



**Klasifikasi Kekuatan Karakter Columbina dalam Permainan *Genshin Impact* Menggunakan Metode Klasifikasi Softmax**

Dionisius Anthoni Pratama<sup>1\*</sup>

<sup>1</sup> Program Studi Statistika, Jurusan Statistika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Jenderal Soedirman, Purwokerto, Indonesia

E-mail Koresponden : [dionisius.pratama@mhs.unsoed.ac.id](mailto:dionisius.pratama@mhs.unsoed.ac.id)

**Abstrak.** *Video game* telah menjadi industri yang banyak diminati. Salah satu *video game* yang populer di kalangan masyarakat yaitu *Genshin Impact*. Setiap pemain dapat mengendalikan karakter dalam lingkungan virtual, mulai dari menjelajah dunia hingga bertempur. Dalam *Genshin Impact* setiap karakter yang bertempur memiliki tingkat kekuatan (*tier*). Menggunakan klasifikasi Softmax, penelitian ini melakukan klasifikasi terhadap karakter Columbina. Setelah menggunakan klasifikasi Softmax kekuatan Columbina berpeluang masuk ke dalam *tier* SS.

**Kata kunci:** Columbina, *Genshin Impact*, klasifikasi, Softmax,

## 1 Pendahuluan

*Video game* telah menjadi industri hiburan yang besar dan terkenal di banyak kalangan masyarakat. Pada *video game*, pemain diperbolehkan untuk berinteraksi dalam lingkungan virtual. Salah satu *video game* yang memperbolehkan pemain berinteraksi dalam lingkungan virtual tersebut yaitu *Genshin Impact*. Dalam lingkungan virtualnya, *Genshin Impact* menyajikan visualisasi geografis seperti perkotaan, pegunungan, gurun, hutan, hingga gunung berapi. Wilayah tersebut disajikan dalam dunia yang disebut dengan *Teyvat*.

Selain itu, pemain akan bermain sebagai karakter hingga dapat mengendalikan karakter. Setiap karakter memiliki peran dan elemen masing-masing. Peran tersebut terbagi menjadi *main-DPS* (*Damage Per Second*) yang berperan untuk bertarung secara signifikan, *sub-DPS* yang berperan untuk bertarung secara pasif, dan *Support* yang bertugas untuk memberi bantuan seperti efek *buff* pada DPS. Elemen karakter terbagi menjadi tujuh elemen utama, yaitu api (*Pyro*), air (*Hydro*), es (*Cryo*), listrik (*Electro*), tanah (*Dendro*), angin (*Anemo*), dan batu (*Geo*). Setiap karakter memiliki tipe senjata yang berbeda-beda, dengan tipe senjata setiap karakter dalam *Genshin Impact* terbagi menjadi pedang kecil (*Sword*), pedang besar (*Claymore*), katalis (*Catalyst*), panah (*Bow*), dan tombak (*Polearm*).

Sistem pertarungan dalam *Genshin Impact* memiliki sistem pertarungan waktu nyata (*real time*).

Pemain dapat menggunakan serangan dasar (*Basic Attack*), serangan elemental (*Elemental Skill/E*), dan serangan *burst* (*Elemental Burst/Q*) untuk melawan musuh. Tak jarang, beberapa karakter memiliki serangan pasif yang unik seperti Raiden Shogun dan Mavuika. Karakter yang memiliki elemental tertentu, dapat memicu serangan elemental tertentu pada setiap serangan dasarnya. Dalam pertarungan, elemental memegang peran penting untuk memainkan reaksi. Serangan karakter yang memiliki elemental berbeda dapat bereaksi membentuk reaksi elemental, tetapi terdapat elemental yang tidak dapat bereaksi.

*Genshin Impact* rutin memperbarui versi permainannya setiap 40 hari. Hingga saat ini *Genshin Impact* telah menyentuh versi 6.1 atau LUNA II. Terhitung dari awal perilisan, September 2020 hingga saat ini, Oktober 2025, *Genshin Impact* telah merilis 105 karakter dalam enam versi yang berbeda. Setiap versinya terdapat fenomena *powercreep*<sup>1</sup> yang membuat kekuatan karakter-karakter lama menjadi tidak relevan. Hal ini dapat membuat adanya sebuah perbedaan antara kekuatan karakter yang dirilis pada awal versi dengan karakter yang dirilis pada versi saat ini.

