

## Evaluasi Dan Prediksi Hasil Pertandingan Dota 2 Menggunakan Algoritma Terbaik Dari Random Forest Atau XGBoost

Rico Satria Fandi<sup>1</sup>, Amalia Anjani Arifiyanti<sup>2</sup>, Seftin Fitri Ana Wati<sup>3</sup>

<sup>1,2,3</sup>Universitas Pembangunan Nasional “Veteran” Jawa Timur, Indonesia

[ricosatriafandi@gmail.com](mailto:ricosatriafandi@gmail.com)

### Abstrak

Penelitian ini bertujuan untuk membandingkan kinerja model algoritma Random Forest dan XGBoost dalam prediksi hasil pertandingan Dota 2. Analisis ini berfokus pada evaluasi kinerja model menggunakan Confusion Matrix dan AUROC untuk menentukan algoritma terbaik. Tujuan penelitian ini adalah untuk menganalisis kinerja kedua algoritma tersebut dan mengimplementasikannya dalam skenario model dengan berbagai pembagian data untuk mencari model terbaik. Pemodelan dilakukan dengan membagi data menjadi tiga skenario: 80:20, 75:25, dan 70:30. Evaluasi model dilakukan menggunakan Confusion Matrix dan AUROC. Hasil penelitian menunjukkan bahwa skenario pembagian data 80:20 dengan algoritma XGBoost memberikan kinerja terbaik. Algoritma XGBoost pada skenario ini mencapai akurasi 52% pada Confusion Matrix dan AUROC sebesar 51.49%. Hasil ini menunjukkan bahwa algoritma XGBoost memberikan keseimbangan optimal antara accuracy, precision, recall, dan f1-score untuk prediksi kedua kelas, yaitu Dire Menang dan Radiant Menang. Penelitian ini menyimpulkan bahwa algoritma XGBoost lebih unggul dibandingkan Random Forest dalam skenario pemodelan prediksi hasil pertandingan Dota 2.

### Sejarah Artikel

Submitted: 18 Juli 2024

Accepted: 23 Juli 2024

Published: 24 Juli 2024

### Kata Kunci

Dota 2;  
Klasifikasi;  
Prediksi;  
Random Forest;  
XGBoost;

## PENDAHULUAN

Perkembangan teknologi dan internet telah mendorong popularitas kompetisi esports, salah satunya Dota 2, sebuah game MOBA (*Multiplayer Online Battle Arena*) yang kompleks [1]. Kompleksitas permainan ini terletak pada banyaknya faktor yang mempengaruhi hasil pertandingan, seperti pemilihan hero dan item [2]. Oleh karena itu, pemodelan untuk memprediksi hasil pertandingan Dota 2 menjadi sangat penting dan menantang untuk dieksplorasi lebih lanjut [3].

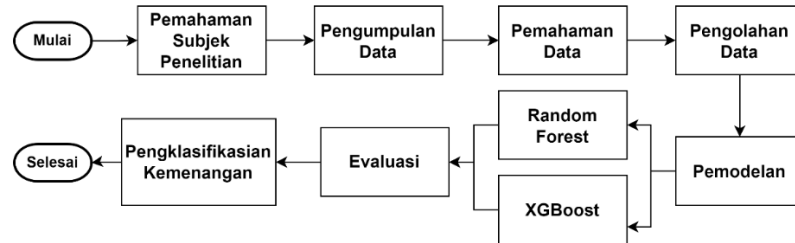
Rumusan masalah dalam penelitian ini mencakup dua pertanyaan utama. Pertama, analisis terkait perbandingan kinerja model algoritma Random Forest dan XGBoost untuk menentukan mana yang lebih baik dalam memprediksi hasil pertandingan Dota 2. Kedua, bagaimana menerapkan dan mengimplementasikan model algoritma Random Forest dan XGBoost menggunakan evaluasi Confusion Matrix dan AUROC untuk mencari model terbaik dalam prediksi hasil pertandingan Dota 2. Kedua pertanyaan ini penting untuk menentukan efektivitas metode yang digunakan dalam memprediksi hasil pertandingan yang kompleks.

Penelitian sebelumnya menunjukkan variasi dalam tingkat akurasi prediksi hasil pertandingan Dota 2 dengan algoritma Decision Tree, Random Forest, dan XGBoost mencapai akurasi hingga 93% [4], sementara penelitian lain menggunakan OpenDota API hanya mencapai akurasi 70% [5]. Ada juga studi yang menggunakan LightGBM dengan akurasi 77,51% [6]. Sementara penelitian [7] menekankan pentingnya data *preprocessing* untuk mengolah dan membersihkan data sebelum dilakukan pemodelan. Namun, ada kesenjangan dalam studi yang komprehensif yang menunjukkan bahwa Random Forest memiliki tingkat akurasi yang lebih tinggi dibandingkan dengan Decision Tree dalam memprediksi keberhasilan imunoterapi dengan tingkat akurasi masing-masing 85.5% dan 84.4% [8].

