Journal of Electrical Engineering and Computer (JEECOM)
Vol. 6, No. 2 (2024), DOI: 10.33650/jeecom.v4i2
p-ISSN: 2715-0410; e-ISSN: 2715-6427

Deteksi Akun *Kaggle Bot* Menggunakan *Linear Regression*

Sudriyanto[1], Muhammad Ali Hafid [2], Moch. Ade Kurniawan [3]
[1,2,3] Universitas Nurul Jadid, Probolinggo, Indonesia

Article Info

Article history:

Diterima 12 Agustus 2024
Revisi 28 September 2024
Diterbitkan 5 Oktober 2024

Keywords:

Kaggle, akun palsu, bot, linear regression, deteksi.

ABSTRAK

Penelitian ini mengkaji permasalahan pemalsuan akun pada *platform Kaggle* dengan fokus pada pengembangan model prediksi menggunakan metode *Linear Regression* untuk mendeteksi akun *bot. Kaggle*, sebagai *platform* terkemuka dalam bidang ilmu data, menghadapi tantangan serius terkait *integritas* data akibat praktik *bot voting* yang berdampak pada keaslian kompetisi dan dataset yang diunggah. Studi ini memanfaatkan dataset *Kaggle Bot Account* yang terdiri dari lebih dari satu juta entri, dengan *variabel independen* mencakup jumlah pengikut, interaksi dengan konten, dan aktivitas pengguna lainnya. Metode *Linear Regression* dipilih karena kemampuannya dalam memodelkan hubungan linear antar variabel, sementara evaluasi kinerja model dilakukan melalui *confusion matrix*. Hasil penelitian menunjukkan bahwa model mampu mengidentifikasi 318 akun palsu dari 143.771 data testing, dengan tingkat akurasi sebesar 0,9968 atau 99,68%. Meskipun demikian, terdapat beberapa kesalahan dalam prediksi akun palsu, yang mengindikasikan perlunya pengembangan lebih lanjut untuk meningkatkan ketepatan deteksi. Kesimpulan penelitian ini menegaskan potensi metode *Linear Regression* dalam mendukung integritas *platform Kaggle* dengan mengurangi dampak negatif akibat keberadaan akun palsu. Penelitian ini memberikan kontribusi signifikan dengan mengeksplorasi karakteristik unik *Kaggle* dan merekomendasikan penelitian lanjutan untuk mengembangkan metode deteksi yang lebih efektif di masa mendatang.

Corresponding Author:
Sudriyanto,
Universitas Nurul Jadid, Krajan, Krampilan Besuk, Probolinggo 67283, Indonesia
Email: sudriyanto@unuja.ac.id

1. PENDAHULUAN

Dalam sebuah penelitian, data memiliki peran yang sangat penting karena menjadi sumber informasi dan angka yang memberikan pemahaman mendalam tentang objek penelitian serta mengungkapkan fenomena yang sedang terjadi [1],[2],[3]. Data dapat diperoleh dari berbagai sumber, baik melalui internet maupun secara langsung dari lokasi penelitian [4],[5]. Namun, proses pengumpulan data sering kali mengalami kendala, yang mengakibatkan terjadinya pemalsuan data dalam penelitian [6]. Dalam beberapa tahun terakhir, banyak laporan menunjukkan adanya praktik bot voting yang menghasilkan data manipulatif dalam kompetisi ilmu data di Kaggle. Pemalsuan data dalam penelitian dapat menimbulkan krisis kredibilitas dalam ilmu pengetahuan [7], di mana hasil penelitian tidak disajikan secara akurat dan bahkan dapat menyesatkan publik jika dipublikasikan. Kaggle merupakan salah satu *platform* terkemuka di bidang Data *Science* dan *Machine Learning* secara global, dengan lebih dari 6000 dataset dan komunitas ilmuwan terbesar saat ini [8]. Banyak data *scientist*, baik pemula maupun yang berpengalaman, aktif menggunakan *Kaggle* karena *platform* ini menawarkan beragam dataset serta lingkungan ilmiah berbasis *web* yang menggunakan teknologi *kontainerisasi* terkini [9]. Namun, perkembangan pesat *Kaggle* dengan segala keunggulannya turut dimanfaatkan oleh pihak-pihak tidak bertanggung jawab untuk kepentingan pribadi dengan membuat akun pengguna palsu (bot). Akun bot ini adalah akun dengan identitas pengguna yang tidak asli. Akun-akun palsu ini merujuk pada individu yang menggunakan media sosial untuk menulis, berpendapat, dan terlibat dalam aktivitas daring tanpa

ISSN: 2715-6427

mengungkapkan identitas pribadi mereka yang sebenarnya [10]. Fenomena akun-akun palsu ini diatur dalam Undang-Undang No. 11 Tahun 2008 Tentang Informasi dan Transaksi Elektronik [11].

Penelitian ini bertujuan untuk mengembangkan dan menghasilkan model prediksi menggunakan metode *Linear Regression* yang dapat membantu para ilmuwan data dalam mendeteksi akun palsu di situs *Kaggle*. Diharapkan, dengan menggunakan metode ini, penelitian dapat memberikan kontribusi penting dalam identifikasi akun palsu di *Kaggle*. Fokus utama penelitian ini adalah pada deteksi akun *bot* di *Kaggle*, dan tidak mencakup deteksi akun palsu di platform lainnya. Data yang digunakan bersumber dari situs *Kaggle*, sementara evaluasi kinerja model akan dilakukan dengan menggunakan *confusion matrix* untuk membandingkan hasil kinerja model *Linear Regression* dengan studi-studi sebelumnya.