Dalam video game, terdapat konsep daftar tingkat untuk menentukan kekuatan karakter berdasarkan kelangkaan, urutan perilisan, elemen, peran, hingga tipe senjata tertentu. Karakter tertinggi merupakan karakter yang mempunyai kekuatan tinggi dibanding dengan karakter-karakter lainnya. Sehingga, pemain akan lebih banyak menggunakan karakter yang mempunyai kekuatan tinggi. Columbina, atau dapat dipanggil dengan “*The Damslette*” merupakan karakter *Genshin Impact* yang akan dirilis pada versi LUNA. Ia dikisahkan sebagai dewi bulan yang lahir di sebuah pulau yang bernama pulau Hiisi. Dalam permainan, Columbina merupakan karakter bintang 5 yang memiliki elemen *Hydro*, mempunyai peran sebagai *support*, dan pengguna senjata *catalyst*.

---

<sup>1</sup> Fenomena dalam sebuah permainan di mana konten baru, seperti karakter atau item, secara bertahap menjadi lebih kuat dan mengungguli konten lama, membuatnya kurang relevan atau bahkan tidak berguna

Penggunaan analisis statistik dalam permainan dapat ditemukan pada penelitian Ellington dkk [1]. Seperti pada penelitian sebelumnya yang telah dilakukan oleh Pratama [9], untuk mengetahui kekuatan sebuah karakter dalam permainan Genshin Impact dapat menggunakan metode klasifikasi. Penggunaan model regresi logistik dalam penelitian ini sejalan dengan penelitian Novianti dan Rizki [8] serta penggunaan fungsi Softmax untuk memprediksi sejalan dengan penelitian Kusnadi dkk [7].

Teknik *data mining* merupakan teknik statistik, matematis, dan kecerdasan buatan untuk melakukan ekstraksi dan memperoleh informasi yang bermanfaat dari kumpulan *data lake*. Setelah *data mining*, dilakukan proses klasifikasi menggunakan metode klasifikasi Softmax, yaitu klasifikasi berdasarkan probabilitas sebuah objek menggunakan regresi logistik multinomial atau regresi yang digunakan untuk data kategorik bertipe nominal pada proses *machine learning*. Penggunaan *software R* sebagai perangkat lunak untuk membantu analisis data memudahkan pemodelan untuk menentukan nilai probabilitas setiap kelas. Fitur penelitian ini berjumlah lima variabel, dengan jumlah kelas sebanyak dua puluh tiga kelas.

## 2 Metodologi

Data yang digunakan pada penelitian ini merupakan data sekunder pada Tabel 1. yang diperoleh dari Game 8, dalam artikel internet berjudul “6.0 Tier List and the Best Characters as of October 2025” untuk karakter yang mempunyai konstelasi 0 (*C0 constellation*). Diperoleh label berupa tingkat kekuatan (*tier*) karakter dengan peran yang berbeda dan terbagi menjadi 6 kelas [1] :

**Tabel 1** Data Karakter

	SS	S	A	B	C	D	Jumlah
Jumlah karakter	17	27	24	29	22	18	137

Penggunaan koding pada setiap kelas fitur seperti tingkat bintang (*star tier*), peran, elemen, dan tipe senjata diberikan pada Tabel 2.

**Tabel 2** Tabel Fitur

	<b>Fitur</b>	<b>Kode</b>	<b>Kelas</b>	<b>Kategori</b>
1	<i>Star Tier (X<sub>1</sub>)</i>	1	1	Bintang 4*
		2	2	Bintang 5*
2	<i>Peran (X<sub>2</sub>)</i>	1	3	Main-DPS
		2	4	Sub-DPS
3	<i>Elemen (X<sub>3</sub>)</i>	3	5	Support
		1	6	Anemo
4	<i>Tipe Senjata (X<sub>4</sub>)</i>	2	7	Geo
		3	8	Electro
5	<i>Urutan Perilisan Karakter (X<sub>5</sub>)</i>	4	9	Dendro
		5	10	Hydro
6		6	11	Pyro
		7	12	Cryo
7		1	13	Sword
		2	14	Claymore
8		3	15	Catalyst
		4	16	Bow
9		5	17	Polearm
		1	18	Versi 1.x
10		2	19	Versi 2.x
		3	20	Versi 3.x
11		4	21	Versi 4.x
		5	22	Versi 5.x
12		6	23	Versi LUNA