Penelitian ini bertujuan untuk menganalisis perbandingan kinerja model algoritma Random Forest dan XGBoost dalam memprediksi hasil pertandingan Dota 2. Tujuan lainnya adalah menerapkan dan mengimplementasikan algoritma Random Forest dan XGBoost pada skenario model dengan menggunakan skenario pembagian data 80:20, 75:25, dan 70:30 hasil evaluasi Confusion Matrix dan AUROC untuk mencari model terbaik. Dengan tujuan ini,

diharapkan penelitian dapat menemukan model prediksi yang lebih akurat dan efisien, memberikan kontribusi pada pengembangan strategi dalam permainan Dota 2, serta penerapan teknik machine learning pada bidang esports.

## METODE



Gambar 1. Alur Metode

### Pemahaman Subjek Penelitian

Pemahaman subjek penelitian yakni video game Dota 2 melibatkan serangkaian kegiatan dalam penelitian dan kajian pustaka dari sumber seperti jurnal, buku, atau dokumentasi yang relevan dengan topik penelitian. Pada penelitian ini, digunakan sumber seperti jurnal, artikel ilmiah, penelitian, dan dokumen resmi dari komunitas Dota 2 dan teknologi terkait. Berdasarkan pemahaman terkait permainan Dota 2, data yang dibutuhkan mencakup pertandingan mulai dari *Patch 7.35* dirilis pada 14 Desember 2023 sampai *Patch 7.35d* dirilis pada 21 Maret 2024. Dalam rentang *Patch 7.35—7.35d*, terdapat 100.000 pertandingan, 124 pahlawan (*hero*), dan 449 perlengkapan (*item*).

### Pengumpulan Data

Pada tahap pengumpulan data, dilakukan dengan cara Request API melalui OpenDota API secara gratis, tanpa memerlukan kunci akses. Namun, terdapat batasan dalam pengumpulan data, yaitu maksimal 2.000 data per hari dan 60 data setiap menit.

### Pemahaman Data

Tahap pemahaman data merupakan proses identifikasi dan pemahaman data dari dataset yang telah diperoleh. Proses ini meliputi pengumpulan data pada tahap sebelumnya serta evaluasi kualitas data. Evaluasi terhadap kualitas data dilakukan untuk menemukan *Missing Value* pada setiap atribut dalam dataset [9]. Atribut yang baik adalah yang tidak memiliki *Missing Value*, sehingga dataset yang bersih dapat digunakan dalam tahap pemodelan. Pada tahap pengolahan data, nilai kosong dihilangkan dan dataset perlengkapan (*item*) disesuaikan dengan rentang *Patch 7.35—7.35d*. Dataset pahlawan (*hero*) sudah relevan dengan rentang *patch* tersebut, sehingga tidak perlu diubah. Pemahaman subjek penelitian data melibatkan proses memahami arti setiap atribut dalam data. Atribut yang menjadi variabel fitur (X) adalah daftar *hero* dan *items* masing-masing tim, sedangkan variabel target (Y) atau class adalah hasil pertandingan (Radiant menang atau Dire menang).

### Pengolahan Data

Tahapan ini mempersiapkan data agar dapat diolah dengan baik, termasuk pengecekan total *Missing Value* dan pemeriksaan tipe data.

### Pemodelan

Proses pemodelan dilakukan dengan algoritma Random Forest dan XGBoost menggunakan library Sklearn. Data dibagi menjadi data training dan data testing berdasarkan

tiga skenario rasio: 80:20, 75:25, dan 70:30. Atribut prediktor mencakup 5 *hero* dan 6 *item* pada masing-masing tim. Objek dari masing-masing class berdasarkan dua model algoritma dibuat dan diuji dengan data testing. Random Forest dipilih karena kemampuannya menangani dataset besar dan kompleks [10], serta memberikan hasil yang stabil dan akurat. Model diimplementasikan dengan memuat objek dari class RandomForestClassifier dari library Sklearn dan mem-fit data training. XGBoost dipilih karena kemampuannya menangani dataset besar dengan efisien dan memberikan hasil yang akurat [11]. Model diimplementasikan dengan memuat objek dari class xgb dari library XGBoost dan mem-fit data training.

## Evaluasi

Tahap evaluasi menghitung nilai evaluasi dari kedua model algoritma yang sudah diimplementasikan. Pengukuran dilakukan dengan confusion matrix yang memuat persentase nilai accuracy, precision, recall, dan f-1 score [12], serta AUROC untuk membandingkan performa model [13].

## Pengklasifikasian Kemenangan

Dalam tahap pengklasifikasi kemenangan performa prediksi hasil pertandingan Dota 2 diuji dengan data kelas *heroes*, *items*, dan pertandingan untuk menghitung nilai akurasi dari dua skenario algoritma pengklasifikasi, yaitu Random Forest dan XGBoost.

## HASIL DAN PEMBAHASAN

Dalam penelitian ini akan diuji dan evaluasi dua algoritma yang berbeda untuk mendapatkan model terbaik. Algoritma dengan Random Forest dan XGBoost di evaluasi menggunakan Confusion Matrix dan AUROC.

