilitas suatu peristiwa, sehingga ketika nilai *entropy* meningkat maka perolehan informasi semakin menurun [5,6].

Rumus menghitung nilai Gain (S,A) sebagai berikut:

$$\text{Entropy}(S) - \sum_{i=1}^n ((-1) * \left| \frac{S_i}{S} \right| * \text{Entropy}(S_i)); \tag{2}$$

dimana,  
S = himpunan kasus  
A = atribut  
n = jumlah partisi atribut A  
|S<sub>i</sub>| = proporsi S<sub>i</sub> terhadap S  
|S| = jumlah kasus dalam S

*Gain* merupakan perbandingan perolehan informasi terhadap informasi yang terkandung, dengan tujuan pemilahan informasi menuju sasaran yang diinginkan.

$$\text{Gain Ratio}(A) = \frac{\text{Gain}(A)}{\text{SplitInfo}(A)}; \tag{3}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

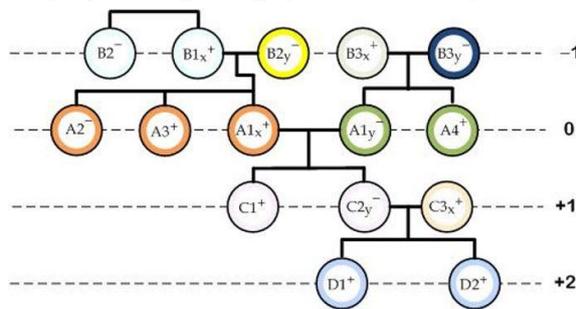
$$\text{Precision} = \frac{TP}{TP + FP} \tag{5}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{6}$$

dimana,  
TP : *True-Positive*, jumlah data dengan kriteria nilai-aktual positif dan nilai-prediksi positif  
TN : *True-Negative*, jumlah data dengan kriteria nilai-aktual negatif dan nilai-prediksi negatif  
FP : *False-Positive*, jumlah data dengan kriteria nilai-aktual negatif dan nilai-prediksi positif  
FN : *False-Negative*, jumlah data dengan kriteria nilai-aktual positif dan nilai-prediksi negatif  
Sedangkan *accuracy* adalah metrik yang mengukur seberapa sering model pembelajaran mesin memprediksi hasilnya dengan tepat; *precision* adalah metrik yang mengukur seberapa sering model pembelajaran mesin memprediksi kelas positif dengan tepat.; *recall* adalah metrik yang mengukur seberapa sering model pembelajaran mesin dengan benar mengidentifikasi kejadian positif (TP, *True-Positive*) dari semua *sample* positif aktual dalam data *set* [13].

### 1.3 Pohon keputusan

Pohon keputusan (*decision tree*) bermanfaat untuk membangun rangkaian solusi yang bersifat konstruktif dan sistematis sehingga memudahkan dalam menuju keputusan yang akan diambil. Secara sederhana pohon keputusan dapat diartikan sebagai sebuah hubungan vertikal antar entitas atau *node* dalam uraian sebuah solusi berjenjang menggunakan kaidah ‘jika-dan-hanya’ untuk hasil ‘ya’ atau ‘tidak’; memiliki atribut berupa simpul akar, cabang, simpul internal, dan simpul daun[14]. Kaidah ini tergolong *non-parametric supervised learning algorithm*, yakni suatu algoritma yang tidak membuat fungsi asumsi awal; sehingga tidak dibatasi dalam membangun *learning function* dari data *training* yang ditetapkan. Teknik *non-parametric* tepat digunakan bila data yang tersedia banyak namun tidak cukup mengetahui *profiling* sebelumnya, dan tidak ingin terlalu khawatir untuk memilih fitur yang tepat.



Gambar 4. Model sederhana pohon keputusan[10]

Luaran dari terapan algoritma C4.5 adalah sebuah pohon keputusan, yakni memberi nilai ‘Ya’ atau ‘Tidak’ untuk sejumlah kondisi dari serangkaian data *training* yang diuji. Adapun kelebihan dalam pemanfaatan pohon keputusan yakni kemampuannya dalam memilah (*break-down*) ke dalam entitas lebih kecil sehingga setiap *steps* yang didapatkan menjadi lebih sederhana. Alhasil pengambilan keputusan dapat lebih mudah dilakukan bahkan solusi yang didapatkan mudah dipahami. Dan pohon keputusan dapat dimanfaatkan untuk melakukan eksplorasi data serta tindakan identifikasi terhadap relasi antar-entitas yang tidak tervisualisasi langsung.