Penelitian ini mengambil banyak inspirasi dan referensi dari studi-studi sebelumnya sebagai dasar perbandingan. Studi melakukan klasifikasi akun palsu di media sosial online menggunakan Algoritma RNN [12]. Hasilnya menunjukkan bahwa penggunaan RNN dalam klasifikasi akun palsu mampu mencapai akurasi tinggi dengan kerugian yang rendah, dengan rata-rata akurasi di atas 80%. Studi [13] bertujuan mendeteksi akun *bot* di *Twitter* dengan menggunakan klasifikasi *decision tree*. Hasilnya menunjukkan kinerja model yang cukup baik, terbukti dari akurasi sebesar 88,84%. Studi ketiga [14] melakukan analisis perbandingan terhadap beberapa algoritma *Machine Learning*, yakni SVM, *Naïve Bayes*, *Random Forest*, dan *Adaptive Boosting* untuk mendeteksi akun palsu di Instagram. Hasilnya menunjukkan bahwa *AdaBoost* adalah algoritma dengan akurasi tertinggi sebesar 92,5%, diikuti oleh *Random Forest* sebesar 91,7%, SVM sebesar 90,7%, dan *Naïve Bayes* sebesar 83,6%. Mustafa dan rekan [15] mengusulkan metode *tweet Similarity Feature* dan *Support Vector Machine* untuk deteksi *buzzer* di *Twitter*, yang menghasilkan akurasi sebesar 89%.

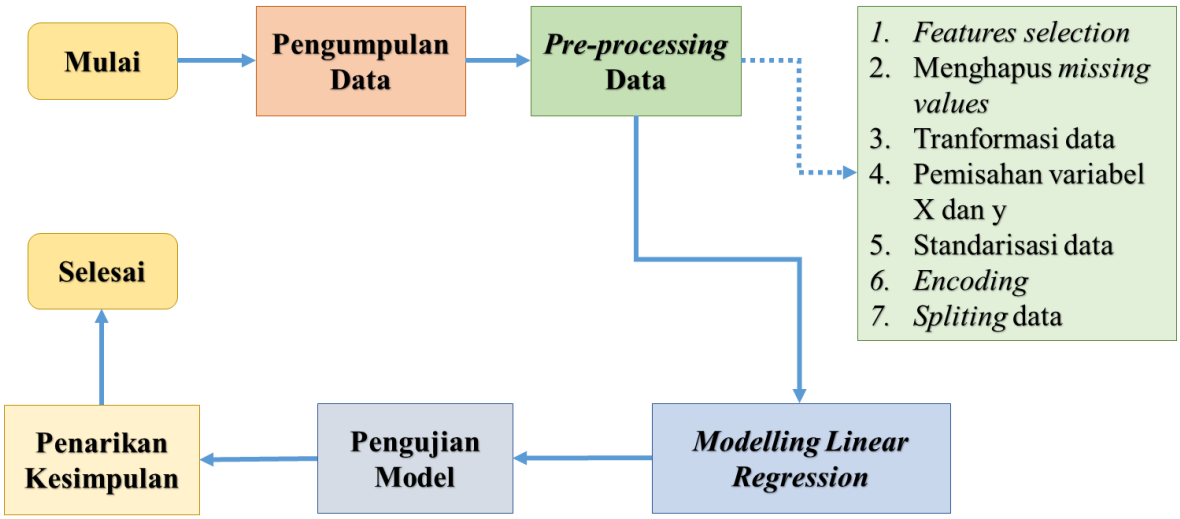
Berdasarkan berbagai penelitian yang telah dilakukan, dapat disimpulkan bahwa penelitian sebelumnya hanya berfokus pada data dan fitur dari *platform* media sosial umum seperti *Twitter* dan *Instagram*. Namun, hingga saat ini belum ada penelitian khusus yang difokuskan untuk mendeteksi akun palsu di *Kaggle*. *Kaggle* memiliki karakteristik dan fitur yang unik dibandingkan dengan *platform* media sosial lainnya, seperti dataset, kompetisi data sains, dan forum diskusi. Oleh karena itu, diperlukan penelitian yang mendalami konteks dan karakteristik khusus dari *Kaggle*, serta mengidentifikasi fitur-fitur yang relevan dalam mendeteksi akun palsu. Selain itu, penelitian sebelumnya cenderung menggunakan algoritma *machine learning* tradisional yang memerlukan waktu pelatihan lebih lama. Maka dari itu, perlu diadakan penelitian untuk menemukan metode yang lebih canggih dan akurat dalam mendeteksi akun palsu di *Kaggle*, seperti menggunakan *regresi linier*.

Linear Regression adalah jenis metode analisis statistik yang digunakan untuk memprediksi hubungan linear antara *variabel independen* dan *variabel dependen* dalam data [16]. Metode ini telah terbukti efektif dalam berbagai aplikasi analisis data dan prediksi di bidang ilmu pengetahuan dan kecerdasan buatan [17]. Keunggulan *Linear Regression* terletak pada kemampuannya untuk menemukan dan memodelkan pola hubungan linier antara variabel-variabel, meskipun pendekatannya lebih sederhana dibandingkan dengan model *non-linear* seperti *Neural Networks*. Penelitian ini menggunakan dataset *Kaggle Bot Account* yang terdiri dari 1.048.574 baris data dengan 9 *variabel independen*, seperti jumlah *followers*, jumlah *following*, jumlah publikasi dataset, jumlah publikasi kode, jumlah diskusi yang diikuti, rata-rata waktu membaca catatan, jumlah total *vote* pada *notebook*, jumlah total *vote* pada data, dan jumlah total *vote* pada komentar. Pemilihan variabel-variabel ini berdasarkan analisis matriks korelasi untuk memastikan relevansinya dalam mendeteksi akun *bot Kaggle*.