## Hasil

### a. Pengumpulan Data

Pengumpulan data dilakukan secara gratis melalui OpenDota API tanpa menggunakan API Key. Namun, terdapat beberapa keterbatasan dalam penggunaan API gratis ini, yaitu adanya batasan 60 request per menit dan maksimal 2.000 data per hari. Batasan ini memastikan bahwa data yang diperoleh akurat dan terbaru. Deskripsi dari masing-masing jenis data yang dikumpulkan mencakup data pertandingan pada Gambar 2 merupakan contoh sampel data mentah dengan format file JSON.

	players	radiant_win	duration	pre_game_duration	start_time	match_id
0	[[{"player_slot": 0, "team_number": 0, "team_sl...	1.0	2556	90	2024-05-20 05:59:33	7746381429
1	[[{"account_id": 169987251, "player_slot": 0, "...	0.0	2395	90	2024-05-20 05:59:23	7746381427
2	[[{"account_id": 464987756, "player_slot": 0, "...	0.0	1573	60	2024-05-20 05:59:38	7746381413
3	[[{"account_id": 201290087, "player_slot": 0, "...	1.0	2551	90	2024-05-20 05:59:38	7746381412
4	[[{"player_slot": 0, "team_number": 0, "team_sl...	1.0	2223	90	2024-05-20 05:59:43	7746381406

Gambar 2. List Data Pertandingan Json

Pada **Error! Reference source not found.** merupakan sampel dataset pahlawan (*hero*) total terdapat 124 pahlawan yang terdapat pada patch 7.35—7.35d. Memberikan ID unik untuk

setiap pahlawan yang diidentifikasi data yang lebih mudah dalam pengembangan atau analisis lebih lanjut.

hero_id	localized_name
0	1 Anti-Mage
1	2 Axe
2	3 Bane
3	4 Bloodseeker
4	5 Crystal Maiden
...	...
119	129 Mars
120	135 Dawnbreaker
121	136 Marci
122	137 Primal Beast
123	138 Muerta

124 rows × 2 columns

**Gambar 3.** Data Pahlawan

Pada **Error! Reference source not found.** merupakan sampel dataset perlengkapan (*item*) total terdapat 449 yang terdapat pada *patch* 7.35—7.35d. Memberikan ID unik untuk setiap item yang diidentifikasi data yang lebih mudah dalam pengembangan atau analisis lebih lanjut.

item_id	item_name
0	1 Blink Dagger
1	2 Blades of Attack
2	3 Broadsword
3	4 Chainmail
4	5 Claymore
...	...
444	4205 Great Healing Lotus
445	4206 Greater Healing Lotus
446	4300 Beloved Memory
447	4301 Scrying Shovel
448	4302 Forebearer's Fortune

449 rows × 2 columns

**Gambar 4.** Data Perlengkapan

### b. Pemahaman Data & Pengolahan Data

Pemahaman data melibatkan proses identifikasi dan pemahaman setiap atribut pada dataset yang diperoleh. Data pertandingan berjumlah 100.000. Memerlukan evaluasi untuk memilih data yang relevan dengan mengkategorikan dataset menjadi pertandingan (*match*) dan pemain (*players*). Proses evaluasi memerlukan transformasi dan eliminasi atribut yang tidak dibutuhkan. Pada **Error! Reference source not found.** menjelaskan detail keterangan dan contoh sampel data terkait atribut pada data pertandingan (*match*).

**Tabel 1.** Keterangan dan Contoh Sampel Data Pertandingan

Nama Atribut	Keterangan	Contoh Data
<i>match_id</i>	Unique setiap pertandingan	7461965918
<i>start_time</i>	Kapan pertandingan selesai	17011132560
<i>duration</i>	Durasi pertandingan	2172
<i>tower_status_radiant</i>	Status tower tim Radiant	1542
<i>tower_status_dire</i>	Status tower tim Dire	452
<i>barracks_status_dire</i>	Status barracks tim Dire	51
<i>barracks_status_radiant</i>	Status barracks tim Radiant	63
<i>first_blood_time</i>	Status waktu <i>first blood</i>	25

Nama Atribut	Keterangan	Contoh Data
<i>game_mode</i>	Status mode pertandingan	22
<i>radiant_win</i>	Status pertandingan yang dihasilkan setiap pertandingan	True = 1 Tim Radiant Menang False = 0 Tim Dire Menang

◆ Pada dataset pertandingan yang terdiri dari 100.000 data dengan 10 atribut, telah dilakukan pengecekan terhadap nilai kosong dan tidak ditemukan nilai kosong dalam dataset **Error! Reference source not found.**

```
df_match.isnull().sum()
match_id      0
start_time    0
duration      0
tower_status_radiant  0
tower_status_dire  0
barracks_status_dire  0
barracks_status_radiant  0
first_blood_time  0
game_mode     0
radiant_win   0
dtype: int64
```