## 2. Metode Penelitian

Metodologi penelitian merupakan sejumlah langkah-langkah yang diambil untuk melakukan penelitian terkait dan menggunakan sebuah teknik atau metode berupa algoritma yang dipilih untuk tujuan pencapaian solusi dari permasalahan yang ditemukan. Dan pada penelitian ini dilakukan serangkaian tahapan berupa melakukan observasi, pengumpulan data, menentukan hipotesis, melakukan eksperimen, dan menarik kesimpulan.

Sejumlah komputer dipilih secara acak dari berbagai literatur di *e-shop*, untuk diuji kompatibilitasnya

dengan tiga *game on-line*, yakni: Resident Evil 4, NFS Heat, dan Pes 2021. Ada dua sistem operasi yang dipilih yakni: Windows 10 dan Windows 11. Untuk prosesor, *main-memory*, dan VGA (*Video Graphic Adapter*) ditunjukkan pada Tabel 1 beserta pengkodeannya untuk tujuan teknis penyebutan.

Tabel 1. Kategori spesifikasi komputer yang diuji beserta *game on-line*

Jenis kategori/spesifikasi	Nama unit	Pengkodean/labeling
Processor	Intel Core i7-860	P01
	Intel Core i7-9700k	P02
	Intel Core i5-6500	P03
	Intel Core i3-3240	P04
	Intel Core i5-10400f	P05
	Intel Core i5-3470	P06
	Intel Core i3-7100	P07
Video Graphic Adapter	GT 730	VGA01
	GTX 750ti	VGA02
	GTX 1650	VGA03
	GTX 1660	VGA04
Game on-line	Pes 2021	G01
	Resident Evil 4	G02
Operating-Systems	NFS Heat	G03
	Windows 10	OS1
	Windows 11	OS1
Random Access Memory (RAM)	8GB	R1
	16GB	R2

Misalkan, pada *game on-line* Pes 2021 dengan spesifikasi berikut:

- 64-bit *processor* dan *operating systems*
- OS: Windows 8.1
- Processor*: Intel Core i5-3470/AMD FX-4350
- RAM: 8GB
- Video-card*: NVIDIA GT 670/AMD Radeon HD 7870, DirectX 11

Misalkan sebuah *local* komputer dengan spesifikasi *processor* adalah Intel Core i5-3470 dengan *main-memory* 8GB, VGA adalah GT 730 dan sistem operasi Windows 10 dilakukan pengujian terhadap *game on-line* NFS Heat, disimpulkan tidak kompatibel.

Tabel 2. Deskripsi kategori dan atribut penjelasannya

Kategori	Penjelasan
Processor	jenis dan model <i>processor</i> pada setiap <i>local</i> komputer yang akan diuji
Video Graphic Adapter	jenis dan model kartu grafis pada <i>local</i> komputer yang akan diuji
Game on-line	nama <i>game on-line</i> pilihan yang akan diuji: Resident Evil 4, PES 2021, NFS Heat
Operating-Systems	sistem operasi yang digunakan pada <i>local</i> komputer yang akan diuji: Windows 10, Windows 11
Random Access Memory (RAM)	kapasitas <i>main-memory</i> yang dimiliki oleh <i>local</i> komputer yang akan diuji: 8GB, 16GB

### 3. Hasil dan Pembahasan

Pengujian kompatibilitas dari kombinasi spesifikasi *local* komputer ditandemkan dengan kebutuhan spesifikasi oleh setiap *game on-line* yang telah ditentukan. Dan telah diinformasikan sebelumnya untuk memudahkan peng-klasifikasian setiap kategori dan masing-masing atributnya maka dilakukan pelabelan, dan ditunjukkan pada Tabel 3.