## 2.1 Regresi Logistik Multinomial

Regresi logistik merupakan regresi yang digunakan untuk menghitung data kategorik, baik nominal maupun ordinal [6]. Probabilitas pada kejadian variabel respon dapat ditentukan berdasarkan kejadian variabel prediktor. Sedangkan, regresi logistik multinomial merupakan regresi logistik yang digunakan untuk menentukan probabilitas variabel dependen, saat terdapat variabel prediktor yang mempunyai lebih dari dua kategorik [6]. Regresi logistik merupakan konsep dasar pada klasifikasi Softmax, yang akan digunakan sebagai metode *machine learning* [4]. Model probabilitas regresi logistik multinomial, yang menjadikan kelas pertama sebagai kelas referensi diberikan sebagai berikut [8]:

$$g_k(x) = x^T \beta_k \quad (1)$$

dengan:

$$\ln\left(\frac{P(y = k|x)}{P(y = 1|x)}\right) = x^T \beta_k \quad (2)$$

sehingga probabilitas objek di kelas ke- $k$ :

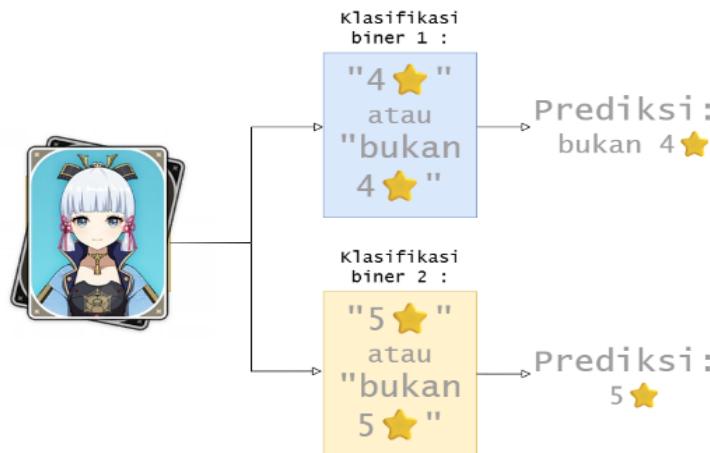
$$P(y = k | x) = \frac{e^{g_k(x)}}{\sum_{i=1}^n e^{g_i(x)}} \quad (3)$$

di mana:

- $P(y = k | x)$  : Probabilitas fitur dalam kelas ke- $k$
- $g_k(x)$  : Model regresi logistik multinomial kelas ke- $k$
- $x$  : Vektor biner untuk fitur regresi ke- $p$  dari kelas ke- $k$
- $n$  : Banyaknya jumlah kelas
- $\beta_k$  : Vektor koefisien dari kelas ke- $k$ .

## 2.2 Konsep Klasifikasi Softmax

Konsep klasifikasi dalam *machine learning* yaitu memprediksi kelas sebuah data dari sekumpulan kelas. Terdapat klasifikasi biner, yaitu klasifikasi yang hanya memprediksi satu dari dua buah kelas [3]. Klasifikasi biner menjadi pondasi dasar untuk klasifikasi Softmax. Berikut adalah gambaran klasifikasi biner dengan variabel ‘bintang (star tier)’ yang mempunyai dua kelas, sebagaimana ditampilkan pada Gambar 1.



Gambar 1 Konsep klasifikasi biner

Klasifikasi Softmax merupakan bentuk machine learning yang menggunakan konsep regresi logistik multinomial, sebagai metode pelatihan mesin untuk melatih mesin. Dalam klasifikasi Softmax, fungsi sigmoid akan mengubah nilai objek dalam suatu kelas menjadi probabilitas [4]. Nilai probabilitas tersebut akan digunakan untuk memprediksi kategori kelas objek. Dalam *machine learning* menggunakan klasifikasi Softmax, setiap kelas dalam fitur dianggap sebagai klasifikasi biner, sehingga setiap kelas variabel kategorik harus

diubah menjadi format biner (*one hot encoding*) [3]. Misal seperti pada **Gambar 1**, fitur bintang (*star tier*) yang mempunyai dua kelas, yaitu bintang 4\* atau bintang 5\* akan dibagi menjadi dua klasifikasi, yaitu karakter yang bintang 4\* atau bukan bintang 4\* dan bintang 5\* atau bukan bintang 5\*. Banyaknya klasifikasi biner pada sebuah fitur akan sama dengan banyaknya kelas pada fitur tersebut.

### 2.3 Evaluasi Metrik Model

Dalam klasifikasi multikelas, model perlu dievaluasi menggunakan metrik-metrik evaluasi. Evaluasi metrik model dilakukan untuk melihat seberapa baik model yang telah dibuat untuk memprediksi data baru. Evaluasi model diperlukan, karena terdapat kelas yang berpindah saat dievaluasi menggunakan model [5]. Metrik evaluasi yang digunakan untuk kelayakan model yaitu *precision* dan *recall*, nilai *accuracy*, nilai F1 score, nilai log-loss, Mattheus *Corelation Coefficients* dan Choheren's Cappa.