**Gambar 5.** Pengecekan *Missing Value* Pertandingan

Dataset tersebut memiliki tipe data numerik, dan atribut *radiant\_win* perlu diubah menjadi tipe data numerik, sesuai dengan transformasi yang ditunjukkan pada **Error! Reference source not found.**

```
df_match.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100000 entries, 0 to 99999
Data columns (total 10 columns):
#   Column                Non-Null Count  Dtype
---  ---                ---
0   match_id              100000 non-null int64
1   start_time            100000 non-null int64
2   duration              100000 non-null int64
3   tower_status_radiant  100000 non-null int64
4   tower_status_dire     100000 non-null int64
5   barracks_status_dire  100000 non-null int64
6   barracks_status_radiant  100000 non-null int64
7   first_blood_time     100000 non-null int64
8   game_mode             100000 non-null int64
9   radiant_win           100000 non-null bool
dtypes: bool(1), int64(9)
memory usage: 7.0 MB
```

**Gambar 6.** Informasi Tipe Data Pertandingan

Kategori data pemain (*players*) diambil dari file mentah json, yang relevan dengan data setiap pemain dalam pertandingan, termasuk informasi tentang pahlawan (*hero*) dan perlengkapan (*item*) yang digunakan. **Error! Reference source not found.** menyajikan keterangan dan contoh sampel data terkait atribut pada data pemain (*players*).

**Tabel 2.** Keterangan dan Contoh Data Pemain

Nama Atribut	Keterangan	Contoh Data
<i>match_id</i>	Unique setiap pertandingan	7461965918
<i>hero_id</i>	Pemilihan pahlawan ( <i>hero</i> ) unique	120
<i>player_slot</i>	Status pemain/pahlawan pada tim Radiant atau Dire	0
<i>gold</i>	Total uang yang dimiliki pemain	3275
<i>kills</i>	Total <i>kill</i> yang dimiliki pemain	3
<i>deaths</i>	Total kematian yang dimiliki pemain	7
<i>assists</i>	Total <i>assists</i> yang dimiliki pemain	1
<i>item_0</i>	Perlengkapan 1 ( <i>item</i> ) yang dimiliki	25
<i>item_1</i>	Perlengkapan 2 ( <i>item</i> ) yang dimiliki	1
<i>item_2</i>	Perlengkapan 3 ( <i>item</i> ) yang dimiliki	174
<i>item_3</i>	Perlengkapan 4 ( <i>item</i> ) yang dimiliki	143
<i>item_4</i>	Perlengkapan 5 ( <i>item</i> ) yang dimiliki	569
<i>item_5</i>	Perlengkapan 6 ( <i>item</i> ) yang dimiliki	108

Dataset ini terdiri dari 1.000.000 data pemain (100.000 pertandingan dengan masing-masing 5 pemain Radiant dan 5 pemain Dire) dengan 13 atribut yang dipilih. Pengecekan terhadap *Missing Value* atau nilai kosong dalam dataset pemain tidak menemukan adanya nilai kosong **Error! Reference source not found.**

```
df_players.isnull().sum()

match_id      0
hero_id       0
player_slot   0
gold          0
kills         0
deaths        0
assists       0
item_0        0
item_1        0
item_2        0
item_3        0
item_4        0
item_5        0
dtype: int64
```

**Gambar 7.** Pengecekan *Missing Value* Pemain

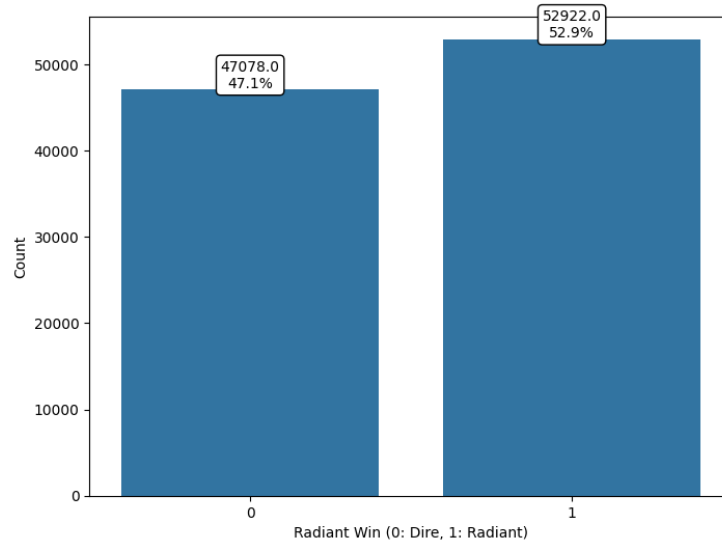
Selain itu, semua tipe data dalam dataset pemain sudah terbentuk numerik terdapat pada **Error! Reference source not found.**

```
df_players.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000000 entries, 0 to 999999
Data columns (total 13 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   match_id    1000000 non-null int64
1   hero_id     1000000 non-null int64
2   player_slot 1000000 non-null int64
3   gold        1000000 non-null int64
4   kills       1000000 non-null int64
5   deaths      1000000 non-null int64
6   assists     1000000 non-null int64
7   item_0      1000000 non-null int64
8   item_1      1000000 non-null int64
9   item_2      1000000 non-null int64
10  item_3      1000000 non-null int64
11  item_4      1000000 non-null int64
12  item_5      1000000 non-null int64
dtypes: int64(13)
```