Mendeteksi akun *bot* di *Kaggle* dengan menggunakan metode *Linear Regression* sangatlah penting untuk menjaga integritas dan kredibilitas data serta komunitas di *platform* tersebut. Langkah ini mendesak dalam mengidentifikasi dan menghapus akun-akun bot yang mencurigakan, sambil memastikan keaslian dataset yang diunggah. Upaya ini tidak hanya melindungi kepentingan pengguna yang berdedikasi dalam meningkatkan keterampilan analisis data, tetapi juga menjaga reputasi peneliti di komunitas *Kaggle*.

2. METODE

Metode penelitian ini memberikan pandangan tentang langkah-langkah yang akan dilakukan dalam penelitian. Tujuannya adalah untuk mempermudah pendeteksian Akun *Bot Kaggle* menggunakan *Linear Regression*. Berdasarkan gambar 1, tahapan dimulai dengan pengumpulan data, diikuti oleh *pre-processing* data mentah untuk mempersiapkannya sebelum tahap pemodelan. Selanjutnya, dilakukan pembangunan model *Linear Regression* dan pengujian model untuk mengevaluasi kinerjanya. Penarikan kesimpulan dilakukan berdasarkan hasil evaluasi model tersebut.



Gambar 1. Metode Penelitian

2.1 Pengumpulan Data

Dataset yang digunakan dalam penelitian ini diperoleh dari sumber publik yang dapat diakses melalui situs *Kaggle.com*, sebuah platform terkemuka untuk berbagi dan mengelola data. Dataset tersebut terdiri dari 1.048.574 catatan data, yang mencakup 16 variabel atribut dan 1 variabel target. Setiap variabel atribut berfungsi untuk memberikan informasi tambahan yang *relevan*, sementara variabel target digunakan untuk memprediksi atau menentukan hasil akhir dalam analisis. Dengan volume data yang cukup besar, dataset ini diharapkan dapat memberikan hasil yang baik dan mendalam dalam penelitian yang dilakukan.

2.2 Pre-processing Data

Data yang diperoleh dari akun *Kaggle Bot* masih kotor sehingga memerlukan proses pra-pemrosesan yang cukup teliti. Langkah ini mencakup seleksi fitur, penanganan nilai yang hilang, transformasi data, pemisahan antara variabel X dan y, standarisasi data, *encoding*, serta pembagian data menjadi data training (80%) dan data testing (20%). Langkah ini bertujuan untuk mempersiapkan data agar dapat dilanjutkan ke tahapan analisis berikutnya dengan lebih terstruktur dan efektif [18].

2.3 Modelling Linear Regression

Linear Regression merupakan salah satu metode analisis statistik yang umum digunakan untuk memodelkan hubungan antara variabel terikat (*dependent variable*) dan satu atau lebih variabel independen (*independent variables*) [19]. Penelitian telah menunjukkan bahwa penggunaan model *Linear Regression* dapat memberikan prediksi yang akurat dalam berbagai konteks, seperti dalam prediksi harga saham [20], perkiraan cuaca [21], dan pengukuran kualitas produk [22]. Metode ini juga berfungsi untuk menguji sejauh mana hubungan sebab-akibat dari faktor penyebab (x) terhadap variabel akibatnya (y). Rumus persamaan regresi linier umumnya adalah sebagai berikut:

$$Y = a + bX \tag{1}$$

Y = Variabel dependen
a = Konstanta
b = Koefisien
X = Variabel Independen

$$b = \frac{n\sum XY - \sum X \sum Y}{n\sum X^2 - (\sum X)^2} \tag{2}$$

$$a = \frac{\sum Y \sum X^2 - \sum X \sum XY}{n\sum X^2 - (\sum X)^2} \tag{3}$$

ISSN: 2715-6427

2.4 Pengujian Model

Tahapan pengujian model bertujuan untuk mengukur tingkat keberhasilan model yang telah dibuat [23]. Proses pengujian ini dilakukan dengan mengaplikasikan model *Linear Regression* dan menggunakan metode *Confusion Matrix* yang menampilkan nilai *precision, recall, f1-score*, serta *support*. Pengujian terhadap model *Linear Regression* dilakukan menggunakan *tools Google Colab* dengan bahasa pemrograman *Python*.

2.5 Penarikan Kesimpulan

Setelah menyelesaikan seluruh tahapan penelitian yang telah disebutkan di atas, pada tahap ini akan dilakukan analisis terhadap uji coba model *Linear Regression* yang telah dibuat. Dengan demikian, dapat disimpulkan apakah implementasi *Linear Regression* dalam mendeteksi Akun *Bot* di *Kaggle* dapat berfungsi dengan baik atau tidak.

3. HASIL DAN PEMBAHASAN

3.1 Pengumpulan Data

Dataset ini merupakan data akun *bot* dari *Kaggle* yang diperoleh dari situs *kaggle.com* dengan total 1.048.574 data *record*. Data ini terdiri dari 16 variabel independen dan 1 variabel dependen. Gambar 2 menunjukkan sampel awal dataset yang belum diolah.

Unnamed	NAME	GENDER	EMAIL_ID	S_GLOGIN	OWER_CO	OWING_C	ASET_CO	ODE_COU	USSION_C	_READ_TI	STRATION	ATION_LO	VOTES_GA	VOTES_GA	VOTES_GA	ISBOT
0	Johnny Ke	Male	jacksonala	FALSE	53	87	5	3	124		81.88.75.1	Argentina	16	10	3	
1	Dwayne L	Male	calvin80@	TRUE	16	67	5		26	24,97		New Zeala	14	5	2	
2		Male	qbrown@	TRUE	44	81	4	17	125	7,75	159.202.1	Costa Rica	16	4	0	FALSE
3	Russell Si	Male	kimberlyv	TRUE	23	114	5	24	67	13,4	196.11.13	Italy	21	10	1	FALSE
4	Jamie Wil	Female	shaunbroo	FALSE	46	112	2	12	63	24,83	159.196.1	Belgium	10	6	2	FALSE
5	Elijah Par	Male	mpearson	FALSE	2	2	0	0	0	0,62	72.175.20	French Po	18	9	2	TRUE
6	Logan Zim	Male	sparkschristy@exam		46	36	0	16	77	22,32	133.206.7	South Georgia and th		9	1	FALSE
7	Erin Herre	Female		FALSE	2	1	0	0	6	1,85	39.214.11	Antarctica	21	3	1	TRUE
8	Matthew	Male	harrisregi	TRUE	50	25	1	7	122		192.177.30.226			7	3	FALSE
9	Michael S	Male	klopez@e	TRUE	65	99	7	19	93	8,79	68.230.13	Saint Lucia	24	7	2	FALSE
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1321187	Susan Wil	Female		FALSE	0	3	0		0	0,53	118.38.25	Cook Islan	24	9	0	TRUE