#### 2.3.1 *Precision* dan *Recall*

*Precision* adalah proporsi perbandingan kelas yang diprediksi secara benar dengan jumlah kelas yang sebenarnya,

$$Precision_k = \frac{TP_k}{TP_k + FP_k}. \quad (4)$$

Dengan:

- |        |   |
|--------|---|
| $TP_k$ | : <i>True Positive</i> dalam kelas ke- $k$  |
| $FP_k$ | : <i>False Positive</i> dalam kelas ke- $k$ |

Sedangkan *Recall* merupakan proporsi perbandingan kelas yang diprediksi secara benar dengan jumlah kelas yang diprediksi.

$$Recall_k = \frac{TP_k}{TP_k + FN_k}. \quad (5)$$

dengan:

- |        |   |
|--------|---|
| $FN_k$ | : <i>False Negative</i> dalam kelas ke- $k$ . |
|--------|---|

#### 2.3.2 Nilai *Accuracy*

*Accuracy* merupakan metrik evaluasi berbentuk probabilitas yang digunakan untuk mengukur seberapa baik model dalam prediksi kelas pada data. Angka probabilitas

*Accuracy* pada model jauh lebih baik digunakan, dibanding angka probabilitas dengan model menebak kelas data secara acak ( $P = \frac{1}{K}$ ), terutama pada klasifikasi kelas yang labelnya memiliki banyak kelas. Persamaan matematis untuk menghitung *Accuracy* diukur pada persamaan (6).

$$Accuracy = \frac{\text{Trace}(\text{Confusion Matrix})}{N}. \quad (6)$$

Sedangkan untuk kasus yang multikelas, digunakan *Weighted Balanced Accuracy* pada persamaan (7).

$$WBA = \sum_K \frac{w_k \cdot TP_k}{TP_k + FN_k}. \quad (7)$$

### 2.3.3 Nilai *F1 Score*

*F1 Score* merupakan rata-rata gabungan antara proporsi kelas yang diprediksi dan kelas yang sebenarnya, atau rata-rata harmonik antara *Precision* dan *Recall* diberikan pada persamaan (8).

$$F1\ Score = \frac{2}{K} \cdot \frac{\sum_K Precision \times \sum_K Recall}{\sum_K Precision + \sum_K Recall}. \quad (8)$$

Untuk kasus klasifikasi yang multikelas, digunakan Macro *F1 Score* dan Micro *F1 Score*. berturut-turut dihitung dengan persamaan (9) dan (10) :

$$Weighted\ F1\ Score = 2 \cdot \frac{Precision \times Recall}{Precision + Recall}. \quad (9)$$

$$Micro\ F1\ Score = 2 \cdot \frac{\sum_K TP_k}{\sum_K Total\ Column_k} = \frac{\sum_K TP_k}{Grand\ Total}. \quad (10)$$

### 2.3.4 Nilai *Log-loss*

Log loss atau Cross Entropy Loss merupakan probabilitas model dalam melakukan kesalahan prediksi. Dalam metrik ini, probabilitas model diukur saat melakukan prediksi kelas yang benar, dengan persamaan (11) .

$$H(p(y), p(\hat{y})) = - \sum_N \sum_K p(Y_i = k | X_i) \log[p(\hat{Y}_i = k | X_i)]. \quad (11)$$

### 2.3.5 *Mattheus Corelation Coefficients*

Koefisien korelasi Mattheus (MCC) merupakan koefisien korelasi antara data yang diprediksi dengan data sebenarnya. Nilai MCC terletak pada interval  $[-1,1]$ , model memiliki korelasi negatif jika mendekati  $-1$ , memiliki korelasi positif jika mendekati  $1$ , dan tidak memiliki korelasi jika mendekati  $0$ . Nilai koefisien korelasi Mattheus dihitung dengan persamaan (12):

$$MCC = \frac{TP_k \cdot N - \sum_K FP_k \cdot FN_k}{\sqrt{(N^2 - (\sum_K FP_k)^2)(N^2 - (\sum_K FN_k)^2)}}. \quad (12)$$