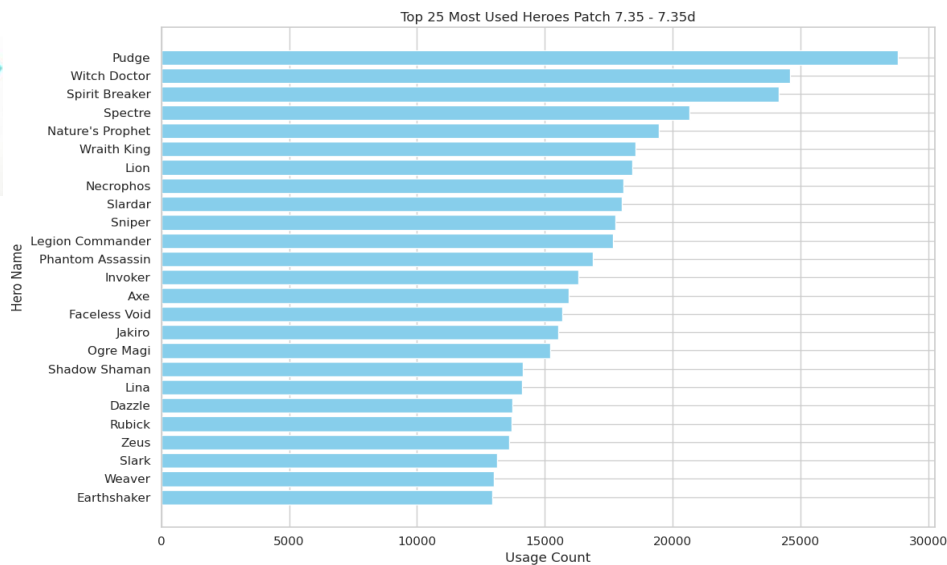
**Gambar 8.** Informasi Tipe Data Pemain

Pemahaman data adalah proses memahami arti dari setiap atribut dalam data. Hal ini penting untuk memilih atribut yang sesuai dalam pengolahan data Pertandingan (*match*) dan Pemain (*players*). Berdasarkan total kemenangan (*radiant\_win*), grafik pada **Error! Reference source not found.** menunjukkan bahwa dari seluruh pertandingan, terdapat 52.922 kemenangan untuk Radiant dan 47.078 kemenangan untuk Dire.



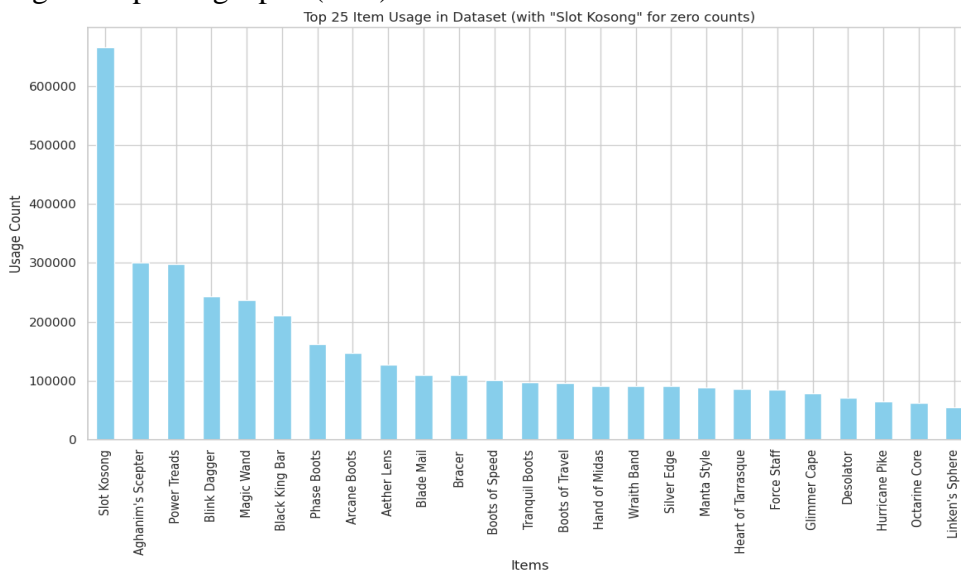
**Gambar 9.** Grafik Ekplorasi Kemenangan

Pada **Error! Reference source not found.** menampilkan grafik yang menunjukkan 25 pahlawan (*hero*) teratas yang paling sering digunakan dalam keseluruhan data pertandingan.



Gambar 10. Grafik Top 25 Pahlawan

Sementara itu, **Error! Reference source not found.** menyajikan grafik yang menunjukkan 25 perlengkapan (*item*) teratas yang paling sering digunakan dalam keseluruhan pertandingan. Slot kosong merupakan ketersediaan perlengkapan (*item*) yang tidak diisi dengan penggunaan perlengkapan (*item*).