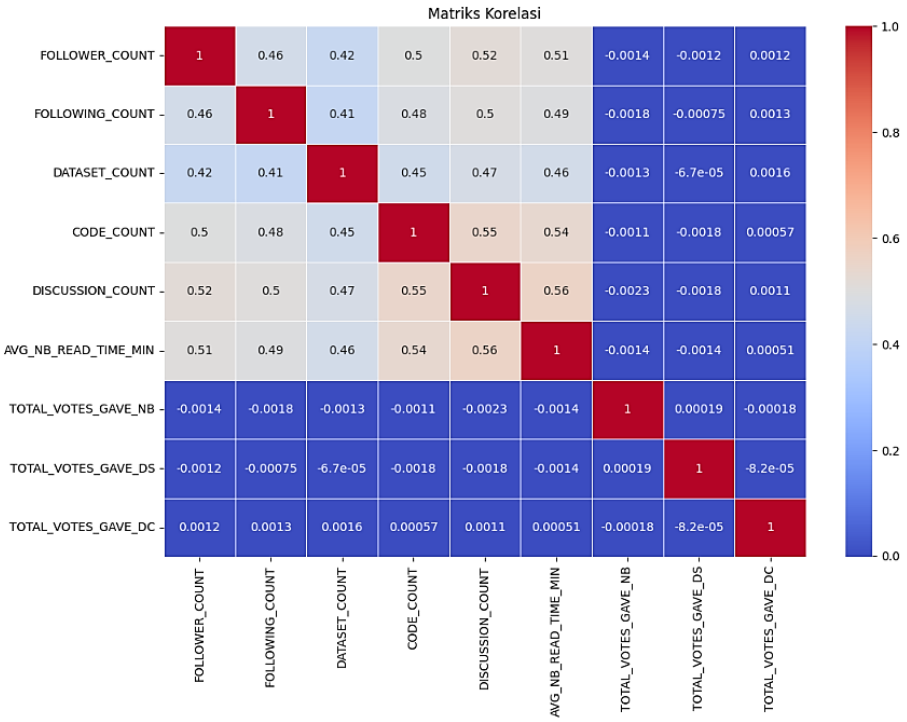
Gambar 2. Sampel Dataset Awal

3.2 Pre-processing Data

Tahap prapemrosesan data merupakan langkah krusial yang harus dilakukan sebelum data digunakan untuk membangun model *prediktif* [23]. Tujuan dari proses ini adalah memastikan bahwa dataset dalam kondisi optimal, bebas dari data yang ambigu atau hilang, yang dapat mempengaruhi kualitas keseluruhan dataset. Prapemrosesan yang tepat akan membantu meningkatkan akurasi model prediksi dengan menyediakan data yang bersih dan siap diolah lebih lanjut [24].

3.2.1 Features Selection dan Missing Values

Pada tahap ini, proses pemilihan fitur akan dilakukan terhadap 16 variabel independen untuk menentukan atribut yang paling penting dalam dataset. Langkah ini diambil karena tidak semua data dan atribut dapat digunakan [25]. Pemilihan fitur bertujuan untuk memilih fitur yang optimal serta mengeliminasi data yang tidak relevan dan berlebihan, menghindari *overfitting*, dan menyederhanakan model. Berdasarkan gambar 2, dalam penelitian ini, pemilihan fitur dilakukan dengan menggunakan matriks korelasi sebagai dasar pemilihan fitur yang optimal. Hasil matrik korelasi dapat dilihat pada gambar 3.



Gambar 3. Matriks Korelasi

Metriks korelasi adalah sebuah tabel yang menampilkan tingkat hubungan antara dua atau lebih variabel dengan rentang nilai antara -1 hingga +1 [26]. Dari Gambar 3, dapat diamati bahwa semakin pekat warna merah, semakin tinggi korelasi antar variabel. Hal ini mengindikasikan adanya hubungan positif yang kuat antara variabel-variabel tersebut. Sebaliknya, warna putih menunjukkan adanya hubungan negatif antar variabel, yang berarti tingkat korelasinya rendah.

Berdasarkan hasil analisis matriks korelasi, terdapat enam variabel independen yang dianggap optimal: *FOLLOWER_COUNT*, *FOLLOWING_COUNT*, *DATASET_COUNT*, *CODE_COUNT*, *DISCUSSION_COUNT*, dan *AVG_NB_READ_TIME_MIN*. Meskipun memiliki tingkat korelasi rendah, variabel *TOTAL_VOTES_GAVE_NB*, *TOTAL_VOTES_GAVE_DS*, dan *TOTAL_VOTES_GAVE_DC* tetap digunakan untuk memperkaya data. Totalnya, terdapat sembilan variabel independen yang digunakan. Variabel *unnamed*: 0 tidak dipilih karena hanya merupakan nomor urut data. Variabel dependen yang digunakan adalah "*ISBOT*" yang terdiri dari dua kelas, yaitu *True* dan *False*. Penjelasan mengenai variabel-variabel ini dapat dilihat pada Tabel 1.