### 2.3.6 *Coheren's Cappa*

Nilai koheren kappa adalah metrik evaluasi yang digunakan untuk mengukur seberapa konsisten antara data yang diprediksi dengan data sebenarnya, yang nilainya ditentukan dengan persamaan (13):

$$\kappa = \frac{TP_k \cdot N - \sum_K FP_k \cdot FN_k}{N^2 - \sum_K FP_k \cdot FN_k}. \quad (13)$$

## 2.4 Langkah-langkah Penelitian

Langkah selanjutnya yang digunakan untuk menentukan kelas objek menggunakan klasifikasi Softmax yaitu:

1. mendapatkan data label dari sumber (*data lake*);
2. enkoding data dengan konsep *one hot encoding*, mengubah setiap kelas dalam fitur menjadi klasifikasi biner. Sehingga, objek dapat dinyatakan sebagai vektor logika;
3. melatih model, dimulai dari menentukan model regresi logistik multinomial;
4. membagi data set menjadi tiga bagian, yaitu *training set* sebesar 50%, *validation set* sebesar 50%, dan *testing set* sebagai data yang akan diprediksi, yaitu tier Columbina;
5. menguji kelayakan prediksi model dengan memperhatikan *Precision*, *Recall*, *Akurasi*, *F1 Score*, *Log-loss*, *MCC*, dan *Cohen's Cappa*;
6. tahap pengujian model *testing set*, yaitu memprediksi kelas karakter Columbina menggunakan model yang sah, dengan menjadikan *tier SS* sebagai pembanding utama; dan
7. menginterpretasikan hasil prediksi model, mendeskripsikan *output* data uji.

### 3 Hasil dan Pembahasan

#### 3.1 Eksplorasi Data Latih

Pada tahap eksplorasi data latih, ditentukan proporsi masing-masing fitur dengan jumlah perbandingan setiap kelasnya. Proporsi terbagi menjadi proporsi data secara keseluruhan, proporsi data set untuk training set, dan proporsi data set untuk validation set. Perhitungan dilakukan menggunakan sintaks yang sudah dibentuk pada pemrograman R.

##### 3.1.1 Proporsi Tingkat Kekuatan (*Tier*)

Menggunakan Rstudio, didapatkan proporsi pada label “*Character Tier*” atau “Kelas Karakter (*Y*)” yang hasilnya tersedia pada Tabel 3 :

**Tabel 3** Proporsi Character Tier (*Y*):

	SS	S	A	B	C	D
Data	0.1240876	0.1970803	0.1678832	0.2189781	0.1605839	0.1313869
Training set	0.1285714	0.2000000	0.1714286	0.2142857	0.1571429	0.1285714
Validation set	0.1194030	0.1940299	0.1641791	0.2238806	0.1641791	0.1343284

Setelah membagi data menjadi data latih dan data validasi, dapat dilihat pada Tabel 3. bahwa proporsi tertinggi pada data latih adalah *tier* B sebesar 21,42857% dan proporsi tertinggi pada data validasi adalah *tier* B sebesar 22,38806%.

##### 3.1.2 Proporsi Fitur

Menggunakan Rstudio, didapatkan proporsi pada fitur “*Star Tier* ( $X_1$ ), Peran ( $X_2$ ), Elemen ( $X_3$ ), Tipe Senjata ( $X_4$ ), dan Urutan Perilisan ( $X_5$ )” yang hasilnya disediakan pada Tabel 4.

**Tabel 4** Data Proporsi Fitur ( $X_i$ )

No	Fitur	Kode	Kelas	Kategori	Proporsi	Proporsi training set	Proporsi validation set
1	<i>Star Tier</i> ( $X_1$ )	1	1	Bintang 4*	0.4160584	0.4285714	0.4029851
		2	2	Bintang 5*	0.5839416	0.5714286	0.5970149
2	Peran ( $X_2$ )	1	3	Main-DPS	0.3284672	0.2857143	0.3731343
		2	4	Sub-DPS	0.3065693	0.3428571	0.2686567
		3	5	Support	0.3649635	0.3714286	0.358209