Gambar 11. Grafik Top 25 Perlengkapan

Pengolahan data dalam data mining bertujuan untuk mempersiapkan data agar dapat diolah dengan baik. Langkah-langkah yang dilakukan meliputi penetapan fitur prediktor dengan menggabungkan atribut dari dataset pertandingan (*match*) dan pemain (*players*), mengkategorikan 5 pemain dari tim Radiant dan 5 pemain dari tim Dire berdasarkan *player\_slot*, serta membuat kolom baru. Selain itu, tipe data *radiant\_win* yang sebelumnya Boolean diubah menjadi numerik, dengan 0 = Dire Team Menang dan 1 = Radiant Team Menang. **Error! Reference source not found.** menampilkan salah satu sampel pada dataset yang terdiri dari atribut-atribut yang mendeskripsikan fitur-fitur serta target untuk sebuah model prediksi Dota 2. Setiap baris dalam tabel ini mewakili sebuah fitur atau target yang akan digunakan dalam model. Fitur-fitur ini mencakup atribut dari pemain dan pertandingan yang relevan, serta target yang menunjukkan hasil pertandingan.

Tabel 3. Keterangan dan Contoh Sampel Dataset

Nama Atribut	Keterangan	Contoh Data
<i>r_hero_1—r_hero_5</i>	Pahlawan ( <i>hero</i> ) yang dipilih dari tim Radiant urutan 1-5	42, 39, 5, 26, 25

Nama Atribut	Keterangan	Contoh Data
<i>d_hero_1—d_hero_5</i>	Pahlawan ( <i>hero</i> ) yang dipilih dari tim Dire urutan 1-5	72, 1, 111, 27, 2
<i>r_1_item_0—</i> <i>r_1_item_5</i>	Perlengkapan ( <i>item</i> ) urutan 1 – 6 yang digunakan oleh pahlawan ( <i>hero</i> ) 1 pada tim Radiant	0, 1, 2, 6, 7, 11
<i>r_2_item_0—</i> <i>r_2_item_5</i>	Perlengkapan ( <i>item</i> ) urutan 1 – 6 yang digunakan oleh pahlawan ( <i>hero</i> ) 2 pada tim Radiant	24, 26, 30, 31, 6, 1
<i>r_3_item_0—</i> <i>r_3_item_5</i>	Perlengkapan ( <i>item</i> ) urutan 1 – 6 yang digunakan oleh pahlawan ( <i>hero</i> ) 3 pada tim Radiant	28, 31, 18, 9, 5, 4
<i>r_4_item_0—</i> <i>r_4_item_5</i>	Perlengkapan ( <i>item</i> ) urutan 1 – 6 yang digunakan oleh pahlawan ( <i>hero</i> ) 4 pada tim Radiant	2, 4, 6, 7, 8, 34
<i>r_5_item_0—</i> <i>r_5_item_5</i>	Perlengkapan ( <i>item</i> ) urutan 1 – 6 yang digunakan oleh pahlawan ( <i>hero</i> ) 5 pada tim Radiant	36, 56, 41, 37, 20, 1
<i>d_1_item_0—</i> <i>d_1_item_5</i>	Perlengkapan ( <i>item</i> ) urutan 1 – 6 yang digunakan oleh pahlawan ( <i>hero</i> ) 1 pada tim Dire	88, 90, 92, 2, 7, 6
<i>d_2_item_0—</i> <i>d_2_item_5</i>	Perlengkapan ( <i>item</i> ) urutan 1 – 6 yang digunakan oleh pahlawan ( <i>hero</i> ) 2 pada tim Dire	36, 26, 24, 20, 19, 2
<i>d_3_item_0—</i> <i>d_3_item_5</i>	Perlengkapan ( <i>item</i> ) urutan 1 – 6 yang digunakan oleh pahlawan ( <i>hero</i> ) 3 pada tim Dire	2, 5, 36, 29, 12, 27
<i>d_4_item_0—</i> <i>d_4_item_5</i>	Perlengkapan ( <i>item</i> ) urutan 1 – 6 yang digunakan oleh pahlawan ( <i>hero</i> ) 4 pada tim Dire	21, 19, 15, 12, 11, 4
<i>d_5_item_0—</i> <i>d_5_item_5</i>	Perlengkapan ( <i>item</i> ) urutan 1 – 6 yang digunakan oleh pahlawan ( <i>hero</i> ) 5 pada tim Dire	14, 13, 11, 36, 57, 92
<i>radiant_win</i>	Tim Dire Menang = 0 Tim Radiant Menang = 1	1

### c. Pemodelan dan Evaluasi

Berdasarkan fitur pemilihan pahlawan (*hero*) dan perlengkapan (*item*) terhadap masing-masing tim Radiant dan Dire berjumlah 70 atribut, sedangkan target terdapat 2 kelas Radiant Menang dan Dire Menang, terkait kinerja prediksi model Random Forest dan XGBoost berdasarkan evaluasi Confusion Matrix dan AUROC disajikan pada **Error! Reference source not found.** dan **Error! Reference source not found.**. Pada **Error! Reference source not found.**, disajikan informasi mengenai hasil dari model yang diuji dengan menggunakan pembagian data menggunakan teknik Confusion Matrix dengan rasio 80:20, 75:25, dan 70:30. Sedangkan pada **Error! Reference source not found.** menyajikan dengan teknik evaluasi AUROC.