Tabel 1. Variabel yang dipilih

No	Variabel	Penjelasan
1	FOLLOWER_COUNT	Total jumlah pengikut dari individu tersebut
2	FOLLOWING_COUNT	Total jumlah individu yang diikuti oleh individu tersebut
3	DATASET_COUNT	Total jumlah dataset yang dibuat oleh individu tersebut
4	CODE_COUNT	Total jumlah notebook yang dibuat oleh individu tersebut
5	DISCUSSION_COUNT	Total jumlah diskusi yang diikuti oleh individu tersebut
6	AVG_NB_READ_TIME_MIN	Rata-rata waktu yang dihabiskan untuk membaca notebook per menit
7	TOTAL_VOTES_GAVE_NB	Total jumlah suara yang diberikan individu kepada notebook

ISSN: 2715-6427

8	TOTAL_VOTES_GAVE_DS	Total jumlah suara yang diberikan individu kepada dataset
9	TOTAL_VOTES_GAVE_DC	Total jumlah suara yang diberikan individu kepada komentar diskusi
10	ISBOT	Variabel target yang menunjukkan apakah individu tersebut adalah bot atau bukan (True/False)

Dataset pada penelitian ini masih memiliki *missing values* dengan jumlah yang cukup signifikan, seperti yang ditunjukkan pada Gambar 4. Oleh karena itu, salah satu cara untuk mengatasinya adalah dengan menghapus baris yang mengandung *missing values*, sehingga data menjadi lebih lengkap dan model dapat membaca data dengan lebih mudah. Hasil dari proses pemilihan fitur dan penanganan *missing values* dapat dilihat pada Gambar 5.

```
Data columns (total 17 columns):
 #   Column                 Non-Null Count    Dtype
---  ------                 --------------    -----
 0   Unnamed: 0             1321188 non-null  int64
 1   NAME                   1243024 non-null  object
 2   GENDER                 1243309 non-null  object
 3   EMAIL_ID               1243374 non-null  object
 4   IS_GLOGIN              1243272 non-null  object
 5   FOLLOWER_COUNT         1243476 non-null  float64
 6   FOLLOWING_COUNT        1242743 non-null  float64
 7   DATASET_COUNT          1242621 non-null  float64
 8   CODE_COUNT             1243262 non-null  float64
 9   DISCUSSION_COUNT       1243466 non-null  float64
 10  AVG_NB_READ_TIME_MIN   1242872 non-null  float64
 11  REGISTRATION_IPV4      1242859 non-null  object
 12  REGISTRATION_LOCATION  1242898 non-null  object
 13  TOTAL_VOTES_GAVE_NB    1243483 non-null  float64
 14  TOTAL_VOTES_GAVE_DS    1243254 non-null  float64
 15  TOTAL_VOTES_GAVE_DC    1243158 non-null  float64
 16  ISBOT                  1242688 non-null  object
dtypes: float64(9), int64(1), object(7)
memory usage: 171.4+ MB
None
Jumlah data missing value per kolom:
Unnamed: 0                  0
NAME                    78164
GENDER                  77879
EMAIL_ID                77814
IS_GLOGIN               77916
FOLLOWER_COUNT          77712
FOLLOWING_COUNT         78445
DATASET_COUNT           78567
CODE_COUNT              77926
DISCUSSION_COUNT        77722
AVG_NB_READ_TIME_MIN    78316
REGISTRATION_IPV4       78329
REGISTRATION_LOCATION   78290
TOTAL_VOTES_GAVE_NB     77705
TOTAL_VOTES_GAVE_DS     77934
TOTAL_VOTES_GAVE_DC     78030
ISBOT                   78500
dtype: int64
```

Gambar 4. Data sebelum *Features Selection* dan *Missing Values*

```
Data columns (total 10 columns):
 #   Column                Non-Null Count   Dtype
---  ------                --------------   -----
 0   FOLLOWER_COUNT        498987 non-null  float64
 1   FOLLOWING_COUNT       498987 non-null  float64
 2   DATASET_COUNT         498987 non-null  float64
 3   CODE_COUNT            498987 non-null  float64
 4   DISCUSSION_COUNT      498987 non-null  float64
 5   AVG_NB_READ_TIME_MIN  498987 non-null  float64
 6   TOTAL_VOTES_GAVE_NB   498987 non-null  float64
 7   TOTAL_VOTES_GAVE_DS   498987 non-null  float64
 8   TOTAL_VOTES_GAVE_DC   498987 non-null  float64
 9   ISBOT                 498987 non-null  object
dtypes: float64(9), object(1)
memory usage: 41.9+ MB
Informasi untuk atribut terpilih:
Jumlah data missing value per kolom untuk atribut yang dipilih:
FOLLOWER_COUNT          0
FOLLOWING_COUNT         0
DATASET_COUNT           0
CODE_COUNT              0
DISCUSSION_COUNT        0
AVG_NB_READ_TIME_MIN    0
TOTAL_VOTES_GAVE_NB     0
TOTAL_VOTES_GAVE_DS     0
TOTAL_VOTES_GAVE_DC     0
dtype: int64
```

Gambar 5. Data setelah *Features Selection* dan *Missing Values*

3.2.2 Data Transformation

Data variabel target, yang masih berupa teks, membuat model tidak dapat memproses data secara langsung. Oleh karena itu, perlu dilakukan konversi tipe data pada variabel "ISBOT" yang semula bertipe objek menjadi integer. Saat melakukan konversi, nilai False diubah menjadi 0 dan nilai True menjadi 1, seperti yang terlihat pada Gambar 6.

```
ISBOT
    0
    0
    1
    0
    0
```

Gambar 6. Data Variabel target (y) setelah Konversi Data

3.2.3 Pemisahan Data Input X dan Data Output y

Data masukan X berisi informasi dari sembilan kolom variabel independen, seperti yang terlihat dalam Gambar 7. Sementara itu, data keluaran y mengandung satu variabel dependen yang memiliki dua label (Benar/Salah). Detail dari data keluaran y terlihat pada Gambar 8.