**Tabel 4** Data Proporsi Fitur ( $X_i$ )(Lanjutan)

No	Fitur	Kode	Kelas	Kategori	Proporsi	Proporsi <i>training set</i>	Proporsi <i>validation set</i>
3	Elemen ( $X_3$ )	1	6	Anemo	0.1459854	0.1	0.1940299
		2	7	Geo	0.1021898	0.1428571	0.05970149
		3	8	Electro	0.1751825	0.1714286	0.1791045
		4	9	Dendro	0.1094891	0.05714286	0.1641791
		5	10	Hydro	0.1386861	0.1571429	0.119403
		6	11	Pyro	0.1605839	0.1857143	0.1343284
		7	12	Cryo	0.1678832	0.1857143	0.1492537
4	Tipe Senjata ( $X_4$ )	1	13	Sword	0.2189781	0.2142857	0.2238806
		2	14	Claymore	0.189781	0.1714286	0.2089552
		3	15	Catalyst	0.2335766	0.2428571	0.2238806
		4	16	Bow	0.1532847	0.1714286	0.1343284
		5	17	Polearm	0.2043796	0.2	0.2089552
5	Urutan Perilisan Karakter ( $X_5$ )	1	18	Versi 1.x	0.3211679	0.6142857	0.01492537
		2	19	Versi 2.x	0.1751825	0.2285714	0.119403
		3	20	Versi 3.x	0.1532847	0.07142857	0.238806
		4	21	Versi 4.x	0.1386861	0.05714286	0.2238806
		5	22	Versi 5.x	0.1605839	0.02857143	0.2985075
		6	23	Versi	0.03649635	0	0.07462687
LUNA							

Setelah membagi data menjadi data latih dan data validasi, dapat dilihat pada Tabel 4. bahwa proporsi tertinggi masing-masing fitur yaitu fitur *Star Tier* data latih sebesar 57,14286% pada kelas Bintang 5\* dan data validasi sebesar 59,70149% pada kelas Bintang 5\*, fitur Peran data latih sebesar 37,14286% pada kelas Support dan data validasi sebesar 37,31343% pada kelas Main-DPS, fitur Elemen data latih sebesar 18,57143% pada kelas Pyro dan Cryo dan data validasi sebesar 17,91045% pada kelas Electro, fitur Tipe Senjata data latih sebesar 24,28571% pada kelas Catalyst dan data validasi sebesar 22,38806% pada kelas Sword dan Catalyst, dan fitur Urutan Perilisan Karakter data latih sebesar 61,42857% pada kelas Versi 1.x dan data validasi sebesar 29,85075% pada kelas Versi 5.x.

### 3.2 Menguji Kemampuan Prediksi Menggunakan Uji Kelayakan Model

Setelah membentuk model, dilakukan pengujian kemampuan model dalam memprediksi data. Data latih yang digunakan adalah data yang digunakan untuk membentuk model sebesar 50% dan data validasi yang digunakan adalah data yang digunakan untuk membentuk model sebesar 50%, sehingga dapat dilihat seberapa konsisten model memprediksi dengan proporsi data yang sama.

### 3.2.1 Tabel Metrik Evaluasi Model

Setelah membentuk model seperti pada persamaan (3) menggunakan fungsi multinom() pada *software R*, perlu dilakukan metrik evaluasi untuk menguji kelayakan model yang telah dibentuk.

**Tabel 5** Tabel F1 Score setiap kelas

Nilai	Training set	Validation set
1. <i>Precision</i>	0.67292	0.51561
2. <i>Recall</i>	0.63906	0.52854
3. <i>Nilai Accuracy</i>		
<i>Accuracy</i>	0.64286	0.52238
<i>Weighted</i>	0.63906	0.52854
<i>Balanced</i>		
<i>Accuracy</i>		
4. <i>F1 Score</i>		
<i>Macro F1 Score</i>	0.63906	0.52854
<i>Micro F1 Score</i>	0.64286	0.52239
5. <i>Log-loss</i>	1.056	1.179
6. <i>Mattheus</i>	0.57112	0.42861
<i>Coefficient</i>		
<i>Corelation</i>		
7. <i>Cohen's Cappa</i>	0.56758	0.42458

### 3.2.2 Interpretasi Metrik Evaluasi

Berdasarkan hasil metrik evaluasi, diperoleh nilai *Precision* data latih adalah sebesar 67,292% dan untuk data validasi adalah sebesar 51,561%. *Precision* mengalami penurunan saat model memprediksi data latih dan data validasi sebesar 15,731%, menandakan model mengalami sedikit *overfitting*.

Berdasarkan hasil metrik evaluasi, diperoleh nilai *Recall* data latih adalah sebesar 63,906% dan untuk data validasi adalah sebesar 52,854%. *Recall* mengalami penurunan saat model memprediksi data latih dan data validasi sebesar 11,052%, menandakan model mengalami sedikit *overfitting*.