Tabel 3. Hasil pengujian kompatibilitas

Processor	RAM	VGA	OS	Game on-line	Simpulan
P01	R1	VGA02	OS2	G01	kompatibel
P02	R2	VGA04	OS1	G02	kompatibel
P03	R2	VGA02	OS2	G03	kompatibel
P02	R2	VGA04	OS1	G02	kompatibel
P04	R1	VGA01	OS2	G01	kompatibel
P05	R1	VGA03	OS1	G02	kompatibel
P05	R1	VGA03	OS1	G02	kompatibel
P07	R1	VGA01	OS1	G03	kompatibel
P05	R1	VGA01	OS1	G03	kompatibel
P01	R2	VGA02	OS1	G01	kompatibel
P01	R2	VGA04	OS1	G02	kompatibel
P07	R2	VGA02	OS2	G02	kompatibel
P05	R1	VGA03	OS2	G02	kompatibel
P01	R1	VGA02	OS1	G03	kompatibel
P07	R2	VGA02	OS1	G03	kompatibel
P02	R1	VGA04	OS1	G02	kompatibel
P01	R2	VGA02	OS1	G03	kompatibel
P07	R1	VGA01	OS2	G03	kompatibel
P06	R1	VGA02	OS1	G02	tidak kompatibel
P03	R1	VGA02	OS1	G03	kompatibel
P07	R2	VGA02	OS2	G01	kompatibel
P05	R1	VGA03	OS2	G03	kompatibel
P04	R1	VGA02	OS1	G02	tidak kompatibel
P03	R2	VGA01	OS1	G01	kompatibel
P05	R1	VGA03	OS2	G03	kompatibel
P07	R2	VGA03	OS1	G02	kompatibel
P03	R2	VGA01	OS2	G01	kompatibel
P02	R2	VGA04	OS1	G02	kompatibel
P05	R1	VGA02	OS1	G02	kompatibel
P01	R1	VGA01	OS2	G01	kompatibel
P02	R1	VGA04	OS1	G02	kompatibel
P02	R2	VGA04	OS1	G03	kompatibel
P03	R1	VGA01	OS1	G03	kompatibel
P05	R2	VGA01	OS1	G02	kompatibel
P04	R1	VGA03	OS2	G03	kompatibel
P04	R1	VGA02	OS1	G01	kompatibel
P06	R1	VGA01	OS1	G03	tidak kompatibel
P07	R2	VGA02	OS1	G01	kompatibel
P03	R1	VGA01	OS1	G02	tidak kompatibel
P06	R1	VGA01	OS1	G02	tidak kompatibel
P02	R2	VGA04	OS1	G02	kompatibel
P06	R1	VGA02	OS1	G02	tidak kompatibel
P02	R2	VGA04	OS1	G02	kompatibel
P04	R1	VGA03	OS2	G02	kompatibel
P01	R1	VGA01	OS2	G01	kompatibel
P05	R1	VGA03	OS1	G01	kompatibel
P06	R1	VGA01	OS1	G02	tidak kompatibel
P02	R1	VGA04	OS1	G02	kompatibel
P07	R1	VGA01	OS2	G03	kompatibel
P04	R1	VGA03	OS2	G03	kompatibel
P06	R1	VGA02	OS1	G02	tidak kompatibel
P03	R1	VGA02	OS2	G01	kompatibel
P01	R2	VGA02	OS2	G01	kompatibel
P01	R1	VGA01	OS1	G03	tidak kompatibel
P06	R1	VGA02	OS1	G02	tidak kompatibel
P04	R1	VGA02	OS1	G03	kompatibel

Terlihat dengan jelas bahwa sebuah spesifikasi *local* komputer apakah kompatibel atau tidak menjalankan *game on-line* bervariasi antar satu spesifikasi dengan

yang lain. Misalkan untuk G02 kompatibel untuk spesifikasi dengan kombinasi: P02, R1, VGA04, dan OS1. Namun G02 tidak kompatibel untuk spesifikasi dengan kombinasi: P06, R1, VGA01, dan OS1.