Tabel 2. Data Input X

follower Count	Following Count	Dataset Count	Code Count	Discussion Count	Avg. NB Read Time (min)	Total Votes Gave NB	Total Votes Gave DS	Total Votes Gave DC
23.0	114.0	5.0	24.0	67.0	13.4	21.0	10.0	1.0
46.0	112.0	2.0	12.0	63.0	24.83	10.0	6.0	2.0
2.0	2.0	0.0	0.0	0.0	40.62	18.0	9.0	2.0
65.0	99.0	7.0	19.0	93.0	8.79	24.0	7.0	2.0
70.0	14.0	5.0	19.0	75.0	12.17	13.0	3.0	2.0

ISSN: 2715-6427

Tabel 3. Data Output y

ISBOT
0.0
0.0
1.0
0.0
0.0

3.2.4 Standarisasi Data

Teknik standarisasi data dilakukan dengan mengubah skala data sehingga memiliki rata-rata rentang = 0 dan standar deviasi = 1. Standarisasi ini menjadikan data seragam dan memudahkan model untuk menerima masukan dengan baik. Data yang dinormalisasi adalah variabel X, dengan hasil normalisasi ditampilkan pada Gambar 9.

```
[[-0.17291357  1.74661698  0.97265515 ...  0.7567307   1.52826323
  -0.44803759]
 [ 0.82753178  1.69593279 -0.22657461 ... -1.62936914 -0.21732431
   0.44614703]
 [-1.08636368 -1.09169763 -1.02606112 ...  0.1059762   1.09186634
   0.44614703]
 ...
 [ 0.43605316  0.45417015  0.17316864 ...  1.62440337  1.09186634
   1.34033166]
 [-0.30340645  0.37814387  1.37239841 ...  0.97364887  1.52826323
   1.34033166]
 [-0.30340645  0.47951225  0.17316864 ...  1.4074852   0.21907257
  -1.34222222]]
```

Gambar 7. Data Standarisasi

3.2.5 *Encoding*

Metode *One Hot Encoding* digunakan untuk mengubah variabel integer menjadi nilai biner, memudahkan komputer dalam pemrosesan data. Hasil *encoding* dapat dilihat dalam Gambar 10.

	False	True
0	0.0	1.0
1	1.0	0.0
2	0.0	1.0
3	0.0	1.0
4	1.0	0.0

Gambar 8. Data setelah *One-Hot Encoding*

3.2.6 Splitting Data

Pada tahap ini, dataset akan dibagi menjadi dua bagian sebelum proses *training* data dimulai. Metode pembagiannya dilakukan dengan membagi keseluruhan dataset menjadi 80% untuk *training* data dan 20% untuk *testing* data. Hasilnya adalah 575.081 baris data untuk training dan 143.771 baris data untuk testing. Selain itu, dalam proses ini, data *training* akan diacak sebanyak 100.000 kali untuk memastikan konsistensi hasil *training*.

3.3 *Modelling* Data

Pada tahap ini, dilakukan pembuatan model menggunakan Algoritma Regresi Linier. Model ini memanfaatkan total 718.852 data record yang dibagi menjadi 575.081 data untuk pelatihan dan 143.771 data untuk pengujian. Regresi Linier diterapkan dengan menggunakan metode *Regresi Ridge*, sebagaimana terlihat pada Gambar 11.

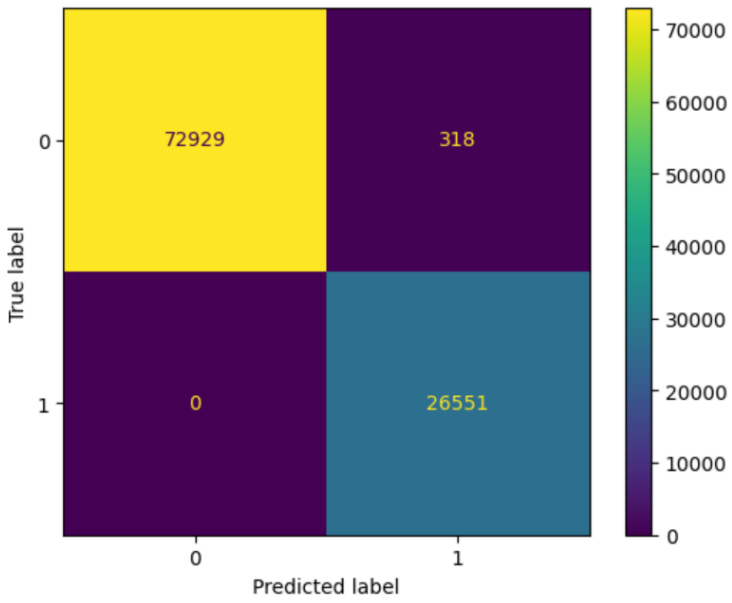
```
# Model Ridge Regression dengan regularisasi dengan alpha=1.0
model = Ridge(alpha=1.0)
```