Berasarkan hasil metrik evaluasi, diperoleh nilai *Accuracy* data latih adalah sebesar 64,286% dan setelah diseimbangkan oleh WBA menjadi 63,609%. Penurunan akurasi data latih, yang sudah diseimbangkan, sebesar 0,677% menandakan bahwa data latih

sebelumnya sudah cukup seimbang. Sedangkan, nilai *Accuracy* data validasi adalah sebesar 52,283% dan setelah diseimbangkan oleh WBA menjadi 52,854%. Kenaikan akurasi data validasi yang sudah diseimbangkan sebesar 0,571% menandakan bahwa data validasi sebelumnya sudah cukup seimbang, tetapi lebih baik jika diseimbangkan menggunakan WBA. Kedua nilai akurasi model jauh lebih baik dibanding model yang menebak kelas objek secara acak (*random guessing*), yang setiap objeknya mempunyai peluang sekitar 16,667% untuk masuk ke dalam suatu kelas. Nilai *Accuracy* data latih dan data validasi mengalami penurunan sebesar 12,003%, sedangkan pada nilai *Accuracy* yang sudah diseimbangkan sebesar 10,755%, menandakan bahwa model mengalami sedikit kendala dalam memprediksi pola kelas positif pada data validasi.

Berdasarkan hasil metrik evaluasi, diperoleh nilai *Macro F1 Score* data latih sebesar 63,906% dan nilai *Micro F1 Score* data latih sebesar 64,286%. Hasil menandakan bahwa terdapat kenaikan sebesar 0,380% pada nilai *F1 Score* data latih yang telah diseimbangkan, tetapi sangat kecil. Dengan demikian nilai agregat setiap kelas pada data latih sudah cukup seimbang. Sedangkan, nilai *Macro F1 Score* data validasi sebesar 52,854% dan nilai *Micro F1 Score* data validasi sebesar 52,239%. Hasil menandakan bahwa terdapat penurunan sebesar 0,615% pada nilai *F1 Score* data validasi yang sudah diseimbangkan, tetapi sangat kecil. Dengan demikian nilai agregat setiap kelas pada data validasi sudah cukup seimbang.

Berdasarkan hasil metrik evaluasi, diperoleh nilai *Log Loss* data latih adalah sebesar 1,056 dan pada data validasi adalah sebesar 1,179. Terdapat kenaikan nilai *Log Loss* sebesar 0,123 yang menandakan model mengalami sedikit *overfitting*. Nilai *Log Loss* model pada data latih dan data validasi dikatakan baik, karena nilai *Log Loss* model masih jauh dari nilai *Log Loss* model ketika melakukan tebak acak, yaitu sebesar 1,792.

Berdasarkan hasil metrik evaluasi, diperoleh *Matthews Correlation Coefficient* (MCC) data latih sebesar 57,112% dan data validasi sebesar 42,861%. Korelasi antara data latih dan data validasi adalah korelasi positif, artinya model menganggap prediksi kelas yang seharusnya adalah benar. Terdapat penurunan korelasi sebesar 14,251% antara data latih dan data validasi yang menandakan adanya *overfitting* ringan. Namun, penurunan angka masih tidak jauh dari rata-rata (sekitar 7,1255% lebih rendah). Nilai MCC pada model masih jauh lebih baik dibanding model yang menebak secara acak. Ketika model menebak secara acak, nilai MCC akan mendekati nol.

Berdasarkan hasil metrik evaluasi, diperoleh nilai Coheren's Kappa data latih adalah sebesar 56,758% dan data validasi sebesar 42,458%. Terdapat penurunan nilai Coheren's Kappa antara data latih dan data validasi sebesar 14,300%. Menandakan bahwa keseimbangan menurun sebesar 14,300% ketika model diberikan data baru, hal ini memberi kesimpulan bahwa model sedikit mengalami *overfitting*.

### 3.3 Prediksi Data Uji

Columbina merupakan karakter bintang 5\*, mempunyai elemen air (*Hydro*), dengan senjata yang digunakan adalah *Catalyst*, mempunyai peran support, dan dirilis pada versi LUNA. Sehingga, enkoding vektornya adalah:

$$Vector_{Columbina} = [1, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1]$$

Setelah dilakukan enkoding, vektor akan diinput ke dalam modelnya, dihasilkan peluang perbandingan antara kelas utama (SS) dan lainnya. Menggunakan *software R*, diperoleh probabilitas masing-masing kelas yang dibandingkan dengan kelas utama yang disajikan pada Tabel 6.