Sebagai contoh dihitung *entropy* untuk P04 sebagai berikut:

Jumlah=7;  
Kompatibel=6;  
Tidak kompatibel=1;

$$Entropy (S) = \left( -\frac{6}{7} * {}^2\log\left(\frac{6}{7}\right) \right) + \left( -\frac{1}{7} * {}^2\log\left(\frac{1}{7}\right) \right) = 0.6287;$$

Dan *gain* kategori *processor* dihitung sebagai berikut:

Jumlah total=56;  
Jumlah P01 =9;  
Jumlah P02 =9;  
Jumlah P03 =7;  
Jumlah P04 =7;  
Jumlah P05= 9;  
Jumlah P06 =7;  
Jumlah P07 =8;  
Gain(S,A) = 0.6287 –

$$\left\{ \left( \frac{7}{56} * 0.62873 + \left( \frac{8}{56} * 0 \right) + \left( \frac{9}{56} * 0 \right) \right) + \left\{ \left( \frac{7}{56} * 0 \right) + \left( \frac{7}{56} * 0.62873 \right) \right\} + \left\{ \left( \frac{9}{56} * 0.5245 \right) + \left( \frac{9}{56} * 0 \right) \right\} \right\} = 0.43546;$$

Secara keseluruhan dirangkum pada Tabel 4 untuk setiap kategori dan atributnya masing-masing, pada parameter *entropy* dan *gain*.

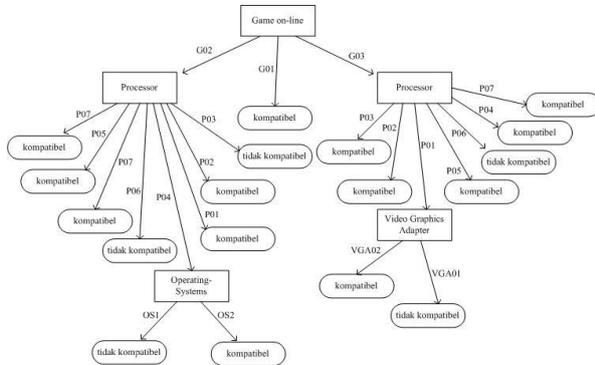
Tabel 4. Hasil perhitungan *entropy* dan *gain* setiap kategori beserta atributnya

	T2	T3	T4	T5	T6
T7	56	46	10	0,676	
T1					0,4355
	P04	7	6	1	0,629
	P07	8	8	0	0
	P05	9	9	0	0
	P06	7	0	7	0
	P03	7	6	1	0,6287
	P01	9	8	1	0,5245
	P02	9	9	0	0
T8					0,0095
	R1	37	27	10	1,0102
	R2	19	19	0	0
T9					0,0116
	VGA01	16	11	5	1,1418
	VGA02	20	15	5	0,9497
	VGA03	10	10	0	0
	VGA04	10	10	0	0
T10					0,0060
	OS1	38	28	10	0,9888
	OS1	18	18	0	0
T11					1,3662
	G02	25	17	8	1,1662
	G03	18	16	2	0,5245
	G01	13	13	0	0

\*alasan teknis agar tabel bisa utuh ditampilkan

T1: Processor  
T2: Jumlah  
T3: Kompatibel  
T4: Tidak Kompatibel

- T5: Entropy
- T6: Gain
- T7: Total
- T8: RAM
- T9: VGA
- T10: Operating Systems
- T11: Game on-line



Gambar 5. Pohon keputusan yang dibangun

Selanjutnya disusun *confusion matrix* sebagai berikut:

$$\begin{bmatrix} 9 & 1 \\ 1 & 3 \end{bmatrix}$$

Dan diperoleh bahwa,  
 True Negative (TN): 9;  
 False Positive (FP): 1;  
 False Negative (FN): 1;  
 True Positive (TP): 3;

$$Accuracy = \frac{(3+9)}{(3+9+1+1)} = 0.8571;$$

$$Precision = \frac{3}{(3+1)} = 0.75;$$

$$Recall = \frac{3}{(3+1)} = 0.75;$$

#### 4. Kesimpulan

Dari berbagai kombinasi kategori beserta atribut tersebut, untuk sejumlah *local* komputer yang telah diuji terhadap tiga jenis *game on-line*; telah memperlihatkan ada yang kompatibel, namun ada juga yang tidak kompatibel. Nilai kompatibilitas terlihat pada kombinasi antar kategori beserta atribut yang dimiliki. Hal ini sesuai dengan teori bahwa kompatibilitas sebuah komputer didasarkan terhadap prioritas pada harmonisasi antara komponen *hardware* dan *software*.