**Tabel 4.** Hasil Evaluasi Kinerja Model Prediksi Confusion Matrix

No	Skenario	Model	Target	Accuracy (%)	Precision (%)	Recall (%)	f1-score (%)	
1	80:20	Random Forest	Dire	52	49	35	40	
			Menang (0)					
			Radiant					
		XGBoost	Menang (1)		52	49	67	59
			Dire					
			Menang (0)					
2	75:25	Random Forest	Dire	51	48	34	40	
			Menang (0)					
			Radiant					
		XGBoost	Menang (1)		52	49	67	59
			Dire					
			Menang (0)					
3	70:30	Random Forest	Dire	52	48	34	40	
			Menang (0)					
			Radiant					
		XGBoost	Menang (1)		52	49	62	58
			Dire					
			Menang (0)					

No	Skenario	Model	Target	Accuracy (%)	Precision (%)	Recall (%)	f1-score (%)
		XGBoost	Radiant Menang (1)	52	53	67	59
			Dire Menang (0)		49	41	45
			Radiant Menang (1)		54	62	58
			Dire Menang (0)				

Model Random Forest, dalam tiga skenario, evaluasi Confusion Matrix menunjukkan akurasi berkisar antara 51%-52%. Precision dan Recall untuk kelas Dire Menang cukup rendah dengan F-1 score sekitar 40%, sedangkan untuk kelas Radiant Menang lebih tinggi dengan F-1 score sekitar 59%. Model XGBoost, dalam tiga skenario, memiliki akurasi yang konsisten di 52%. Precision dan Recall untuk kelas Dire Menang lebih tinggi dibandingkan dengan Random Forest, dengan F-1 score sekitar 45%.

Model XGBoost lebih konsisten dalam hal akurasi dan memberikan performa yang lebih baik dalam precision dan recall untuk kelas Dire Menang. Untuk kelas Radiant Menang, performa kedua model cukup bersaing dengan F1-score yang hampir serupa. Skenario terbaik untuk kedua model adalah 80:20, di mana precision dan recall lebih seimbang untuk kedua kelas. Berdasarkan analisis ini, model XGBoost dengan skenario 80:20 merupakan pilihan terbaik karena memberikan keseimbangan terbaik antara akurasi, precision, recall, dan F1-score untuk kedua kelas.

**Tabel 5.** Hasil Evaluasi Kinerja Model Prediksi AUROC

Model	Skenario	AUROC (%)	AUROC Plot
Random Forest	80:20	50.79	
	75:25	50.51	
	70:30	50.70	
XGBoost	80:20	51.49	
	75:25	51.47	
	70:30	51.31	

Kedua model, Random Forest dan XGBoost, menunjukkan performa yang sangat dekat dengan nilai 50% dalam hal AUROC, yang menunjukkan bahwa kemampuan mereka dalam membedakan kelas positif dan negatif sangat terbatas. Meskipun XGBoost menunjukkan

performa yang sedikit lebih baik dibandingkan Random Forest, peningkatan ini tidak signifikan. Evaluasi menunjukkan bahwa algoritma XGBoost dengan skenario 80:20 memberikan performa yang sedikit lebih baik dibandingkan dengan model Random Forest sebelumnya, dengan nilai AUROC sedikit di atas 0.5.

### Pembahasan

Berdasarkan hasil evaluasi model Random Forest dan XGBoost menunjukkan performa yang berbeda dalam memprediksi hasil pertandingan antara tim Radiant dan Dire. Hasil dari Confusion Matrix menunjukkan bahwa model Random Forest memiliki akurasi antara 51%-52% dengan *precision* dan *recall* yang rendah untuk kelas Dire Menang menghasilkan *F1-score* sekitar 40%, sementara untuk kelas Radiant Menang, *F1-score* mencapai sekitar 59%. Model XGBoost, di sisi lain, menunjukkan akurasi konsisten sebesar 52% dengan *precision* dan *recall* yang lebih baik untuk kelas Dire Menang, menghasilkan *score* sekitar 45%, serta performa hampir serupa dengan Random Forest untuk kelas Radiant Menang. skenario 80:20 memberikan hasil terbaik untuk kedua model, dengan keseimbangan terbaik antara *precision* dan *recall* untuk kedua kelas.

Evaluasi AUROC mengungkapkan bahwa kemampuan kedua model dalam membedakan kelas positif dan negatif sangat terbatas, dengan nilai mendekati 50%. Meskipun XGBoost sedikit lebih unggul dibandingkan Random Forest dalam AUROC, peningkatan ini tidak signifikan. Oleh karena itu, model XGBoost dengan skenario 80:20 direkomendasikan sebagai pilihan terbaik.