**Tabel 6** Probabilitas Logit

	S vs SS	A vs SS	B vs SS	C vs SS	D vs SS
Probabilitas	0.7700927	2.967957e-10	0.2299073	6.766244e-11	2.126617e-11

Berdasarkan fungsi Softmax pada persamaan (3) dan bantuan *software R*, diperoleh nilai probabilitas masing-masing kelasnya yang disajikan pada Tabel 7.

**Tabel 7** Probabilitas yang Diaktifkan dengan Fungsi Softmax

	SS	S	A	B	C	D
Probabilitas	0.5	0.3850463	1.483979e-10	0.1149537	3.383122e-11	1.063308e-11

Berdasarkan Tabel 7, probabilitas tertinggi terdapat pada kelas SS sebesar 50,000%. Diikuti dengan kelas S sebesar 38,504% dan kelas B sebesar 11,495%. Objek tidak mungkin masuk ke kelas A, C, dan D karena probabilitas kedua kelas hampir mendekati 0. Dengan demikian, Columbina berpeluang besar masuk ke kelas SS. Namun, tidak menutup kemungkinan akan masuk ke kelas S atau B.

#### 4 Kesimpulan dan Saran

Berdasarkan hasil analisis data yang bersumber dari Game 8 : “6.0 Tier List and the Best Characters as of October 2025” dan *machine learning* menggunakan metode klasifikasi Softmax, diperoleh informasi bahwa Columbina berpeluang besar dikategorikan sebagai karakter dengan kelas SS dan tidak menutup kemungkinan akan dikategorikan sebagai kelas S atau B. Namun, Columbina tidak mungkin dikategorikan sebagai kelas A, C, dan D.

Untuk penelitian selanjutnya yang masih berkaitan dengan *machine learning* klasifikasi dan Genshin Impact, dapat menggunakan perangkat lunak selain R, dan menggunakan metode yang berbeda seperti fungsi Softmax berbasis regresi logistik ordinal (*ordinal Softmax*). Jika akan menganalisis data yang berkaitan dengan karakter Genshin Impact, diharapkan artikel dapat diselesaikan kurang dari 40 hari (terhitung dari update Genshin Impact setiap *patch*) mengingat setiap update ada kemungkinan MiHoYo akan merilis karakter baru, sehingga data karakter akan selalu diperbarui.

#### UCAPAN TERIMA KASIH

Kepada seluruh pihak yang terlibat dan membantu penelitian ini.

#### Referensi

- [1] Ellington C., Pramanik P., Robinson K. H. PlayMyData: *A Statistical Analysis of a Video Game Dataset on Review Scores and Gaming Platforms*, Analytics 31(4), pp. 1 – 26, 2025.
- [2] Game 8, *6.0 Tier List and the Best Characters as of October 2025*, <https://game8.co/games/Genshin-Impact/archives/297465>, (16 Oktober 2025).
- [3] GeekforGeeks. *One Hot Encoding in Machine Learning*. <https://www.geeksforgeeks.org/machine-learning/ml-one-hot-encoding/> (16 Oktober 2025)
- [4] Gomez R., *Understanding Categorical Cross-Entropy Loss, Binary Cross-Entropy Loss, Softmax Loss, Logistic Loss, Focal Loss and all those confusing names*, Raúl Gómez blog, [https://gombru.github.io/2018/05/23/cross\\_entropy\\_loss/](https://gombru.github.io/2018/05/23/cross_entropy_loss/) (16 Oktober 2025)

- [5] Grandini M., Bagli E., Visani G., *Metrics for Multi-Class Classification: An Overview*. CRIF S.p.A. arXiv:2008.05756. <https://arxiv.org/abs/2008.05756>
- [6] Hosmer W. D., Lemeshow S., *Applied Logistic Regression* 2<sup>nd</sup> ed., A Wiley-Interscience Publication. 260-262, 2000
- [7] Kusnadi R., Yusuf, Andriantony, Yaputra A. R, Caintan M. *Analisis Sentimen Terhadap Game Genshin Impact Menggunakan Bert*. RABIT : Jurnal Teknologi dan Sistem Informasi Univrab 6(2), pp. 122-129, 2021
- [8] Novianti & Rizki, W. S., *Analisis Regresi Logistik Multinomial Pada Pemilihan Alat Kontrasepsi Wanita*, Buletin Ilmiah Math. Stat. dan Terapannya 8(4), pp. 751 – 758, 2019.
- [9] Pratama P. F. M, *Mengetahui Tingkat Kekuatan Karakter Klee pada Game Genshin Impact Menggunakan Metode Klasifikasi dengan Algoritma Naïve Bayes*, Journal of Data Mining and Information Systems 1(2), pp. 72-77, 2023