Berdasar hasil pengujian menunjukkan bahwa model pembelajaran mesin akurat memprediksi sebesar 85.71%. Adapun model pembelajaran mesin dapat

memprediksi kelas positif sebesar 75%. Dan diperoleh bahwa 75% model pembelajaran mesin dengan benar mengidentifikasi kejadian positif. Alhasil model pembelajaran mesin yang didesain telah optimal dalam menunjukkan simpulan sejumlah *sampling local* komputer yang diuji apakah kompatibel terhadap tiga jenis *game on-line* yang ditentukan tersebut.

Untuk selanjutnya dapat dikembangkan pada *object-object* dinamis seperti arah angin terhadap pertumbuhan suatu jenis tanaman. Dan hal yang lebih penting lagi agar dibandingkan terhadap lingkungan interaktif seperti *speech* atau *interactive communication*.

#### Ucapan Terimakasih

Ucapan terima kasih kepada Laboratorium Multimedia dan Pengolahan Informasi, Institut Digital Ekonomi (IDE) LPKIA Bandung atas penyediaan infrastruktur saat pengolahan data dan pengujian berlangsung.

#### Daftar Rujukan

- [1] E. Yan-Tao, T., "Computational model of selective attention for machine vision based on adapted entropy," 2015.
- [2] V. S. G. Mcgraw, R. Bonett, H. Figueroa, "Security Engineering for Machine Learning," *Comput. Mag. Artic.*, vol. 52, no. 8, 2019.
- [3] M. Melladia, D. E. Putra, and L. Muhelni, "Penerapan Data Mining Pemasaran Produk Menggunakan Metode Clustering," *J. Tek. Inf. dan Komput.*, vol. 5, no. 1, p. 160, 2022, doi: 10.37600/tekinkom.v5i1.458.
- [4] S. G. A. Müller, *Introduction to Machine Learning with Python: A Guide for Data Scientists 1st Edition*. O'Reilly Publisher.
- [5] D. Cote, "Using machine learning in communication networks," *J. Opt. Commun. Netw.*, vol. 10, no. 10, 2018.
- [6] Melladia, T. Informatika, U. Nahdlatul, U. Sumatera, F. Chaining, and Z. Plastik, "Aplikasi Sistem Pakar Mendeteksi Zat Berbahaya Pada Plastik Menggunakan Metode Backward Chaining," pp. 19–28.
- [7] A. N. S. Z. Gustiana, "Combination of C 4.5 Algorithm and Profile Matching for Determining University Students Graduation," 2021.
- [8] K. R. A. K. U.M. Ashwin Kumar, "Data Preparation by CFS: An Essential Approach for Decision Making Using C 4.5 for Medical Data Mining," 2013.
- [9] M. X. W. Zhu, "A novel method for detecting disk filtration attacks via the various machine learning algorithms," *China Communications Magazine Article*, 2020.
- [10] S.N.M.P. Simamora, "Model Pohon Relasi Untuk Mendeteksi Jalur Kedekatan Hubungan Antar-Entitas," *Koran Monitor. Edisi 16 Januari 2023*, 2023.

- <https://koranmonitor.com/uncategorized/model-pohon-relasi-untuk-mendeteksi-jalur-kedekatan-hubungan-antar-entitas/>. 138., 2023.
- [11] S.N.M.P. Simamora, *Modul Belajar Praktis Algoritma dan Pemrograman*. Yogyakarta: Penerbit Deepublish, 2016.
- [12] P. Aplikasi and E. B. Web, "Jurnal Tefsin," vol. 1, no. 1, pp. 13–18, 2023.
- [13] S.N.M.P. Simamora, "Identifikasi Awal Kode Tanda Tangan Dengan Pendekatan Pemodelan Koordinat," *J. Nas. Komputasi dan Teknol. Inf.*, vol. 6, no. 1, pp. 122–138., 2023.
- [14] Melladia, "Algoritma Genetika Menentukan Jalur Jalan dengan Lintasan Terpendek ( Shortest Path )," pp. 112–117.