### SIMPULAN DAN SARAN

Penelitian ini mengungkapkan bahwa algoritma XGBoost mampu memanfaatkan fitur pemilihan pahlawan (*hero*) dan perlengkapan (*item*) secara efektif, menunjukkan kinerja superior dengan rata-rata akurasi tertinggi pada skenario 80:20 untuk prediksi hasil pertandingan Dota 2. Implementasi model terbaik XGBoost dilakukan menggunakan framework Flask dengan tampilan frontend HTML dan Tailwind CSS. Evaluasi menggunakan Confusion Matrix dan AUROC menunjukkan bahwa XGBoost memberikan keseimbangan terbaik antara akurasi, presisi, dan *f1-score* untuk kedua kelas Dire Menang dan Radiant Menang, dengan nilai AUROC 51.49%.

Penelitian selanjutnya disarankan untuk mengumpulkan data pertandingan lebih baru, khususnya dari High Skill MMR 8000+, dan mempertimbangkan penggunaan FACET pada *Patch 7.36*. Disarankan juga menggunakan dataset mentah dengan fitur lebih lengkap seperti role *hero*, mengeksplorasi model algoritma lain selain Random Forest dan XGBoost.

### DAFTAR PUSTAKA

- [1] A. Torres-Toukoumidis, *Esports and the Media: Challenges and Expectations in a Multi-Screen Society*. 2022. doi: 10.4324/9781003273691.
- [2] J. Hamari and M. Sjöblom, "What is eSports and why do people watch it?," *Internet Res.*, vol. 27, no. 2, pp. 211–232, 2017, doi: 10.1108/IntR-04-2016-0085.
- [3] R. A. Haristu and P. H. P. Rosa, "Penerapan Metode Random Forest untuk Prediksi Win Ratio Pemain Player Unknown Battleground," *MEANS (Media Inf. Anal. dan Sist.*, vol. 4, no. 2, pp. 120–128, 2019, doi: 10.54367/means.v4i2.545.
- [4] Stanlly, F. A. Putra, and N. N. Qomariyah, "DOTA 2 Win Loss Prediction from Item and Hero Data with Machine Learning," *Proc. 2022 IEEE Int. Conf. Ind. 4.0, Artif. Intell. Commun. Technol. IAICT 2022*, pp. 204–209, 2022, doi: 10.1109/IAICT55358.2022.9887525.
- [5] C. H. Ke *et al.*, "DOTA 2 match prediction through deep learning team fight models,"

- IEEE Conf. Comput. Intell. Games, CIG*, vol. 2022-August, pp. 96–103, 2022, doi: 10.1109/CoG51982.2022.9893647.
- [6] V. J. Hodge, S. Devlin, N. Sephton, F. Block, P. I. Cowling, and A. Drachen, “Win Prediction in Multiplayer Esports: Live Professional Match Prediction,” *IEEE Trans. Games*, vol. 13, no. 4, pp. 368–379, 2021, doi: 10.1109/TG.2019.2948469.
- [7] G. A. Sandag, “Model Prediksi Kemenangan Tim dalam Game League of Legend Menggunakan Algoritma Decision Tree,” *J. Komput. Terap.*, vol. 7, no. 1, pp. 42–52, 2021.
- [8] F. Y. Pamuji and V. P. Ramadhan, “Komparasi Algoritma Random Forest dan Decision Tree untuk Memprediksi Keberhasilan Immunotherapy,” *J. Teknol. dan Manaj. Inform.*, vol. 7, no. 1, pp. 46–50, 2021, doi: 10.26905/jtmi.v7i1.5982.
- [9] S. Hagedorn, S. Kläbe, and K. U. Sattler, “Putting Pandas in a Box,” *11th Annu. Conf. Innov. Data Syst. Res. CIDR 2021*, 2021.
- [10] Z. Jin, J. Shang, Q. Zhu, C. Ling, W. Xie, and B. Qiang, “RFRSF: Employee Turnover Prediction Based on Random Forests and Survival Analysis,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12343 LNCS, pp. 503–515, 2020, doi: 10.1007/978-3-030-62008-0\_35.
- [11] H. Mo, H. Sun, J. Liu, and S. Wei, “Developing window behavior models for residential buildings using XGBoost algorithm,” *Energy Build.*, vol. 205, p. 109564, 2019, doi: 10.1016/j.enbuild.2019.109564.
- [12] H. R. Ngemba *et al.*, “Penerapan Algoritma K-Nearest Neighbor untuk Menentukan Potensi Ekspor Komoditas Pertanian di Provinsi Sulawesi Tengah,” *J. Teknol. dan Manaj. Inform.*, vol. 9, no. 2, pp. 151–160, 2023, doi: 10.26905/jtmi.v9i2.10235.
- [13] R. D. Riley, B. Van Calster, and G. S. Collins, “A note on estimating the Cox-Snell R<sup>2</sup> from a reported C statistic (AUROC) to inform sample size calculations for developing a prediction model with a binary outcome,” *Stat. Med.*, vol. 40, no. 4, pp. 859–864, 2021, doi: 10.1002/sim.8806.