Gambar 9. Model *Ridge Regression*

Berdasarkan Gambar 11, dapat diketahui bahwa model = *Ridge*(alpha=1.0) digunakan untuk membuat sebuah model *Ridge Regression* dalam *Python*, menggunakan pustaka *scikit-learn*. Di dalam konteks ini, *Ridge* adalah sebuah kelas yang merupakan implementasi dari *Ridge Regression*. *Ridge Regression* adalah salah satu metode regresi yang digunakan untuk mengatasi masalah multikolinearitas dalam model regresi, dengan cara menambahkan regularisasi L2 pada fungsi tujuan (*objective function*) yang akan dioptimalkan.
Dalam *Ridge Regression*, parameter *alpha* (yang diset di sini sebagai 1.0) mengontrol kekuatan regularisasi yang diterapkan pada model. Regularisasi L2 membantu mencegah *overfitting* dengan menambahkan penalti terhadap magnitudo koefisien dalam model, sehingga mendorong nilai *koefisien* mendekati nol tetapi tidak secara langsung menghilangkannya. Nilai alpha yang lebih tinggi akan menghasilkan penalti yang lebih besar, yang mengarah pada model yang lebih disesuaikan dengan data (dengan risiko sedikit *overfitting*) tetapi kurang sensitif terhadap variabel yang tidak relevan.
Tujuan dari menggunakan *Ridge Regression* adalah untuk meningkatkan *generalisasi* model, dengan cara mengurangi *varians* (*overfitting*) pada data yang dilatih, sekaligus mempertahankan akurasi yang baik terhadap data yang tidak terlihat (data uji). Dengan menggunakan *scikit-learn*, kita dapat dengan mudah membangun model, menyesuaikan data pelatihan, dan kemudian menggunakannya untuk membuat prediksi pada data baru.

3.4 Pengujian Model

Setelah proses pelatihan model selesai, langkah selanjutnya adalah melakukan pengujian untuk mengevaluasi kinerja model menggunakan *Confusion Matrix*. Data yang digunakan untuk pengujian merupakan 20% dari total dataset yang berjumlah 143.771 data testing. Berdasarkan hasil *Confusion Matrix* seperti yang terlihat pada Gambar 12, dari 143.771 data *testing*, teridentifikasi 318 akun palsu dan 143.453 akun asli, dengan tingkat akurasi mencapai 99.68%. Hasil evaluasi *Confusion Matrix* menunjukkan bahwa meskipun model *Linear Regression* mampu dengan baik dalam memprediksi akun asli, namun masih terdapat beberapa kesalahan dalam memprediksi akun palsu.



Gambar 10. Confusion Matrix Data Testing

4. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan pada dataset *Kaggle Bot Account* yang terdiri 9 *variabel independent* dan 1 *variabel dependent* dengan total 718.852 data, diperoleh hasil yang sangat baik dalam mendeteksi akun palsu *Kaggle*, dengan nilai akurasi pada data *testing* 0,9968 atau 99,68%. Berdasarkan hasil yang diperoleh, dapat disimpulkan bahwa model yang dilatih memiliki performa yang sangat baik dalam

mendeteksi akun palsu di *Kaggle*. Tingkat akurasi yang tinggi menunjukkan bahwa model ini mampu mengklasifikasikan akun dengan tepat dalam mendeteksi akun bot di *Kaggle*. Kesimpulan penelitian ini didasarkan pada analisis data *Kaggle Bot Account*. Oleh karena itu, penelitian lanjutan diperlukan untuk mendeteksi akun *bot* di *Kaggle* dengan metode penelitian yang berbeda, melibatkan sampel yang lebih besar, serta mengembangkan implementasi penelitian yang lebih komprehensif dan mendalam. Penelitian lanjutan akan membantu menguji keandalan model ini dalam berbagai kondisi dan meningkatkan ketepatan deteksi akun bot di platform lain yang serupa. Dengan demikian, hasil penelitian ini memberikan dasar yang kuat untuk pengembangan model yang lebih efektif dan efisien dalam mendeteksi akun palsu di masa mendatang.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada semua pihak yang telah memberikan dukungan dalam penelitian ini. Ucapan terima kasih khusus disampaikan kepada LP3M Universitas Nurul Jadid atas dukungan dan bantuan finansial yang telah diberikan. Tanpa dukungan tersebut, penelitian ini tidak akan terwujud. Selain itu, penulis juga berterima kasih kepada rekan-rekan dan keluarga yang telah memberikan motivasi dan inspirasi sepanjang proses penelitian.

REFERENSI

[1] M. Sigala, A. Beer, L. Hodgson, and A. O'Connor, "Big data for measuring the impact of tourism economic development programmes: A process and quality criteria framework for using big data," in *Big Data and Innovation in Tourism, Travel, and Hospitality: Managerial Approaches, Techniques, and Applications*, 2019. doi: 10.1007/978-981-13-6339-9_4.

[2] G. Nguyen *et al.*, "Machine Learning and Deep Learning frameworks and libraries for large-scale data mining: a survey," *Artif Intell Rev*, vol. 52, no. 1, 2019, doi: 10.1007/s10462-018-09679-z.

[3] C. Zhang, C. Diao, and T. Guo, "GeoAI for Agriculture," in *Handbook of Geospatial Artificial Intelligence*, 2023. doi: 10.1201/9781003308423-16.

[4] M. Mittal, K. Kumar, and S. Behal, "Deep learning approaches for detecting DDoS attacks: a systematic review," 2023. doi: 10.1007/s00500-021-06608-1.

[5] R. W. D. Pramita, N. Rizal, and R. B. Sulistyan, *Metode Penelitian Kuantitatif*. 2021.

[6] L. Liu, "Intelligent Detection and Diagnosis of Power Failure Relying on BP Neural Network Algorithm," *Comput Intell Neurosci*, vol. 2022, 2022, doi: 10.1155/2022/3758660.

[7] K. Hayawi, S. Saha, M. M. Masud, S. S. Mathew, and M. Kaosar, "Social media bot detection with deep learning methods: a systematic review," 2023. doi: 10.1007/s00521-023-08352-z.

[8] O. Ozdemir, R. L. Russell, and A. A. Berlin, "A 3D Probabilistic Deep Learning System for Detection and Diagnosis of Lung Cancer Using Low-Dose CT Scans," *IEEE Trans Med Imaging*, vol. 39, no. 5, 2020, doi: 10.1109/TMI.2019.2947595.

[9] Kaggle, "Kaggle: Your Home for Data Science," Website.

[10] A. M. Priyatno, M. M. Muttaqi, F. Syuhada, and A. Z. Arifin, "Deteksi bot spammer twitter berbasis time interval entropy dan global vectors for word representations tweet's hashtag," *Register: Jurnal Ilmiah Teknologi Sistem Informasi*, vol. 5, no. 1, 2019, doi: 10.26594/register.v5i1.1382.

[11] V. F. Jahriyah, Moch. Tommy Kusuma, Kuni Qonitazzakiyah, and Muh. Ali Fathomi, "Kebebasan Berekspresi di Media Elektronik Dalam Perspektif Pasal 27 Ayat (3) Undang- Undang Nomor 19 Tahun 2016 Perubahan Atas Undang- Undang Nomor 11 Tahun 2008 Tentang Informasi dan Pelayanan Transaksi Elektronik (UU ITE)," *Sosio Yustisia: Jurnal Hukum dan Perubahan Sosial*, vol. 1, no. 2, 2021, doi: 10.15642/sosyus.v1i2.96.

[12] P. Wanda, M. E. Hiswati, M. Diqi, and R. Herlinda, "Re-Fake: Klasifikasi Akun Palsu di Sosial Media Online menggunakan Algoritma RNN," *Prosiding Seminar Nasional Sains Teknologi dan Inovasi Indonesia (SENASTINDO)*, vol. 3, 2021, doi: 10.54706/senastindo.v3.2021.139.

[13] H. Kurniawan, "Deteksi Twitter Bot menggunakan Klasifikasi Decision Tree," *Jurnal Sustainable: Jurnal Hasil Penelitian dan Industri Terapan*, vol. 9, no. 1, 2020, doi: 10.31629/sustainable.v9i1.2347.

[14] S. Sheikhi, "An efficient method for detection of fake accounts on the instagram platform," 2020. doi: 10.18280/ria.340407.

[15] A. Mustofa *et al.*, "Twitter Buzzer Detection System Using Tweet Similarity Feature And Support Vector Machine," *NJCA (Nusantara Journal of Computers and Its Applications)*, vol. 8, no. 1, 2023, doi: 10.36564/njca.v8i1.306.

[16] M. Pal and P. Bharati, "Introduction to Correlation and Linear Regression Analysis," in *Applications of Regression Techniques*, 2019. doi: 10.1007/978-981-13-9314-3_1.

[17] S. Rath, A. Tripathy, and A. R. Tripathy, "Prediction of new active cases of coronavirus disease (COVID-19) pandemic using multiple linear regression model," *Diabetes and Metabolic Syndrome: Clinical Research and Reviews*, vol. 14, no. 5, 2020, doi: 10.1016/j.dsx.2020.07.045.

[18] F. Nur Fajri, A. Tholib, and W. Yuliana, "Application of Machine Learning Algorithm for Determining Elective Courses in Informatics Study Program," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 8, no. 3, 2022, doi: 10.28932/jutisi.v8i3.3990.

[19] Sarmanu, *Dasar Metodologi Penelitian Kuantitatif Kualitatif dan Statistika*. 2017.

[20] E. P. Ariesanto Akhmad, "Data Mining Menggunakan Regresi Linear untuk Prediksi Harga Saham Perusahaan Pelayaran," *Jurnal Aplikasi Pelayaran dan Kepelabuhanan*, vol. 10, no. 2, 2020, doi: 10.30649/japk.v10i2.83.

[21] A. Luthfiarta, A. Febriyanto, H. Lestiawan, and W. Wicaksono, "Analisa Prakiraan Cuaca dengan Parameter Suhu, Kelembaban, Tekanan Udara, dan Kecepatan Angin Menggunakan Regresi Linear Berganda," *JOINS (Journal of Information System)*, vol. 5, no. 1, 2020, doi: 10.33633/joins.v5i1.2760.

[22] B. A. Wisudaningsi, I. Arofah, and K. A. Belang, "Pengaruh Kualitas Pelayanan Dan Kualitas Produk Terhadap Kepuasan Konsumen Dengan Menggunakan Metode Analisis Regresi Linear Berganda," *STATMAT : JURNAL STATISTIKA DAN MATEMATIKA*, vol. 1, no. 1, 2019, doi: 10.32493/sm.v1i1.2377.

[23] S. Sudriyanto, "Optimizing Neural Networks Using Particle Swarm Optimization (PSO) Algorithm for Hypertension Disease Prediction," *JEECOM Journal of Electrical Engineering and Computer*, vol. 5, no. 2, 2023, doi: 10.33650/jeecom.v5i2.6759.

[24] S. Sudriyanto, A. Khairi, and A. S. Hikam, "Penerapan Algoritma K-Means Untuk Clustering Santri Pra-Sejahtera Di Yayasan Bantuan Sosial (Ybs) Az-Zainiyyah Pondok Pesantren Nurul Jadid," *NJCA (Nusantara Journal of Computers and Its Applications)*, vol. 8, no. 1, 2023, doi: 10.36564/njca.v8i1.234.

[25] R. Hidayad, R. A. Ronaldo, R. A. Prasetiyo, and S. A. Edho Wicaksono, "Optimasi Parameter Support Vector Machine Menggunakan Algoritma Genetika untuk Meningkatkan Prediksi Pergerakan Harga Saham," *COREAI: Jurnal Kecerdasan Buatan, Komputasi dan Teknologi Informasi*, vol. 3, no. 1, 2022, doi: 10.33650/coreai.v3i1.3859.

[26] S. Soewignjo, Sediono, M. F. F. Mardianto, and E. Pusporani, "Prediksi Harga Saham Bank BCA (BBCA) Pasca Stock Split dengan Artificial Neural Network dengan Algoritma Backpropagation," *G-Tech: Jurnal Teknologi Terapan*, vol. 7, no. 4, 2023, doi: 10.33379/gtech.v7i4.3363.